AYI NEDJIMI Consultants

Sparse Autoencoders et Interprétabilité Mécanistique

Catégorie : Intelligence Artificielle Lecture : 15 min Publié le : 28/02/2026 Auteur : Ayi NEDJIMI

Techniques d'interprétabilité mécanistique pour auditer les décisions internes des LLM : sparse autoencoders, circuit analysis, probing et feature...

Table des Matières

Votre organisation est-elle prête à faire face aux attaques basées sur l'IA ? Techniques d'interprétabilité mécanistique pour auditer les décisions internes des LLM : sparse autoencoders, circuit analysis, probing et feature... Dans un contexte où l'intelligence artificielle transforme les pratiques de cybersécurité, la maîtrise de ia sparse autoencoders interpretabilite devient un avantage stratégique pour les équipes techniques. Nous abordons notamment : table des matières, 1 introduction : pourquoi ouvrir la boîte noire des llm et 2 qu'est-ce que

l'interprétabilité mécanistique. Les professionnels y trouveront des recommandations actionnables, des commandes prêtes à l'emploi et des stratégies de mise en œuvre adaptées aux environnements d'entreprise.

1 Introduction : Pourquoi ouvrir la boîte noire des LLM

Les **grands modèles de langage** (LLM) sont devenus des composants critiques de l'infrastructure numérique mondiale. GPT-4o traite des millions de requêtes médicales quotidiennes, Claude assiste des cabinets juridiques dans l'analyse de contrats, et Gemini aide les ingénieurs à écrire du code déployé en production. Pourtant, ces systèmes restent fondamentalement des **boîtes noires** : nous savons ce qu'ils font (prédire le prochain token), mais nous comprenons mal pourquoi ils le font. Cette opacité n'est plus acceptable en 2026, alors que les régulateurs (AI Act européen, NIST AI RMF) exigent une **transparence et une auditabilité** des systèmes d'IA déployés dans des contextes à haut risque.

L'**interprétabilité mécanistique** (mechanistic interpretability) est le domaine de recherche qui vise à comprendre les mécanismes computationnels internes des réseaux de neurones — non pas simplement observer quels inputs produisent quels outputs, mais comprendre *comment* le modèle transforme l'information à travers ses couches, ses têtes d'attention et ses neurones pour arriver à une décision. Cette discipline, longtemps cantonnée aux laboratoires de recherche académique, connaît en 2025-2026 une accélération spectaculaire grâce à une percée technique majeure : les **sparse autoencoders (SAE)**.

Les SAE permettent de décomposer les représentations internes d'un LLM en **features interprétables** — des unités sémantiques compréhensibles par un humain. Là où un neurone individuel encode typiquement un mélange confus de concepts (phénomène de *superposition*), un SAE peut extraire des features correspondant à des concepts nets : "ce texte parle de la Tour Eiffel", "l'auteur exprime du sarcasme", "le code contient une vulnérabilité SQL". Cette capacité à lire dans les pensées du modèle ouvre des perspectives bouleversants pour la **sécurité, l'audit et la gouvernance** des systèmes d'IA.

Enjeu fondamental : L'interprétabilité mécanistique n'est pas un exercice académique mais une nécessité opérationnelle. En 2026, elle permet de détecter des **backdoors dans les modèles**, de vérifier que les guardrails de sécurité fonctionnent comme prévu, d'identifier les biais systémiques encodés dans les représentations internes, et de comprendre pourquoi un modèle produit des hallucinations dans des contextes spécifiques.

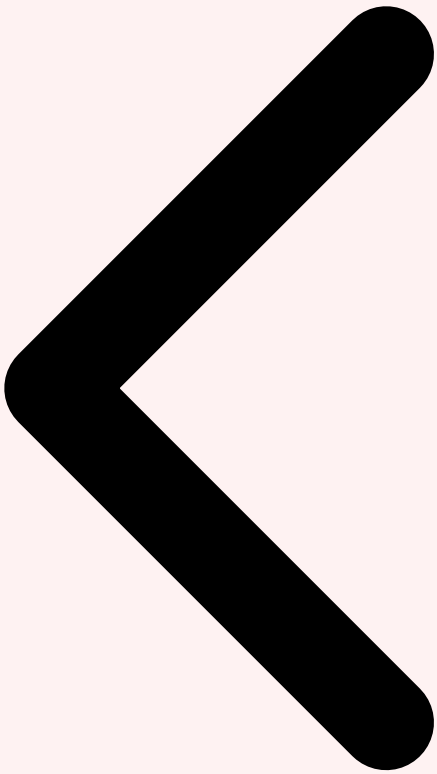
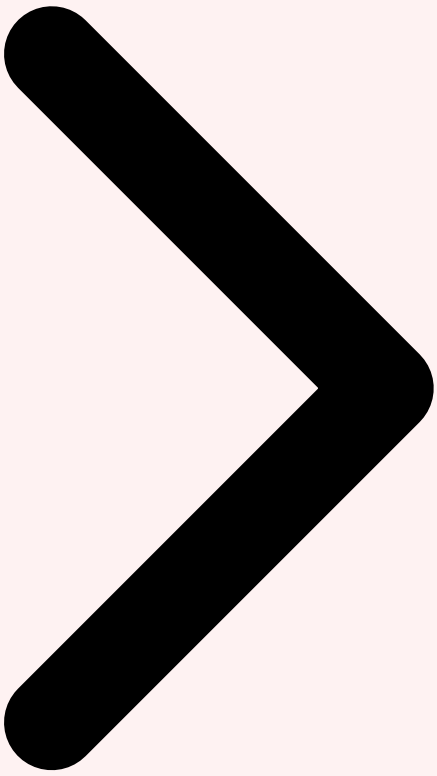


