Integración de grafos de conocimiento educativos con modelos de lenguaje masivos mediante arquitecturas RAG para evaluación personalizada en la enseñanza de programación

Alumno: Adrián Bueno Junquero. Director: Prof. Dr. José Luis Fernández Vindel

Resumen—La generación aumentada con grafos de conocimiento (GraphRAG) y el *fine tuning* eficiente en parámetros se promueven como vías para convertir los grandes modelos de lenguaje (LLM) en generadores fiables de retroalimentación formativa de programación; sin embargo, la literatura rara vez separa sus contribuciones ni comprueba si las mejoras reportadas se sostienen fuera del banco de pruebas que las produjo. Presentamos *EKG-Python*, un grafo de conocimiento educativo canónico para Python introductorio (157 conceptos; 1772 tripletas aseveradas que se expanden a 4786 bajo un cierre OWL 2 RL; 30 enlaces `skos:exactMatch` verificados hacia Wikidata; cero violaciones SHACL), y un canal de retroalimentación GraphRAG construido en torno a él. Mediante un diseño factorial 2×2 comparamos un LLM base (A), un modelo ajustado con QLoRA (B), GraphRAG (C) y su híbrido (D) sobre un banco de pruebas sintético reservado (n=50) y sobre entregas reales de estudiantes procedentes de un corpus público de reparación de código universitario (n=60). En distribución, el híbrido D encabeza las métricas objetivas independientes del juez ($0{,}76$ de exactitud de categoría, $0{,}54$ de exactitud de concepto), y el *fine tuning* y el anclaje en el grafo mejoran facetas complementarias y casi ortogonales de la retroalimentación. Sobre código real, la ordenación objetiva *se invierte*, pues el modelo base A alcanza $0{,}802$ de relevancia de corrección frente al $0{,}323$ de D (Friedman p=$8{,}1 \times 10^{-15}$), un fallo de generalización que atribuimos a la homogeneidad intraplantilla medida en los datos sintéticos. Un estudio de fiabilidad de las puntuaciones de calidad muestra que un juez LLM tiene una validez de criterio débil frente a un panel humano de diez personas (Pearson $0{,}203$, negativa en el eje de la explicación divulgativa), mientras que el panel *promediado* resulta fiable (ICC(2,k) $= 0{,}831$) pese a la baja concordancia por evaluador (Fleiss' κ=$0{,}096$). Concluimos que las ganancias de la aumentación dependen de qué se recupera y de cuán realista es el código evaluado, y que los veredictos automatizados de calidad deben anclarse en métricas objetivas inmunes al juez.

Palabras clave—Grafo de conocimiento educativo, GraphRAG, generación aumentada por recuperación, grandes modelos de lenguaje, retroalimentación formativa, enseñanza de la programación, OWL 2 RL, *fine tuning* con QLoRA, fiabilidad entre evaluadores, generalización.

1. Introducción

La retroalimentación formativa es una de las palancas más potentes del aprendizaje del estudiante, pero en la enseñanza de la programación no escala. Las cohortes superan con frecuencia una proporción de cincuenta estudiantes por instructor, mientras que un solo comentario individualizado sobre una entrega exige de quince a veinte minutos de atención experta. La respuesta clásica, el análisis estático combinado con pruebas unitarias, escala sin esfuerzo pero verifica el comportamiento en lugar de la comprensión, puesto que puede informar de que una prueba ha fallado sin explicar *por qué* el modelo conceptual subyacente es erróneo. Una revisión sistemática de la retroalimentación automatizada para ejercicios de programación halló que la inmensa mayoría de los sistemas ofrece únicamente retroalimentación diagnóstica básica en lugar de la retroalimentación conceptual y explicativa que la literatura de las ciencias del aprendizaje identifica como eficaz [1], [2]. Una brecha estructural separa, pues, la retroalimentación barata de la que enseña.

Los grandes modelos de lenguaje (LLM) parecen cerrar esta brecha, pues generan explicaciones individualizadas y fluidas a demanda. Dos técnicas de adaptación dominan la práctica. El *fine tuning* eficiente en parámetros, ejemplificado por la adaptación de bajo rango (LoRA) y su variante cuantizada QLoRA, especializa de forma económica un modelo base hacia un registro de retroalimentación concreto [3], [4], [5]. La generación aumentada por recuperación (RAG), en cambio, condiciona el modelo con contexto externo recuperado en tiempo de inferencia y ancla la salida en una fuente autorizada más allá de la memoria paramétrica [6]. Cuando esa fuente es un grafo de conocimiento (KG) curado [7], el canal GraphRAG resultante puede, en principio, ofrecer una retroalimentación que es a la vez pedagógicamente enmarcada y trazable hasta conceptos de dominio explícitos [8], [9], [10]. En un contexto educativo, un grafo de conocimiento puede codificar la estructura de prerrequisitos y las concepciones erróneas documentadas de

• *Este trabajo ha sido elaborado en primer lugar en el marco de la asignatura de Web Semántica y Enlazado de Datos, con la intención de servir de base para el Trabajo de Fin de Máster del Máster Universitario en Investigación en Inteligencia Artificial, Universidad Nacional de Educación a Distancia (UNED), Madrid, España.*
Correo electrónico: abueno166@alumno.uned.es

Manuscrito entregado en junio de 2026.

un currículo, de modo que la retroalimentación sobre una función recursiva defectuosa se ancle en el concepto de caso base y sus prerrequisitos [11], [12], [13].

La aplicación directa de los LLM a esta tarea choca, sin embargo, con dos obstáculos. El primero es la alucinación, la producción por los LLM de afirmaciones sin respaldo en fuente alguna, un modo de fallo especialmente dañino en la instrucción, donde una explicación incorrecta deseduca activamente [14], [15]. El segundo es un problema de *medición*. Como la evaluación humana de la retroalimentación generada es lenta y costosa, el campo delega en buena medida el juicio de calidad en otro LLM que actúa como evaluador automatizado —el paradigma del "LLM como juez" [16], [17]—, cuya concordancia con los docentes humanos rara vez se comprueba [18], [19], [20], [21]. Si las ganancias reportadas solo las certifica un juez de esta clase, pueden reflejar las idiosincrasias del evaluador más que un valor pedagógico genuino.

Este artículo se organiza en torno a una pregunta principal que el trabajo previo tiende a dejar implícita — *¿cuándo, y bajo qué evaluación, aumentar un LLM con un grafo de conocimiento o con* fine tuning *mejora la retroalimentación de programación?*—, que desglosamos en cuatro cuestiones más concretas. La primera atiende a si puede construirse un grafo de conocimiento educativo validado y enlazado de forma verificable; la segunda, a si cada aumentación y su combinación elevan la calidad objetiva frente a un modelo base y si esa ventaja se transfiere al código real de estudiantes; la tercera, a la fiabilidad de un juez LLM frente a un panel humano; y la cuarta, a si un panel humano promediado constituye un instrumento fiable y discriminante aun con baja concordancia individual. Abordamos esta agenda con tres compromisos que la literatura combina pocas veces. Primero, separamos las dos aumentaciones en un diseño factorial 2×2 limpio en lugar de reportar un único "mejor sistema". Segundo, evaluamos tanto *en distribución*, sobre un banco de pruebas sintético reservado, como *fuera de distribución*, sobre código real de estudiantes, y tratamos la transferencia como la prueba primaria de una afirmación, no como un añadido. Tercero, anclamos cada veredicto comparativo en métricas objetivas de verdad de referencia, inmunes a cualquier juez LLM, y medimos la validez de criterio del juez frente a un panel humano en vez de darla por supuesta.

Preguntas de investigación: **PI1:** ¿Es posible construir un grafo de conocimiento educativo validado con SHACL y cerrado bajo OWL 2 RL que cumpla objetivos estructurales explícitos (≥ 150 conceptos) y enlace de forma verificable con una base de conocimiento abierta? **PI2:** ¿Mejoran el anclaje en el grafo, el *fine tuning* y su combinación la calidad objetiva de la retroalimentación respecto a un LLM base, y se transfiere alguna ventaja en distribución al código real de estudiantes? **PI3:** ¿Cuán fiable es un juez LLM de esta retroalimentación frente a un panel humano? **PI4:** ¿Es un panel humano promediado un instrumento fiable y discriminante incluso cuando la concordancia por evaluador es baja?

Contribuciones: (i) *EKG-Python*, un grafo de conocimiento educativo canónico y cerrado bajo razonamiento para Python introductorio, con enlaces externos verificados y cero violaciones SHACL, junto con la justificación de diseño de su corrección semántica, en particular el uso deliberado de `skos:exactMatch` en lugar de `owl:sameAs`. (ii) Un canal de retroalimentación GraphRAG que combina un vinculador de conceptos basado en el árbol de sintaxis abstracta (AST) con un generador de retroalimentación en español ajustado con QLoRA, y una comparación factorial de cuatro vías que aísla cada aumentación. (iii) Una evaluación en dos distribuciones que documenta tanto una victoria en distribución de los sistemas aumentados como una inversión de la ordenación estadísticamente significativa fuera de distribución, que atribuimos al sobreajuste a las plantillas. (iv) Un estudio de fiabilidad que muestra que el juez LLM *no* es un instrumento absoluto válido, mientras que el panel humano *promediado* es fiable y discrimina los sistemas. De ahí se sigue un mensaje metodológico. En la investigación sobre retroalimentación automatizada de programación, las ganancias de la aumentación son condicionales y los veredictos de calidad deben descansar en evidencia inmune al juez.

2. TRABAJO RELACIONADO

Esta sección sitúa nuestra propuesta en el cruce de cuatro líneas de investigación que rara vez se consideran de forma conjunta. Primero repasamos el uso de grafos de conocimiento en la enseñanza de la programación; después, la retroalimentación automatizada y basada en LLM; a continuación, la recuperación aumentada y su variante sobre grafos junto con sus protocolos de evaluación; y, por último, la validez del paradigma del LLM como juez y la fiabilidad de las puntuaciones. En cada caso señalamos qué tomamos prestado y en qué nos apartamos del trabajo previo.

2.1. Grafos de conocimiento en la enseñanza de la programación

Los grafos de conocimiento se han estudiado ampliamente como tecnología general, con entidades, relaciones tipadas, esquemas y restricciones ontológicas sobre RDF ya bien consolidadas [7], [22]. En educación, revisiones recientes catalogan aplicaciones que abarcan la construcción de ontologías, el modelado del aprendiz y la recomendación de recursos, a la vez que señalan una evaluación de cara al usuario débil y una transferencia entre dominios limitada [13], [23]. Una línea complementaria modela currículos y estructuras de competencias de forma explícita mediante ontologías curriculares [11] y "árboles de habilidades" de prerrequisitos [12], y la construcción de grafos asistida por LLM es en sí misma un tema activo [24]. Nos apoyamos en la pila estándar RDF/RDFS/OWL/SHACL/SPARQL y en el vocabulario SKOS [25], [26], [27], [28], [29], [30], [31] en lugar de reinventarla. Frente a este cuerpo de trabajo, EKG-Python se distingue menos por la novedad del vocabulario que por ser un artefacto *validado y cerrado bajo razonamiento*, con objetivos estructurales explícitos y enlaces externos verificados, empleado aguas abajo como sustrato de recuperación.

