Marco de validación de los LLM

"Las consecuencias de que la IA vaya mal son graves, por lo que debemos ser proactivos en lugar de reactivos".
Elon Musk[94]

Marco

Los modelos de lenguaje de gran escala (LLM) ofrecen un gran potencial para transformar diversos sectores y aplicaciones, pero también conllevan riesgos significativos que deben abordarse. Estos riesgos incluyen la generación de información errónea o alucinaciones, la perpetuación de sesgos, la dificultad para olvidar la información aprendida, preocupaciones éticas y de equidad, problemas de privacidad por uso indebido, dificultades en la interpretación de los resultados, y la potencial creación de contenido malicioso, entre otros.

Dado el impacto potencial de estos riesgos, es necesario validar exhaustivamente los LLM antes de su despliegue en entornos de producción. De hecho, la validación de los LLM no es solo una buena práctica, sino también un requisito regulatorio en muchas jurisdicciones. En Europa, la propuesta de AI Act exige una evaluación y mitigación de los riesgos de los sistemas de IA[95], mientras que, en Estados Unidos, el marco de gestión de riesgos de IA del NIST[96] y el AI Bill of Rights destacan la importancia de comprender y abordar los riesgos inherentes a estos sistemas.

La validación de los LLM puede partir de los principios establecidos en la disciplina de riesgo de modelo, que se centra[97] en evaluar y mitigar los riesgos derivados de errores, deficiente implementación o mal uso de los modelos. Sin embargo, en el caso de la IA, y particularmente de los LLM, es necesario adoptar una perspectiva más amplia que abarque los otros riesgos que comportan. Un enfoque integral de validación es esencial para garantizar un despliegue seguro y responsable de los LLM.

Este enfoque holístico se plasma en un marco de validación multidimensional para los LLM, que cubre aspectos clave (Fig. 9) como el riesgo de modelo, la gestión de datos y privacidad, la ciberseguridad, los riesgos legales y de cumplimiento normativo, los riesgos operativos y tecnológicos, la ética y la reputación, y el riesgo de proveedor, entre otros. Al abordar todos estos aspectos de manera sistemática, las organizaciones pueden identificar y mitigar de manera proactiva los riesgos asociados con los LLM, sentando las bases para aprovechar su potencial de manera segura y responsable.

En los LLM, esta evaluación de riesgos se puede anclar en las siguientes dimensiones usadas en la disciplina de riesgo de modelo, adaptando los tests en función de la naturaleza y el uso del LLM:

- **Datos de entrada:** comprensión del texto[98], calidad del dato[99].
- **Solidez conceptual y diseño del modelo:** selección del modelo y sus componentes (p. ej., metodologías de *fine-tuning*, conexiones a bases de datos, RAG[100]), y comparación con otros modelos[101].

[94] Elon Musk (n. 1971), CEO de X, SpaceX, Tesla. Empresario sudafricano-estadounidense, conocido por fundar o cofundar empresas como Tesla, SpaceX y PayPal, dueño de X (anteriormente Twitter), red social que tiene su propio LLM, llamado Grok.

[95] European Parliament (2024) AI Act Art. 9: "Se establecerá, aplicará, documentará y mantendrá un sistema de gestión de riesgos en relación con los sistemas de IA de alto riesgo. El sistema de gestión de riesgos [...] comprenderá [...] la estimación y evaluación de los riesgos que puedan surgir cuando el sistema de IA de alto riesgo se utilice de acuerdo con su finalidad prevista, y en condiciones de uso indebido razonablemente previsibles".

[96] NIST (2023): "La decisión de encargar o desplegar un sistema de IA debe basarse en una evaluación contextual de las características de fiabilidad y los riesgos, impactos, costes y beneficios relativos, y debe ser informada por un amplio conjunto de partes interesadas".

[97] Management Solutions (2014). Model Risk Management: Quantitative and Qualitative Aspects.

