



ILUSTRACIÓN: VAN SAIYAN



CARMEN TORRIJOS
Lingüista computacional y
especialista en IA aplicada
a la comunicación

SOBRE CÓMO
LAS TRIBUS
DEL LENGUAJE
NATURAL
Y FORMAL
FIRMARON
LA PAZ



Palabras
clave:
Lenguaje natural,
lenguaje formal,
IA generativa,
programación,
tecnología,
educación

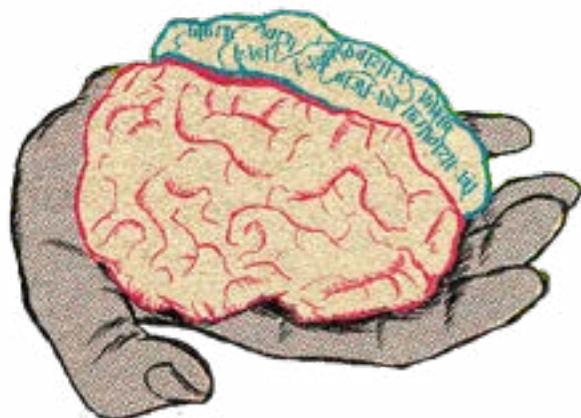
A, Orii, B

La IA generativa trata de allanarnos el camino a la información y la producción, pero en su frenética llegada no ha sido consciente de la existencia de dos tribus bien diferenciadas: la que solo utiliza el lenguaje natural, y la que también puede comunicarse con los sistemas en lenguaje formal.

On how the natural and formal language tribes signed peace BANK A, BANK B

Generative AI is trying to pave the way to information and production, but in its frantic arrival it has not been aware of the existence of two distinct tribes: the one that only uses natural language, and the one that can also communicate with systems in formal language.

Keywords: Natural language, formal language, generative AI, programming, technology, education.



Había una vez un río, con un pueblo en cada orilla. Los habitantes de una orilla, en la que vivía la mayoría de la gente, se comunicaban únicamente en lenguaje natural, aunque en distintos idiomas. Escribían novelas y poemas, pero también leyes, manuales y tratados. En la otra orilla, mucho menos poblada, los habitantes utilizaban a diario este lenguaje, pero también comprendían y utilizaban el lenguaje formal cuando era necesario. Programaban, estudiaban matemáticas y lógica y combinaban ambos lenguajes para vivir, trabajar o incluso hacer bromas. No entendían que los habitantes de la otra orilla no hicieran lo mismo, pero al mismo tiempo disfrutaban de atesorar un lenguaje privado. En la orilla del lenguaje natural, el lenguaje formal era considerado como algo complicado y ajeno, que les generaba una gran pereza. Los adolescentes elegían muy pronto en qué orilla querían vivir y, a partir de ese momento, un río ancho y profundo y muy pocos puentes hacían difícil la comunicación con la otra orilla.

Así, las relaciones entre ambos pueblos llevaban siglos estancadas cuando la inteligencia artificial llegó para instalarse en la región. Para poder desarrollarla, era necesario conformar equipos con profesionales de uno y otro lado del río, que pronto descubrieron que tenían dificultades para entenderse en las cuestiones más básicas del proyecto. La gente de la orilla del lenguaje formal empezó a construir por su cuenta un sistema de codificación que permitía a los nativos del lenguaje natural utilizar sus herramientas y producir más rápido, y así nació algo a lo que llamaron inteligencia artificial generativa. Por primera vez pusieron en circulación una moneda de uso común entre los pueblos, a la que llamaron prompt. Sin embargo, después de tantos siglos, las fronteras entre las dos formas de comunicación no eran tan fáciles de neutralizar. En ese sistema, aún en pruebas, las capas superpuestas de un lenguaje y otro se confundían y transparentaban.

Orilla A: lenguaje natural

La historia que acabamos de contar es más alegórica que ficticia. La revolución de la IA generativa, que nos permite hacer peticiones a un chatbot para asistimos en tareas que antes nos llevaban mucho más tiempo, como escribir un informe o ilustrar un artículo, no es solo tecnológica. Es también social y cultural, y en el ámbito económico abre nuevos perfiles profesionales, nuevos nichos de mercado y toda una nebulosa de consecuencias y efectos donde por ahora solo vemos dos o tres pasos más allá.

