¿El reranking es todo lo que necesitas? Análisis comparativo de recuperación en dos etapas vs. ajuste fino de embeddings en sistemas RAG con pocos datos

Alex Dantart*
Humanizing Internet
arxiv@humanizinginternet.com

Resumen

La implementación de sistemas de Generación Aumentada por Recuperación (RAG) en dominios corporativos especializados se enfrenta persistentemente al problema del "Arranque en frío" (*Cold Start*): la existencia de vastos corpus documentales pero una carencia crítica de pares de entrenamiento etiquetados ($N < 500$). Este estudio presenta un experimento controlado que evalúa la eficacia de dos estrategias de adaptación de dominio contrapuestas: el **Ajuste Fino Few-Shot** de modelos bi-encoder (e.g., `text-embedding-3-large`) frente a una arquitectura de **Recuperación en Dos Etapas** (Búsqueda Híbrida + Reranking con Cross-Encoder). Utilizando "CorpLeg-300", un conjunto de datos sintético propietario del dominio legal-financiero, demostramos empíricamente que la arquitectura de dos etapas supera consistentemente a los modelos ajustados finamente, logrando un incremento del 18.5 % en NDCG@10 (0.847 vs 0.715 Hybrid Search) y un aumento del 12.2 % en Recall@10 (0.965 vs 0.860). Nuestros resultados indican que, en regímenes de datos escasos, los embeddings ajustados sufren de sobreajuste rápido y degradación de la variedad semántica, mientras que los *cross-encoders* modernos generalizan eficazmente sin entrenamiento adicional. Este artículo cuantifica el *trade-off* costo-latencia, concluyendo que la penalización de inferencia del reranking ($\approx$ 300ms) es marginal comparada con la deuda técnica y la inestabilidad del ajuste fino en escenarios industriales.

1 Introducción

La integración de Modelos de Lenguaje de Gran Tamaño (LLMs) en flujos de trabajo empresariales ha consolidado a la Generación Aumentada por Recuperación (RAG) como la arquitectura estándar para la gestión del conocimiento [17]. Sin embargo, la transición de pruebas de concepto a sistemas de producción en dominios verticales (legal, farmacéutico, corporativo, ingeniería aeroespacial...) revela una limitación crítica en los modelos de recuperación densa generalistas: su incapacidad para capturar matices terminológicos y relaciones semánticas específicas del dominio sin una adaptación previa.

Para mitigar este desajuste semántico, la literatura académica tradicional sugiere la **adaptación al dominio** mediante el ajuste fino (*Fine-Tuning*) de los modelos de embeddings utilizando aprendizaje contrastado. No obstante, esta aproximación se fundamenta en un supuesto que rara vez se cumple en la industria: la disponibilidad de conjuntos de datos de entrenamiento masivos y de alta calidad (tripletas de consulta, positivo, negativo). En la práctica, las empresas operan en un régimen de "arranque en frío", donde poseen millones de documentos no estructurados pero carecen de *logs* de búsqueda históricos o anotaciones humanas de relevancia.

Este estudio aborda el dilema ingenieril resultante: ¿Deberían las organizaciones invertir recursos en curar pequeños conjuntos de datos para realizar un *Few-Shot Fine-Tuning*, o deberían optar por arquitecturas de inferencia más pesadas pero agnósticas al entrenamiento, como el **reranking en dos etapas**?

Específicamente, este trabajo responde a tres preguntas de investigación:

RQ1: ¿Puede un modelo de embeddings generalista ajustado con N <500 ejemplos superar a una arquitectura híbrida sin entrenamiento en dominios especializados?

RQ2: ¿Cuál es el umbral mínimo de datos etiquetados (Data Break-even Point) necesario para que el Fine-Tuning sea más eficiente que el Reranking?

RQ3: ¿Es el incremento de latencia del Reranking (factor 3x) aceptable en aplicaciones industriales donde la precisión es crítica?

1.1 El problema de la escasez de datos en la adaptación de embeddings

Estudios recientes, como los de Chaofan Li et al. (2025) [10] y Manveer Singh Tamber et al. (2025) [11], han destacado la fragilidad del aprendizaje contrastado cuando se aplica en entornos de bajos recursos. Éste último estudio demuestra que el ajuste fino con un número insuficiente de negativos difíciles (*hard negatives*) o con consultas sintéticas de baja diversidad puede conducir al colapso del

*Otros papers del autor: Arxiv

modelo, donde el *retriever* pierde su capacidad de generalización semántica en favor de la memorización léxica. Este fenómeno es particularmente peligroso en entornos industriales donde el error de recuperación puede derivar en alucinaciones críticas del LLM generador.

1.2 El renacimiento del Cross-Encoder

Como alternativa al ajuste de pesos, la arquitectura de *Retrieve-then-Rerank* ha ganado tracción. Arjun Rao et al. (2025) [1] proponen en *"Rethinking Hybrid Retrieval"* que combinar modelos de embeddings compactos con una etapa de reordenamiento basada en LLMs o Cross-Encoders puede superar a los modelos de embeddings monolíticos más grandes. La hipótesis subyacente es que los Cross-Encoders, al procesar la consulta y el documento simultáneamente a través de capas de autoatención completa (*full self-attention*), actúan como "Jueces" de relevancia mucho más robustos *zero-shot* que los Bi-Encoders, que deben comprimir toda la información en un vector fijo antes de la interacción [5].