2.2. Retroalimentación de programación automatizada y basada en LLM

Los LLM han acelerado el paso de la calificación automática de aprobado/suspenso hacia una retroalimentación formativa rica. La revisión sistemática de la retroalimentación

automatizada sigue siendo la caracterización de referencia de la brecha [1], y la teoría de la retroalimentación subraya que una retroalimentación eficaz debe abordar el modelo conceptual del aprendiz [2]. El trabajo reciente apunta exactamente a esto: revisiones de la calificación automática basada en LLM [32], marcos de retroalimentación pedagógica multifacética [33], generación de pistas de siguiente paso sin revelar soluciones [34], [35], sistemas de retroalimentación integrados en el aula [36], retroalimentación para la refactorización [37] y estudios de la implicación del estudiante con la retroalimentación generada [38]. Las revisiones de LLM para código y de LLM con verificación formal enmarcan el problema de la evaluación [39], [40], y el *fine tuning* de revisión de código estructurado por dimensiones está estrechamente alineado con nuestra estructura de tipo de error/concepto/explicación/sugerencia [41]. Nuestro generador QLoRA se apoya en el linaje del *fine tuning* eficiente [3], [4], [5], [42], [43], pero difiere en que no presuponemos la superioridad del sistema ajustado, sino que comprobamos si transfiere al código real de estudiantes.

2.3. RAG, GraphRAG y su evaluación

RAG surgió como una forma de anclar la generación en evidencia recuperada y reducir la alucinación [6], [14], [15]. GraphRAG extiende esto a fuentes estructuradas en grafo, y dos revisiones consolidan el área, ambas subrayando que los datos en grafo exigen una recuperación específica para grafos [8], [9], [10]. En educación, GraphRAG se ha combinado con grafos de doble estructura de conocimiento para la recomendación de rutas de aprendizaje [44], y RAG se ha empleado para explicar violaciones SHACL [45]. La evaluación de RAG ha convergido en métricas sin referencia y orientadas a la fidelidad, como RAGAS [46], que adoptamos para sondear el anclaje. Nuestro uso de GraphRAG es inusual al emparejarlo con un generador ajustado y al medir el anclaje con una métrica propia de citación de conceptos y con la fidelidad estilo RAGAS, lo que pone de manifiesto la tensión entre ambas.

2.4. Validez del LLM como juez y fiabilidad de las puntuaciones

La fiabilidad de cualquier puntuación se cuantifica con estadísticos establecidos: el κ de Fleiss para la concordancia categórica entre muchos evaluadores [47], el α de Krippendorff para la fiabilidad ordinal [48], [49], el coeficiente de correlación intraclase ICC(2,k) para la fiabilidad de las puntuaciones *promediadas* [50], [51], [52], el análisis de Bland–Altman para la concordancia entre dos métodos [53] y las bandas de Landis–Koch para interpretar κ [54]. Para comparar varios sistemas sobre los mismos elementos usamos la prueba de Friedman con comparaciones post-hoc siguiendo a Demšar [55]. El paradigma del LLM como juez [16], [17] se cuestiona cada vez más: hay estudios que miden la consistencia de las puntuaciones [18], muestran que las decisiones de diseño alteran la fiabilidad [19], cuantifican el sesgo de puntuación [20] y argumentan desde la teoría de la medición que tales jueces fallan tanto en validez como en fiabilidad [21], con remedios como paneles de jueces disjuntos [56] o una prueba formal de sustitución del anotador alternativo [57]. La mayor parte de esta literatura mide la consistencia entre jueces; comparativamente poca mide la validez de criterio juez–humano para una tarea generativa concreta con un panel real, que es lo que reportamos aquí.

3. El grafo de conocimiento EKG-Python y la arquitectura GraphRAG

Cuatro artefactos componen el enfoque: el grafo de conocimiento educativo canónico, un canal de recuperación y generación GraphRAG, un protocolo de *fine tuning* con QLoRA y un diseño factorial de cuatro sistemas.

3.1. Ingeniería de EKG-Python

EKG-Python, el núcleo semántico del proyecto, contiene **157** conceptos y **1772** tripletas aseveradas. Al materializar el cierre OWL 2 RL, el grafo se expande a **4786** tripletas, añadiendo 3014 enunciados inferidos. Los 157 conceptos se organizan como individuos tipados en siete subclases de concepto (principios transversales, paradigmas, estructuras de control, módulos de biblioteca, estructuras de datos, funciones integradas y tipos de datos), junto con temas y concepciones erróneas documentadas, sobre un esquema de **20** clases OWL, **21** propiedades de objeto y **7** propiedades de tipo de dato, elaborado siguiendo la práctica establecida de ingeniería de ontologías en Protégé [58]. El vocabulario propio `pyedu:` se reserva para términos genuinamente específicos del dominio, mientras que los recursos se acuñan bajo `pyr:` y todo lo demás se toma prestado de vocabularios consolidados. Ningún individuo se tipa a mano como `pyedu:Concepto`, ya que la pertenencia a la clase se propaga íntegramente mediante `rdfs:subClassOf`, de modo que la consulta diagnóstica `?x a pyedu:Concepto` devuelve **0** filas sin inferencia y exactamente **157** con inferencia OWL 2 RL. Este contraste $0 \rightarrow 157$ es la demostración central de que es el razonamiento, y no la aseveración manual, lo que puebla la clase de conceptos (Fig. 1).

Una decisión semántica deliberada rige los enlaces externos. Los 30 enlaces verificados hacia Wikidata [59] usan `skos:exactMatch`, no `owl:sameAs`. Bajo OWL 2 RL, `owl:sameAs` fundiría cada entidad externa en la clase de conceptos e inflaría el recuento inferido más allá de los 157 conceptos nativos; `skos:exactMatch` registra la correspondencia sin fusión de identidad y mantiene el recuento en los 157 semánticamente correctos. Articulamos además la jerarquía con 19 enlaces `skos:broader` y modelamos relaciones N-arias por reificación, en particular la clase `pyedu:EvaluacionActividad` (10 instancias), que registra la procedencia de cada aseveración evaluativa.

Validamos el grafo frente a diez formas SHACL de nodo y de propiedad [60] (todo concepto debe pertenecer a un tema, los valores de dificultad deben caer dentro de rango, los ejercicios deben llevar metadatos obligatorios, entre otras). SHACL reporta **0** violaciones sobre el grafo canónico (`sh:conforms` verdadero) y, como control negativo, **6** violaciones sobre un grafo deliberadamente malformado, lo que demuestra que las restricciones tienen un poder discriminante real. Un `CONSTRUCT` que aplana el cierre transitivo de prerrequisitos produce 340 tripletas sin inferencia y 356 con OWL 2 RL, y una consulta federada `SERVICE` recupera del punto de acceso en vivo de Wikidata las descripciones

en español de los conceptos enlazados. El grafo se serializa en Turtle y va y vuelve sin pérdida a través de JSON-LD, N-Triples y RDF/XML [61]. Esto responde afirmativamente a la PI1.

3.2. El canal GraphRAG

Dada una entrega de un estudiante, el canal produce retroalimentación estructurada en tres etapas. Primero, un vinculador de conceptos analiza el AST de la entrega (el módulo `ast` de Python) aumentado con métricas de complejidad (`radon`) y mapea las construcciones de código a conceptos de EKG; la vinculación basada en el AST sustituye a una heurística previa de subcadenas que producía coincidencias espurias. Segundo, un recuperador indexa los 157 conceptos con incrustaciones de frase `nomic-embed-text` (servidas localmente a través de Ollama) y devuelve un subgrafo de los k mejores (los conceptos coincidentes junto con sus prerrequisitos, conceptos relacionados y concepciones erróneas asociadas) expandido mediante SPARQL sobre un almacén RDF [62] con razonamiento nativo OWL 2 RL y soporte SHACL. Tercero, el subgrafo recuperado se serializa en el prompt de generación, y el generador emite retroalimentación en español con una estructura fija de cinco partes: tipo de error, concepto afectado, explicación divulgativa (popular), explicación técnica y una sugerencia accionable. Esta estructura refleja las dimensiones sobre las que la retroalimentación se puntúa después. Todos los modelos se ejecutan localmente por soberanía de datos y coste marginal nulo.

3.3. El protocolo de *fine tuning* con QLoRA

El generador ajustado (el adaptador del Sistema B, reutilizado por el híbrido D) parte de una base `Qwen2.5-Coder-7B-Instruct` [43] adaptada con QLoRA, esto es, cuantización NF4 de 4 bits de la base congelada con adaptadores de bajo rango (rango 16, α=32) entrenados encima [4], [3], en una sola NVIDIA RTX 5090. Los datos de entrenamiento proceden de un corpus sintético de retroalimentación dividido de modo que los esqueletos estructurales quedan fuera del entrenamiento con solapamiento cero, una medida anti-fuga. Una ejecución sin regularización sobreajustó, con la pérdida de evaluación en reservado *ascendiendo* época tras época (1,297, 1,380, 1,410). Añadir NEFTune (α=5) [42] y un dropout de LoRA de 0,1 invirtió la tendencia. La pérdida en reservado *desciende* a 1,051 y luego a 1,028, con una exactitud de token en reservado que alcanza el **0.842** (Fig. 2). Se selecciona como canónico el punto de control de la segunda época.

3.4. El diseño factorial A/B/C/D

Comparamos cuatro sistemas que cruzan el *fine tuning* con el anclaje en el grafo. **A** es el LLM base (`llama3.1:8b`) [63] sin ninguna aumentación. **B** es el modelo ajustado con QLoRA sin el grafo. **C** es GraphRAG sobre el LLM base (anclaje sin *fine tuning*). **D** es el híbrido, el modelo ajustado con GraphRAG. El diseño aísla la contribución marginal del *fine tuning* ($A \to B$, $C \to D$), del anclaje ($A \to C$, $B \to D$) y de su combinación ($A \to D$), y sustenta todos los resultados comparativos que siguen (Fig. 3).

4. DISEÑO EXPERIMENTAL

Esta sección detalla el marco empírico con el que ponemos a prueba las preguntas de investigación. Describimos primero los dos conjuntos de evaluación (el banco de pruebas sintético en distribución y el corpus de código real fuera de distribución) junto con la auditoría de diversidad que motiva su uso conjunto, y a continuación los modelos comparados, el juez automatizado, el panel humano y el protocolo de medición que articula las métricas objetivas, las del juez y las de anotación humana.