Table des Matières Introduction Interprétabilité

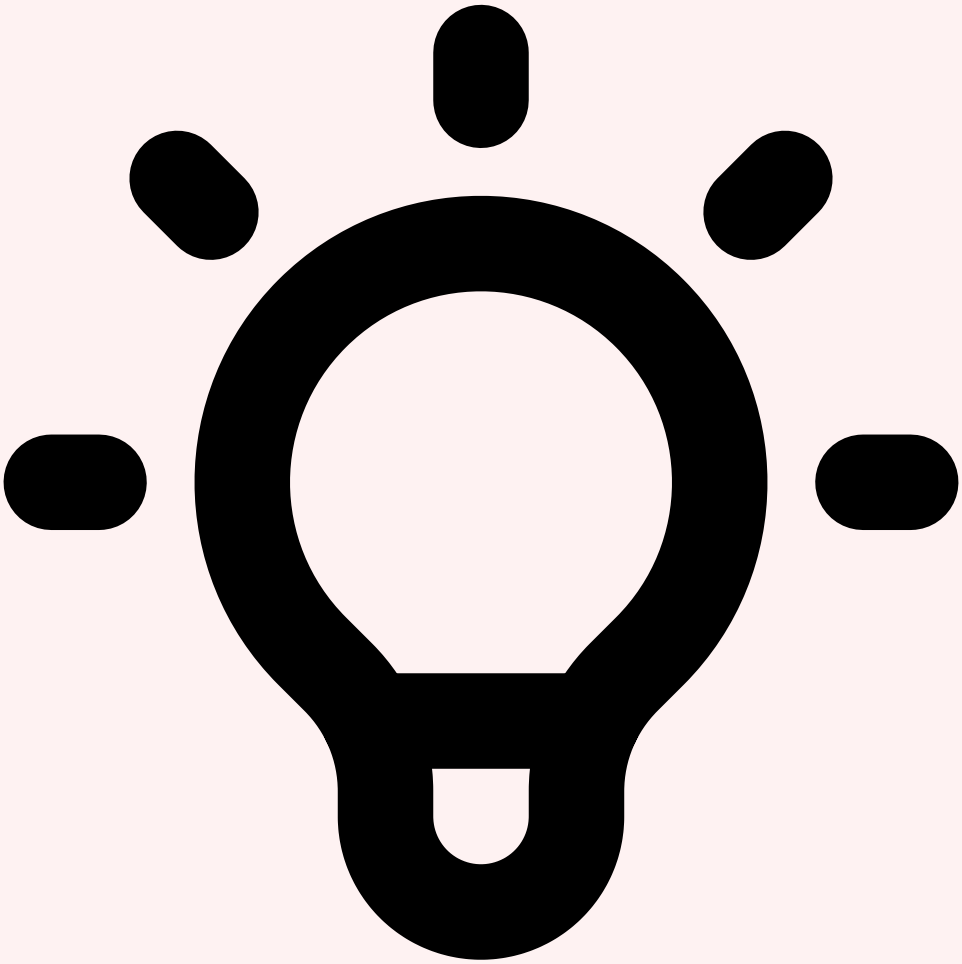


Critere	Description	Niveau de risque
Confidentialite	Protection des donnees d'entrainement et des prompts	Eleve
Integrite	Fiabilite des sorties et detection des hallucinations	Critique
Disponibilite	Resilience du service et gestion de la charge	Moyen
Conformite	Respect du RGPD, AI Act et politiques internes	Eleve

2 Qu'est-ce que l'interprétabilité mécanistique

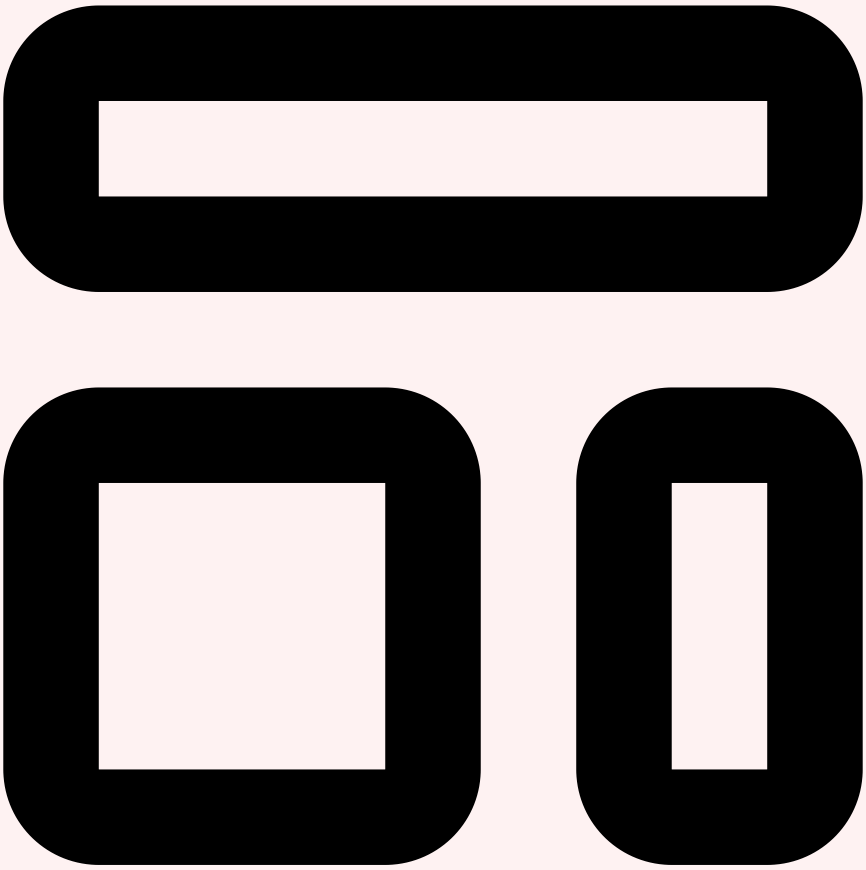
L'interprétabilité mécanistique se distingue fondamentalement des approches d'explicabilité traditionnelles (LIME, SHAP, attention visualization) qui opèrent au niveau des **inputs et outputs** du modèle. Ces méthodes, dites *post-hoc*, traitent le modèle comme une boîte noire et tentent d'approximer son comportement par des modèles plus simples. L'interprétabilité mécanistique, en revanche, ouvre la boîte noire et examine directement

les **mécanismes computationnels internes** — les circuits de neurones, les patterns d'attention, les transformations de représentation — qui produisent le comportement observé.



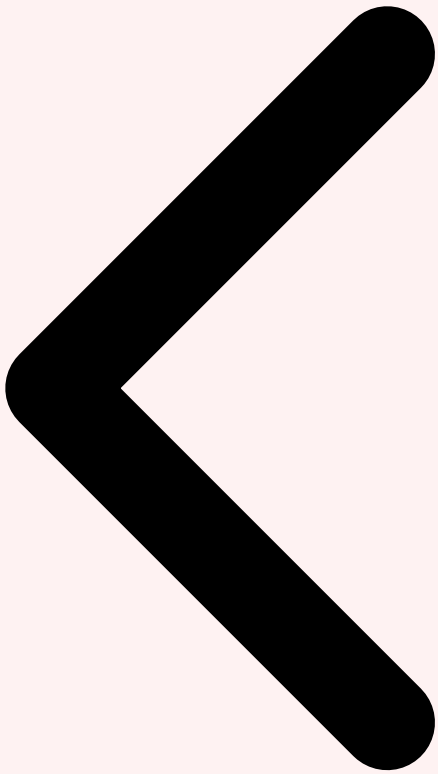
Le problème de la superposition

Le défi central de l'interprétabilité mécanistique est le phénomène de **superposition** (superposition hypothesis). Un LLM représente bien plus de concepts que le nombre de dimensions dans ses couches cachées. Un modèle avec des couches de 4096 dimensions encode potentiellement des centaines de milliers de features sémantiques distinctes. Comment est-ce possible ? Par superposition : chaque neurone participe à l'encodage de **multiples concepts simultanément**, de manière similaire à la manière dont un hologramme encode une image 3D dans un support 2D. Un seul neurone peut être partiellement activé par "textes en français", "questions médicales", et "expressions formelles" simultanément. Cela rend l'interprétation directe des neurones individuels pratiquement impossible — et c'est précisément le problème que les sparse autoencoders résolvent.