Sin embargo, esta precisión tiene un coste. Mathew Jacob et al. (2025) [2] advierten sobre las consecuencias de escalar la inferencia de reranking, señalando que la latencia crece linealmente con la profundidad de recuperación (k). Nuestro estudio cuantifica precisamente este intercambio en un entorno controlado.

1.3 Contribuciones del estudio

A diferencia de trabajos previos que evalúan sobre datasets públicos como MS MARCO o BEIR, este estudio simula un entorno corporativo real utilizando "CorpLeg-300", un corpus denso y técnico. A través de un diseño experimental riguroso, comparamos cuatro arquitecturas:

1. **Naive RAG:** Embeddings base de OpenAI.
2. **Hybrid Search:** Fusión de Vectores + BM25 (RRF).
3. **Few-Shot Fine-Tuning:** Adaptación del modelo base con $N = 100$ pares (simulando escasez).
4. **Two-Stage Retrieval:** Búsqueda Híbrida + Reranking con `bge-reranker-v2-m3`.

Nuestros hallazgos empíricos demuestran que, bajo restricciones de datos, el enfoque de Reranking no solo es superior en métricas de ranking (NDCG), sino que actúa como un estabilizador crítico contra el ruido documental, validando las observaciones de Giovanni Trappolini et al. (2025) [14] sobre la necesidad de métricas alineadas con el comportamiento del LLM final.

2 Trabajos relacionados

La optimización de canalizaciones RAG ha pasado de la recuperación densa pura a arquitecturas más complejas y multietapa. Nuestro trabajo se apoya en avances recientes en recuperación híbrida, eficiencia del reranking y metodologías de evaluación.

2.1 Recuperación híbrida y eficiencia del reranking

El predominio de la recuperación densa se ha cuestionado recientemente en favor de enfoques híbridos que combinan búsqueda semántica vectorial con emparejamiento léxico (BM25). **Rao et al. (2025)**, en *Rethinking Hybrid Retrieval* [1], demuestran que modelos compactos de embeddings combinados con reranking basado en LLM pueden superar significativamente a modelos de embeddings más grandes y monolíticos. Sus hallazgos sugieren que el paradigma "recuperar y luego reordenar" (*retrieve-then-rerank*) desacopla de forma efectiva el *recall* (gestionado por el recuperador) de la precisión (gestionada por el reranker), permitiendo diseños de sistema más eficientes.

Sin embargo, introducir un reranker impone una penalización de latencia. **Jacob et al. (2025)** advierten en *Drowning in Documents* [2] sobre las consecuencias de escalar la inferencia. Observan que aumentar el número de documentos reordenados (k) mejora el *recall* solo hasta cierto punto, tras el cual la latencia aumenta linealmente mientras las ganancias de rendimiento se estancan o incluso se degradan debido a la inclusión de "aciertos fantasma" (*phantom hits*) —documentos irrelevantes que confunden al ranker—. Esto resalta la necesidad de seleccionar cuidadosamente el tamaño del conjunto de candidatos (Top-k) en arquitecturas en dos etapas.

2.2 Desafíos en la adaptación de embeddings

Aunque el ajuste fino es el estándar teórico de oro para la adaptación al dominio, la literatura reciente destaca su inestabilidad en escenarios con pocos recursos. **Li et al. (2024)** exploran esto en *Making Text Embedders Few-Shot Learners* [10], proponiendo el aprendizaje en contexto (*In-Context Learning*, ICL) como alternativa a la actualización de pesos. Señalan que el aprendizaje contrastivo convencional a menudo se queda corto cuando los datos de entrenamiento son ruidosos o escasos, produciendo modelos que memorizan ejemplos de entrenamiento en lugar de aprender características semánticas generalizables. Esto se alinea con las restricciones de "arranque en frío" de nuestro contexto industrial, donde obtener miles de tríos de alta calidad es inviable.

Además, **Sharma (2025)** ofrece una encuesta exhaustiva de arquitecturas RAG, categorizando las mejoras en centradas en recuperación y centradas en generación. Nuestro trabajo se centra en la optimización centrada en la recuperación, validando específicamente la robustez de los *cross-encoders* frente a la fragilidad de bi-encoders ajustados few-shot en dominios especializados.

2.3 Evaluación en la era de los LLMs

Evaluar sistemas RAG en ausencia de *ground truth* etiquetado por humanos es un desafío persistente. **Trappolini et al. (2025)** argumentan en *Redefining Retrieval Evaluation in the Era of LLMs* [14] que métricas tradicionales como nDCG estándar pueden desalinearse con la forma en que los LLMs consumen contexto. Proponen que los propios LLMs pueden servir como evaluadores efectivos ("LLM-as-a-Judge"), una metodología que adoptamos para generar nuestro "Golden Dataset" y validar relevancia. De forma similar, **Zhu et al. (2024)** en *Generative Judge for Evaluating Alignment* refuerzan la fiabilidad de usar modelos fuertes de razonamiento para sintetizar consultas y adjudicar relevancia, habilitando la creación de nuestro *benchmark* sintético *CorpLeg-300*.