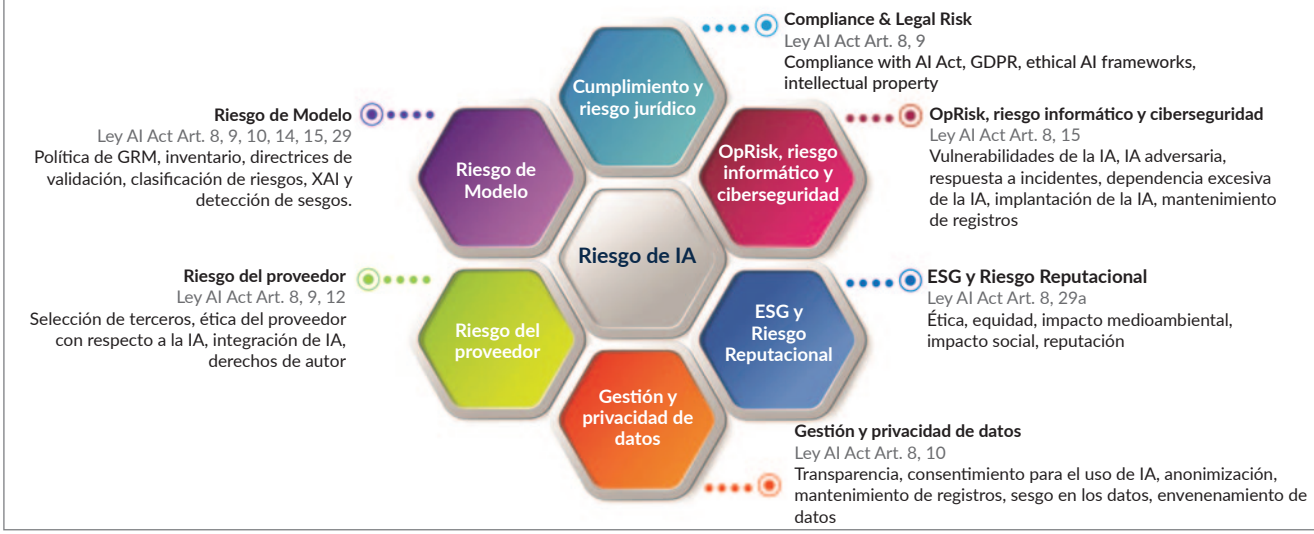
[98] Imperial et al. (2023).

[99] Wettig et al. (2024).

[100] RAG (Retrieval-Augmented Generation) es una técnica avanzada en la que un modelo de lenguaje busca información relevante de una fuente externa antes de generar texto. Esto enriquece las respuestas con conocimientos precisos y actuales, combinando inteligentemente la búsqueda de información y la generación de texto. Al integrar datos de fuentes externas, los modelos RAG, como los RAG-Token y RAG-Sequence propuestos (Lewis et al., 2020), ofrecen respuestas más informadas y coherentes, minimizando el riesgo de generar contenido inexacto o 'alucinaciones'. Este avance representa un paso significativo hacia modelos de inteligencia artificial más confiables y basados en evidencia real.

[101] Khang (2024).

Fig. 9. Riesgos asociados a la IA y referencia regulatoria en el AI Act.



- **Evaluación del modelo y análisis de sus resultados:** privacidad y seguridad de los resultados[102], precisión del modelo[103], consistencia[104], robustez[105], adaptabilidad[106], interpretabilidad (XAI)[107], ética, sesgos y equidad[108], toxicidad[109], comparación contra modelos *challenger*.

- **Implementación y uso:** revisión humana en el uso (incluyendo el monitoreo de usos indebidos), resolución de errores, escalabilidad y eficiencia, aceptación del usuario.

- **Gobernanza**[110] **y ética**[111]: marco de gobierno de la IA generativa, incluyendo los LLM.

- **Documentación**[112]: completitud de la documentación del modelo.

- **Cumplimiento regulatorio**[113]: evaluación de los requisitos regulatorios (p. ej., AI Act).

Para garantizar el uso efectivo y seguro de los modelos de lenguaje, es fundamental realizar una evaluación de riesgos que considere tanto el modelo en sí como su uso específico. Esto asegura que, independientemente de su origen (*in-house* o de un proveedor) o personalización (*fine-tuning*), el modelo funcione adecuadamente en su contexto de uso, cumpliendo con los estándares de seguridad, ética y regulación necesarios.

Técnicas de validación

Cuando una organización se plantea implementar un LLM para un caso de uso específico, puede ser beneficioso adoptar un enfoque integral que abarque las dimensiones clave del ciclo de vida del modelo: datos, diseño, evaluación, implementación y uso. Asimismo, de manera transversal, resulta necesario evaluar el cumplimiento de la normativa aplicable, como el AI Act en la Unión Europea.