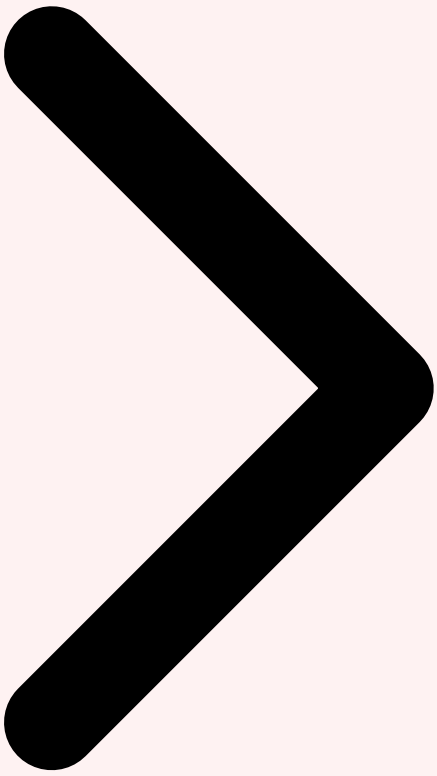


Résidu streams et architecture des transformers

Pour comprendre l'interprétabilité mécanistique, l'architecture des transformers sous l'angle du **residual stream**. Dans un transformer, chaque token est représenté par un vecteur qui traverse l'ensemble des couches du modèle via des connexions résiduelles. Chaque couche (attention + MLP) lit depuis le residual stream, effectue une computation, et écrit le résultat dans le residual stream. Ce flux résiduel constitue le **bus de communication principal** du modèle — toute l'information pertinente transite par ce vecteur. Les têtes d'attention implémentent des opérations de copie d'information entre positions (par exemple, copier l'information sur le sujet grammatical vers la position du verbe), tandis que les couches MLP effectuent des transformations non linéaires qui encodent des connaissances factuelles et des règles de raisonnement. L'interprétabilité mécanistique cherche à comprendre **quelles informations** sont encodées dans le residual stream à chaque point, et **quelles opérations** chaque composant du modèle effectue sur ces informations. Pour approfondir, consultez PLAM : Agents IA Personnalisés Edge et Déploiement Sécurisé.



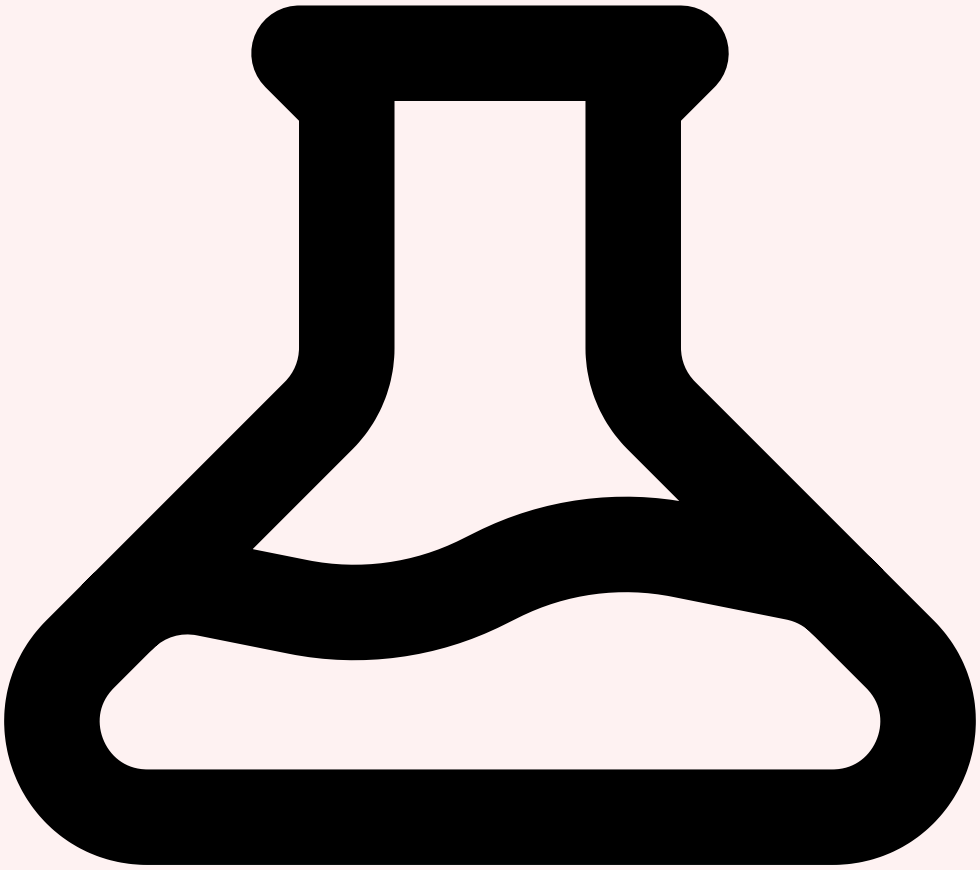
Introduction Interprétabilité Mécanistique Sparse Autoencoders



Notre avis d'expert
Chez Ayi NEDJIMI Consultants, nous constatons que la majorité des organisations sous-estiment les risques liés aux modèles de langage déployés en production. La sécurité des LLM ne se limite pas au prompt engineering : elle exige une approche systémique couvrant les embeddings, les pipelines de données et les mécanismes de contrôle d'accès aux API.

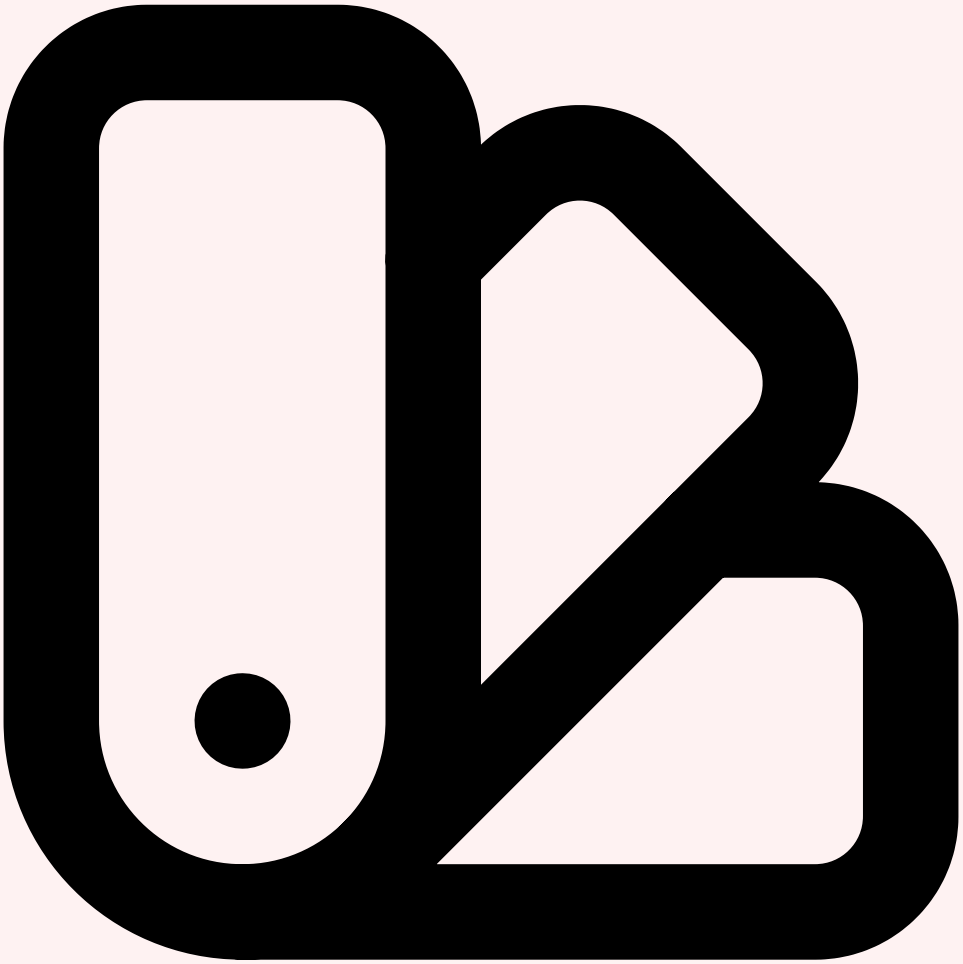
3 Sparse Autoencoders (SAE)