La interacción entre el código informático y el lenguaje se ha revelado en su máxima expresión precisamente en las herramientas de IA generativa

Los complementos o plug-in para ChatGPT son un buen ejemplo¹. No muchas marcas los han descubierto todavía, pero los casos de uso siguen creciendo y empiezan a comportarse como las nuevas app. El complemento de Expedia², por ejemplo, te permite preparar un viaje con datos reales, como horarios de vuelos, alojamientos disponibles y tarifas. Si estás preparando un viaje a la República Checa, puedes pedirle que te sugiera hoteles para dormir en Praga, que ajuste la búsqueda a tu precio máximo por persona y noche, y que te recomiende los mejores restaurantes típicos en Brno o dónde beber cerveza en Pilsen. También puede ayudarte con la organización de los horarios y el plan de ruta, y aquí no hay riesgo de que nos cuele falsos datos: podemos confiar, porque la información no proviene del modelo de lenguaje, sino del plug-in, que nos facilita los enlaces a la web de Expedia para que podamos comprobarlos. Nuestra confianza en los resultados estará, por tanto, en relación directa con nuestra confianza en la marca y no en el sistema de IA. Otro complemento interesante es el de Amazon Gift Chooser, que te ayuda a elegir el mejor regalo siempre y cuando le des información sobre la persona a la que quieres regalar, sus gustos y tu límite de gasto. Con cada idea, añadirá un enlace a Amazon para que puedas comprarlo directamente si te convence, aunque ahora mismo solo trabaja con Amazon EE. UU. y con precios en dólares. El objetivo siempre es el mismo, y no es pequeño: permitir a los usuarios combinar las capacidades de conversación GPT con la oferta de valor de una marca, en una plataforma web que sobrepasó los 100 millones de usuarios solo dos meses después de su lanzamiento³. Todas las peticiones a estos sistemas se realizan únicamente en lenguaje natural, en un nuevo género literario a medio camino entre la instrucción y la descripción al que llamamos prompt.

Accedemos a estos complementos de dos maneras, ambas sencillas una vez que estamos en la pista: la primera es dentro del propio ChatGPT, en su versión de pago, habilitando las funcionalidades beta. Cuando selecciona-

mos GPT-4, aparece una pestaña plug-in donde podemos encontrar un plug-in store, una tienda al más puro estilo Google Play o Apple Store, con un catálogo de extensiones gratuitas que podemos instalar y usar sin más. La segunda opción, más accesible si no tenemos la versión de pago, es en forma de extensión para Google Chrome, que se instalará en nuestro navegador y se activará cuando abramos ChatGPT como una capa adicional. Entre estas últimas se encuentran algunas muy prácticas, como editGPT, que te permite aplicar un control de cambios similar al de Word para saber en qué mejora el chatbot tus textos, o Prompt Genius, un administrador de prompts destinado a cubrir algunas carencias de funcionalidad en ChatGPT. Este genio incorporado te permite algo fundamental, que es guardar tus prompts favoritos en la propia interfaz para no tener que reescribirlos cada vez. Por ejemplo, si has dado con el prompt ideal para tus notas de prensa o tus menús semanales, puedes lanzarlos cada vez que lo necesites sin tener que escribirlos de nuevo.

Fue precisamente trasteando con Prompt Genius cuando encontré algo que me llamó la atención: la posibilidad de definir variables dentro de los prompts. Tardé un tiempo en entender por qué una funcionalidad que reconocía tan natural y necesaria me resultaba a la vez tan extraña en el contexto, tan estridente y fuera de lugar como un iglú en una playa tropical: en el ejemplo, en medio de una frase en español, había una palabra encerrada entre dobles llaves ({}).

Orilla B: lenguaje formal

Empezando por el principio: una variable es un concepto conocido para cualquier estudiante de los cursos más básicos de iniciación a la programación. Es un elemento contenedor que no es fijo como el resto del código, como indica su nombre, sino que su contenido puede cambiar con

¹ OpenAI: "ChatGPT plug-ins" en openai.com (marzo 2023). Disponible en: <https://openai.com/blog/chatgpt-plugins>

² Expedia Group: "ChatGPT wrote this press release — no, it didn't, but it can now assist with travel planning in the Expedia app" en <https://www.expediagroup.com/investors/news-and-events/financial-releases/news/news-details/2023/Chatgpt-Wrote-This-Press-Release-No-It-Didn-But-It-Can-Now-Assist-With-Travel-Planning-In-The-Expedia-App/default.aspx>