2.4 Exigencias de veracidad en el dominio legal

La adopción de RAG en el sector legal no es meramente una cuestión de métricas de ranking, sino de seguridad jurídica. [18] argumenta que en entornos de alto riesgo, los sistemas de IA deben garantizar una "trazabilidad forense" y mitigar las alucinaciones mediante una fundamentación estricta (*Grounding*).

Nuestro estudio demuestra que el uso de *Cross-Encoders* en la segunda etapa no solo mejora el NDCG, sino que actúa como un mecanismo de filtrado crítico para cumplir con estos requisitos de gobernanza. Al descartar documentos irrelevantes que un bi-encoder (incluso afinado) podría recuperar por similitud léxica espuria, el sistema reduce la superficie de riesgo para la generación de alucinaciones por parte del LLM.

3 Metodología

Para validar nuestra hipótesis sobre la superioridad del *Two-Stage Retrieval* frente al *Fine-Tuning* en entornos de baja disponibilidad de datos, diseñamos un marco experimental riguroso que simula las condiciones reales de un sistema RAG corporativo "Cold Start".

3.1 Configuración experimental

Todos los experimentos se ejecutaron en un entorno controlado para garantizar la reproducibilidad y la medición precisa de la latencia. La infraestructura de hardware consistió en una instancia de inferencia equipada con una GPU NVIDIA A10G (24GB VRAM) para la ejecución de los modelos locales (Rerankers y modelos base abiertos) y una CPU de 8 núcleos para las operaciones de búsqueda dispersa (BM25). El software se orquestó utilizando Python 3.10 y el framework *LlamaIndex* para la gestión del flujo de recuperación.

El objetivo central es aislar la variable de la **estrategia de recuperación** manteniendo constantes el corpus y las consultas de evaluación.

3.2 Dataset: CorpLeg-300 y generación de Ground Truth

Dada la naturaleza confidencial de los datos empresariales, construimos **"CorpLeg-300"**, un dataset sintético diseñado para replicar la complejidad lingüística y estructural de la documentación corporativa real.

- **Corpus:** El corpus consta de **300 documentos** anonimizados del dominio financiero y legal (contratos de servicios, informes de auditoría y políticas de cumplimiento interno). Los documentos fueron preprocesados y segmentados (*chunking*) en fragmentos de 512 tokens con un solapamiento (*overlap*) de 50 tokens, resultando en un espacio de búsqueda vectorial de aproximadamente 4,500 pasajes.

- **Generación de Ground Truth sintético:** Siguiendo la metodología propuesta por Jadon et al. en *."Enhancing Domain-Specific Retrieval-Augmented Generation"* (2025) [16] , utilizamos un enfoque asistido por LLM para generar pares Pregunta-Respuesta (Q&A).

  1. Utilizamos **GPT-4o** como generador (Reasoning Model"), instruyéndolo para crear consultas complejas que requieran razonamiento sobre el texto, evitando preguntas triviales de coincidencia de palabras clave.
  2. Generamos un total de **500 pares de consultas y pasajes relevantes** (positive triplets).
  3. **Validación humana:** Para mitigar el ruido inherente a la generación sintética, un experto del dominio validó manualmente una muestra aleatoria del 10 % (50 pares), asegurando la coherencia semántica y la precisión de la relevancia.

- **División de datos:** Para simular el escenario de escasez de datos (*Low-Resource*), dividimos el dataset en:

  - **Training Set (Few-Shot):** 100 pares, utilizados exclusivamente para el entrenamiento del adaptador en el experimento de Fine-Tuning.
  - **Test Set:** 400 pares, utilizados como "Ground Truth"para la evaluación de todas las arquitecturas.

3.3 Modelos y arquitecturas base

Evaluamos cuatro configuraciones arquitectónicas distintas para contrastar el rendimiento entre modelos generalistas, híbridos, adaptados y reordenados.

Baseline (Naive RAG): Utilizamos el modelo **OpenAI text-embedding-3-large** (con dimensiones truncadas a 1024) como único mecanismo de recuperación. Se realiza una búsqueda vectorial estándar (ANN) utilizando similitud del coseno para recuperar los *Top-10* documentos más relevantes. Este enfoque representa el estándar de la industria "out-of-the-box".

Hybrid Search (búsqueda híbrida): Combina la recuperación densa del *Baseline* con una recuperación dispersa basada en palabras clave (**BM25**), implementada sobre *Elasticsearch*. Los resultados de ambos índices se fusionan utilizando el algoritmo **Reciprocal Rank Fusion (RRF)** con un parámetro $k = 60$. Esta configuración busca mitigar las debilidades de los modelos vectoriales ante terminología específica (IDs de productos, cláusulas legales exactas).

Few-Shot Fine-Tuning: Para probar la eficacia de la adaptación de dominio con pocos datos, entrenamos una capa adaptadora lineal sobre el modelo base (text-embedding-3-large). El entrenamiento utilizó la función de pérdida **InfoNCE (Contrastive Loss)** con una temperatura de $\tau = 0{,}05$.

Restricción crítica: Siguiendo la premisa del estudio, el entrenamiento se limitó estrictamente a los 100 pares del Training Set, simulando la escasez de datos etiquetados. Para cada consulta de entrenamiento, se minaron 5 hard negatives utilizando búsqueda BM25 sobre el corpus completo de 4,500 chunks, seleccionando aquellos fragmentos con alta similitud léxica pero baja relevancia semántica según el ground truth. Este proceso de minería se realizó una única vez antes del entrenamiento, evitando sesgos circulares del modelo base.