Les **sparse autoencoders** (SAE) sont la percée technique qui a transformé l'interprétabilité mécanistique d'un domaine de recherche théorique en un outil pratique d'audit de modèles. Publiés simultanément par Anthropic (Bricken et al., 2023) et des chercheurs indépendants, les SAE résolvent le problème de la superposition en apprenant à décomposer les activations d'un modèle en **features monosémantiques** — chaque feature correspondant à un concept unique et interprétable.



Architecture et fonctionnement mathématique

Un SAE est un autoencoder avec deux propriétés distinctives : un **hidden layer sur-dimensionné** (typiquement 8x à 64x la dimension d'entrée) et une **contrainte de sparsité** forte. L'encodeur projette les activations du modèle depuis l'espace original (par exemple, 4096 dimensions) vers un espace beaucoup plus grand (par exemple, 131072 features), puis une fonction d'activation (ReLU ou TopK) assure que seule une petite fraction des features est active pour chaque input. Le décodeur reconstruit les activations originales à partir de cette représentation sparse. L'entraînement minimise la perte de reconstruction tout en maximisant la sparsité, forçant chaque feature à capturer un concept distinct et réutilisable. La contrainte de sparsité est cruciale : elle empêche les features d'encoder des mélanges de concepts comme le font les neurones du modèle original. Le résultat est un **dictionnaire de features** où chaque entrée correspond idéalement à un concept sémantique unique.



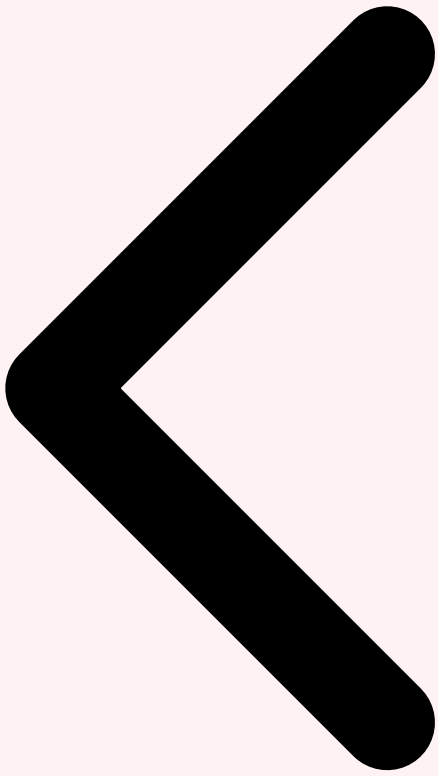
Résultats majeurs d'Anthropic et OpenAI

Les résultats publiés par **Anthropic** en 2024 sur Claude 3 Sonnet ont constitué un tournant pour le domaine. En entraînant un SAE sur les activations du résidu médian du modèle, les chercheurs ont extrait des **millions de features interprétables** couvrant un spectre extraordinaire de concepts : des features correspondant à des entités spécifiques (la Golden Gate Bridge, le code Python, les erreurs grammaticales en français), des concepts abstraits (le sarcasme, la tromperie, la sécurité), et des patterns structurels (début de liste, fin de paragraphe, transition argumentative). Plus remarquable encore, la manipulation causale de ces features — en augmentant ou supprimant artificiellement l'activation d'une feature — modifie de manière prévisible le comportement du modèle. Activer fortement la feature "Golden Gate Bridge" fait que Claude mentionne le pont dans chaque réponse. Supprimer une feature liée à la sécurité affaiblit les refus du modèle face à des requêtes dangereuses. Ces résultats démontrent que les features SAE ne sont pas de simples corrélations statistiques mais des **composants causaux** du fonctionnement du modèle.

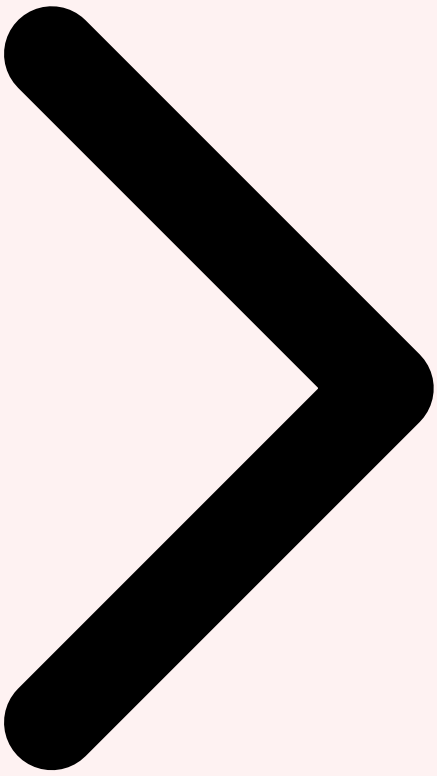
Comment garantir que vos modèles de machine learning ne deviennent pas des vecteurs d'attaque ?

Considerations pratiques avancees

OpenAI a publié des résultats comparables sur GPT-4 en 2024, utilisant des SAE avec 16 millions de features et confirmant la scalabilité de l'approche aux modèles les plus grands. Leurs travaux ont particulièrement mis en évidence la capacité des SAE à identifier des features liées à la **tromperie et à la manipulation** — des features qui s'activent lorsque le modèle génère du texte factuellement incorrect tout en le présentant avec confiance. Cette découverte ouvre la voie à des systèmes de détection d'hallucinations basés sur l'inspection des représentations internes plutôt que sur la vérification externe des faits. En 2026, les SAE sont entraînés sur les principaux modèles open source (Llama 3.1, Mistral, Gemma 2) et les dictionnaires de features sont partagés comme des ressources communautaires sur des plateformes dédiées.