³ Hu, K: "ChatGPT sets record for fastest-growing user base - analyst note" en Reuters (febrero 2023). Disponible en: <https://www.reuters.com/technology/chatgpt-sets-record-fastest-growing-user-base-analyst-note-2023-02-01/>

cada ejecución. Ya no ocurre con tanta frecuencia, pero en los primeros tiempos de los envíos masivos por correo electrónico era habitual recibir de vez en cuando un email con el mensaje "Te damos la bienvenida <NOMBRE>:". Esta sensación de desencanto al encontrarnos con las tripas mismas de la informática se produce porque estamos ante una variable, el nombre de la persona receptora, una información cambiante en cada mensaje que se extrae de una base de datos en la que cada correo electrónico lleva asociado un nombre de pila, que se insertará automáticamente. La extensión Prompt Genius traslada este concepto de las variables informáticas al prompt y permite incluir en él una o más, marcadas en este caso con dobles llaves ({}). Es una manera rústica de incluir una señal gráfica para que el sistema sepa que esa palabra es un comodín que puede adquirir distintos valores. Por ejemplo, imagina que has dado con el prompt perfecto para que ChatGPT escriba por ti una guía turística para tus fines de semana de escapada. Has invertido un tiempo en crear ese prompt, pero hay contenido que cada vez es diferente, como la ciudad que vas a visitar. Es tan sencillo como dejar la palabra {{ciudad}} entre llaves, y Prompt Genius te preguntará cuál es en cada nueva ejecución, por lo que tendrás el prompt preparado en cuestión de segundos. Lo que estamos haciendo no es más que acumular facilitadores adicionales sobre una herramienta ya de por sí facilitadora como es ChatGPT. Nos obsesiona, desde hace varios años, poder poner cualquier contenido que queramos al alcance de una simple petición en nuestro idioma, y caminamos en una trayectoria sin vuelta atrás que nos lleva a allanar más y más el terreno.

Desde luego estamos ante una aproximación ganadora, salvo porque lo que llamamos nuestro idioma está mucho más limitado de lo que creemos. El lenguaje natural es el que nos permite expresar nuestras ideas y emociones y crear conexiones sociales, algo que es y va a seguir siendo



4 Álvarez Mellado, Elena: "Lenguaje inclusivo: algunas claves lingüísticas" en [elDiario.es](https://www.eldiario.es/opinion/zona-critica/lenguaje-inclusivo-claves-linguisticas_129_8021559.html) (julio 2021). Disponible en: https://www.eldiario.es/opinion/zona-critica/lenguaje-inclusivo-claves-linguisticas_129_8021559.html
 5 FundéuRAE: "La arroba (@) para el género, uso inadecuado", en [fundeu.es](https://www.fundeu.es/recomendacion/arroba/) (febrero 2011). Disponible en: <https://www.fundeu.es/recomendacion/arroba/>
 6 https://es.wikipedia.org/wiki/Ken_Thompson

prioridad por pura supervivencia. Ahora bien, a diferentes velocidades vamos llegando a momentos evolutivos en los que este sistema de expresión encuentra límites. Un ejemplo es el uso del lenguaje inclusivo en castellano, un idioma que ha visto brotar un conflicto social de una dualidad de género de herencia indoeuropea, y de la ausencia de un género neutro que pudiera rescatarnos⁴. Y aquí, en una sociedad occidental en la que las visiones de progreso valoran el bilingüismo como un tesoro de las personas y los pueblos, batallamos a diario para no tener que traspasar las fronteras entre el lenguaje natural y el formal. Este último lo estamos utilizando sin darnos cuenta, por ejemplo, cuando indicamos la opcionalidad con una barra (/) los alumnos/as, un gesto moderno, seguramente no anterior al siglo XVIII y totalmente ajeno al lenguaje natural, pero útil para expresar dos opciones equivalentes en el mínimo espacio posible. Por un lado, la convención de la barra (/) ha experimentado un fuerte crecimiento desde la llegada de Internet y la mensajería instantánea. En el lenguaje informal por escrito es eficaz en lo que comunica, es breve, resuelve. Y, sin embargo, en un lenguaje más elevado tendemos a evitarla por poco elegante, como en un artículo o en un libro. Ocurre también

con el rechazo generalizado al uso de la arroba (@)⁵ para marcar la inclusividad, una solución escrita también breve y eficaz, pero que hemos descartado de plano y que ya apenas vemos en comunicaciones, publicidad o prensa. ¿Por qué? Es sencillo. En el caso de la barra de opcionalidad, cuando nos la encontramos leyendo en voz alta nos vemos en la obligación de marcarla con una pausa o un gesto leve. En el caso de la arroba, nuestro cerebro colapsa. Mi sensación de incomodidad al encontrar una palabra entre llaves en medio de un prompt, si bien no es verbalizada muy a menudo, es común a gran parte de los hablantes: más allá de las letras y nuestros signos de puntuación habituales, los corchetes y llaves, como los asteriscos o el

porcentaje, son símbolos que percibimos como ajenos al lenguaje natural, venidos de otro planeta, de alguna manera no del todo válidos para la comunicación entre las personas. Como si fueran propiedad más bien del mundo de los números, o de las máquinas.