4.1. Conjuntos de datos y su diversidad

Usamos dos conjuntos de evaluación. El banco de pruebas *sintético* reservado (n=50) se extrae de esqueletos estructurales no vistos durante el *fine tuning*; para cada caso se conocen la categoría de error prevista y el concepto afectado, de modo que las métricas principales se anclan en la verdad de referencia y son independientes de cualquier juez. Una auditoría de diversidad cuantifica una amenaza a la validez. Aunque el corpus subyacente tiene un 53 % de cadenas objetivo únicas, la similitud coseno media *dentro* de un esqueleto es de 0,962 frente a una media global de 0,604, por lo que los objetivos "únicos" difieren en gran medida en los nombres de variables. Esta homogeneidad intraesqueleto es la causa raíz cuantificada del sobreajuste documentado más adelante, y motiva el segundo conjunto de datos.

El conjunto *real* (n=60) es una prueba de generalización fuera de distribución extraída del corpus público de reparación "intro_prog" de CS1 de la Dublin City University. Aquí, el diff defectuoso→corrección del docente es la verdad de referencia, y la métrica objetiva *relevancia de corrección* es la fracción de tokens de la corrección real a los que hace referencia la retroalimentación. Por seguridad, el código nunca se ejecuta, y solo se aplican `ast.parse` y el cálculo de diferencias.

4.2. Modelos, juez y protocolo

Los sistemas A y C usan `llama3.1:8b`; B y D usan el adaptador Qwen ajustado. El juez LLM es `qwen2.5:32b` [64]. Todos los modelos se ejecutan localmente en una sola RTX 5090 bajo Python 3.12, con la semilla aleatoria fijada en 11. Señalamos, y controlamos, el solapamiento parcial dentro de la misma familia entre el juez (Qwen) y los sistemas B/D, que es una de las razones por las que el veredicto se ancla en métricas objetivas inmunes al juez. Combinamos tres clases de medición. Las métricas *objetivas* se anclan en la verdad de referencia: exactitud de categoría y exactitud de concepto en el conjunto sintético, y relevancia de corrección en el conjunto real. Dos puntuaciones basadas en reglas (identificación y trazabilidad, 1–5) son recodificaciones afínes deterministas de la exactitud de categoría y de concepto, y se reportan como escalas alternativas, no como confirmaciones independientes. Las puntuaciones del *juez* evalúan tres dimensiones cualitativas (explicación divulgativa, explicación técnica, sugerencia) en 1–5; la estabilidad del juez se sondea con un panel multifamilia (Qwen, Llama, Gemma). La fidelidad al subgrafo recuperado se mide con una métrica propia de citación de conceptos y con la fidelidad a nivel de afirmación y la precisión de contexto al estilo RAGAS [46].

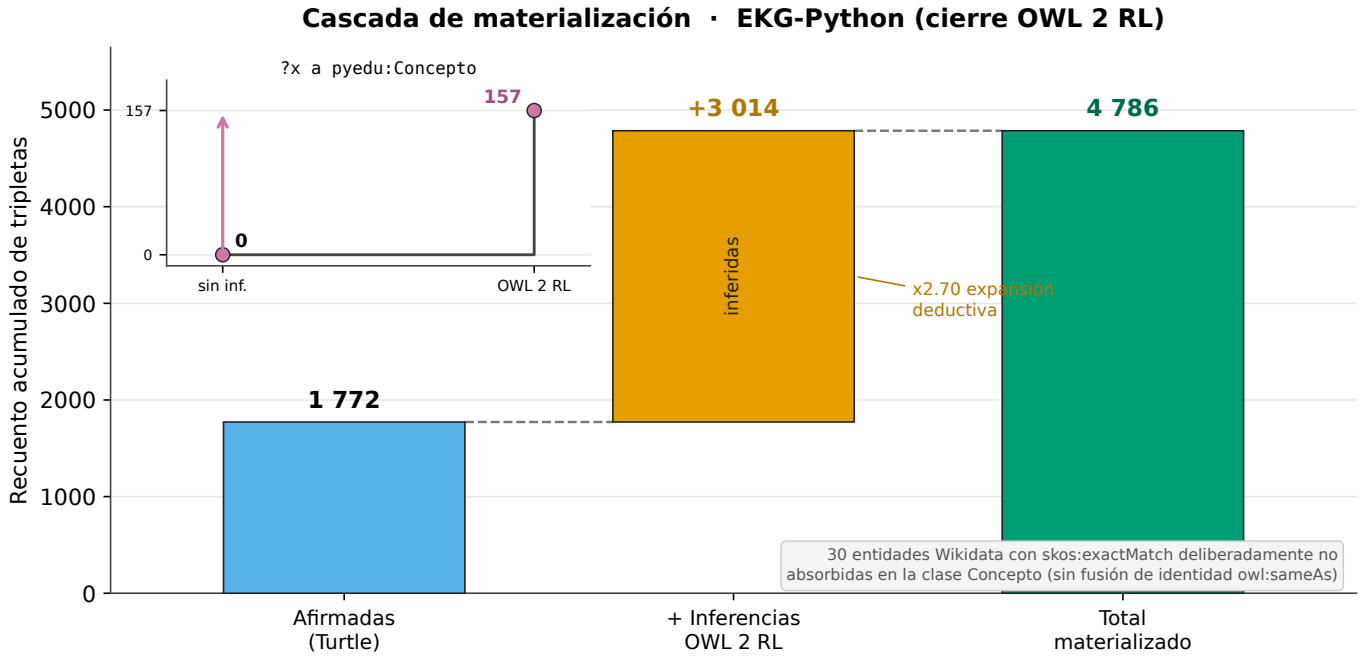


Figura 1. Cascada de materialización de EKG-Python. Las barras siguen el recuento acumulado de tripletas desde los 1772 enunciados aseverados, pasando por las +3014 inferencias añadidas por el cierre OWL 2 RL, hasta las 4786 tripletas finales; un escalón insertado anota el contraste de razonamiento complementario, en el que la consulta diagnóstica `?x a pyedu:Concepto` salta de 0 a 157 filas una vez habilitada la inferencia, mientras que las 30 entidades de Wikidata enlazadas con `skos:exactMatch` *no* se absorben deliberadamente en la clase de conceptos.

La anotación *humana* es el instrumento de criterio. Diez revisores (siete programadores profesionales y tres docentes universitarios, identificados únicamente como R01–R10 a petición suya) evaluaron a ciegas la retroalimentación de los cuatro sistemas sobre los 50 casos reservados en las mismas tres dimensiones cualitativas, con los sistemas anonimizados y barajados. Cada revisor produjo 200 filas (50 casos $\times$ 4 sistemas), cada una con tres puntuaciones Likert, para un total de 2000 filas y **6000** valoraciones humanas a nivel de dimensión frente a las 600 celdas del juez. La concordancia se reporta con el κ de Fleiss, el α de Krippendorff y el ICC(2,k); la validez de criterio humano↔juez, con las correlaciones de Pearson y Spearman y el análisis de Bland–Altman, siendo el consenso humano la media por celda de los diez revisores [47], [48], [50], [53].

5. Resultados

Presentamos los hallazgos en el mismo orden que las preguntas de investigación. Comenzamos por el rendimiento en distribución sobre el banco de pruebas sintético, seguimos con la generalización al código real de estudiantes y la inversión de la ordenación que allí emerge, y cerramos con el estudio de fiabilidad que contrasta la estabilidad del juez LLM con la del panel humano promediado, complementado por una línea de reconstrucción independiente.

5.1. Banco de pruebas en distribución (PI2)

En el banco de pruebas sintético reservado, el híbrido D encabeza las métricas objetivas, con **0.76** de exactitud de categoría y **0.54** de exactitud de concepto, por delante de B, C y el base A (Tabla 1). El patrón factorial resulta informativo. El *fine tuning* (B) es lo que más mejora la clasificación del tipo de error (categoría 0,70), el anclaje en el grafo (C) es lo que más mejora la exactitud de concepto (0,48) y la trazabilidad, y el híbrido, que combina ambos, gana o empata en todas las dimensiones. Las dos aumentaciones mejoran, por tanto, dimensiones distintas y en buena medida independientes de la retroalimentación, sin competir por la misma. El análisis inferencial (Friedman con post-hoc de Wilcoxon corregido por Holm) confirma que todos los sistemas aumentados superan al base A de forma significativa en las métricas objetivas (exactitud de categoría $p < 0{,}001$; IC del 95 % por bootstrap para $D - A$ de $[+0{,}36, +0{,}64]$), pero la ventaja de D sobre B *no* es significativa con este tamaño muestral ($p \approx 0{,}18$ para categoría). Un análisis de anclaje añade matices. D es casi el doble de fiel al subgrafo recuperado que C en la métrica propia de citación de conceptos (0,65 frente a 0,35), aunque cita menos conceptos ($\approx 2{,}5$ frente a > 5 por caso), mientras que a nivel de afirmaciones atómicas la fidelidad estilo RAGAS de C y D es comparable (0,62 frente a 0,58) y la precisión de contexto es idéntica (0,375), lo que señala al recuperador, y no al generador, como el lugar de la mejora.

5.2. Generalización al código real de estudiantes (PI2)

Sobre código real de estudiantes la ordenación **se invierte** (Tabla 2; Fig. 4). El base A alcanza **0.802** de relevancia de corrección y C alcanza 0,733, mientras que el ajustado B cae a 0,433 y el híbrido D a **0.323**. La inversión es estadísticamente robusta. Una prueba de Friedman da χ^2=68,7, p=8,1 $\times$ 10^{-15} (W de Kendall = 0,218), y sobrevive a la deduplicación a 54 casos únicos (p=7,6 $\times$ 10^{-14}). Las