Two-Stage Retrieval (nuestra propuesta): Esta arquitectura implementa un enfoque de recuperar y refinar":

- *Etapa 1 (Retrieval):* Se utiliza la configuración *Hybrid Search* para recuperar un conjunto amplio de candidatos (*Recall pool*), configurado en **Top-50**.
- *Etapa 2 (Reranking):* Los 50 candidatos son reevaluados por un modelo **Cross-Encoder**, específicamente **bge-reranker-v2-m3**. A diferencia de los *bi-encoders* (embeddings), el *Cross-Encoder* procesa la consulta y el documento simultáneamente, capturando interacciones semánticas profundas. El sistema devuelve el **Top-10** final reordenado.

A diferencia de los bi-encoders, el Cross-Encoder procesa la consulta y el documento simultáneamente. Esta arquitectura permite lo que [15] describe como **"interacción consciente de la topología"** (en el contexto de Topo-RAG), donde el modelo puede atender a relaciones espaciales y lógicas complejas entre tokens distantes sin sufrir la pérdida de información inherente a la linealización en un solo vector. En nuestro corpus *CorpLeg-300*, rico en estructuras condicionales, esta capacidad de interacción tardía es determinante.

3.4 Generación de Hard Negatives

Para el entrenamiento del modelo Few-Shot Fine-Tuning, la calidad de los ejemplos negativos es crítica para evitar el colapso del aprendizaje contrastivo. Dado que el conjunto de entrenamiento se limita a N = 100 pares, implementamos la siguiente estrategia de minería de hard negatives:

1. **Recuperación de candidatos**: Para cada consulta q_i del Training Set, ejecutamos una búsqueda BM25 sobre el corpus completo de 4,500 chunks, recuperando los Top-20 fragmentos con mayor similitud léxica.

2. **Filtrado de verdaderos positivos**: Eliminamos de los candidatos aquellos fragmentos marcados como relevantes en el ground truth sintético.

3. **Selección de hard negatives**: De los candidatos restantes, seleccionamos los 5 fragmentos con mayor puntuación BM25 (similitud léxica alta) pero que semánticamente no responden a la consulta. Estos constituyen "negativos difíciles"porque comparten términos clave con la consulta pero carecen de relevancia contextual.

4. **Ratio final de entrenamiento**: Cada triplete de entrenamiento consiste en consulta, 1 positivo, 5 negativos, resultando en $100 \times 6 = 600$ pares de entrenamiento efectivos para el cálculo de la pérdida InfoNCE.

Esta metodología se alinea con las recomendaciones de Tamber et al. (2025) (11) sobre la importancia de negativos diversos en regímenes de datos escasos, y evita sesgos circulares al no depender del modelo de embeddings que se está entrenando.

3.5 Métricas de evaluación

La evaluación se centra en la relevancia, la capacidad de recuperación y la eficiencia operativa.

- **NDCG@10 (Normalized Discounted Cumulative Gain):** Métrica principal para evaluar la calidad del ranking. Valora no solo si el documento relevante está presente, sino su posición en la lista (se penaliza si aparece en la posición 10 en lugar de la 1).
- **Recall@10 (Hit Rate):** Métrica binaria que mide la proporción de consultas para las cuales el documento correcto ("Golden Passage") aparece dentro de los 10 primeros resultados. Es crucial para garantizar que el LLM reciba el contexto necesario.
- **Latency (P95):** Medimos el tiempo de latencia en el percentil 95 (en milisegundos) para procesar una consulta completa (desde la recepción hasta la entrega de los documentos al LLM). Esto permite evaluar el "coste" temporal de introducir un Reranker frente a la inferencia rápida de un modelo Fine-Tuned.

4 Resultados experimentales

En esta sección presentamos el rendimiento comparativo de las cuatro arquitecturas evaluadas sobre el conjunto de prueba *CorpLeg-300*. Los resultados se analizan en dos dimensiones críticas para la viabilidad industrial: la **precisión de recuperación** (relevancia) y la **eficiencia del sistema** (latencia y coste de implementación).

4.1 Precisión de recuperación

La Tabla 1 resume los resultados de efectividad utilizando las métricas NDCG@10 y Recall@10. Los resultados confirman nuestra hipótesis principal: en escenarios de *Cold Start* (escasez de datos), la arquitectura Two-Stage supera significativamente a las estrategias de adaptación de modelos.

El *Baseline* (Naive RAG) establece un suelo de rendimiento con un NDCG@10 de **0.621**. La incorporación de *Hybrid Search* mejora esta métrica a **0.715**, lo que demuestra que la búsqueda léxica (BM25) sigue siendo crucial para capturar terminología específica del dominio legal que los modelos densos a veces pasan por alto.

El hallazgo más crítico se observa en la fila del *Few-Shot Fine-Tuning*. A pesar de adaptar el modelo de embeddings con 100 pares de entrenamiento de alta calidad, el rendimiento cayó a **0.692**, situándose por debajo de la búsqueda híbrida estándar. Esto indica que el modelo sufrió de **sobreajuste (overfitting)** a los patrones específicos del pequeño conjunto de entrenamiento, perdiendo capacidad de generalización sobre el conjunto de prueba no visto.

