mdt.

Ex Datis, Claritas

RAG: cómo conseguir que la IA Generativa sea aún más lista

La RAG (Retrieval-Augmented Generation) hace más fiable la IA generativa. Lo expliqué en mi sección de tecnología de «No son horas», en Onda Cero.

Manuel Delgado Tenorio · 17 de octubre de 2024

Publicado originalmente en www.manueldelgado.com/blog/rag-como-conseguir-que-la-ia-generativa



En mi sección de Tecnología del programa "No son horas" de Onda Cero, traté la semana pasada sobre la RAG (Retrieval-Augmented Generation) y cómo hace más fiable la IA Generativa.

Si quieres escuchar la sección de la semana pasada, puedes hacerlo en la web de Onda Cero. Avanza hasta casi el final, al punto 02:32:45.

La RAG (*Retrieval-Augmented Generation*, o generación aumentada por recuperación) es una técnica que, antes de que el modelo responda, busca en los documentos de tu organización la información relevante y se la entrega como contexto. Así, la respuesta se apoya en tus propios datos y no solo en lo que el modelo aprendió durante su entrenamiento.

En la actualidad, las empresas buscan implementar soluciones de inteligencia artificial generativa que ofrezcan respuestas rápidas, precisas y alineadas con sus necesidades internas.

Además, en el punto de madurez que nos encontramos, ya (casi) todas las organizaciones son conscientes de las limitaciones y los riesgos que se derivan del uso de modelos lingüísticos de gran tamaño (LLM) genéricos.

¿Para qué sirve RAG?

RAG es la técnica que nos evita los enormes costes y complejidad de entrenar un LLM desde cero para que comprenda todos los matices y conocimientos específicos de nuestra organización.

Debemos tener en cuenta que los grandes LLM se basan fundamentalmente en textos disponibles públicamente en Internet, por lo que no saben nada de la documentación o los datos que tu empresa usa internamente.

> *Puedes pedir a ChatGPT que te dé una explicación genérica de cómo se pasan las notas de gastos en la mayoría de las empresas, pero no puedes esperar que te diga cómo debes hacerlo en tu empresa, porque no lo sabe.*

Mediante RAG, por ejemplo, una empresa de comercio electrónico puede conseguir que su chatbot de atención al cliente consulte instantáneamente el inventario de productos y ofrezca información precisa sobre la disponibilidad o características de los artículos. En sectores como el legal o el financiero, RAG permite a los sistemas de IA referenciar normativas o políticas internas actualizadas para garantizar que las respuestas estén alineadas con las regulaciones vigentes.

¿Y cómo funciona?

A efectos prácticos, RAG es como añadir tu catálogo de productos o las políticas de la empresa al LLM. El “truco” está en que esa documentación adicional no se incorpora realmente al modelo, sino que se anexa automáticamente a las instrucciones/preguntas (*prompts*) lanzadas por los usuarios, con lo que el LLM la usa como contexto para generar la respuesta.

Así, el LLM aporta la capacidad general para interpretar las preguntas y producir las respuestas, y esa capacidad general se ve mejorada o aumentada (de ahí su nombre) mediante el contexto proporcionado mediante RAG.

La información adicional se mantiene separada del modelo, almacenada en bases de datos, en un formato optimizado para su uso con el LLM (si te interesa saber más, busca "*bases de datos vectoriales*"). A medida que esa información cambia (documentos que se actualizan, nuevos datos que se generan, políticas que quedan obsoletas...), lo que se actualiza es la información guardada en esas bases de datos, no el modelo de IA generativa.

Alternativas a RAG

Recordemos que entrenar desde cero un LLM puede llegar a costar decenas o incluso cientos de millones de euros. Por tanto, crear su propio LLM carece de sentido para la inmensa mayoría de las organizaciones.

Sin embargo, algo que muchas organizaciones sí pueden plantearse es re-entrenar (*fine-tune*) un modelo existente. Aunque depende del grado de ambición de la iniciativa, el coste de ese *fine-tuning* puede ser muy moderado y, con ello, conseguimos un modelo personalizado, listo para usarse como base de múltiples otros proyectos y capaz de alcanzar un buen rendimiento.

Eso sí, el *fine-tuning* de modelos tiene dos desventajas principales:

- **Requiere mayores conocimientos de creación de LLM que el RAG,** incluido un nivel mayor de exigencia para verificar que el modelo resultante no ha incurrido en pérdidas de entrenamiento o validación respecto al modelo base.
- Incluso con LLM "pequeños", **el tiempo de re-entrenamiento suele ser elevado.** En los casos más pequeños, se alargará durante horas. En proyectos ambiciosos, el tiempo se puede reducir con mayores o mejores recursos de computación, pero eso eleva el coste. Todo esto descarta esta técnica para aquellos casos de uso en los que la documentación con la que queremos mejorar el modelo es numerosa y se actualiza con mucha frecuencia.

Para la mayoría de las situaciones, RAG presenta hoy día el mejor equilibrio entre precisión/relevancia de las respuestas, rendimiento y costes.

PARA SEGUIR LEYENDO
Sobre los riesgos que la RAG no resuelve, Los LLM son inseguros por naturaleza. Y todo lo que he escrito sobre el tema, en IA generativa.



Sobre el autor

Manuel Delgado Tenorio

Manuel Delgado Tenorio es asesor independiente de estrategia de datos e inteligencia artificial, profesor e investigador. Cofundó y vendió una *startup* de inteligencia artificial, fue socio de la práctica de Data & Analytics para servicios financieros en EY y responsable de Data Analytics para el sur de Europa en AWS. Es profesor asociado en IE University, donde imparte la primera asignatura de grado sobre Customer Lifetime Value en España, investiga sobre el CLV en su doctorado en la Universidad Complutense de Madrid y es autor del libro "Customer Lifetime Value (CLV): No todos tus clientes valen lo mismo".

manueldelgado.com · LinkedIn · X

Recibe cada artículo en tu correo con la newsletter Ex Datis, Claritas: manueldelgado.com/newsletter/suscribete