En cada una de estas dimensiones, dos grupos de técnicas complementarias permiten realizar una validación más completa (Fig. 10):

- Métricas de evaluación cuantitativas (tests): se trata de pruebas cuantitativas estandarizadas que miden el desempeño del modelo en tareas específicas; *benchmarks* y métricas predefinidas para evaluar distintos aspectos del rendimiento del LLM después del preentrenamiento, o durante las etapas de *fine-tuning* o *instruction-tuning* (es decir, técnicas de aprendizaje por refuerzo), optimización, ingeniería de *prompts*, o recuperación y generación de información. Algunos ejemplos incluyen la precisión en la creación de resúmenes, la robustez ante ataques adversarios o la consistencia en la respuesta ante *prompts* similares.

- Evaluación humana: implica el juicio cualitativo por parte de expertos y usuarios finales; por ejemplo, la revisión de una muestra concreta de los *prompts* y las respuestas del LLM por un ser humano para identificar errores.

La validación de un uso específico de un LLM, por tanto, se lleva a cabo mediante una combinación de técnicas cuantitativas (tests) y cualitativas (evaluación humana). Para cada caso de uso concreto, es necesario diseñar un enfoque de validación a medida, que consistirá en una selección de algunas de estas técnicas.

[102] Nasr (2023).
[103] Liang (2023).
[104] Elazar (2021).
[105] Liu (2023).
[106] Dun (2024).
[107] Singh (2024).
[108] NIST (2023), Oneto (2020) y Zhou (2021).
[109] Shaikh (2023).
[110] Management Solutions (2014). Model Risk Management.
[111] Oneto (2020).
[112] NIST (2023).
[113] European Parliament (2024). AI Act.

Fig. 10. Pruebas de evaluación de LLM.

Dimensiones	Aspectos validados	Descripción	Métricas de validación (ejemplos)	Evaluación humana (ejemplos)
1. Datos de entrada	**1.1 Calidad de dato**	Grado de calidad de la modelización o de los datos de aplicación	• Flesch-Kinkaid Grade	• Revisión caso a caso
2. Diseño del modelo	**2.1 Diseño del modelo**	Elección de modelos y metodología adecuadas	• Revisión de los elementos del LLM: RAG, filtros de entrada o salida, definición de *prompts*, *fine-tuning*, optimización, etc. • Comparación contra otros LLM	• Pruebas A/B
3. Evaluación del modelo	**3.1 Privacidad y seguridad**	Respeto de la confidencialidad y no regurgitación de información personal	• Data leakage • PII tests, K-anonymity	• Registros • *Hacking* ético
	3.2 Precisión	Corrección y pertinencia de las respuestas del modelo	• Q&A: SummaQA, Word error rate • Recuperación de información: SSA, nDCG • Resumen: ROUGE • Traducción: BLEU, Ruby, ROUGE-L • Otros: Sistemas de QA, nivel de *overrides*, nivel de alucinaciones, etc. • Benchmarks: XSUM, LogiQA, WikiData, etc.	• *Backtest* de forzajes • Revisión caso a caso
	3.3 Consistencia	Respuestas uniformes a consultas similares	• Cosine similarity • Jaccard similarity index	• Revisión caso a caso • Pruebas A/B
	3.4 Robustez	Resiliencia a la información adversa o engañosa	• Generación de texto adversario (TextFooler), patrones Regex • Benchmarks de ataques adversarios (PromptBench), número de refusals	• *Hacking* ético • Simulacros de incidentes
	3.5.Adaptabilidad	Capacidad para aprender o adaptarse a nuevos contextos	• Rendimiento del LLM ante datos nuevos por Zero/One/Few-shot learning	• Pruebas A/B • Revisión caso a caso
	3.6 Explicabilidad	Comprensión del proceso de toma de decisiones	• SHAP • Puntuaciones de explicabilidad	• *Hacking* ético • *Focus groups*
	3.7 Sesgos y equidad	Respuestas sin sesgo demográfico	• AI Fairness 360 toolkit • WEAT score, paridad demográfica, asociaciones de palabras, etc. • *Benchmarks* de sesgos (BBQ, etc.)	• *Hacking* ético • *Focus groups*
	3.8 Toxicidad	Propensión a generar contenidos nocivos	• Perspective API, Hatebase API • *Toxicity benchmarks* (RealToxicityPrompts, BOLD, etc.)	• *Hacking* ético • *Focus groups*
4. Implementación y uso	**4.1 Revisión humana y seguridad de uso**	Exclusión de sugerencias perjudiciales o ilegales e inclusión de una revisión humana (‘*human-in-the-loop*’)	• Protocolos de riesgos, evaluaciones de seguridad • Control humano	• *Hacking* ético • *Focus groups*
	4.2 Recuperación y gestión de errores	Capacidad para recuperarse de errores y gestionar entradas inesperadas	• Tests de recuperación del sistema • Métricas de procesamiento de errores	• Simulacros de incidentes
	4.3 Escalabilidad	Mantenimiento del rendimiento con más datos o usuarios	• *Stress testing* del sistema, Apache Jmeter, etc. • *Benchmarks* de escalabilidad	• Simulacros de incidentes • Pruebas A/B
	4.4 Eficiencia	Utilización de recursos y velocidad de respuesta	• *Time-to-first-byte* (TTFB), uso de GPU/CPU, inferencia de emisiones, memoria, latencia	• Simulacros de incidentes
	4.5 Aceptación del usuario	Pruebas de aceptación de usuario	• *Checklist* de requisitos de usuario, *opt-out* del usuario • Satisfacción del usuario (Net Promoter Score, CSAT)	• UX *tracking* • Pruebas A/B