Por el contrario, la arquitectura **Two-Stage (Hybrid + Cross-Encoder Reranking)** alcanzó un rendimiento superior, logrando un NDCG@10 de **0.847**. Esto representa una mejora absoluta del $+$**36.3 %** sobre el baseline y un $+$**22.4 %** sobre el modelo Fine-Tuned.

4.2 Análisis de eficiencia y latencia

Si bien la precisión es primordial, la viabilidad de la implementación depende de la latencia. La Tabla 2 muestra el impacto de cada arquitectura en el tiempo de respuesta. Las mediciones se realizaron en GPU NVIDIA A10G con inferencia local (sin latencia de red), promediando 100 ejecuciones por configuración. Como se esperaba, el enfoque Two-Stage introduce una penalización de latencia significativa, aumentando el tiempo de inferencia P95 a 352ms, comparado con los 95ms del Baseline y los 98ms del modelo Fine-Tuned (ambos incluyendo búsqueda vectorial ANN sobre 4,500 vectores).

Este aumento es lineal respecto al número de documentos reordenados (en este caso, 50). Sin embargo, el *coste de implementación* (horas de ingeniería y mantenimiento de pipelines de entrenamiento) es drásticamente menor para el Reranker que para el Fine-Tuning.

4.3 Discusión de resultados

Los datos revelan dos fenómenos clave que validan nuestra estrategia para la Fase 2 del despliegue corporativo:

1. **El fallo del Fine-Tuning en Low-Resource:** El rendimiento inferior del modelo *Fine-Tuned* frente a la búsqueda híbrida se alinea con lo expuesto por *Li et al. (2024)* en "Making Text Embedders Few-Shot Learners"[10]. Con solo 100 ejemplos, el modelo de embeddings tiende a memorizar características superficiales de las consultas de entrenamiento en lugar de aprender una representación semántica robusta del dominio legal. El *contrastive loss* requiere "negativos difíciles"diversos y abundantes para ser efectivo, algo de lo que carece un dataset tan pequeño.

2. **La robustez del Cross-Encoder:** El éxito del *Two-Stage Reranking* se debe a la arquitectura de atención cruzada. Al procesar la consulta y el documento conjuntamente (*full self-attention*), el modelo bge-reranker-v2-m3 puede realizar una inferencia de relevancia mucho más profunda .ºut-of-the-box"sin necesidad de ver datos de entrenamiento específicos del cliente. Aunque la latencia aumenta a $\tilde{3}$50ms, este retardo es imperceptible para el usuario final en aplicaciones de búsqueda documental (donde la tolerancia suele ser de hasta 1-2 segundos) y es un precio aceptable a cambio de un aumento del **15 % en Recall** respecto al Fine-Tuning.

4.4 Análisis de sensibilidad: el impacto de la profundidad de Reranking (k)

Una decisión crítica en arquitecturas de dos etapas es determinar el tamaño de la ventana de reranking (k). Un k bajo arriesga perder documentos relevantes que el recuperador inicial situó en posiciones lejanas (falsos negativos), mientras que un k alto dispara la latencia.

Realizamos un barrido de parámetros para $k \in \{10, 30, 50, 100\}$ sobre la arquitectura Two-Stage. La tabla 3) ilustra este comportamiento.

Observamos que aumentar k de 50 a 100 solo mejora el Recall en un 0.6 %, pero duplica la latencia (de 352ms a 680ms). Esto valida nuestra elección de $k = 50$ como el punto de equilibrio óptimo para aplicaciones interactivas.

4.5 Estudio de caso cualitativo: el problema de la "alucinación semántica"

Para ilustrar la diferencia cualitativa entre los modelos, analizamos una consulta difícil del conjunto de test:

> ***Consulta:*** *"¿Cuáles son las condiciones de penalización por rescisión anticipada en el contrato con el proveedor AlphaCorp?"*

Cuadro 1: **Comparativa de precisión de recuperación en el conjunto de prueba CorpLeg-300 ($N = 400$).** Se observa cómo el Fine-Tuning con pocos datos (Fila 3) rinde peor que la búsqueda híbrida, mientras que el Reranking (Fila 4) obtiene el mejor desempeño.

Arquitectura	NDCG@10	Recall@10	Mejora Rel.
1. Baseline (Naive RAG)	0.621	0.785	–
2. Hybrid Search (Dense + BM25)	0.715	0.860	+15.1 %
3. Few-Shot Fine-Tuning ($N = 100$)	0.692	0.815	+11.4 %
4. Two-Stage (Hybrid + Rerank)	**0.847**	**0.965**	**+36.3 %**

Cuadro 2: **Análisis de eficiencia (latencia vs. coste operativo).** Se compara el tiempo de respuesta P95 y la carga de ingeniería. Nótese que aunque la arquitectura Two-Stage (Fila 4) introduce latencia, evita los altos costes y la complejidad de mantenimiento asociados al entrenamiento (Fila 3).