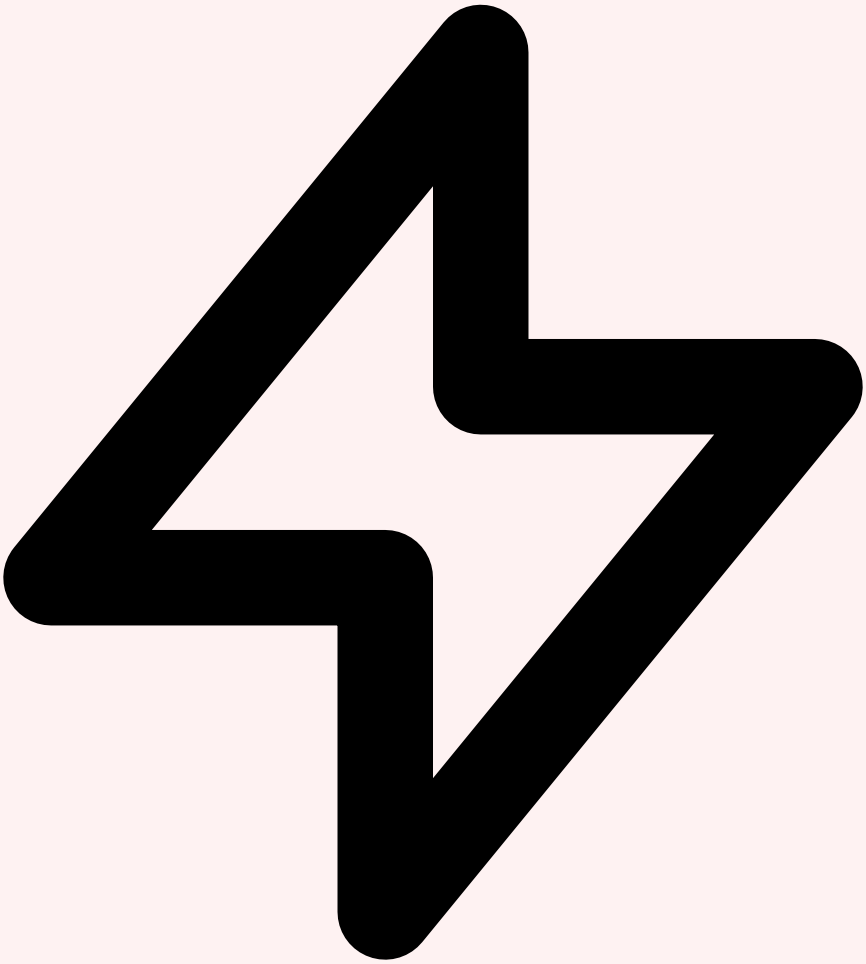
Interprétabilité Sparse Autoencoders Circuit Analysis



Cas concret
En février 2024, une entreprise de Hong Kong a perdu 25 millions de dollars après qu'un employé a été trompé par un deepfake vidéo lors d'une visioconférence. Les attaquants avaient recréé l'apparence et la voix du directeur financier à l'aide de modèles d'IA générative, démontrant les risques concrets de cette technologie en contexte corporate.

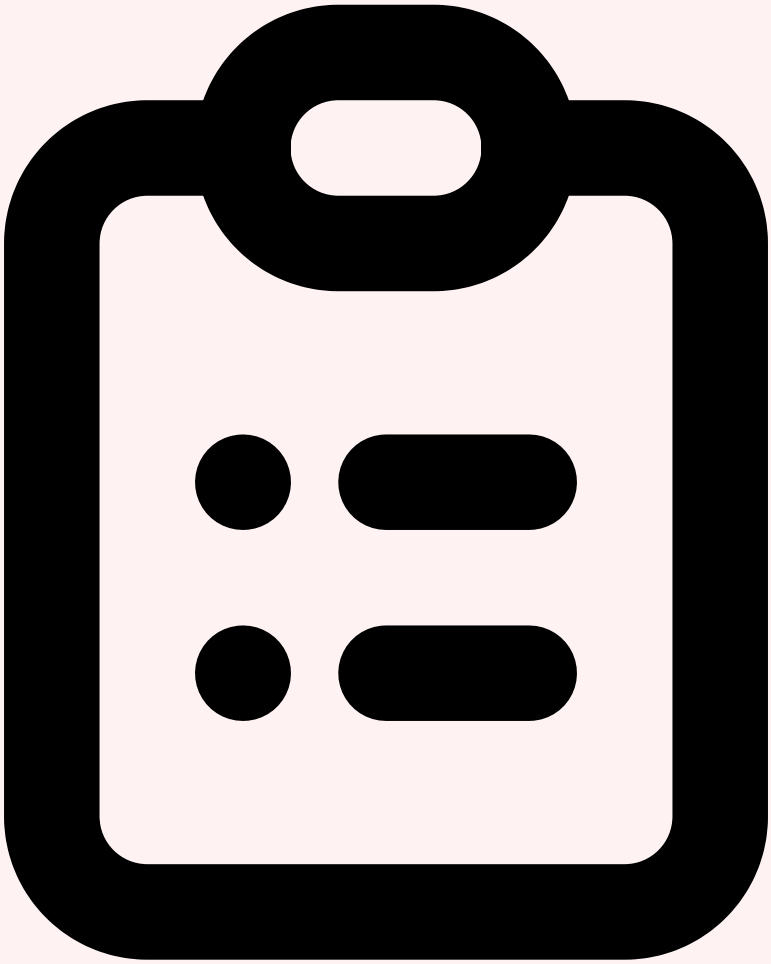
4 Circuit analysis et probing

Au-delà de l'identification de features individuelles, l'interprétabilité mécanistique cherche à comprendre comment ces features **interagissent entre elles** pour produire des comportements complexes. L'analyse de circuits (circuit analysis) et le probing sont les deux méthodologies complémentaires qui permettent cette compréhension structurelle.



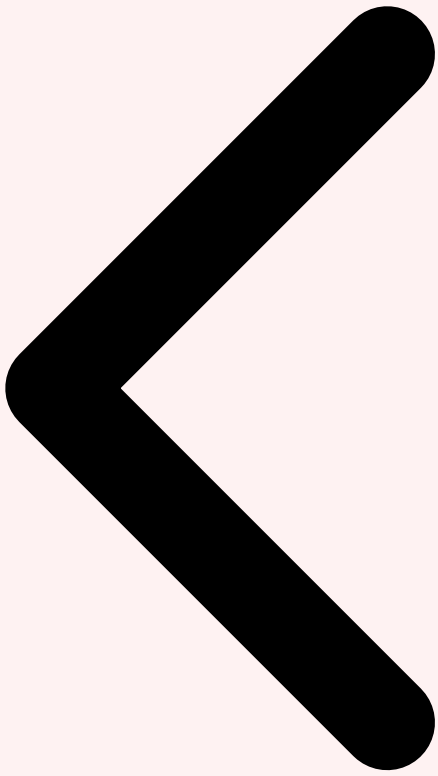
Circuit discovery et analyse causale

L'**analyse de circuits** consiste à identifier les sous-réseaux minimaux — appelés circuits — responsables d'un comportement spécifique du modèle. La méthodologie utilise l'**ablation causale** (path patching, activation patching) : en désactivant sélectivement des composants du modèle (têtes d'attention, neurones MLP, connexions spécifiques) et en observant l'impact sur le comportement cible, on peut identifier les composants essentiels et ceux qui sont redondants. Par exemple, le **circuit d'induction** (induction circuit) — l'un des premiers circuits identifiés dans les transformers — est un circuit à deux couches qui implémente la copie de patterns : si le modèle a vu "AB...A", le circuit prédit "B" comme prochain token. Ce circuit repose sur des **têtes d'attention précédentes** (previous token heads) qui copient l'information du token précédent et des **têtes d'induction** (induction heads) qui utilisent cette information pour prédire. La découverte de circuits similaires pour des comportements plus complexes — raisonnement factuel, détection de sentiment, génération de code — est l'objectif de la recherche actuelle en 2026.