Entre tanto, al otro lado del río, las expresiones regulares llevan décadas siendo una herramienta de inmensa utilidad para los programadores, irreductibles a pesar del tiempo y la llegada de técnicas mucho más avanzadas. Se trata de conjuntos de caracteres que representan de forma simbólica un patrón de búsqueda en un texto, y que en los años 60 el matemático Ken Thompson introdujo entre las funcionalidades de un editor de texto llamado QED⁶. Por ejemplo, imagina que quieres encontrar números de teléfonos móviles españoles en un millón de textos de Internet. Tendrías que indicar en la expresión regular que estás buscando tres conjuntos de tres dígitos, separados por espacios o no, y marcar que el primer dígito del primer conjunto pueda ser tanto un 6 como un 7, porque muchos teléfonos en España empiezan ya por un 7. La expresión podría quedar más o menos así:

```
\b[67]\d{2}[-\s]?d{3}[-\s]?d{3}\b
```

Nada parecido a la secuencia de nueve números de nuestro teléfono, pero con una eficacia del cien por cien para detectar ese, o cualquier otro. La opcionalidad en este caso se marca con corchetes "[67]", mientras que la expresión "\d" sirve para indicar que en ese lugar se espera un dígito cualquiera. Entre llaves "{}" indicamos el número de dígitos que queremos encontrar en cada conjunto, junto con otros símbolos para identificar un espacio ("s") o marcar el inicio y fin de la regla ("b"). Estas expresiones son complejas de aprender y de crear, pero una vez dominadas despliegan todo su potencial. Una convención tan arbitraria como nuestras palabras o nuestros fonemas, pero más eficiente, que en la actualidad aún sorprende por cómo sostiene desde las sombras grandes proyectos de software.

El puente

Que el grueso de la población digitalizada tuviera un conocimiento básico en expresiones regulares, o que se incluyeran de forma natural como aprendizaje desde el instituto, podría mejorar la eficiencia en la búsqueda de información, pero también promover un mayor entendimiento entre los expertos en programación y el

Nadie es hablante nativo del lenguaje formal. Generalizar el hábito de aprenderlo y utilizarlo, o fomentar el pensamiento numérico y lógico, es una misión colectiva

resto de disciplinas, además de fomentar el desarrollo de habilidades valiosas, como el pensamiento lógico y sintético. En absoluto se trata de sustituir el lenguaje natural. La motivación es más bien ampliar sus límites democratizando una herramienta que ha probado con creces su valor. La interacción entre el código informático y el lenguaje, hasta ahora en orillas opuestas de un río con muy pocos puentes, se ha revelado en su máxima expresión precisamente en las herramientas de IA generativa. Poco a poco vamos adquiriendo la costumbre de interactuar con ellas sin código (las llamamos no-code o low-code). Vamos normalizando el texto a texto, texto a imagen, texto a música, texto a vídeo, integramos el prompting en nuestras vidas como nuestra nueva interfaz de trabajo. Con los tres o cuatro trucos que ruedan por las redes vamos aprendiendo a pedir a los sistemas lo que necesitamos, sin mayor esfuerzo. Pero es precisamente esta revolución del texto-a-todo la que ha puesto ante nuestros ojos una nueva colisión entre lenguajes. En un camino supuestamente allanado, aparecen diminutos baches donde el militante del lenguaje natural se encuentra con el código sin querer.

La manifestación más reciente de todo esto, y quizá el puente más moderno y mejor trazado hasta la fecha en este río, es el Code Interpreter que OpenAI anunció el 6

La revolución de la IA generativa no es solo tecnológica. Es también social y cultural

de julio de 2023⁷. Esta funcionalidad, útil por igual para programadores y no programadores, es una evolución natural de ChatGPT que nos permite subir ficheros, interactuar con ellos y obtener resultados más allá de la conversación escrita. Podemos, por ejemplo, subir un fichero de datos macroeconómicos y obtener gráficas automáticas, o pedirle que nos genere un código QR a partir de un enlace a una web. Podemos pedirle que redacte un informe a partir de unos datos y exportar después el resultado en PDF, o que extraiga el texto que aparece en una imagen. Ahora bien, esta magia revolucionaria sería imposible sin un paso intermedio: la máquina traduce nuestra petición a código en un lenguaje de programación muy concreto, Python, que podemos ver generarse ante nuestros ojos, y es el propio código el que genera el resultado, que después el sistema nos devuelve en un enlace descargable. La razón de que OpenAI haya decidido no hacer el proceso cien por cien transparente para los usuarios no programadores es probablemente maximizar su público, y esto ha generado una coyuntura sin precedentes. Por un lado, los habitantes de la orilla del lenguaje natural tendrán que habituarse a ver molestos fragmentos de código precediendo sus resultados. Por otro, los habitantes de la orilla del código ahora pueden crearlo sin escribirlo, cambiando el esfuerzo de lugar para construir en lenguaje natural el mejor prompt posible. Es significativo también que OpenAI decidiera, semanas después del lanzamiento de este módulo, cambiar el nombre de Code Interpreter, evidente para una orilla pero no para la otra, a Advanced Data Analysis. Mucho más explicativo, mucho más para todo el mundo, mucho más puente.