La selección exacta de técnicas dependerá de las características particulares del caso de uso; y, en concreto, varios factores importantes a tener en cuenta para decidir las técnicas más adecuadas son:

- El nivel de riesgo y la criticidad de las tareas que se confiarán al LLM.
- Si el LLM está abierto al público (y por tanto el *hacking* ético cobra especial relevancia) o si su uso se limita al ámbito interno de la organización.
- Si el LLM procesa datos personales.
- La línea de negocio o servicio que utilizará el LLM.

Un análisis cuidadoso de estos *drivers* permitirá construir un marco de validación robusto y adaptado a las necesidades de cada uso de un LLM.

Métricas de evaluación cuantitativas

Aunque es un campo de estudio emergente, existe una amplia gama de métricas cuantitativas para evaluar el rendimiento de los LLM. Algunas de estas métricas son adaptaciones de las utilizadas en modelos tradicionales de aprendizaje automático, como la precisión, la exhaustividad (*recall*), la puntuación F1 o el área bajo la curva ROC (AUC-ROC). Otras métricas han sido diseñadas específicamente para evaluar aspectos únicos de los LLM, como la coherencia del texto generado, la fidelidad a los hechos o la diversidad del lenguaje.

En este sentido, ya existen marcos holísticos de testeo cuantitativo de LLM en entornos de programación en Python, que facilitan la implementación de muchas de las métricas cuantitativas de validación, por ejemplo:

- **LLM *Comparator***[114]: herramienta creada por investigadores de Google para la evaluación automática y comparación de LLM, que revisa la calidad de las respuestas de los LLM.
- **HELM**[115]: evaluación holística de los modelos del lenguaje, que compila métricas de evaluación a lo largo de siete dimensiones (precisión, calibración, robustez, equidad, sesgos, toxicidad y eficiencia) para una serie de escenarios predefinidos.
- **ReLM**[116]: sistema de validación y consulta de LLM mediante uso del lenguaje, incluyendo evaluaciones de modelos lingüísticos, memorización, sesgos, toxicidad y comprensión del lenguaje.

En la actualidad, ciertas técnicas de validación, como los métodos de explicabilidad (XAI) basados en SHAP, algunas métricas como ROUGE[117] o los análisis de imparcialidad mediante paridad demográfica, aún no cuentan con umbrales predefinidos ampliamente aceptados. En estos casos, es tarea de la comunidad científica y de la industria seguir investigando para establecer criterios claros que permitan una validación robusta y estandarizada.

[114] Kahng (2024).
[115] Liang (2023).
[116] Kuchnik (2023).
[117] Duan (2023).

Fig. 11. Algunas técnicas de evaluación humana de LLM.

