CAPTUM Y XAI: IDEA GENERAL

Todos los métodos de Captum buscan responder la **misma pregunta**:

"¿Qué tanto contribuye cada entrada (píxel, feature o token) al resultado del modelo?"

La diferencia entre métodos está en **cómo calculan esa "contribución"**.
Algunos usan derivadas (gradientes), otros comparan activaciones con una referencia (*baseline*), y otros integran ambos enfoques.

Saliency Maps (Gradiente simple)

Intuición: mide *cuánto cambia la salida si alteramos ligeramente el input*.

 Fórmula mental:

"Si muevo un píxel un poquito, ¿cuánto cambia la probabilidad de la clase?"

💡 Es como ver *la sensibilidad local* del modelo.
Si el gradiente en una zona es alto → el modelo "reacciona" fuerte ahí.

📸 En imágenes: destaca los bordes o trazos donde el modelo es más sensible.
⚠️ Limitación: puede ser ruidoso o poco estable, porque depende de una sola derivada (sin promedio).

Integrated Gradients (IG)

Intuición: mide *la contribución acumulada* de cada input al pasar de una baseline (por ejemplo, imagen en negro) al input real.

 Fórmula mental:

"¿Qué tanto se activa el modelo mientras voy pintando gradualmente la imagen desde cero hasta completa?"

💡 Es como encender una imagen poco a poco y medir cómo sube la confianza del modelo.
La suma de los efectos por píxel da la **atribución total**.

📸 En imágenes: muestra zonas que **efectivamente construyen la evidencia** para la clase.
✅ Estable y teóricamente sólido (cumple propiedades como *completitud*).
⚙️ Requiere definir una buena *baseline* (lo "neutro").

DeepLIFT (Deep Learning Important FeaTures)

Intuición: compara cuánto cambia cada neurona (o input) respecto a una baseline, **no por gradiente sino por diferencia de activaciones**.

🧮 Fórmula mental:

"¿Cuánto se activó este píxel comparado con lo que habría pasado si fuera neutro?"

💡 Más robusto que los gradientes puros (no se anula en saturaciones).
Usa una idea parecida a IG, pero computacionalmente más rápida.
Se propaga *deltas* (cambios) capa por capa hacia atrás.

📸 En imágenes: genera mapas más limpios y menos dependientes del ruido.
🧮 En datos tabulares: útil cuando hay valores de referencia (ej. 0 o media).

Gradient SHAP

Intuición: combina *SHAP values* con *gradientes*.
En lugar de muestrear todas las combinaciones posibles como SHAP, estima el efecto de cada feature promediando gradientes con pequeñas perturbaciones.

🧮 Fórmula mental:

"Tomo varios puntos entre la baseline y el input, aplico ruido y gradientes, y promedio los efectos."

💡 Se parece a IG pero con *muestreo estocástico* → más suave, más interpretable.

📸 Resultado: heatmaps más suaves que Saliency, pero con mayor costo de cálculo.

Layer Attribution (Layer Grad-CAM, Layer IG, etc.)

Intuición: mide qué **capas internas o filtros** activan más la predicción.

 Fórmula mental:

"¿Qué parte de la red 'vio' la evidencia más fuerte para esta clase?"

💡 En CNNs: muestra qué regiones de una imagen activan los mapas de características internos.
📸 Ideal para entender *qué "ve" cada capa* (útil para enseñanza o depuración de redes profundas).

COMPARATIVO INTUITIVO

Método	Idea central	Usa gradientes	Usa baseline	Resultados típicos	Analogía
Saliency	Sensibilidad inmediata	✅	❌	Ruidosos, pero rápidos	Termómetro instantáneo
Integrated Gradients	Contribución acumulada	✅	✅	Suaves, interpretables	Encender imagen poco a poco
DeepLIFT	Diferencia de activaciones	❌	✅	Limpios y rápidos	Comparación antes/después
Gradient SHAP	Gradientes promediados	✅	✅	Suaves, estables	SHAP "con esteroides"
Layer Attribution / Grad-CAM	Foco por capas	✅	Opcional	Mapas visuales	"Dónde mira el modelo"