En esta era de la inteligencia artificial nos encontramos ante tecnologías tan capaces, con un potencial tan brillante, que nos sitúan una y otra vez en un espejo frente a nuestras propias limitaciones. Algunas de ellas son naturales, predeterminadas —nuestra capacidad de cálculo, mínima, a años luz de la artificial—, pero otras son adquiridas o inconscientemente autoimpuestas. Un condicionamiento cultural omnipresente, aunque muy poco evidenciado hasta ahora, ha bloqueado a lo largo de los siglos a buena parte

de la población la inclinación a utilizar el lenguaje formal cuando es necesario. Un instrumento tan humano y útil como la lengua, que descartamos porque cometemos el error de entenderlo como ajeno a la especie.

No sabemos si caminamos hacia un mundo donde todos programaremos o hacia un mundo no-code en el que programarán los de siempre, pero, más allá de la incertidumbre, no podemos discutir la utilidad de una mirada híbrida, capaz de comunicarse en ambos lenguajes sin tantos complejos. Quizá es un buen momento para revisar si nuestros itinerarios educativos, tan cuidadosamente contruidos, están actuando como restricciones que han perdido su sentido original. No solo la división tradicional entre ciencias y letras o la elección exclusiva de asignaturas en el bachillerato son parte de un sistema caducado, también los mensajes constantes en los que la gente adulta nos posicionamos —a veces con vergüenza, a veces con un orgullo mal entendido— en una u otra orilla.

Nadie es hablante nativo del lenguaje formal. Generalizar el hábito de aprenderlo y utilizarlo, o fomentar el pensamiento numérico y lógico, es una misión colectiva que va más allá de construir puentes sobre un río, y que es tan capacitadora e integradora como el bilingüismo natural. Consiste en dejar de decir que no todo el mundo vale para todo, y atreverse más a intentarlo. Tiene mucho más que ver con perder el miedo a nadar.

Bibliografía

- Álvarez Mellado, E.: "Lenguaje inclusivo: algunas claves lingüísticas" en *elDiario.es* (julio 2021). Disponible en: https://www.eldiario.es/opinion/zona-critica/lenguaje-inclusivo-claves-linguisticas_129_8021559.html
- Expedia Group: "ChatGPT wrote this press release — no, it didn't, but it can now assist with travel planning in the Expedia app" en *Expedia.com* (abril 2023). Disponible en: <https://www.expediagroup.com/investors/news-and-events/financial-releases/news/news-details/2023/Chatgpt-Wrote-This-Press-Release--No-It-Didnt-But-It-Can-Now-Assist-With-Travel-Planning-In-The-Expedia-App/default.aspx>
- FundéuRAE: "La arroba (@) para el género, uso inadecuado", en *fundeu.es* (febrero 2011). Disponible en: <https://www.fundeu.es/recomendacion/arroba/>
- Hu, K.: "ChatGPT sets record for fastest-growing user base - analyst note" en *Reuters* (febrero 2023). Disponible en: <https://www.reuters.com/technology/chatgpt-sets-record-fastest-growing-user-base-analyst-note-2023-02-01/>
- OpenAI: "ChatGPT plug-ins" en *openai.com* (marzo 2023). Disponible en: <https://openai.com/blog/chatgpt-plugins>
- Yildirim, S.: "ChatGPT Code Interpreter: how it saved me hours of work", en *Towards Data Science* (julio 2023). Disponible en: <https://towardsdatascience.com/chatgpt-code-interpreter-how-it-saved-me-hours-of-work-3c65a8dfa935#>

⁷ Yildirim, Soner: "chatGPT Code Interpreter: how it saved me hours of work", en *Towards Data Science* (julio 2023). Disponible en: <https://towardsdatascience.com/chatgpt-code-interpreter-how-it-saved-me-hours-of-work-3c65a8dfa935#>