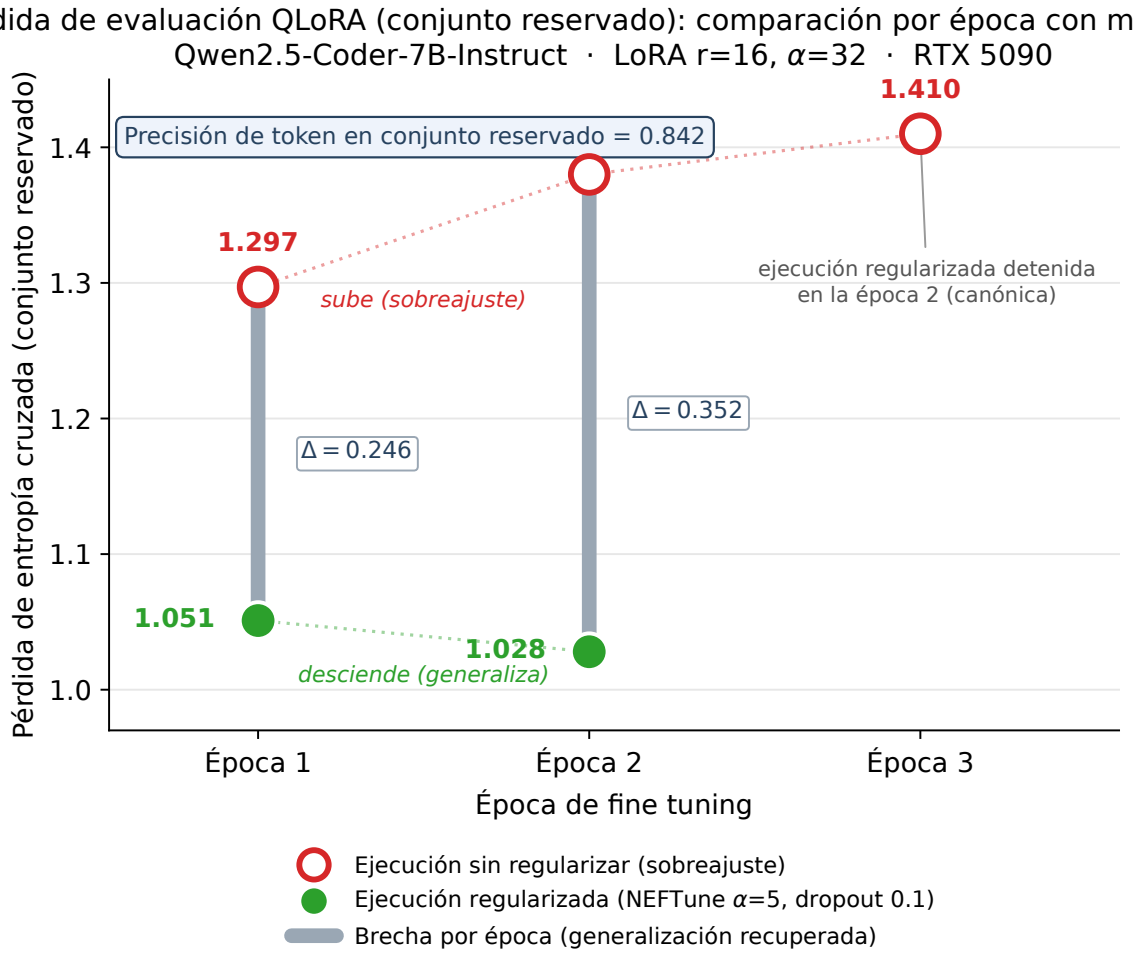


Figura 2. Pérdida de evaluación en reservado del generador QLoRA como comparación de mancuerna conectada a lo largo de las épocas de entrenamiento. La ejecución sin regularizar (marcadores huecos) deriva *al alza* ($1{,}297 \to 1{,}380 \to 1{,}410$), la firma del sobreajuste, mientras que la ejecución regularizada con NEFTune α=5 y dropout de LoRA $0{,}1$ (marcadores rellenos) desciende a $1{,}051$ y luego a $1{,}028$; la separación entre los extremos emparejados visualiza la generalización recuperada. Fuente: registros de *fine tuning* en reservado.

Cuadro 1
Banco de pruebas sintético reservado (n=50). Las métricas objetivas (categoría, concepto) están ancladas en la verdad de referencia y son independientes del juez; las dimensiones cualitativas son puntuaciones del juez LLM (1–5). El híbrido D encabeza o empata en todas.

Métrica	A (base)	B (QLoRA)	C (GraphRAG)	D (híbrido)
Exactitud de categoría	0.26	0.70	0.36	**0.76**
Exactitud de concepto	0.18	0.50	0.48	**0.54**
Identificación (1–5)	2.04	3.80	2.44	**4.04**
Trazabilidad (1–5)	1.72	3.00	2.92	**3.16**
Expl. divulgativa (juez)	3.52	3.38	3.20	**3.66**
Técnica (juez)	3.02	3.44	2.44	**3.62**
Sugerencia (juez)		≈ 2,9 (empate a cuatro)		

Cuadro 2
Código real de estudiantes, CS1 de la DCU de Dublín (n=60). La ordenación objetiva de relevancia de corrección se invierte respecto al banco de pruebas sintético (Friedman p=$8{,}1 \times 10^{-15}$).

Métrica	A (base)	B (QLoRA)	C (GraphRAG)	D (híbrido)
Relevancia de corrección (obj.)	**0.802**	0.433	0.733	0.323
Expl. divulgativa (juez)	3.72	3.18	3.45	3.17
Técnica (juez)	3.03	2.40	2.68	2.47
Sugerencia (juez)	1.97	2.80	2.62	**2.92**

comparaciones post-hoc de Wilcoxon corregido por Holm y de Nemenyi coinciden en que A y C superan significativamente a B y D ($A \approx C$ no significativo). Esta es evidencia directa y confirmada estadísticamente de sobreajuste a las plantillas sintéticas, dado que la ventaja en distribución de los sistemas ajustados no se transfiere a los errores reales. Señalamos una salvedad en ambas direcciones. La métrica de relevancia de corrección premia repetir los tokens literales de la corrección, y los B/D ajustados se entrenaron para dar retroalimentación conceptual en español que, por diseño, no reproduce nombres de variables, de modo que parte de la brecha refleja el *estilo* de la retroalimentación más que su corrección. La señal objetiva, con todo, se sostiene y restringe la afirmación central a "la aumentación ayuda en distribución", no a "el híbrido es el mejor en general".

5.3. Estabilidad del juez y fiabilidad del panel humano (PI3, PI4)

El juez automatizado resulta poco estable. Un panel de jueces multifamilia (Qwen, Llama, Gemma) muestra una concordancia entre jueces débil, con un κ de Fleiss = $0{,}213$

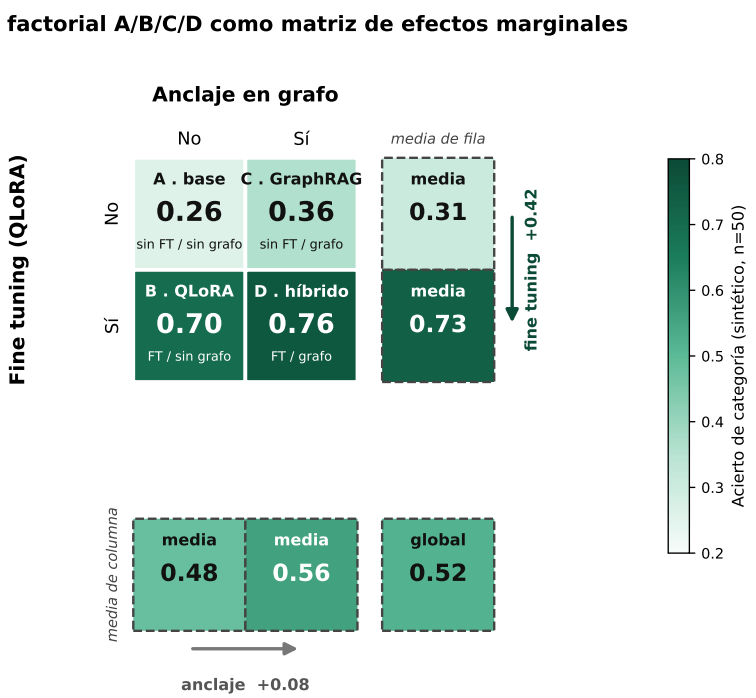


Figura 3. El factorial 2×2 como matriz de efectos marginales. Las cuatro celdas contienen la exactitud de categoría sintética de A, B, C y D cruzando el *fine tuning* con QLoRA (filas) con el anclaje en el grafo (columnas); los márgenes de fila y columna reportan el efecto principal de cada factor como la media sobre su nivel (el *fine tuning* eleva la media de $0{,}31$ a $0{,}73$; el anclaje, de $0{,}48$ a $0{,}56$) y dejan explícito que el *fine tuning* es el efecto principal dominante sobre esta métrica objetiva, mientras que el anclaje aporta una elevación menor y aditiva. Fuente: `resultados_abcd_n50`.

Cuadro 3
Concordancia humana entre anotadores por dimensión (diez revisores, 6000 valoraciones). Los coeficientes por evaluador son bajos, pero la fiabilidad del panel promediado ICC(2,k) es alta.

Coeficiente	Global	Divulg.	Técnica	Sugerencia
κ de Fleiss	0.096	0.071	0.143	0.032
α de Krippendorff	0.318	0.216	0.507	0.116
ICC(2,k) concord. abs.	**0.831**	0.736	**0.910**	0.605

a nivel global, prácticamente ausente en las dimensiones divulgativa y técnica y solo apenas positiva en la dimensión de sugerencia. El juez Qwen es sistemáticamente el más estricto ($\approx 3{,}1$) y Llama/Gemma los más indulgentes ($4{,}1$–$4{,}3$). Una comprobación pareada independiente reproduce el κ ponderado de Cohen Qwen–Llama en $0{,}118$. Las puntuaciones cualitativas del juez dependen, por tanto, fuertemente del juez elegido.

La fiabilidad humana ofrece, en cambio, un panorama distinto. Entre los diez revisores, la concordancia categórica por evaluador también es débil (κ de Fleiss $= 0{,}096$; α de Krippendorff $= 0{,}318$), pero el panel *promediado* es fiable, con un ICC(2,k) $=$ **0.831** (IC del $95\,\%$ $[0{,}80, 0{,}86]$). Estos estadísticos no se contradicen entre sí. La granularidad de 1–5 deprime la concordancia categórica entre individuos, mientras que el promediado cancela el ruido idiosincrásico. Los humanos concuerdan más en la dimensión técnica (ICC $0{,}910$, α $0{,}507$) y menos en la dimensión de sugerencia (ICC $0{,}605$, α $0{,}116$), como muestran la Tabla 3 y la Fig. 5.

El panel humano promediado distingue los cuatro sistemas en todas las dimensiones, con la prueba de Friedman significativa en todas (χ^2=122,3, 137,8 y 92,6 para las dimensiones divulgativa, técnica y de sugerencia; todas $p < 0{,}001$) y los seis contrastes pareados significativos por dimensión bajo pruebas de Wilcoxon corregidas por Holm. Las medias de consenso trazan un perfil interpretable y transversal (Fig. 6). En la dimensión técnica encabezan los sistemas ajustado e híbrido (D $4{,}20$, B $3{,}82$) por encima del base y GraphRAG (A $2{,}37$, C $2{,}58$), mientras que en la dimensión divulgativa (C $3{,}77$, A $3{,}60$, B $3{,}02$, D $2{,}83$) y en la dimensión de sugerencia (C $3{,}64$, A $3{,}23$, D $3{,}03$, B $2{,}90$) la ordenación favorece a C y A. Tal comportamiento es el esperable de un instrumento utilizable y confirma la PI4.

La validez de criterio del juez es pobre. Frente al consenso humano, el juez LLM no se valida como instrumento absoluto (Tabla 4; Fig. 7). La correlación de Pearson global es de solo $0{,}203$ (Spearman $0{,}189$). En la dimensión de la explicación divulgativa es *negativa* ($-0{,}203$), de modo que allí el juez ordena los casos de forma inversa a los humanos, mientras que solo la dimensión técnica alcanza un moderado $0{,}363$ y la dimensión de sugerencia es prácticamente nula ($0{,}041$). El análisis de Bland–Altman da un sesgo medio casi nulo ($+0{,}139$, humano menos juez) pero unos límites de concordancia de $[-2{,}19, +2{,}47]$, que abarcan casi todo el rango 1–5, de modo que el juez y el panel no pueden tratarse como intercambiables a nivel de una celda individual. La correlación se calcula sobre los 33 de 50 casos que tanto los humanos como el juez puntuaron. En la puntuación absoluta, por tanto, la PI3 recibe una respuesta negativa.