Técnicas de evaluación humana

Mientras que las métricas de evaluación cuantitativa son implementables de forma más directa debido a la multitud de recursos *online* y publicaciones de los últimos años, las técnicas de evaluación humana[118] son variadas y deben ser construidas en función de la tarea específica[119] que esté realizando el LLM, e incluyen (Fig. 11):

- ***Backtest* de los forzajes del usuario:** contabilizar y medir la importancia de las modificaciones humanas en los resultados del LLM (p. ej., cuántas veces un gestor comercial debe modificar manualmente los resúmenes de llamadas a clientes que ha realizado un LLM).
- **Revisión caso a caso:** comparar una muestra representativa de respuestas del LLM con las expectativas del usuario («*ground truth*").
- ***Hacking* ético *(Red Team)*:** manipular los *prompts* para forzar al LLM a producir resultados no deseados (p. ej., regurgitación de información personal, contenido ilegal, tests de penetración, explotación de vulnerabilidades).
- **Testeo A/B:** comparación para evaluar dos versiones del LLM (A y B), o de un LLM frente a un ser humano.
- ***Focus groups:*** recabar opiniones de diversos usuarios sobre el comportamiento del LLM, p. ej., en materia de ética, adecuación cultural, discriminación, etc.
- **Experiencia del usuario *(UX tracking)*:** observar y evaluar las interacciones de los usuarios con el LLM a lo largo del tiempo o en tiempo real.
- **Simulacros de incidentes:** simular escenarios adversos para probar la respuesta del LLM (p. ej., prueba de estrés, comprobación de copias de seguridad, medición del tiempo de recuperación, etc.).
- **Mantenimiento de registros:** revisar los diarios y registros del sistema LLM, garantizando el cumplimiento de la normativa y la traza de auditoría.

[118] Datta, Dickerson (2023).
[119] Guzmán (2015).

Benchmarks de evaluación de LLM

La mayoría de los modelos de inteligencia artificial generativa, incluidos los LLM, se someten a pruebas utilizando *benchmarks* públicos que evalúan su desempeño en una variedad de tareas relacionadas con la comprensión y el uso del lenguaje natural. Estas pruebas sirven para medir cómo maneja el LLM tareas específicas y refleja el entendimiento humano. Algunos de estos *benchmarks* incluyen:

- GLUE/SuperGLUE: evalúa la comprensión del lenguaje a través de tareas que miden la capacidad de un modelo para entender el texto.
- Eleuther AI Language Model Evaluation Harness: realiza una evaluación "few-shot" de los modelos, es decir, su precisión con muy pocos ejemplos de entrenamiento.
- ARC (AI2 Reasoning Challenge): pone a prueba la habilidad del modelo para responder preguntas de ciencia que requieren razonamiento.
- HellaSwag: evalúa el sentido común del modelo a través de tareas que requieren predecir el final coherente de una historia.
- MMLU (Massive Multitask Language Understanding): prueba la precisión del modelo en una amplia gama de tareas para evaluar su comprensión multitarea.
- TruthfulQA: desafía al modelo a discernir entre información verdadera y falsa, evaluando su habilidad para manejar datos verídicos.
- Winogrande: otra herramienta para evaluar el sentido común, similar a HellaSwag pero con diferentes métodos y énfasis.
- GSM8K: evalúa la capacidad lógico-matemática del modelo a través de problemas de matemáticas diseñados para estudiantes.

Nuevas tendencias

El campo de la validación de LLM se encuentra en constante evolución, impulsado por los rápidos avances en el desarrollo de estos modelos y por la creciente conciencia sobre la importancia de garantizar su fiabilidad, equidad y alineación con la ética y la regulación.

A continuación, se presentan algunas de las principales tendencias emergentes en este ámbito:

- **Explicabilidad de los LLM:** a medida que los LLM ganan en complejidad y opacidad, crece la demanda de mecanismos que permitan entender y explicar su funcionamiento interno. Las técnicas de XAI (*eXplainable* AI) como SHAP, LIME o la atribución de importancia a los tokens de entrada están ganando protagonismo en la validación de LLM. Aunque para los modelos tradicionales hay una variedad de técnicas *post-hoc* disponibles para comprender el funcionamiento de los modelos a nivel local y global[120] (p. ej., Anchors, PDP, ICE), y ha proliferado la definición e implementación de modelos inherentemente interpretables por construcción, la implementación de estos principios para los LLM no está todavía resuelta.

