2025 | 16·20
GIJÓN | JUNIO
9º CONGRESO FORESTAL ESPAÑOL

9CFE-1895

Actas del Noveno Congreso Forestal Español
Edita: **Sociedad Española de Ciencias Forestales. 2025**.
ISBN: **978-84-941695-7-1**

Organiza

Ejemplos de aplicación de los modelos del lenguaje (LLM) en gestión forestal: gestión de la documentación y creación de herramientas de procesado

BENGOA, J.

Junta de Castilla y León. Dirección General de Patrimonio Natural y Política Forestal.

Resumen

Este trabajo presenta algunas aplicaciones de los modelos del lenguaje (LLM), que pueden facilitar tareas y ahorrar tiempo en nuestro día a día. Una de las aplicaciones, a menudo subestimada, es el uso de técnicas de búsqueda semántica y recuperación avanzada de información (RAG) para acceder de manera eficiente a nuestra propia documentación, la cual también puede, a su vez, enriquecerse mediante estos modelos.

Además, la combinación de LLM con asistentes virtuales convierte a estos modelos en herramientas que van más allá del procesamiento de lenguaje natural, permitiendo abordar tareas complejas con datos diversos, incluida la información espacial. Por ejemplo, es posible crear asistentes que faciliten la interacción con capas en QGIS o el procesado de datos Lidar.

Si bien los LLM suelen utilizarse a través de interfaces web y apps móviles (como GPT, Claude o Gemini), su verdadero potencial se despliega al trabajar con sus API. Además, el uso en local de modelos como Llama 3.1, Mistral 7B o phi 3.1 también abre interesantes posibilidades.

A medida que estas herramientas se desarrollan, su uso se vuelve más sencillo y útil, como lo demuestran las esperadas versiones multimodales de GPT-4 y Gemini, o las menos esperadas novedades en razonamiento avanzado de OpenAI-o1.

Palabras clave

Inteligencia Artificial, modelos del lenguaje, LLM, búsqueda semántica, RAG, assistants.

1. Introducción

En los últimos años, los modelos de lenguaje (LLM) han revolucionado la forma en que interactuamos con la tecnología, facilitando tareas cotidianas y profesionales, como es la propia elaboración de esta comunicación. Si, he usado LLMs para que me ayuden con esta comunicación y eso no me ha impedido dar al texto un enfoque personal, quizás excesivo, a conciencia de que es poco habitual en este tipo de trabajos.

Con el lanzamiento de chatGPT, el interés por estos modelos ha pasado a primer

plano. Si bien ha sido OpenAI la que ha protagonizado buena parte de los avances en este campo, eso no ha impedido que la competencia haya lanzado con éxito sus propios productos (Gemini, Claude, Llama, Mistral, etc.). Los avances han venido de la mano de un mejor comportamiento de los modelos, de una mejor experiencia de usuario, de la multimodalidad (texto, imagen y voz/sonido) y del razonamiento avanzado.

El cambio reciente más significativo ha sido el relacionado con el razonamiento (dedicando más tiempo de computación antes de generar salidas) y se ha presentado como un paso significativo hacia la AGI. Más allá de la trascendencia de los avances en los modelos del lenguaje, hay algo que es a la vez una virtud y un problema en este campo: todo ocurre a un ritmo vertiginoso y lo que un día es una novedad, al mes puede quedar obsoleto. Sirvan de ejemplo los últimos lanzamientos de OpenAI.

Esta empresa lanzó un avance del modelo o1 ("o1-preview" y "o1-mini") el 12 de septiembre de 2024, tres días antes de que se cerrara el plazo para presentar los resúmenes a este congreso lo que me permitió citarlo en el resumen. En diciembre, esta empresa cambió su estilo de comunicación y en su particular calendario de adviento lanzó 12 eventos casi seguidos (los "12 días de Open AI") para anunciar novedades y productos. El primer día (6 de diciembre), Sam Altman y sus compañeros anunciaban la versión oficial de o1. Pues bien, el doceavo día (20 de diciembre), el mismo Sam con otros dos colegas de Open AI presentó o3: la revolución de los LLM. Si parpadeas, te lo pierdes: o1, con 2 semanas de vida, ya era historia.

Este texto está escrito el 17 de enero de 2025 y, previsiblemente, algunas novedades que cita estarán caducas bastante antes de la fecha del congreso (junio de 2025).