Conviene situar la prueba del anotador alternativo en su contexto. El procedimiento formal de sustitución Alt-Test [57] (ε=0,15) devuelve una tasa de victoria global de $1{,}0$ (probabilidad de ventaja media $0{,}696$), nominalmente

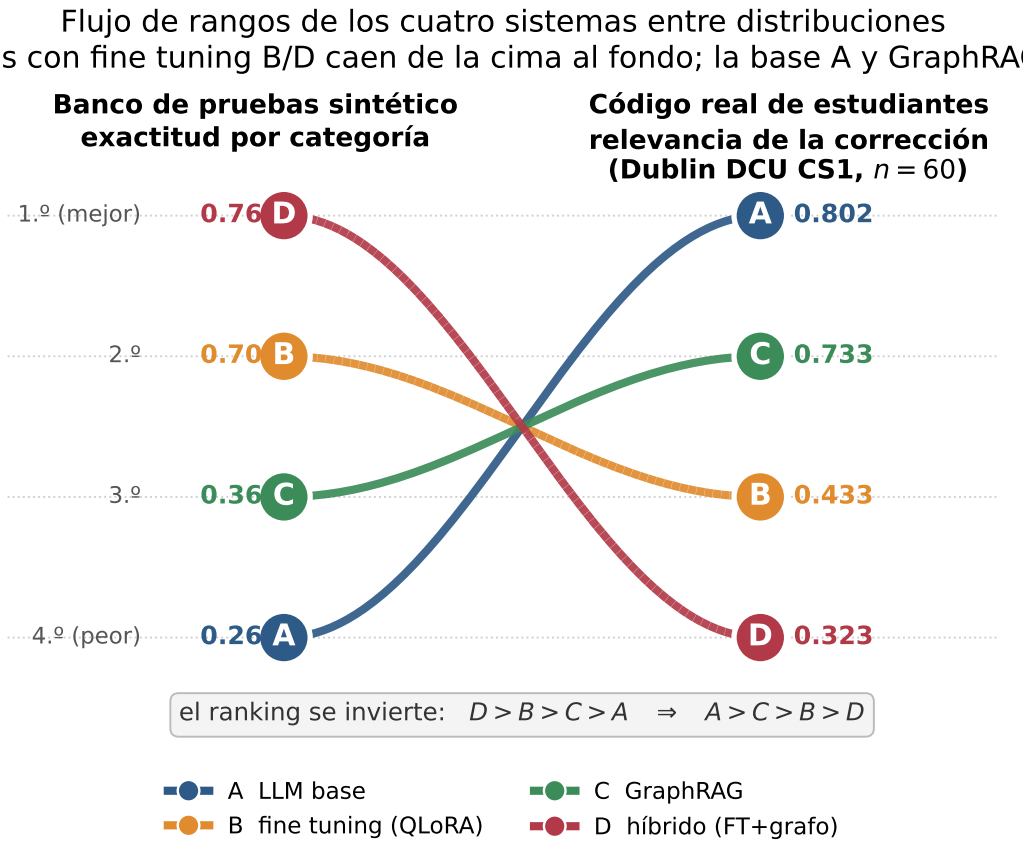


Figura 4. Diagrama de flujo de rangos (bump) de la posición de los cuatro sistemas en la métrica objetiva principal a través de las dos distribuciones. A la izquierda, los sistemas se ordenan por la exactitud de categoría sintética ($D > B > C > A$); a la derecha, por la relevancia de corrección en código real ($A > C > B > D$). Las líneas que se cruzan hacen visible el fallo de generalización: los sistemas ajustado e híbrido caen de los dos primeros puestos a los dos últimos, mientras que los sistemas base y GraphRAG ascienden. Fuente: `resultados_abcd_n50` y la ejecución de CS1 de la DCU de Dublín.

Cuadro 4
Fiabilidad y validez de criterio humano↔juez. El panel promediado es fiable; la validez de criterio del juez es débil y en parte negativa.

Magnitud	Valor
κ de Fleiss entre jueces (multifamilia)	0.213
κ de Fleiss humano entre evaluadores (n=10)	0.096
α de Krippendorff humano (ordinal)	0.318
ICC(2,k) humano, concordancia absoluta	**0.831**
r de Pearson humano↔juez	0.203
dimensión de explicación divulgativa	−0,203
dimensión de explicación técnica	+0,363
Sesgo de Bland–Altman (humano − juez)	+0,139
Casos solapados (humano ∩ juez)	33/50

"sustitución justificada" (por dimensión 1,0/0,6/0,7). Esto *no* contradice la débil validez de criterio, porque la Alt-Test es *relativa* a un panel que de por sí concuerda muy poco (κ de Fleiss = 0,096), de modo que "igualar a un anotador típico" es un listón bajo que un juez internamente consistente supera, mientras que "igualar al consenso" (validez de criterio) sigue siendo difícil. El κ ponderado humano↔juez coincide (global +0,161, *negativo* en la dimensión divulgativa, −0,173). La Alt-Test no rescata al juez como instrumento absoluto; presentamos ambos lado a lado para que el veredicto favorable no se lea de forma aislada.

5.4. Una línea de reconstrucción independiente

Un experimento de reconstrucción separado y autocontenido (que no debe confundirse con el estudio canónico A/B/C/D) alineó el generador con optimización de preferencias ORPO sin referencia y volvió a puntuar cuatro variantes sobre diez casos reservados con el mismo juez `qwen2.5:32b` (Fig. 8). Esto matiza aún más la tesis. La recuperación basada en pasajes *no* supera al base (3,467 y 3,567 frente a 3,733), y solo la recuperación semántica quirúrgica sobre el grafo ontológico (ORPO-v4, 4,233) lo supera, aún por debajo de un techo humano de 4,463 y con la mejora atribuible a la recuperación, no al *fine tuning*. La lectura reconciliada no es "la aumentación gana", sino que su valor depende críticamente de *qué* se recupera y de *cuán real* es el código evaluado.

6. Discusión

Interpretamos a continuación los resultados en dos planos. Primero discutimos bajo qué condiciones la aumentación (el anclaje en el grafo y el *fine tuning*) mejora la retroalimentación, conciliando la victoria en distribución con la inversión observada sobre código real. Después extraemos las implicaciones metodológicas para confiar en la evaluación educativa automatizada y el papel que en ella deben tener el juez LLM y el panel humano.

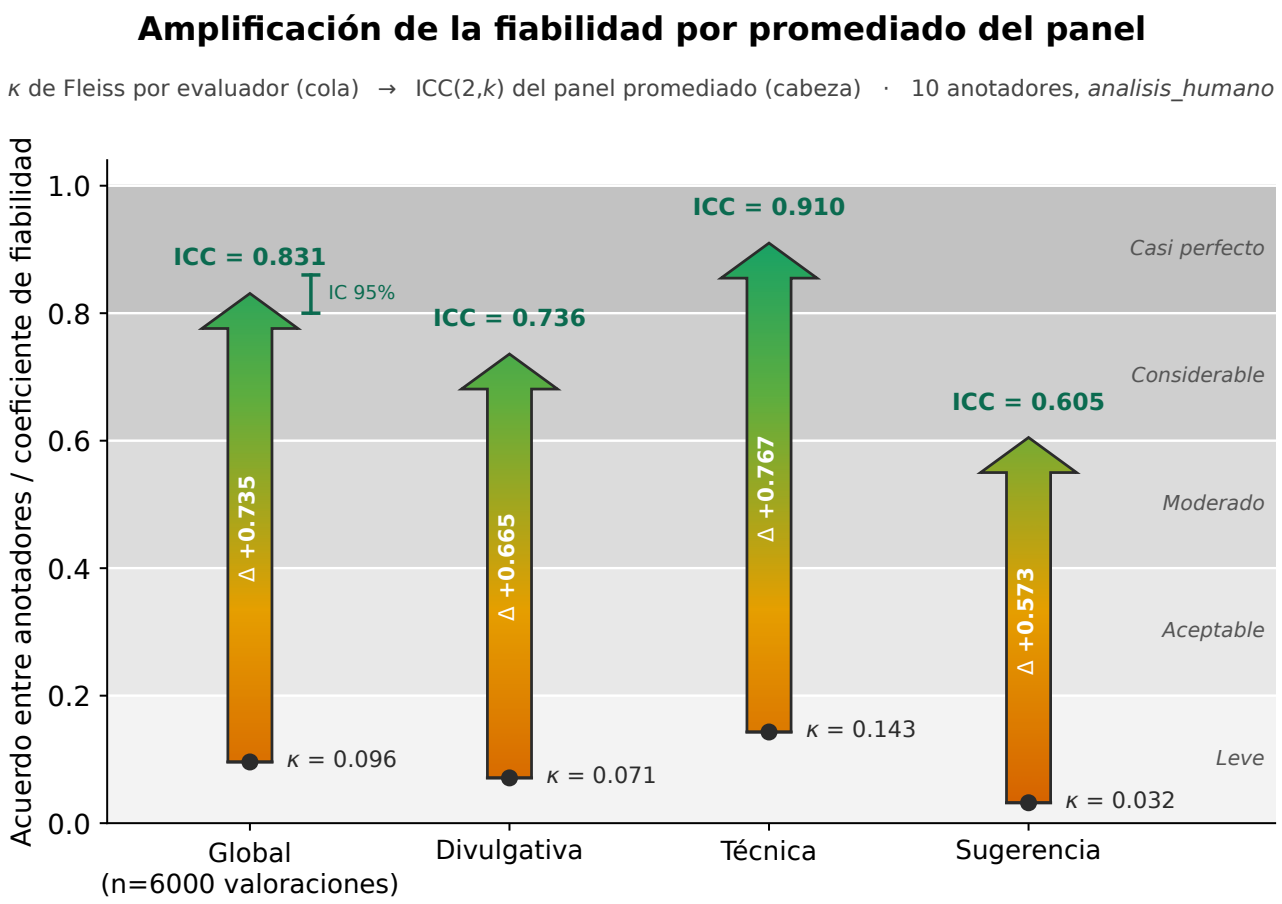


Figura 5. Amplificación de la fiabilidad por promediado del panel. Para cada dimensión, una flecha ascendente conecta el bajo κ de Fleiss por evaluador (cola de la flecha) con el alto ICC(2,k) del panel promediado (cabeza de la flecha), sobre bandas de interpretación de Landis–Koch sombreadas. La dimensión técnica es la más amplificada ($0{,}143 \rightarrow 0{,}910$) y la dimensión de sugerencia la que menos ($0{,}032 \rightarrow 0{,}605$); la flecha global termina en $0{,}831$. Fuente: `analisis_humano`.

6.1. ¿Cuándo ayuda la aumentación?