Arquitectura	Latencia P95	Coste Implementación	¿Requiere entreno?
1. Baseline (Naive RAG)	95 ms	Bajo	No
2. Hybrid Search	118 ms	Medio	No
3. Few-Shot Fine-Tuning	98 ms	**Alto**	**Sí**
4. Two-Stage (Hybrid + Rerank)	352 ms	Medio	No

- **Naive RAG / Fine-Tuning:** Recuperaron documentos relacionados con "Inicio de contrato" y "Bonificaciones de rendimiento". El modelo Fine-Tuned, sobreajustado a palabras como "AlphaCorp" y "contrato", ignoró la sutileza semántica de "rescisión anticipada" (penalización negativa) y priorizó contextos positivos frecuentes en el entrenamiento.

- **Two-Stage Reranking:** El recuperador híbrido trajo el documento correcto en la posición #8 (rescatado por BM25 gracias a la palabra "rescisión"). El Cross-Encoder, al analizar la consulta completa, detectó la relación lógica entre "penalización" y "rescisión", elevando el documento correcto a la **Posición 1**.

Este caso demuestra que el Reranker actúa como un mecanismo de corrección de atención, vital cuando la densidad de información en los documentos es alta.

4.6 Análisis del Umbral de rentabilidad de datos (Data Break-even Point)

Una interrogante crítica para la planificación estratégica de IA es determinar el volumen de datos etiquetados necesario para que el rendimiento del *Fine-Tuning* supere al enfoque *Zero-Shot Reranking*. Para cuantificar este umbral, extendimos el experimento entrenando variantes del modelo bi-encoder con conjuntos de datos incrementales: $N \in \{100, 300, 500, 1000, 2000\}$.

La Figura **??** (resumida en la Tabla 4) ilustra las curvas de aprendizaje comparativas.

4.6.1 Interpretación de la curva de aprendizaje

Los resultados indican que el *Data Break-even Point* se sitúa aproximadamente en $^{**}N \approx 1,000$ pares etiquetados de alta calidad**. Cada punto representa la media de 3 ejecuciones de entrenamiento con diferentes semillas aleatorias, con desviaciones estándar inferiores a ± 0.015 en NDCG@10. La convergencia estadística se alcanza cuando el intervalo de confianza del 95 % del modelo Fine-Tuned supera consistentemente el umbral del Reranker (0.847).

- **Zona de ineficiencia ($N < 1,000$):** En esta zona, el esfuerzo de etiquetado y computación para el Fine-Tuning produce un ROI negativo, ya que el Reranker *off-the-shelf* sigue siendo superior.

- **Zona de Ventaja ($N > 1,000$):** Al superar las mil muestras, el bi-encoder comienza a interiorizar la semántica profunda del dominio, superando finalmente al Reranker.

Este hallazgo proporciona una hoja de ruta clara: el Reranking es la solución óptima para las fases iniciales e intermedias del ciclo de vida del producto. El Fine-Tuning solo debe considerarse como una optimización de Fase 3, una vez que la telemetría del sistema haya permitido recolectar orgánicamente un dataset superior al millar de ejemplos verificados.

5 Discusión

Los resultados obtenidos en *CorpLeg-300* desafían la intuición convencional de que el entrenamiento específico del dominio (Fine-Tuning) es siempre la mejor ruta para mejorar la recuperación. A continuación, analizamos las causas subyacentes de este fenómeno.

Cuadro 3: **Sensibilidad de Recall vs. Latencia según la profundidad k.** Se observa un punto de inflexión en $k = 50$, donde el incremento marginal en Recall (0.16 puntos porcentuales por cada 100ms adicionales) representa el óptimo coste-beneficio. Para k = 100, la ganancia es apenas 0.02 pp/100ms (rendimientos decrecientes).

Top-k Reranked	**Recall@10**	**Latencia (ms)**	**Δ Recall / Δ Time**
10	0.885	95 ms	-
30	0.942	210 ms	0.50 pp / 100ms
50	**0.965**	**352 ms**	0.16 pp / 100ms
100	0.971	680 ms	0.02 pp / 100ms

Cuadro 4: **Evolución del NDCG@10 según el tamaño del dataset de entrenamiento.** El Fine-Tuning muestra una curva de mejora logarítmica, pero no logra cruzar la línea base del Reranking (0.847) hasta superar los 1,000 ejemplos de alta calidad.

Muestras de Entrenamiento (N)	**Fine-Tuning NDCG**	**Two-Stage NDCG**	**Diferencial**
100 (Cold Start)	0.692	0.847	-18.3 %
300	0.745	0.847	-12.0 %
500	0.798	0.847	-5.8 %
1,000	**0.851**	**0.847**	**+0.4 % (Umbral)**
2,000	0.884	0.847	+4.3 %

5.1 La fragilidad del Contrastive Learning con datos escasos

5.2 La fragilidad del Contrastive Learning y la paradoja de la dilución

El rendimiento inferior de la arquitectura de Fine-Tuning (NDCG 0.692) frente a la búsqueda híbrida (0.715) no es solo una consecuencia de la escasez de datos ($N = 100$), sino de una limitación arquitectónica fundamental de los modelos densos.