Dejando aparte la estrategia de comunicación de Open AI, que no entiendo, está claro que no podemos seguir el ritmo que marcan las tecnológicas en su carrera por estar en primera línea. Si es conveniente saber qué está ocurriendo en este campo, pero no podemos adaptar nuestra forma de trabajar con cada novedad que se publica.

Por fortuna, no son las novedades ni un análisis de la evolución de los LLM lo que quiero contar en esta comunicación, sino que quiero hablar de algo que les acompaña, habitualmente en segundo plano, más dirigido a la comunidad de desarrolladores, y son diversas herramientas que nos permiten sacar partido de los LLMs, más allá del uso habitual en forma de chat en el navegador del PC o en la app del móvil.

2. Antecedentes

En los últimos años he vivido varios cambios en mi relación con la inteligencia artificial. Empezamos hablando de machine learning; era la época de scipy y scikit-learn, y de SVM, KNN y DT para clasificar datos. El panorama cambió notablemente cuando algunas tecnológicas lanzaron sus librerías: Google sacó tensorflow en 2015 y Meta lanzó pytorch en 2016. Estas herramientas "democratizaban" la IA al hacerla más accesible, sobre todo con la llegada de tensorflow 2 (2019). Era la época de "todo en abierto" en IA y la publicación de muchos modelos, incluso con código, lo facilitó todo: era posible construir, adaptar y entrenar modelos con los últimos avances de la comunidad científica y usarlos en nuestras modestas aplicaciones prácticas.

La llegada de los transformers en 2017 cambió bastantes cosas. Esta aportación del equipo de Google Research permitió a OpenAI hincar el diente a una tarea que se resistía: el procesamiento del lenguaje natural (NLP). Pocos años después nacieron los grandes modelos del lenguaje (LLM). En este campo, no hay nada que me haya impresionado más que las primeras experiencias con GPT-3: yo diría que el gran cambio ocurrió en 2020, con esta versión de GPT. El lanzamiento de chat-GPT abrió este campo al gran público el 30 de noviembre de 2022 y se desató la carrera por ir ocupando los nichos de mercado que se vislumbraban en el horizonte.

Pero este salto cualitativo en la potencialidad de la IA, vino acompañado, al menos en mi caso, de un cambio en la forma de interaccionar con ella: cada vez era menos "desarrollador" y más "usuario a secas". No es tanto porque no hubiera recursos y herramientas; como por una mezcla de razones. De hecho, las dos librerías ya clásicas de IA (tensorflow y pytorch) acabaron incorporando los transformers, como era de esperar, y además se comprobó que estos modelos funcionaban no solo con series temporales, para los cuales estaban pensados sino, también, con imágenes (sorprendente).

Desarrollar da más autonomía, en la medida en que podemos crear o adaptar herramientas para nuestras necesidades. En cambio, ser "usuarios a secas" nos deja más a merced de las tecnológicas, convirtiéndonos más en producto de consumo y consumidores de productos. En esta comunicación intento mostrar que es posible revertir un poco esta tendencia en nuestra interacción con la IA y, en concreto con los LLM: podemos sacarles partido más allá de nuestra interacción en forma de chat, de buscador o de sabelotodo.

3. El verdadero potencial del NLP

El título es un poco pretencioso, pero tiene como objeto llamar la atención sobre una distinción que puede parecer sutil, y no lo es. Los LLM se han entrenado con ingentes cantidades de información y tienen muy altas capacidades no solo en lo que se refiere a la información que conocen, sino también a su capacidad para interaccionar con la documentación que les proporcionemos y con nosotros mismos usando el lenguaje natural. Y es esta función de nexo la que quiero poner

en valor en esta comunicación.

Habitualmente sacamos partido de los amplios conocimientos de los LLM (o, recientemente, su capacidad para buscarlos en internet), pero le damos menos importancia a lo más llamativo para mí, que es su dominio del lenguaje natural y las posibilidades que abre esta capacidad como herramienta para interaccionar.

Hace más de dos décadas que tenemos toda la información que busquemos a golpe de click y son muchas las herramientas que nos facilitan la búsqueda y consulta, empezado por el propio buscador de Google que revolucionó esta tarea y terminando por los foros, tutoriales, wikis, algunas redes sociales, repositorios documentales (ArXiv, Internet Archive, etc.), stackoverflow, github, researchgate, IDEs, etc.