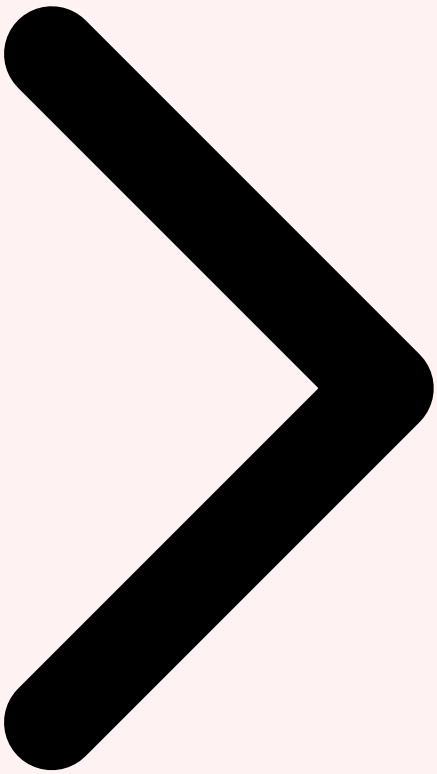


Linear probing et représentations internes

Le **probing** est une technique complémentaire qui consiste à entraîner de petits classificateurs linéaires (probes) sur les activations intermédiaires du modèle pour déterminer quelles informations sont encodées à chaque couche. Par exemple, on peut entraîner un probe pour détecter si le modèle "sait" que le texte contient une erreur factuelle, en classifiant les activations de chaque couche comme "factuel" ou "non-factuel". Si le probe atteint une haute accuracy à partir d'une certaine couche, cela signifie que l'information est représentée de manière **linéairement décodable** dans le residual stream — le modèle a encodé cette connaissance de manière structurée. Le probing a révélé que les LLM encodent des informations remarquablement riches dans leurs représentations internes : la structure syntaxique des phrases, les relations sémantiques entre entités, la véracité des affirmations, le registre de langue, et même des représentations spatiales et temporelles d'un monde modélisé. Ces découvertes sont directement exploitables pour l'audit de sécurité : un probe détectant les features de "confiance injustifiée" dans les couches tardives pourrait servir de **détecteur d'hallucinations en temps réel**. Pour approfondir, consultez Gouvernance Globale de l'IA 2026 : Alignement International.