Los dos bancos de pruebas cuentan una historia coherente una vez que se tiene en cuenta la auditoría de diversidad. En distribución, la aumentación funciona. El *fine tuning* mejora la clasificación del tipo de error, el anclaje en el grafo eleva la identificación de conceptos y la trazabilidad, y el híbrido combina ambos, con todos los sistemas aumentados superando significativamente al LLM base. Fuera de distribución, la ordenación se invierte, y la inversión es estadísticamente significativa. Lo que reconcilia ambas es la homogeneidad intraesqueleto medida en los datos sintéticos de entrenamiento (coseno $0{,}962$ dentro de los esqueletos frente a $0{,}604$ a nivel global). El generador ajustado aprendió el *formato* de un pequeño conjunto de plantillas en lugar de una competencia de retroalimentación generalizable, de modo que su ventaja se evapora sobre el corpus real heterogéneo. Este es un diagnóstico de sobreajuste, y probar fuera de distribución es lo que convierte una sospecha en un resultado medido. La línea de reconstrucción independiente afina la lección, ya que la recuperación solo ayuda cuando es *nativa del grafo y quirúrgica*, no cuando el contexto se inyecta como pasajes planos. En conjunto, ambos hallazgos reformulan la contribución de GraphRAG en educación, de "añade un grafo" a "recupera el subgrafo correcto, y demuéstralo sobre código real".

6.2. Confiar en la evaluación educativa automatizada

El estudio de fiabilidad lleva un mensaje metodológico que trasciende este sistema. No debe presuponerse válido un juez LLM de retroalimentación de programación, pues entre familias es inestable y, frente a un panel humano, su validez de criterio es débil e incluso negativa en la dimensión más subjetiva. Al mismo tiempo, el fallo no es de la medición humana, sino del sucedáneo de un único juez; el panel *promediado* de diez personas es un instrumento fiable (ICC(2,k) $= 0{,}831$) que discrimina los sistemas, aunque la concordancia por evaluador sea baja. La recomendación práctica es doble: anclar los veredictos comparativos en métricas objetivas e inmunes al juez allí donde exista verdad de referencia, y tratar las puntuaciones del juez LLM en las dimensiones subjetivas como señales indicativas que han de corroborarse mediante un panel humano *agregado* en lugar de un único evaluador automatizado.

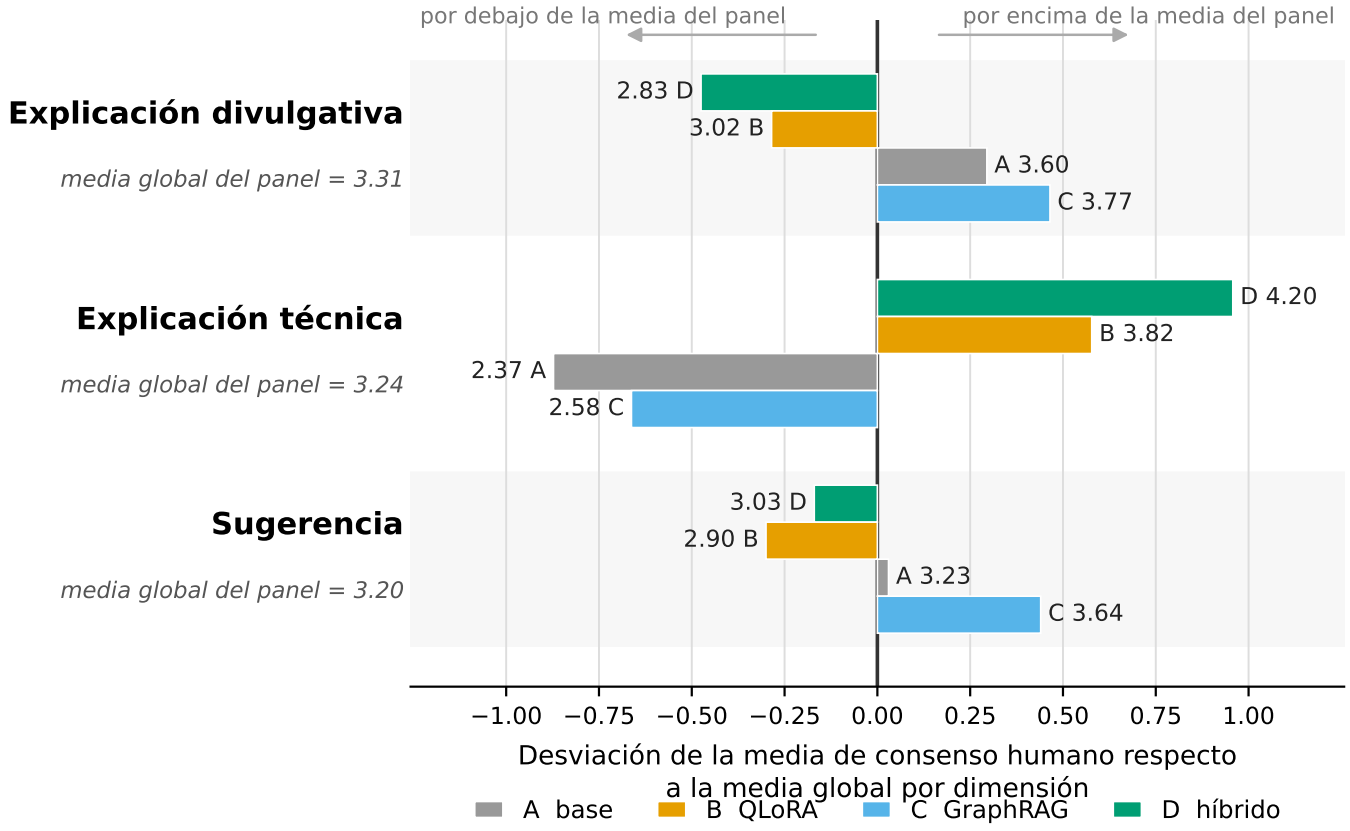


Figura 6. Especialización por dimensión de los cuatro sistemas como gráfico de barras divergentes. La media de consenso humano de cada sistema se muestra como una desviación (a izquierda/derecha de la media global del panel por dimensión) para las dimensiones divulgativa, técnica y de sugerencia; el patrón en espejo entre la fila técnica (B, D positivos) y las filas divulgativa/sugerencia (A, C positivos) es la complementariedad cuantificada del *fine tuning* y el anclaje en el grafo. Fuente: medias de consenso de `analisis_humano`.

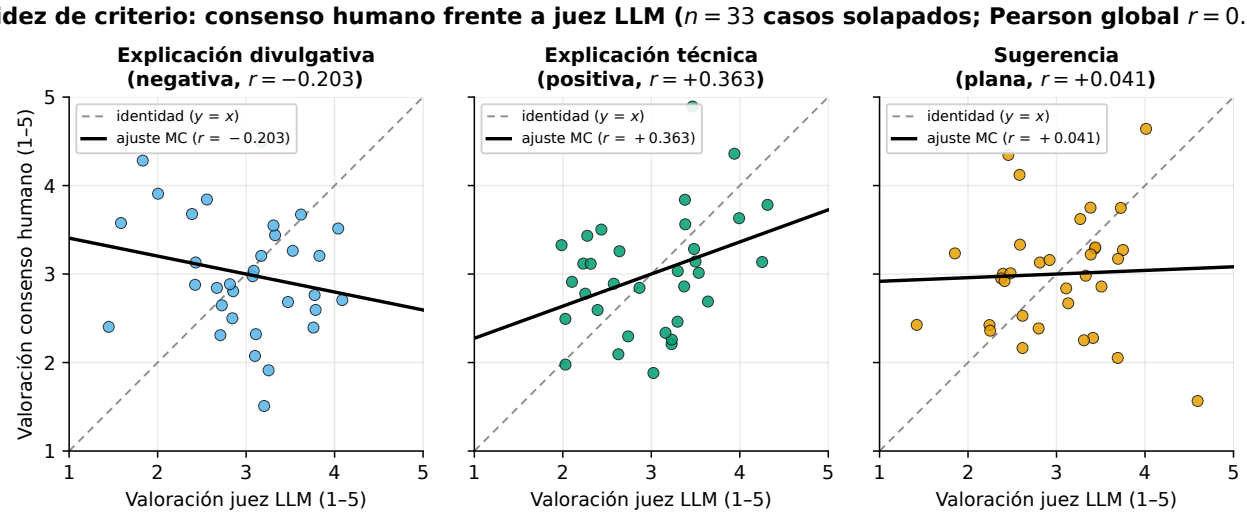


Figura 7. Múltiplos pequeños de dispersión de validez de criterio: valoración de consenso humano (y) frente a la valoración del juez LLM (x) sobre los 33 casos solapados, un panel por dimensión cualitativa con un ajuste por mínimos cuadrados por dimensión. El panel técnico tiene pendiente positiva (r=0,363); el panel divulgativo tiene pendiente *negativa* (r= − 0,203); el panel de sugerencia es plano (r=0,041). El contraste con una hipotética línea de identidad expone cuán lejos está el juez de seguir el consenso humano. Fuente: `analisis_humano` (33 casos comunes).

7. Limitaciones

Varias limitaciones acotan las afirmaciones. El banco de pruebas sintético es pequeño (n=50) y generado a partir de plantillas, por lo que sus cifras absolutas son indicativas, no concluyentes, y la auditoría de diversidad muestra que son homogéneas intraplantilla. La métrica objetiva de relevancia de corrección sobre código real premia el solapamiento

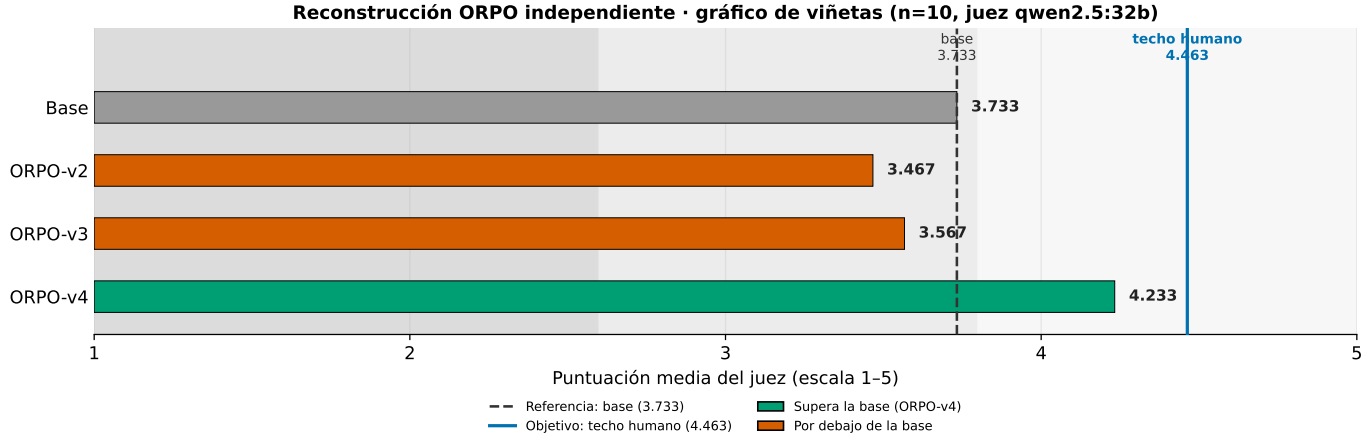


Figura 8. Línea de reconstrucción ORPO independiente como gráfico de viñeta en la escala 1–5 del juez. Las barras horizontales dan la puntuación media de cada variante (Base $3{,}733$; ORPO-v2 $3{,}467$; ORPO-v3 $3{,}567$; ORPO-v4 $4{,}233$); una línea de referencia discontinua marca el nivel base y una línea de objetivo continua marca el techo humano ($4{,}463$). Solo ORPO-v4, que recupera de forma quirúrgica sobre la ontología, supera el base, y ninguna alcanza el techo humano. Fuente: ejecución de reconstrucción independiente ($n{=}10$). Este experimento se reporta por separado y nunca se mezcla con las métricas A/B/C/D.