Con frecuencia hemos usado el lenguaje natural para las búsquedas, pero sólo con la llegada de los LLM podemos decir que la interacción responde verdaderamente al lenguaje natural y, además, se le saca más partido a la búsqueda semántica y la IA es capaz de combinar la información de varias fuentes.

Otra iniciativa relacionada con esta función nexo de los LLM es la de usar el lenguaje natural para otras interacciones con el mundo digital (era lo que pretendía Alexa). Esta aspiración siempre ha estado ahí, pero la tecnología no estaba madura para que funcionara suficientemente bien. Ahora sí está madura y eso abre un campo para crear herramientas que aprovechen ese potencial.

4. Objetivos

Esta comunicación tiene como objetivo mostrar algunas herramientas que permiten sacar partido a los modelos de lenguaje (LLM) en el campo de la administración y la gestión forestal, más allá de la interacción habitual en forma de chat.

5. Ámbitos y herramientas

Está claro que la IA acabará integrándose en todos los ámbitos de nuestra vida laboral y personal. Y es conveniente trabajar desde ya en ello, que tengamos la iniciativa de subirnos al carro no tanto porque lo que hagamos ahora vaya a ser definitivo, como para ganar una experiencia y unos conocimientos que mejoren nuestra relación futura con la IA. Esta reflexión, que está hecha para la IA en general, vale igualmente para los LLM en los que me centro en esta comunicación.

Hay ámbitos en los que el desembarco la IA parece más previsible y lo primero que viene a nuestra mente son las tareas administrativas más rutinarias. Sin embargo, no hay que confundir la aportación de la IA con la necesaria digitalización de muchos ámbitos de nuestro trabajo y, sobre todo, la integración y optimización de esa digitalización. Yo creo que queda bastante por recorrer en digitalización y lo digo con la perspectiva de más de 20 años trabajando en una administración y la comprobación de que la digitalización de la información y su gestión, que ya se planteaba hace más de dos décadas, no ha alcanzado, ni de lejos, el desarrollo que debiera.

Centrándome en el campo que más conozco, el de la cartografía, el hecho es que hemos alcanzado una buena implantación de Qgis entre nuestros técnicos, toda la cartografía es digital, pero eso no significa que hayamos alcanzado una verdadera digitalización de la gestión de la cartografía, porque los datos y los procedimientos no están armonizados, al menos en el ámbito de la gestión forestal. Las buenas intenciones de la directiva INSPIRE, que emanaban de los ideólogos de Bruselas, es posible que responda a las necesidades de las instituciones europeas, pero queda bastante lejos de la realidad y complejidad de la gestión del territorio.

Este inciso no tiene otro objetivo que el de diferenciar entre la digitalización de la información y la incorporación de la IA. Ésta última requiere aquella, pero no es lo mismo. Si hay algo necesario e importante en este ámbito, es una adecuada digitalización; de hecho, la IA es un añadido que, posiblemente, encuentre más cabida en facilitar pequeñas tareas del día a día que en una optimización del funcionamiento de la Administración. Un poco lo contrario de lo que flota en el imaginario colectivo. Bueno, creo que la realidad va a tener un poco de cada.

Hecha esa distinción, he de decir que la primera motivación para esta comunicación fue la utilización de los LLM en el ámbito de la gestión de información con base cartográfica, en concreto, dentro del proyecto dasolidar que estamos implementando en la Junta de Castilla y León. Sin embargo, las reflexiones y la correspondiente organización de las ideas que requiere escribir una comunicación me han llevado a plantear el texto de una forma un poco más abierta.

Las herramientas que quiero mostrar en esta comunicación son de dos tipos:

- Herramientas de búsqueda de información en documentación propia (RAG, búsqueda semántica, asistente de file search).
- La llamada a funciones (function calling), que permite ejecutar código en nuestros PC como respuesta a instrucciones dadas en lenguaje natural. La idea es que podamos ejecutar algo que sabemos que hace nuestra aplicación, pero que no nos acordamos exactamente cómo se hace. Pasa de vez en cuando, ¿no?

Si tuviera que hablar del uso de la IA en mi trabajo, no serían estas dos herramientas las que traería, porque hay otras tareas relacionadas con la gestión del medio natural en las que está más que probada la utilidad de la IA y en las que

venimos trabajando desde hace años. Pero prefiero dejarlas a un lado para no perder el foco. Esas “otras tareas” son las relacionadas con el procesado de datos alfanuméricos y geográficos: sacar partido a los datos para identificar patrones y obtener sistemas expertos, interpretar y clasificar datos geoespaciales (teledetección, Lidar, ortofotos), etc. Y hay otros ámbitos que podríamos explorar, por ejemplo, usar la IA para integrar diferentes fuentes de información geográfica, mejorando nuestra capacidad de análisis.