- **Uso de LLM para explicar LLM:** una tendencia emergente consiste en utilizar un LLM para generar explicaciones sobre el comportamiento o las respuestas de otro LLM. En otras palabras, se emplea un modelo del lenguaje para interpretar y comunicar de forma más comprensible el razonamiento subyacente de otro modelo. Para enriquecer estas explicaciones, se están desarrollando herramientas[121] que incorporan además técnicas de análisis post-hoc.

- **Técnicas de interpretabilidad *post-hoc***: estas técnicas se basan en la interpretabilidad de los resultados en la etapa posterior al entrenamiento o *fine-tuning*, y permiten identificar qué partes de la entrada han influido más en la respuesta del modelo (importancia de características), encontrar ejemplos similares en el conjunto de datos de entrenamiento (similitud basada en *embeddings*) o diseñar *prompts* específicos que guíen al modelo hacia explicaciones más informativas (estrategias de *prompting*).

- **Puntuaciones por atribución:** como parte de la interpretabilidad post-hoc, se están desarrollando técnicas[122] que permiten identificar qué partes del texto de entrada tienen mayor influencia en la respuesta generada por un LLM. Ayudan a entender qué palabras o frases son más importantes para el modelo. Existen diferentes métodos para calcular estas puntuaciones:

  - Métodos basados en el gradiente: analizan cómo cambian los gradientes (una medida de sensibilidad) para cada palabra al retroceder por la red neuronal.

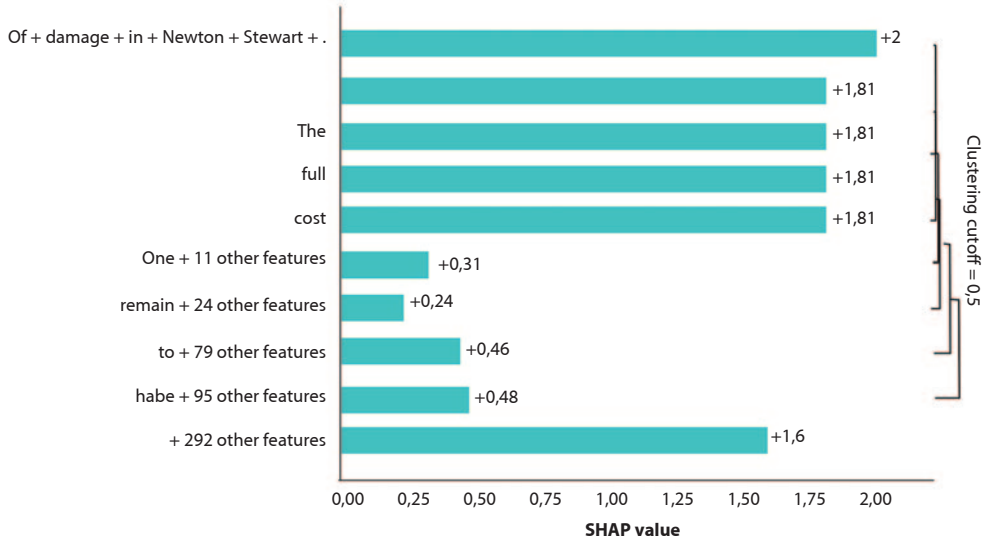
  - Métodos basados en perturbaciones: modifican ligeramente el texto de entrada y observan cómo cambia la respuesta del modelo.

  - Interpretación de métricas internas: utilizan métricas calculadas por el propio modelo, como los pesos de atención en los *transformers*, para determinar la importancia de cada palabra.

[120] Management Solutions (2023). Explainable Artificial Intelligence.
[121] Wang (2024).
[122] Sarti (2023).

Fig. 12. Implementación de valores de SHAP para resumen de textos.

Resumen de la salida: "Todavía se está evaluando el coste total de los daños en Newton Stewart, una de las zonas más afectadas . La Primera Ministra, Nicola Sturgeon, visitó la zona para inspeccionar los daños. El líder adjunto del Partido Laborista Escocés, Alex Rowley, estuvo el lunes en Hawick para ver la situación de primera mano. Afirmó que era importante aplicar correctamente el plan de protección contra las inundaciones".



Un ejemplo de puntuación por atribución es la aplicación de la técnica SHAP para proporcionar una medida cuantitativa de la importancia de cada palabra para la salida del LLM, lo que facilita su interpretación y comprensión (Fig. 12).