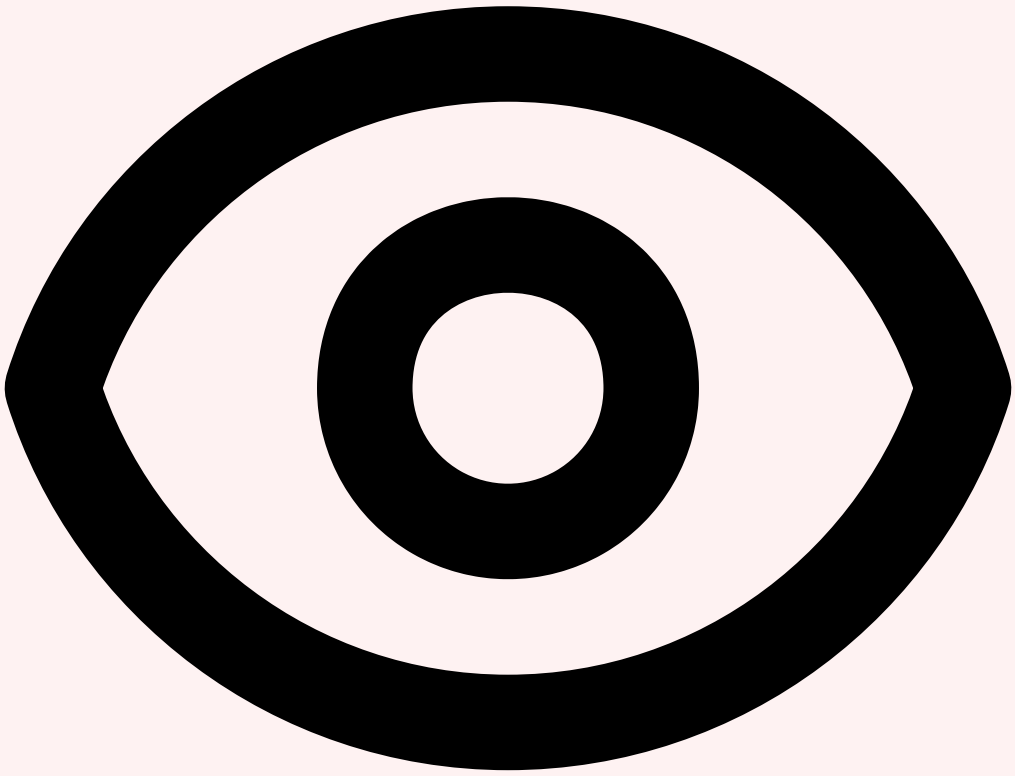


Sparse Autoencoders Circuit Analysis Feature Visualization



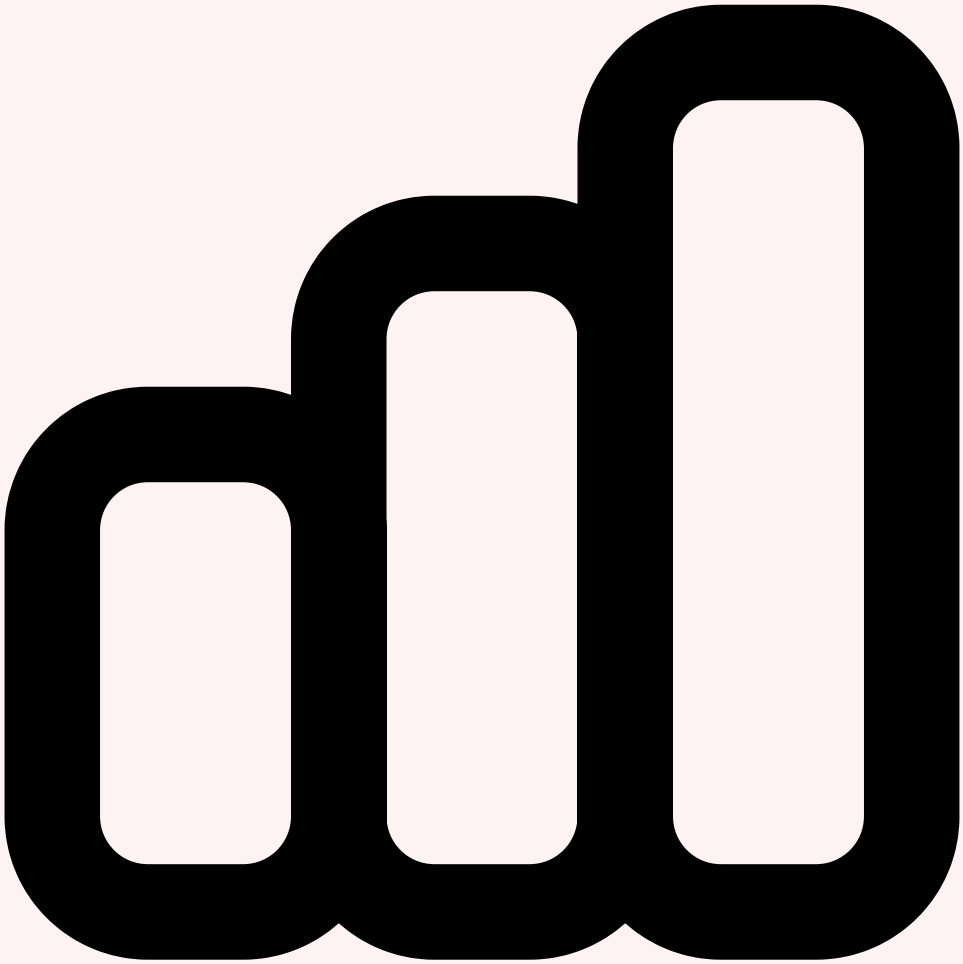
5 Feature visualization

La **visualisation de features** est le processus qui rend les découvertes de l'interprétabilité mécanistique accessibles et exploitables par des auditeurs humains. Elle transforme les données mathématiques abstraites (vecteurs de poids, activations, corrélations) en représentations visuelles et textuelles interprétables.



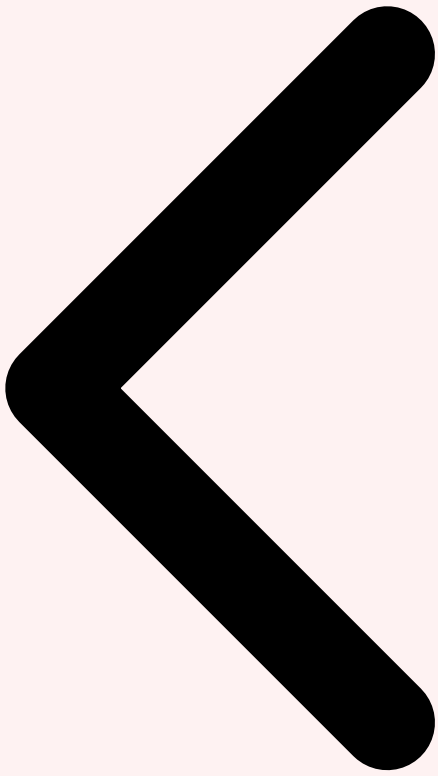
Max activating examples et feature dashboards

La méthode la plus directe pour comprendre une feature SAE est d'examiner ses **max activating examples** — les textes du corpus qui activent le plus fortement cette feature. Pour chaque feature, on collecte les 20-50 exemples avec la plus haute activation et on identifie visuellement le concept commun. Anthropic a publié **Neuronpedia**, une plateforme interactive permettant d'explorer des millions de features SAE avec leurs exemples d'activation, leurs descriptions auto-générées, et des métriques de qualité. Les **feature dashboards** modernes intègrent plusieurs vues complémentaires : la distribution des activations (fréquence et intensité), les contextes d'activation (avant/après le token déclencheur), les corrélations avec d'autres features (features qui co-activent), et l'impact causal sur la sortie du modèle (comment la manipulation de cette feature modifie les prédictions). Ces dashboards permettent à un auditeur de comprendre en quelques minutes ce qu'une feature encode, quand elle s'active, et quel est son impact sur le comportement du modèle.

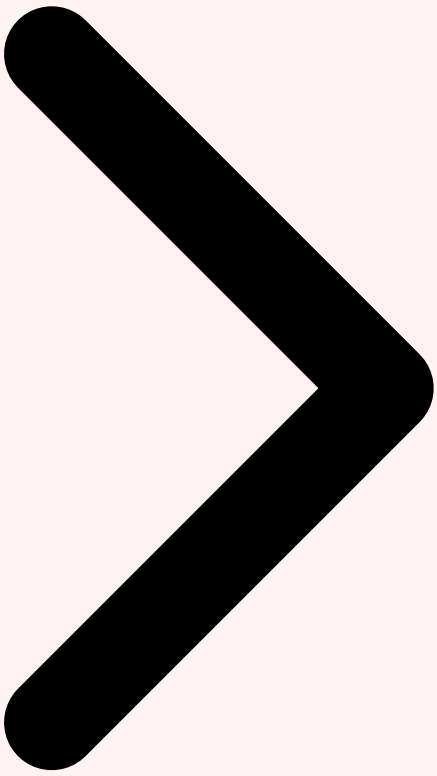


Logit lens et visualisation des prédictions couche par couche

Le **logit lens** est une technique de visualisation qui projette le residual stream à chaque couche intermédiaire vers l'espace de vocabulaire final (via la matrice d'unembedding) pour observer comment les prédictions du modèle **évoluent à travers les couches**. On peut ainsi voir qu'aux couches précoces, le modèle prédit des tokens sémantiquement proches mais grammaticalement incorrects, et qu'aux couches tardives, la prédiction se raffine vers le token syntaxiquement et sémantiquement approprié. Le **tuned lens**, une extension qui ajoute une projection apprise par couche, améliore la fidélité de cette visualisation. Ces outils sont essentiels pour comprendre *quand* le modèle prend ses décisions : la détection de toxicité se fait-elle aux couches précoces ou tardives ? Le modèle "sait-il" qu'il hallucine avant de générer le token erroné ? Ces questions, auxquelles le logit lens et ses extensions permettent de répondre, ont des implications directes pour la conception de guardrails plus efficaces.