6. Consulta de documentación propia con lenguaje natural.

Sea cual sea nuestra área de gestión, a diario descargamos o recibimos información o documentación que guardamos de la forma más ordenada posible. Lo mismo pasa con notas personales, correos electrónicos o instrucciones que nos anotamos sobre cómo hemos hecho algo. El problema es que, pasado un tiempo, cuando queremos consultar algo, no recordamos bien en qué documento estaba o donde guardamos el documento en cuestión. A mí me pasa y muchas veces me resulta más fácil volver a buscar esa información en internet, si la saqué de ahí, que encontrar algo que descargué y guardé en su momento. Y es que los buscadores de internet son los mayores expertos en esta tarea, y con la IA, además integran la información y la sirven cocinada. Incluso para buscar algo de nuestra propia organización suele ser más fácil utilizar un buscador de internet que hacerlo con nuestros propios buscadores institucionales, aun cuando éstos sólo se ocupan de lo que tenemos en nuestro domino.

Pero, eso solo vale si lo que buscamos está en internet. Si queremos tirar de documentación interna de nuestra organización o personal nuestra, esa solución no vale. Y aquí es donde vienen las herramientas destinadas a bucear en nuestra propia documentación. Cualquier cosa que se diga en este sentido seguro que es un sacrilegio para cualquier documentalista, pero la IA siempre ha tenido algo de sacrilegio en su ADN y desde sus inicios; si no, que lo digan los estadísticos.

Yo me planteé esta cuestión pensando en mi propia documentación, incluidas notas personales y correos electrónicos, que son un poco el registro de mi actividad profesional. Sin embargo, este tipo de técnicas valen igualmente para la información administrativa, por ejemplo, de expedientes, y para informes, manuales técnicos, proyectos, memorias o instrucciones.

El objetivo es que podamos preguntar: "¿Qué actuaciones se han realizado en los últimos años para controlar la procesionaria en los pinares de León?" y el sistema sea capaz de localizar y sintetizar la información relevante de múltiples documentos. Eso requiere, claro está, que dicha información exista y esté accesible al LLM (los LLM son capaces de responder a preguntas referentes a la información con la que se han entrenado, pero no sobre información privada a la que nunca han tenido acceso).

La estrategia que parece más obvia para cumplir este objetivo es la de hacer un ajuste fino del LLM, alimentando ese entrenamiento final con nuestra documentación. Sin embargo, esta opción no acaba de funcionar bien y esto puede ser porque esa documentación no gana el protagonismo que nos conviene, y la modificación de parámetros que se produce en el ajuste fino queda diluida, como ocurriría con nuestra documentación dentro del ingente volumen de información con que se ha entrenado el modelo.

Otra alternativa consiste en lanzar la pregunta al modelo, pero acompañada de información de contexto en la que -presumiblemente- está la respuesta. Suena un poco a trampa, porque al LLM le damos la pregunta y una “chuleta” para darle pistas: esa es la estrategia. La gracia está en que la generación de esa “información de contexto” la hace el sistema la buscando en nuestra documentación sin que el usuario se entere.

Esta es la alternativa que mejor funciona, con la limitación de que la cantidad de información que puede acompañar a nuestra pregunta es limitada (es lo que se llama la ventana de contexto). De hecho, el tamaño de la ventana de contexto ha sido uno de los caballos de batalla de los LLMs desde los tiempos de GPT-3 con 2K tokens (unas 4 páginas de texto) hasta el momento actual, con los 32K tokens de GPT4, los 128K de GPT4-o y los 200K de Gemini (unas 500 páginas de texto).

Para llevar a cabo esta estrategia se puede usar la técnica denominada RAG (Recuperación Avanzada de Información). Requiere que la documentación que va a servir de fuente de información sea preprocesada: se trocea en fragmentos manejables y con significado propio (chunks) y se vectoriza y almacena vectorizado. El sistema busca entonces los fragmentos que mayor relación tienen con la pregunta y los lanza al LLM a modo de contexto junto con la pregunta que queremos hacer. De esta forma el LLM puede procesar ese contexto y elaborar una respuesta muy precisa y fiel a nuestra documentación.