literal de tokens y, por tanto, penaliza en parte el registro conceptual que los sistemas ajustados fueron entrenados para producir; la inversión debe leerse como evidencia en contra de la *transferencia de la ventaja en distribución*, no como prueba de que el modelo base sea pedagógicamente superior. El juez comparte familia de modelos con los sistemas B y D, razón por la cual el veredicto descansa en métricas inmunes al juez. La correlación de validez de criterio se calcula sobre los 33 de 50 casos puntuados tanto por humanos como por el juez; cerrar este solapamiento a $50/50$ es el único paso pendiente. La retroalimentación se genera en español para un currículo específico, y el panel humano, aunque incluye tres docentes universitarios, es una muestra de conveniencia de diez revisores. Por último, los objetivos de la propuesta original del proyecto (p. ej., $\geq$ $85\,\%$ de exactitud diagnóstica) no se alcanzan, ya que el mejor sistema llega a $0{,}76$ de exactitud de categoría en distribución, y reportamos únicamente lo que se midió.

8. Conclusión

Presentamos EKG-Python, un grafo de conocimiento educativo validado y cerrado bajo razonamiento para Python introductorio, y un canal de retroalimentación GraphRAG construido en torno a él, y empleamos un diseño factorial 2×2 para preguntar cuándo la recuperación sobre el grafo de conocimiento y el *fine tuning* mejoran realmente la retroalimentación de programación. En distribución, las dos aumentaciones mejoran aspectos distintos y el híbrido encabeza; sobre código real de estudiantes, la ordenación objetiva se invierte, un fallo de generalización estadísticamente significativo que se atribuye al sobreajuste a las plantillas. Un estudio de fiabilidad muestra que un juez LLM no es un instrumento absoluto válido, mientras que un panel humano promediado es fiable y discriminante. La contribución es, por tanto, tan metodológica como arquitectónica. Las ganancias de la aumentación en la retroalimentación educativa son condicionales a qué se recupera y a cuán realista es la evaluación, y los veredictos automatizados de calidad deben anclarse en evidencia inmune al juez y corroborarse mediante un juicio humano agregado. El trabajo futuro reemplazará los datos de entrenamiento generados por plantillas con retroalimentación real diversa, cerrará el solapamiento de casos humano↔juez y extenderá EKG-Python a más currículos y lenguajes.

Agradecimientos

El autor agradece el desarrollo de este Trabajo Fin de Máster bajo la dirección del Prof. Dr. José Luis Fernández Vindel, por su orientación sostenida y su lectura crítica a lo largo de todo el proyecto, así como a los diez revisores voluntarios (R01–R10) por su labor de anotación a ciegas. Este trabajo se realizó en el marco del Máster Universitario en Investigación en Inteligencia Artificial de la UNED.

Disponibilidad de los datos

El artefacto EKG-Python (serialización en Turtle, formas SHACL, consultas SPARQL), el arnés de evaluación, las salidas de evaluación sintética y real, la configuración del adaptador QLoRA y los registros anonimizados de anotación humana están disponibles para el tribunal y la dirección del trabajo previa solicitud, y se planea su depósito en el repositorio institucional de la UNED. Las valoraciones humanas no contienen datos personales y se publican únicamente de forma agregada.

Referencias

[1] H. Keuning, J. Jeuring, and B. Heeren, "A systematic literature review of automated feedback generation for programming exercises," *ACM Transactions on Computing Education*, vol. 19, no. 1, pp. 1–43, 2019.

[2] J. Hattie and H. Timperley, "The power of feedback," *Review of Educational Research*, vol. 77, no. 1, pp. 81–112, 2007.

[3] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen, "LoRA: Low-rank adaptation of large language models," arXiv preprint arXiv:2106.09685, 2021. [Online]. Available: https://arxiv.org/abs/2106.09685

[4] T. Dettmers, A. Pagnoni, A. Holtzman, and L. Zettlemoyer, "QLoRA: Efficient finetuning of quantized LLMs," arXiv preprint arXiv:2305.14314, 2023. [Online]. Available: https://arxiv.org/abs/2305.14314

[5] M. Yang, J. Chen, J. Tao, Y. Zhang, J. Liu, J. Zhang, Q. Ma, H. Verma, R. Zhang, M. Zhou, I. King, and R. Ying, "Low-rank adaptation for foundation models: A comprehensive review," arXiv preprint arXiv:2501.00365, 2025. [Online]. Available: https://arxiv.org/abs/2501.00365

[6] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W.-t. Yih, T. Rocktäschel, S. Riedel, and D. Kiela, "Retrieval-augmented generation for knowledge-intensive NLP tasks," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 33, 2020, pp. 9459–9474.

[7] A. Hogan, E. Blomqvist, M. Cochez, C. d'Amato, G. de Melo, C. Gutierrez, S. Kirrane, J. E. Labra Gayo, R. Navigli, S. Neumaier, A.-C. N. Ngomo, A. Polleres, S. M. Rashid, A. Rula, L. Schmelzeisen, J. Sequeda, S. Staab, and A. Zimmermann, "Knowledge graphs," *ACM Computing Surveys*, vol. 54, no. 4, pp. 1–37, 2021.

[8] D. Edge, H. Trinh, N. Cheng, J. Bradley, A. Chao, A. Mody, S. Truitt, and J. Larson, "From local to global: A graph RAG approach to query-focused summarization," arXiv preprint arXiv:2404.16130, 2024. [Online]. Available: https://arxiv.org/abs/2404.16130

[9] B. Peng, Y. Zhu, Y. Liu, X. Bo, H. Shi, C. Hong, Y. Zhang, and S. Tang, "Graph retrieval-augmented generation: A survey," arXiv preprint arXiv:2408.08921, 2024. [Online]. Available: https://arxiv.org/abs/2408.08921

[10] H. Han, Y. Wang, H. Shomer, K. Guo, J. Ding, Y. Lei, M. Halappanavar, R. A. Rossi, S. Mukherjee, X. Tang, Q. He, Z. Hua, B. Long, T. Zhao, N. Shah, A. Javari, Y. Xia, and J. Tang, "Retrieval-augmented generation with graphs (GraphRAG)," arXiv preprint arXiv:2501.00309, 2025. [Online]. Available: https://arxiv.org/abs/2501.00309

[11] A. Christou, C. D. Jaldi, J. Zalewski, H. K. McGinty, P. Hitzler, and C. Shimizu, "An ontology for representing curriculum and learning material," arXiv preprint arXiv:2506.05751, 2025. [Online]. Available: https://arxiv.org/abs/2506.05751

[12] H. Bijl, "Structuring competency-based courses through skill trees," arXiv preprint arXiv:2504.16966, 2025. [Online]. Available: https://arxiv.org/abs/2504.16966

[13] K. Qu, K. C. Li, B. T.-M. Wong, M. M.-F. Wu, and M. Liu, "A survey of knowledge graph approaches and applications in education," *Electronics*, vol. 13, no. 13, p. 2537, 2024. [Online]. Available: https://www.mdpi.com/2079-9292/13/13/2537

[14] Z. Ji, N. Lee, R. Frieske, T. Yu, D. Su, Y. Xu, E. Ishii, Y. J. Bang, A. Madotto, and P. Fung, "Survey of hallucination in natural language generation," *ACM Computing Surveys*, vol. 55, no. 12, pp. 1–38, 2023.

[15] L. Huang, W. Yu, W. Ma, W. Zhong, Z. Feng, H. Wang, Q. Chen, W. Peng, X. Feng, B. Qin, and T. Liu, "A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions," arXiv preprint arXiv:2311.05232, 2023. [Online]. Available: https://arxiv.org/abs/2311.05232

[16] L. Zheng, W.-L. Chiang, Y. Sheng, S. Zhuang, Z. Wu, Y. Zhuang, Z. Lin, Z. Li, D. Li, E. P. Xing, H. Zhang, J. E. Gonzalez, and I. Stoica, "Judging LLM-as-a-judge with MT-Bench and chatbot arena," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 36, 2023, pp. 46 595–46 623.

[17] H. Li, Q. Dong, J. Chen, H. Su, Y. Zhou, Q. Ai, Z. Ye, and Y. Liu, "LLMs-as-Judges: A comprehensive survey on LLM-based evaluation methods," arXiv preprint arXiv:2412.05579, 2024. [Online]. Available: https://arxiv.org/abs/2412.05579

[18] V. Hackl, A. E. Müller, M. Granitzer, and M. Sailer, "Is GPT-4 a reliable rater? evaluating consistency in GPT-4 text ratings," arXiv preprint arXiv:2308.02575, 2023. [Online]. Available: https://arxiv.org/abs/2308.02575

[19] Y. Yamauchi, T. Yano, and M. Oyamada, "An empirical study of LLM-as-a-judge: How design choices impact evaluation reliability," arXiv preprint arXiv:2506.13639, 2025. [Online]. Available: https://arxiv.org/abs/2506.13639

[20] Q. Li, S. Dou, K. Shao, C. Chen, and H. Hu, "Evaluating scoring bias in LLM-as-a-judge," arXiv preprint arXiv:2506.22316, 2025. [Online]. Available: https://arxiv.org/abs/2506.22316

[21] K. Chehbouni, M. Haddou, J. C. K. Cheung, and G. Farnadi, "Neither valid nor reliable? investigating the use of LLMs as judges," arXiv preprint arXiv:2508.18076, 2025. [Online]. Available: https://arxiv.org/abs/2508.18076

[22] S. Pan, L. Luo, Y. Wang, C. Chen, J. Wang, and X. Wu, "Unifying large language models and knowledge graphs: A roadmap," *IEEE Transactions on Knowledge and Data Engineering*, 2024.