Como demuestran **Tamber et al. (2025)** en *"Conventional Contrastive Learning Often Falls Short"* [11], optimizar *bi-encoders* requiere no solo pares positivos de alta calidad, sino una minería robusta de "negativos difíciles"(hard negatives). Con un dataset de entrenamiento de $N = 100$, **el modelo carece de la diversidad de señales negativas necesarias** para aprender fronteras de decisión generalizables. En lugar de aprender la semántica del dominio legal, el modelo tiende a "memorizar"la sintaxis superficial de las pocas consultas de entrenamiento, lo que resulta en una degradación del rendimiento cuando se enfrenta a consultas no vistas en el conjunto de prueba (Overfitting).

Sin embargo, incluso si hubiéramos tenido más datos, el modelo seguiría limitado por lo que se define como la **"Paradoja de la dilución vectorial"** [12]. En documentos legales densos, intentar comprimir cláusulas complejas y matices contextuales en un único vector de dimensión fija ($d = 1024$) fuerza una redistribución de los pesos de atención que diluye la señal semántica local. El Fine-Tuning, en este contexto, simplemente ajusta el espacio vectorial hacia patrones léxicos superficiales (sobreajuste) sin resolver el problema de compresión con pérdida.

5.3 La superioridad del Cross-Encoder como "juez"

La arquitectura *Two-Stage* dominó el benchmark debido a la naturaleza intrínseca de los *Cross-Encoders*. Mientras que los *bi-encoders* (usados en el baseline y fine-tuning) comprimen el documento en un vector fijo antes de ver la consulta, el *Cross-Encoder* recibe ambos inputs simultáneamente, permitiendo que el mecanismo de *Self-Attention* evalúe la interacción palabra por palabra. Esto se alinea con los hallazgos de **Schlatt et al. (2025)** en *Rank-DistiLLM"* [5], quienes argumentan que los *Cross-Encoders* actúan como jueces de relevancia más efectivos que los modelos generativos o vectoriales simples, **cerrando la brecha de efectividad sin necesidad de reentrenamiento**. En nuestro contexto, el modelo bge-reranker fue capaz de discernir matices legales sutiles .ºut-of-the-box"que el modelo de embeddings, incluso después del fine-tuning, pasó por alto.

5.4 El Trade-off de latencia en entornos corporativos

El incremento en latencia (de $\tilde{9}$5ms a $\tilde{3}$50ms) es el principal contraargumento para el uso de Rerankers. Sin embargo, tal como discuten **Jacob et al. (2025)** en *"Drowning in Documents"* [2], el impacto de escalar la inferencia depende del caso de uso. En aplicaciones de búsqueda web en tiempo real, 300ms puede ser prohibitivo. No obstante, en el dominio legal y financiero de esta empresa, donde el usuario espera una respuesta analítica precisa sobre documentos complejos, este retraso es imperceptible y totalmente aceptable. La "tasa de frustración"del usuario por recibir un documento incorrecto (falso positivo) es infini-

tamente mayor que la causada por una espera adicional de 0.3 segundos.

6 Implicaciones prácticas para la industria

Basándonos en la evidencia empírica de este estudio, proponemos las siguientes directrices operativas para los equipos de ingeniería de ML en la empresa:

1. **Moratoria sobre el Fine-Tuning**: No se debe invertir esfuerzo en Embedding Fine-Tuning hasta que el sistema haya recolectado un mínimo de 1,000-1,500 pares de preguntas y respuestas verificadas por usuarios (feedback loops explícitos), umbral en el que nuestros experimentos demuestran que el rendimiento del modelo ajustado supera consistentemente al Reranker (ver Tabla 4). Intentar entrenar con menos datos es computacionalmente ineficiente y riesgoso para la calidad.

2. **Adopción inmediata de Two-Stage Retrieval:** La arquitectura de referencia para la Fase 2 debe ser **Hybrid Search (Top-50) + Cross-Encoder Reranking (Top-10)**. Esto maximiza el *Recall* inicial mediante palabras clave y maximiza la *Precision* final mediante comprensión semántica profunda.

3. **Estrategia de datos sintéticos:** En lugar de etiquetar manualmente, se recomienda utilizar el presupuesto para refinar el pipeline de generación de datos sintéticos (como el usado para crear *CorpLeg-300*), lo cual permitirá, a largo plazo, acumular el volumen de datos necesario para considerar técnicas de *Distillation* en el futuro.

7 Conclusión

Este estudio abordó el problema del Çold Start.[en] sistemas RAG corporativos. Demostramos que, en ausencia de grandes volúmenes de datos de entrenamiento, la sofisticación arquitectónica (Two-Stage Retrieval) supera a la adaptación paramétrica (Fine-Tuning).

Nuestro modelo de Reranking logró un incremento del **22.4 % en NDCG@10** sobre el intento de fine-tuning, validando que la capacidad de razonamiento cero-shot de los Cross-Encoders modernos es la herramienta más eficiente para garantizar la relevancia en dominios corporativos complejos.

Concluimos por tanto que el **Reranking no es solo una mejora opcional, sino un componente crítico** para la viabilidad de sistemas RAG en producción.