Esto del vectorizado o embedding[1] es clave en el NLP: consiste en convertir un texto en un vector de n dimensiones, de forma que los modelos no trabajan con texto, sino con sus correspondientes vectores de números (embeddings). La magia está en que esos vectores se construyen en función del significado del texto del que proceden y eso es lo que puede interpretar el modelo. Eso requiere que la dimensión del vector sea suficiente, por ejemplo, 1536 dimensiones (text-embedding-3-small), o 3072 (text-embedding-3-large) para modelos “sencillos” y bastantes más para los más avanzados (p. ej. 12.288 para GPT4).

Lo más interesante es que este sistema no requiere una reorganización de nuestra documentación existente. A diferencia de los sistemas tradicionales, donde la eficacia de la búsqueda dependía crucialmente de una catalogación y etiquetado adecuados, el RAG puede trabajar con documentos sin organizar (siempre y cuando puedan ser preprocesados para trocearlos y convertirlos en embeddings). En todo caso, es el sistema el que divide automáticamente los documentos en

fragmentos manejables (chunks), el que los vectoriza, el que busca los más relevantes para la pregunta formulada y el que remite todo ello al LLM. Todo esto ocurre sin que el usuario sea consiente de ello: él hace una pregunta y recibe una respuesta, con la particularidad de que está elaborada a partir de su documentación, incluso a partir de información dispersa en diferentes documentos.

En resumen, si necesitamos información sobre cómo se han abordado los clareos en una determinada especie, estos sistemas están destinados a que podamos mantener una conversación natural con el sistema: "¿Qué criterios se han seguido en los expedientes que incluyen clareos en masas jóvenes de pino silvestre?" "¿En qué zonas se han aplicado estos tratamientos?" La idea es que, si esa información existe, el sistema sea capaz de localizarla y sintetizarla, incluso extrayéndola de diferentes documentos. Además, siempre puede proporcionar las referencias a los documentos originales para su consulta detallada.

7. Ejecución de tareas con lenguaje natural.

Una de las posibilidades que abren los LLM y que tiene mucho recorrido por delante, es la utilización del lenguaje natural para ejecutar tareas en nuestros PCs. Se trata de una aspiración antigua, que las compañías de software han intentado desarrollar en múltiples ocasiones, pero con resultados no del todo satisfactorios. Por supuesto, eso está cambiando con los LLM.

La interacción con voz existe desde hace bastante tiempo, pero debía hacerse con órdenes más o menos predeterminadas; cuando hablo de interacción con lenguaje natural me refiero a eso, al lenguaje que usamos con otras personas, que es bastante más abierto. Por otra parte, cuando hablo de interacción, no me refiero a que el LLM nos responda aportando texto, imágenes o sonido/voz, sino a que ejecute tareas en aplicaciones que están instaladas en nuestro PC, como puede ser Qgis (ajenas al proveedor del servicio).

Algo parecido, pero con un dispositivo creado a tal fin en lugar de nuestro PC, es Alexa, que nació igualmente con intención de tener una interacción en lenguaje natural, pero que no lo era del todo. Además, sus capacidades se restringen a lo que puede hacer este dispositivo: reproducir música, activar recordatorios, tareas de domótica, etc.

La ejecución de código o instrucciones en nuestro PC a partir de lenguaje natural no debe confundirse con lo que se denomina *code interpreter*, que crea y ejecuta código como respuesta a órdenes en lenguaje natural, pero en el lado del proveedor, no en nuestros sistemas (hay una diferencia importante en términos de seguridad). Tampoco hay que confundir esto con la creación de código a partir de lenguaje natural, que es una gran aportación de los LLM, madura y omnipresente entre desarrolladores.

La llegada de los LLM ha transformado esta aspiración en una realidad práctica, abriendo un abanico de posibilidades. El cambio fundamental radica en la capacidad de estos modelos para entender el contexto y las intenciones del usuario y transformarlo en órdenes y parámetros que se envían a nuestro PC. Cuando este recibe esos inputs (nombres de funciones y parámetros) se encarga de ejecutarlos, siempre y cuando así lo tenga previsto. A este proceso se le denomina *function calling* o llamada a funciones. En definitiva, estamos hablando de un puente entre lenguaje natural y código.