- **Validación continua y monitorización en producción:** más allá de la evaluación puntual antes del despliegue, se extiende la práctica de realizar un seguimiento continuo del comportamiento de los LLM una vez que están en uso, al igual que se hace con modelos tradicionales. Esto permite detectar posibles desviaciones o degradaciones en su rendimiento a lo largo del tiempo, así como identificar sesgos o riesgos no previstos inicialmente.

- **Validación colaborativa y participativa:** se promueve una mayor implicación de diversos stakeholders en el proceso de validación, incluyendo no solo a expertos técnicos sino también a usuarios finales, reguladores, auditorías externas y representantes de la sociedad civil. Esta participación plural permite incorporar diferentes perspectivas y fomenta la transparencia y la responsabilidad.

- **Validación ética y alineada con la regulación:** más allá de las métricas de rendimiento, se otorga cada vez más importancia a evaluar si el comportamiento de los LLM es ético y está alineado con los valores humanos y con la regulación. Esto implica analizar cuestiones como la equidad, la privacidad, la seguridad, la transparencia o el impacto social de estos sistemas.

- **Machine unlearning:** se trata de una técnica emergente[123] que permite "desaprender" información conocida de un LLM sin reentrenarlo desde cero. Esto se consigue, por ejemplo, adaptando los hiperparámetros del modelo a los datos que se desea desaprender. Se puede usar el mismo principio para eliminar los sesgos que se hayan identificado. Así, se obtiene un modelo que mantiene su conocimiento general, pero ha eliminado los sesgos problemáticos, mejorando su equidad y alineación ética de forma eficiente y selectiva. Actualmente se están explorando varios métodos de machine unlearning, como el *gradient 4scent*[129], el uso de *fine-tuning*[125] o la modificación selectiva de determinados pesos, capas o neuronas del modelo[126].

[123] Liu (2024).
[124] Jang (2022).
[125] Yu (2023).
[126] Wu (2023)

SHAP (SHapley Additive exPlanations) aplicado a un LLM

SHAP es un método de explicabilidad *post-hoc* basado en la teoría de juegos cooperativos. Asigna a cada característica (token) un valor de importancia (valor Shapley) que representa su contribución a la predicción del modelo.

Formalmente, sea x = (x_1,…,x_n) una secuencia de tokens de entrada. La predicción del modelo se denota como f(x). El valor Shapley φ para el token x_i se define como:

$$\phi_i = \sum_{\{S \subseteq N_i\}} \frac{\{|S|!\,(n-|S|-1)!\}}{\{n!\}} [f(S \cup \{i\}) - f(S)]$$

donde N es el conjunto de todos los tokens, S es un subconjunto de tokens, y f(S) es la predicción del modelo para el subconjunto S.

Intuitivamente, el valor Shapley φi captura el impacto promedio del token xi en la predicción del modelo, considerando todos los subconjuntos posibles de tokens.

Ejemplo: se considera un LLM entrenado para clasificar correos electrónicos corporativos como "importante" o "no importante". Dado el vector de tokens de entrada:

x = [El, informe, financiero, del, Q2, muestra, un, aumento, significativo, en, los, ingresos, y, la, rentabilidad]

El modelo clasifica el correo como "importante" con f(x) = 0.85.

Aplicando SHAP, se obtienen los siguientes valores Shapley:

φ_1 = 0.01 (El)

φ_2 = 0.2 (informe)

φ_3 = 0.15 (financiero)

φ_4 = 0.02 (del)

φ_5 = 0.1 (Q2)

φ_6 = 0.05 (muestra)

φ_7 = 0.01 (un)

φ_8 = 0.15 (aumento)

φ_9 = 0.1 (significativo)

φ_{10} = 0.01 (en)

φ_{11} = 0.02 (los)

φ_{12} = 0.12 (ingresos)

φ_{13} = 0.01 (y)

φ_{14} = 0.02 (la)

φ_{15} = 0.08 (rentabilidad)

Interpretación: los tokens "informe" (0.2), "financiero" (0.15), "aumento" (0.15) e "ingresos" (0.12) tienen las mayores contribuciones a la clasificación del correo como "importante". Esto sugiere que el LLM ha aprendido a asociar estos términos con la importancia del mensaje en un contexto empresarial.