[23] S. Wu, Y. Cao, R. Li, J. Cui, H. Qian, B. Jiang, and W. Zhang, "A comprehensive exploration of personalized learning in smart education: From student modeling to personalized recommendations," arXiv preprint arXiv:2402.01666, 2024. [Online]. Available: https://arxiv.org/abs/2402.01666

[24] H. Bian, "LLM-empowered knowledge graph construction: A survey," arXiv preprint arXiv:2510.20345, 2025. [Online]. Available: https://arxiv.org/abs/2510.20345

[25] R. Cyganiak, D. Wood, and M. Lanthaler, "RDF 1.1 concepts and abstract syntax," W3C, W3C Recommendation, 2014. [Online]. Available: https://www.w3.org/TR/rdf11-concepts/

[26] D. Brickley and R. V. Guha, "RDF schema 1.1," W3C, W3C Recommendation, 2014, edited by Guus Schreiber and Dan Brickley. [Online]. Available: https://www.w3.org/TR/rdf-schema/

[27] W3C OWL Working Group, "OWL 2 web ontology language document overview (second edition)," W3C, W3C Recommendation, 2012. [Online]. Available: https://www.w3.org/TR/owl2-overview/

[28] B. Motik, B. Cuenca Grau, I. Horrocks, Z. Wu, A. Fokoue, and C. Lutz, "OWL 2 web ontology language profiles (second edition)," W3C, W3C Recommendation, 2012. [Online]. Available: https://www.w3.org/TR/owl2-profiles/

[29] H. Knublauch and D. Kontokostas, "Shapes constraint language (SHACL)," W3C, W3C Recommendation, 2017. [Online]. Available: https://www.w3.org/TR/shacl/

[30] S. Harris and A. Seaborne, "SPARQL 1.1 query language," W3C, W3C Recommendation, 2013. [Online]. Available: https://www.w3.org/TR/sparql11-query/

[31] A. Miles and S. Bechhofer, "SKOS simple knowledge organization system reference," W3C, W3C Recommendation, 2009. [Online]. Available: https://www.w3.org/TR/skos-reference/

[32] T. A. N. Frederick Eneye, C. F. Ijezue, A. Imam Amjad, M. Amjad, S. Butt, and G. Castañeda-Garza, "Advances in auto-grading with large language models: A cross-disciplinary survey," in *Proceedings of the 20th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2025)*. Association for Computational Linguistics, 2025, pp. 477–498. [Online]. Available: https://aclanthology.org/2025.bea-1.35/

[33] V. Sahu, G. R. Gupta, R. Borikar, and N. Mane, "Autograder+: A multi-faceted AI framework for rich pedagogical feedback in programming education," arXiv preprint arXiv:2510.26402, 2025. [Online]. Available: https://arxiv.org/abs/2510.26402

[34] A. Birillo, E. Artser, A. Potriasaeva, I. Vlasov, K. Dzialets, Y. Golubev, I. Gerasimov, H. Keuning, and T. Bryksin, "One step at a time: Combining LLMs and static analysis to generate next-step hints for programming tasks," arXiv preprint arXiv:2410.09268, 2024. [Online]. Available: https://arxiv.org/abs/2410.09268

[35] T. Phung, J. Cambronero, S. Gulwani, T. Kohn, R. Majumdar, A. Singla, and G. Soares, "Generating high-precision feedback for programming syntax errors using large language models," arXiv preprint arXiv:2302.04662, 2023. [Online]. Available: https://arxiv.org/abs/2302.04662

[36] N. Scholz, M. H. Nguyen, A. Singla, and T. Nagashima, "Partnering with AI: A pedagogical feedback system for LLM integration into programming education," arXiv preprint arXiv:2507.00406, 2025. [Online]. Available: https://arxiv.org/abs/2507.00406

[37] A. Khairnar, A. Rajoju, and E. F. Gehringer, "Teaching code refactoring using LLMs," arXiv preprint arXiv:2508.09332, 2025. [Online]. Available: https://arxiv.org/abs/2508.09332

[38] S. Jacobs, M. Kempf, and N. Kiesler, "That's not the feedback i need! student engagement with GenAI feedback in the tutor Kai," arXiv preprint arXiv:2506.20433, 2025. [Online]. Available: https://arxiv.org/abs/2506.20433

[39] N. Huynh and B. Lin, "Large language models for code generation: A comprehensive survey of challenges, techniques, evaluation, and applications," arXiv preprint arXiv:2503.01245, 2025. [Online]. Available: https://arxiv.org/abs/2503.01245

[40] G. Dolcetti and E. Iotti, "A dual perspective review on large language models and code verification," *Frontiers in Computer Science*, vol. 7, p. 1655469, 2025.

[41] Y. Yu, G. Shi, X. Wu, H. He, X. Gu, Q. Zhao, K. Liu, Q. Wang, Z. Tian, H. Shen, and G. Rong, "Fine-tuning LLMs to analyze multiple dimensions of code review: A maximum entropy regulated long chain-of-thought approach," arXiv preprint arXiv:2509.21170, 2025. [Online]. Available: https://arxiv.org/abs/2509.21170

[42] N. Jain, P.-y. Chiang, Y. Wen, J. Kirchenbauer, H.-M. Chu, G. Somepalli, B. R. Bartoldson, B. Kailkhura, A. Schwarzschild, A. Saha, M. Goldblum, J. Geiping, and T. Goldstein, "NEFTune: Noisy embeddings improve instruction finetuning," arXiv preprint arXiv:2310.05914, 2023. [Online]. Available: https://arxiv.org/abs/2310.05914

[43] B. Hui, J. Yang, Z. Cui, J. Yang, D. Liu, L. Zhang, T. Liu, J. Zhang, B. Yu, K. Lu *et al.*, "Qwen2.5-Coder technical report," arXiv preprint arXiv:2409.12186, 2024. [Online]. Available: https://arxiv.org/abs/2409.12186

[44] X. Cheng, Z. Zhang, J. Wang, L. Fang, C. He, Q. Guan, S. Pan, and W. Luo, "GraphRAG-induced dual knowledge structure graphs for personalized learning path recommendation," arXiv preprint arXiv:2506.22303, 2025. [Online]. Available: https://arxiv.org/abs/2506.22303

[45] G. C. Publio and J. E. Labra Gayo, "xpSHACL: Explainable SHACL validation using retrieval-augmented generation and large language models," arXiv preprint arXiv:2507.08432, 2025. [Online]. Available: https://arxiv.org/abs/2507.08432

[46] S. Es, J. James, L. Espinosa-Anke, and S. Schockaert, "RAGAS: Automated evaluation of retrieval augmented generation," arXiv preprint arXiv:2309.15217, 2023. [Online]. Available: https://arxiv.org/abs/2309.15217

[47] J. L. Fleiss, "Measuring nominal scale agreement among many raters," *Psychological Bulletin*, vol. 76, no. 5, pp. 378–382, 1971.

[48] K. Krippendorff, *Content Analysis: An Introduction to Its Methodology*, 4th ed. Thousand Oaks, CA: SAGE Publications, 2019.

[49] G. Marzi, M. Balzano, and D. Marchiori, "K-Alpha calculator–Krippendorff's alpha calculator: A user-friendly tool for computing Krippendorff's alpha inter-rater reliability coefficient," *MethodsX*, vol. 12, p. 102545, 2024. [Online]. Available: https://pmc.ncbi.nlm.nih.gov/articles/PMC11636850/

[50] P. E. Shrout and J. L. Fleiss, "Intraclass correlations: Uses in assessing rater reliability," *Psychological Bulletin*, vol. 86, no. 2, pp. 420–428, 1979.

[51] K. O. McGraw and S. P. Wong, "Forming inferences about some intraclass correlation coefficients," *Psychological Methods*, vol. 1, no. 1, pp. 30–46, 1996.

[52] K. Ruan, X. Wang, J. Hong, P. Wang, Y. Liu, and H. Sun, "Evaluating LLMs' divergent thinking capabilities for scientific idea generation with minimal context," arXiv preprint arXiv:2412.17596, 2024. [Online]. Available: https://arxiv.org/abs/2412.17596

[53] J. M. Bland and D. G. Altman, "Statistical methods for assessing agreement between two methods of clinical measurement," *The Lancet*, vol. 327, no. 8476, pp. 307–310, 1986.

[54] J. R. Landis and G. G. Koch, "The measurement of observer agreement for categorical data," *Biometrics*, vol. 33, no. 1, pp. 159–174, 1977.

[55] J. Demšar, "Statistical comparisons of classifiers over multiple data sets," *Journal of Machine Learning Research*, vol. 7, pp. 1–30, 2006.

[56] P. Verga, S. Hofstatter, S. Althammer, Y. Su, A. Piktus, A. Arkhangorodsky, M. Xu, N. White, and P. Lewis, "Replacing judges with juries: Evaluating LLM generations with a panel of diverse models," arXiv preprint arXiv:2404.18796, 2024. [Online]. Available: https://arxiv.org/abs/2404.18796

[57] N. Calderon, R. Reichart, and R. Dror, "The alternative annotator test for LLM-as-a-judge: How to statistically justify replacing human annotators with LLMs," in *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (ACL)*, 2025, pp. 16 051–16 075. [Online]. Available: https://aclanthology.org/2025.acl-long.782/

[58] M. DeBellis, "A practical guide to building owl ontologies using protégé 5.5 and plugins," Stanford Center for Biomedical Informatics Research (Protégé project), Tutorial Edition 3.2, 2021, revised from the Protégé 4 tutorial by Matthew Horridge.

[59] D. Vrandečić and M. Krötzsch, "Wikidata: A free collaborative knowledgebase," *Communications of the ACM*, vol. 57, no. 10, pp. 78–85, 2014.

[60] A. Sommer and N. J. Car, "pySHACL: A Python validator for SHACL," Computer software, RDFLib. [Online]. Available: https://github.com/RDFLib/pySHACL

[61] RDFLib Team, "RDFLib: A Python library for working with RDF," Computer software. [Online]. Available: https://github.com/RDFLib/rdflib

[62] Ontotext, "GraphDB documentation," Software documentation, Ontotext AD, version 11.1; RDF triplestore. Accedido en 2026. [Online]. Available: https://graphdb.ontotext.com/documentation/

[63] A. Grattafiori *et al.*, "The Llama 3 herd of models," arXiv preprint arXiv:2407.21783, 2024, llama Team, AI @ Meta. [Online]. Available: https://arxiv.org/abs/2407.21783

[64] A. Yang, B. Yang, B. Zhang, B. Hui, B. Zheng, B. Yu, C. Li, D. Liu, F. Huang, H. Wei, H. Lin, J. Yang, J. Tu, J. Zhang, J. Yang, J. Yang, J. Zhou, J. Lin, K. Dang, K. Lu, K. Bao, K. Yang, L. Yu, M. Li, M. Xue, P. Zhang, Q. Zhu, R. Men, R. Lin, T. Li, T. Tang, T. Xia, X. Ren, X. Ren, Y. Fan, Y. Su, Y. Zhang, Y. Wan, Y. Liu, Z. Cui, Z. Zhang, and Z. Qiu, "Qwen2.5 technical report," arXiv preprint arXiv:2412.15115, 2024. [Online]. Available: https://arxiv.org/abs/2412.15115