Esta funcionalidad requiere desarrollo para definir dichas funciones y para explicar al LLM qué funciones tiene a su disposición y qué parámetros debe tranferirle en json. Es decir, el LLM no escribe código, como hace el *code interpreter*, sino que traduce el lenguaje natural al lenguaje de unas funciones que tenemos que configurar previamente.

En el ámbito de los GIS esta capacidad puede usarse para extraer información de nuestras bases de datos alfanuméricas y geográficas sin necesidad de conocer las órdenes concretas que hay que ejecutar. Se trata, por ejemplo, de preguntar: ¿Cuál es el volumen de madera que hay en los montes de UP del municipio de Íscar? Y este tipo de herramientas son capaces de convertir esa pregunta en llamada a funciones escritas en python a las que se les trasfieren determinados parámetros. Al ejecutarse esas funciones se realiza la consulta en cuestión a las bases de datos geográficas que tenemos en nuestro sistema (en este caso, dasolidar) y devuelven el resultado al LLM para que responda a dicha pregunta.

El lanzamiento de esta utilidad parte de OpenAI tuvo lugar en 2023, pero fue en 2024 cuando la mejoró e integró en los asistentes. En diciembre de 2024 con el anuncio de la API de o1 se incorporó esta funcionalidad a o1.

OpenAI ha sido pionera en desarrollar estas capacidades con sus asistentes, pero la comunidad de desarrolladores está siguiendo rápidamente este camino. LM Studio ya permite implementar estas funcionalidades en local, lo que abre la puerta a aplicaciones que requieren mayor privacidad o control sobre los datos.

Detrás de las novedades de OpenAI o de otras tecnológicas vienen siempre las de la competencia y las de la comunidad de desarrolladores, que las lleva a las versiones en abierto de diversos productos y herramientas. Por ejemplo, LM Studio ha sacado este mes (6 de enero) su versión 0.3.6 que incluye una API de llamada a funciones compatible con la de OpenAI (con su misma notación, pero llamando a modelos descargados en local). Todavía es una versión beta, pero seguro que, en poco tiempo, estará plenamente operativa.

Esta utilidad de los LLM debe llevar asociada una reflexión sobre la seguridad: ¿queremos habilitar a los LLM para que puedan controlar nuestros dispositivos, nuestros ordenadores, nuestros coches o nuestros sistemas en general? ¿Hasta qué

punto? No es solo una cuestión de seguridad frente a malware o ataques externos; la pregunta va más allá, pero eso mejor dejarlo para un episodio de Black Mirror, que seguro que lo plantea con más imaginación e ingenio que yo.

En esta primera etapa, OpenAI se ha cuidado muy mucho de que el desarrollador tenga todo el control de lo que puede ejecutar el LLM: éste no puede crear código y ejecutarlo en nuestro PC, eso sólo puede hacerlo en sus servidores, en entornos controlados (asistentes de *code interpreter*). Para que se ejecute código en nuestro ordenador, nosotros debemos tenerlo previsto, es decir, nosotros mantenemos el control. Este enfoque permite a los LLM realizar tareas complejas y específicas sin comprometer la seguridad del sistema en el que operan.

La idea es que vayamos hacia sistemas más intuitivos y potentes, que nos permitan centrarnos en los aspectos más creativos y estratégicos de nuestro trabajo, mientras que las tareas más mecánicas o complejas se ejecutan con menos esfuerzo.

8. Uso de modelos on-line y en local

Los LLM se ofrecen y utilizan habitualmente on-line porque su uso requiere unos recursos que la mayor parte de los usuarios no tiene en sus PCs y menos en los móviles (especialmente VRAM). Además, este es el modelo de negocio más habitual de los proveedores, entre otras cosas porque pueden monetizarlo mediante suscripción o pago por servicio. Esto permite a los proveedores tener un mayor control sobre lo que ofrecen con las actualizaciones que correspondan y disponer del feedback que les resulte de utilidad para sus propósitos.

El uso de estos servicios on-line presenta algunas limitaciones. La primera es el coste: el acceso a través de API se factura generalmente por tokens (unidades de texto procesado), lo que puede suponer un gasto. La verdad es que el precio es económico para el servicio que dan, pero todo depende de la cuantía del uso. La segunda consideración es la privacidad: cuando enviamos información a estos servicios, esta viaja por internet y se procesa en servidores externos, lo que puede ser problemático cuando manejamos datos sensibles o información confidencial o de carácter personal.