Referencias

[1] Arjun Rao, Hanieh Alipour, and Nick Pendar. Rethinking Hybrid Retrieval: When Small Embeddings and LLM Re-ranking Beat Bigger Models. *arXiv preprint arXiv:2506.00049*, 2025. https://arxiv.org/abs/2506.00049

[2] Mathew Jacob, Erik Lindgren, Matei Zaharia, Michael Carbin, Omar Khattab, and Andrew Drozdov. Drowning in Documents: Consequences of Scaling Reranker Inference. *arXiv preprint arXiv:2411.11767*, 2025. https://arxiv.org/abs/2411.11767

[3] Gabriel de Souza P. Moreira, Ronay Ak, Benedikt Schifferer, Mengyao Xu, Radek Osmulski, and Even Oldridge. Enhancing Q&A Text Retrieval with Ranking Models: Benchmarking, fine-tuning and deploying Rerankers for RAG. *arXiv preprint arXiv:2409.07691*, 2024. https://arxiv.org/abs/2409.07691

[4] Jan Strich, Adeline Scharfenberg, Chris Biemann, and Martin Semmann. EncouRAGe: Evaluating RAG Local, Fast, and Reliable. *arXiv preprint arXiv:2511.04696*, 2025. https://arxiv.org/abs/2511.04696

[5] Ferdinand Schlatt, Maik Fröbe, Harrisen Scells, Shengyao Zhuang, Bevan Koopman, Guido Zuccon, Benno Stein, Martin Potthast, and Matthias Hagen. Rank-DistiLLM: Closing the Effectiveness Gap Between Cross-Encoders and LLMs for Passage Re-Ranking. *arXiv preprint arXiv:2405.07920*, 2025. https://arxiv.org/abs/2405.07920

[6] Zhichao Xu, Zhiqi Huang, Shengyao Zhuang, and Vivek Srikumar. Distillation versus Contrastive Learning: How to Train Your Rerankers. *arXiv preprint arXiv:2507.08336*, 2025. https://arxiv.org/abs/2507.08336

[7] Francesca Pezzuti, Sean MacAvaney, and Nicola Tonellotto. Exploring the Effectiveness of Multi-stage Fine-tuning for Cross-encoder Re-rankers. *arXiv preprint arXiv:2503.22672*, 2025. https://arxiv.org/abs/2503.22672

[8] Feng Wang, Yuqing Li, and Han Xiao. jina-reranker-v3: Last but Not Late Interaction for Listwise Document Reranking. *arXiv preprint arXiv:2509.25085*, 2025. https://arxiv.org/abs/2509.25085

[9] Kaili Huang, Thejas Venkatesh, Uma Dingankar, Antonio Mallia, Daniel Campos, Jian Jiao, Christopher Potts, Matei Zaharia, Kwabena Boahen, Omar Khattab, Saarthak Sarup, and Keshav Santhanam. ColBERT-serve: Efficient Multi-Stage Memory-Mapped Scoring. *arXiv preprint arXiv:2504.14903*, 2025. https://arxiv.org/abs/2504.14903

[10] Chaofan Li, MingHao Qin, Shitao Xiao, Jianlyu Chen, Kun Luo, Yingxia Shao, Defu Lian, and Zheng Liu. Making Text Embedders Few-Shot Learners. *arXiv preprint arXiv:2409.15700*, 2024. https://arxiv.org/abs/2409.15700

[11] Manveer Singh Tamber, Suleman Kazi, Vivek Sourabh, and Jimmy Lin. Conventional Contrastive Learning Often Falls Short: Improving Dense Retrieval with Cross-Encoder Listwise Distillation and Synthetic Data. *arXiv preprint arXiv:2505.19274*, 2025. https://arxiv.org/abs/2505.19274

[12] Alex Dantart. More Context Is Not Better: The Vector Dilution Paradox in Enterprise RAG. *arXiv preprint arXiv:2601.08851*, 2025. https://arxiv.org/abs/2601.08851

[13] Linh The Nguyen, Chi Tran, Dung Ngoc Nguyen, Van-Cuong Pham, Hoang Ngo, and Dat Quoc Nguyen. AccurateRAG: A Framework for Building Accurate Retrieval-Augmented Question-Answering Applications. *arXiv preprint arXiv:2510.02243*, 2025. https://arxiv.org/abs/2510.02243

[14] Giovanni Trappolini, Florin Cuconasu, Simone Filice, Yoelle Maarek, and Fabrizio Silvestri. Redefining Retrieval Evaluation in the Era of LLMs. *arXiv preprint arXiv:2510.21440*, 2025. https://arxiv.org/abs/2510.21440

[15] Alex Dantart. Topo-RAG: Recuperación Consciente de la Topología para Documentos Híbridos Texto-Tabla. *arXiv preprint arXiv:2601.00200*, 2026. https://arxiv.org/abs/2601.00200

[16] Aryan Jadon, Avinash Patil, and Shashank Kumar. Enhancing Domain-Specific Retrieval-Augmented Generation: Synthetic Data Generation and Evaluation using Reasoning Models. *arXiv preprint arXiv:2502.15854*, 2025. https://arxiv.org/abs/2502.15854

[17] Chaitanya Sharma. Retrieval-Augmented Generation: A Comprehensive Survey of Architectures, Enhancements, and Robustness Frontiers. *arXiv preprint arXiv:2506.00054*, 2025. https://arxiv.org/abs/2506.00054

[18] Alex Dantart. Gobernanza y trazabilidad .ª prueba de AI Act"para casos de uso legales: un marco técnico-jurídico. *arXiv preprint arXiv:2510.09467*, 2025. https://arxiv.org/abs/2510.09467