La alternativa es utilizar modelos en local, como Llama, Mistral, Gemma, Phi, etc. (y sus variantes). Estos modelos se pueden descargar y ejecutar en nuestros propios equipos, lo que ofrece ventajas significativas en términos de privacidad y control. Una vez descargados, la información procesada no sale de nuestros sistemas. Además, no hay costes recurrentes por uso.

El principal desafío de los modelos locales es el requisito de hardware. Aunque existen versiones adelgazadas (modelos "cuantizados") que pueden funcionar en

ordenadores “normales”, su rendimiento suele ser inferior al de los modelos on-line. En mi caso, con un core i7 con 8 núcleos y 32 GB de RAM (sin GPU dedicada) puedo cargar y consultar los modelos Llama o Mistral con 7B parámetros y cuantizados (Q4), que ocupan entre 4 y 5 GB (no así otros, de más de 10 GB). Es decir, podemos usar versiones con muchos menos parámetros que las más eficientes y además cuantizadas. Otro problema es que su funcionamiento es muy lento, entre otras cosas porque no se pueden cargar en una VRAM verdadera (a falta de VRAM, el PC usa la RAM para emularla). Tarda uno o varios minutos en hacer lo que un modelo on-line hace en segundos.

Las herramientas para utilizar modelos en local han mejorado significativamente. Aplicaciones como LM Studio u Ollama han simplificado el proceso de descarga, instalación y uso de estos modelos. Algunas incluso ofrecen una API compatible con la de OpenAI, lo que facilita la transición entre modelos en línea y locales.

Aunque con carácter los modelos on-line proporcionan resultados mucho mejores, el uso de modelos en local tiene su sitio para casos concretos por razones de confidencialidad o en tareas que puedan ser satisfechas con modelos más modestos y que no requieran inmediatez.

9. Conclusiones

Los modelos de lenguaje (LLM) están transformando nuestra forma de interactuar con la información y las herramientas digitales, ofreciendo nuevas posibilidades en el ámbito de la gestión forestal:

- La aplicación de LLM en la gestión forestal va más allá del uso común como chatbots o asistentes generales, pudiendo integrarse en flujos de trabajo específicos mediante APIs y funciones personalizadas.
- Las técnicas de RAG y búsqueda semántica permiten aprovechar mejor nuestra documentación técnica existente, facilitando el acceso y la recuperación de información relevante de manera más eficiente.
- La capacidad de los LLM para entender y generar instrucciones en lenguaje natural abre nuevas posibilidades para la automatización de tareas y la creación de interfaces más intuitivas para herramientas técnicas complejas.
- La elección entre modelos en línea o locales debe basarse en las necesidades específicas de cada caso, considerando factores como privacidad, recursos computacionales disponibles y requisitos de rendimiento.
- Es fundamental mantener un equilibrio entre la adopción de nuevas tecnologías y la estabilidad de los procesos de trabajo, evitando la dispersión que puede provocar el rápido ritmo de innovación en este campo.

10. Agradecimientos

A Carlos Santana Vega por los IA Notebook y Data Coffee y de su primera época.

11. Créditos

No he incluido referencias en esta comunicación como expresión de un cambio de estilo en la comunicación. Bien es cierto que todos los avances en IA han llegado de la mano de publicaciones científicas, pero más cierto es que los verdaderos protagonistas han sido los que convertido los átomos o los aminoácidos de las publicaciones científicas en las moléculas o las proteínas de la IA. Mis verdaderas fuentes están dispersas en internet, en cientos de sitios, repositorios de código, blogs, vídeos, etc. He estado tentado de citar alguna publicación científica de relevancia, esas que fueron seminales o fundacionales; eso siempre queda bien. Pero no hay que olvidar que son enanos a hombros de gigantes.

OpenAI, Nvidia, Google, Deep Mind, Microsoft, Meta, Anthropic, Mistral, Stability AI, Huggingface, etc. son los grandes protagonistas y durante bastantes años han dado una lección de comunicación y, muchas de ellas, también de *open source* (que empezó a quebrarse con el lanzamiento de chatGPT). Sus blogs, videos, código, documentos, etc. así como el de numerosos desarrolladores, son las referencias de la IA, y para saber lo que pasa en el mundo IA … *attention is all you need*.

---

[1] Se suele traducir como incrustación, pero ya la entiendo como una vectorización o una traducción o transducción de texto a vectores.