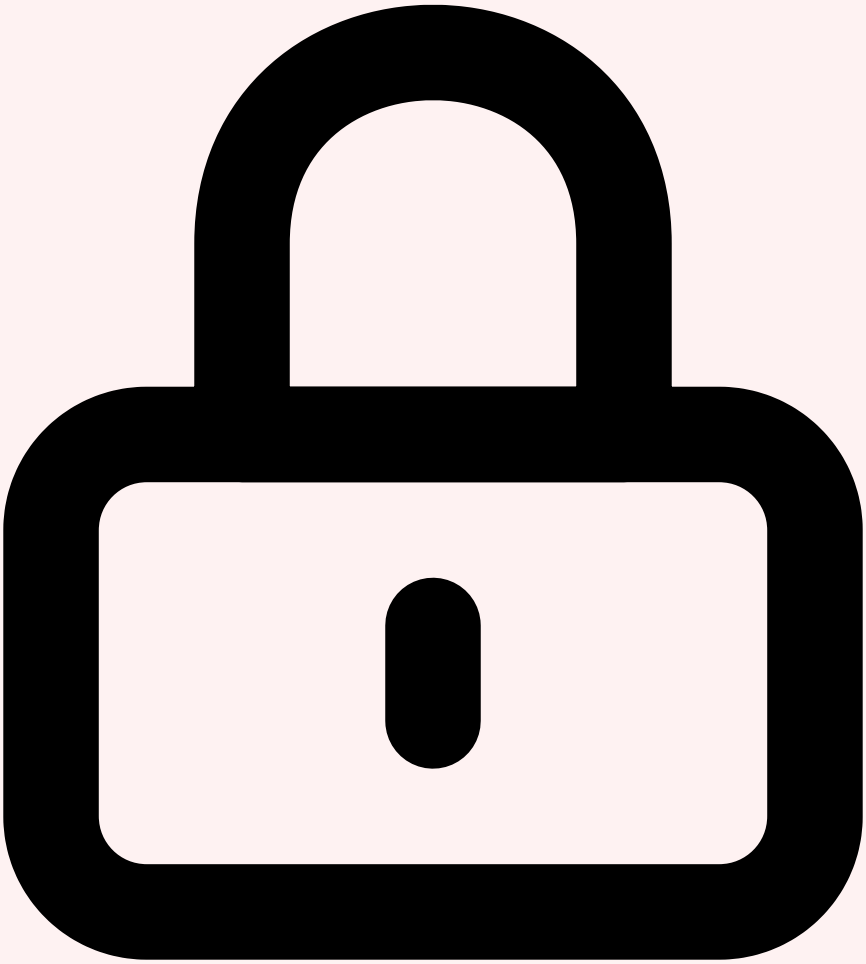


Circuit Analysis Feature Visualization Audit de Modèles



6 Applications en audit de modèles

Les techniques d'interprétabilité mécanistique ne sont plus confinées aux laboratoires de recherche. En 2026, elles trouvent des applications concrètes dans l'**audit de sécurité, la conformité réglementaire et la gouvernance** des modèles d'IA déployés en production.



Détection de backdoors et comportements cachés

Les SAE permettent de scanner un modèle pour détecter des **features anormales ou suspectes**. Un modèle backdooré (via data poisoning lors du fine-tuning) contient nécessairement des features qui encodent le trigger de la backdoor et le comportement malveillant associé. En analysant systématiquement les features SAE et en identifiant celles dont le pattern d'activation est inhabituel (activations très rares mais très fortes, corrélation avec des patterns de trigger), un auditeur peut détecter des backdoors qui seraient invisibles aux tests comportementaux classiques. De manière similaire, les features SAE peuvent révéler des **comportements déceptifs appris** — features qui s'activent lorsque le modèle "décide" de produire une réponse trompeuse plutôt qu'honnête. La capacité à inspecter les intentions internes du modèle, plutôt que simplement observer ses outputs, constitue un changement de cadre dans l'audit de sécurité IA.

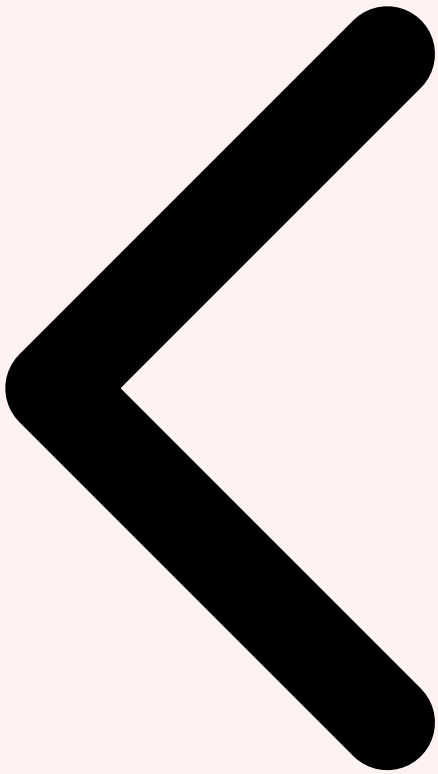
Détection de biais et conformité réglementaire

L'interprétabilité mécanistique permet de **détecter et quantifier les biais** encodés dans les représentations internes des modèles. En examinant les features SAE liées au genre, à l'ethnicité, à l'âge ou à d'autres caractéristiques protégées, un auditeur peut déterminer comment ces concepts interagissent avec les features de compétence, de confiance, de risque et de décision. Par exemple, si la feature "candidat masculin" co-active systématiquement avec la feature "compétence technique élevée" dans un modèle utilisé pour le screening de CV, cela révèle un biais de genre qui serait difficile à détecter par des tests comportementaux limités. Cette capacité d'**audit interne** est directement pertinente pour la conformité à l'AI Act européen (Article 10 sur la qualité des données et les biais) et pour les évaluations d'impact sur les droits fondamentaux requises pour les systèmes d'IA à haut risque.

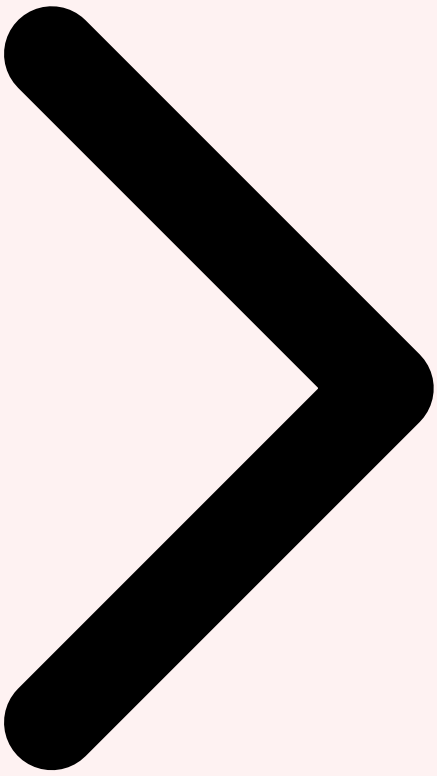
Applications pratiques de l'interprétabilité mécanistique en 2026 : Pour approfondir, consultez La Fin des Moteurs.

- ✓**Détection de backdoors :** scanning des features SAE pour identifier des triggers et comportements cachés

- ✓**Audit de biais :** analyse des corrélations entre features protégées et features de décision
- ✓**Détection d'hallucinations :** monitoring des features de confiance vs. features de connaissance
- ✓**Vérification de guardrails :** inspection des circuits de refus pour confirmer leur robustesse
- ✓**Forensique post-incident :** analyse des activations internes pour comprendre pourquoi un modèle a produit un output dangereux

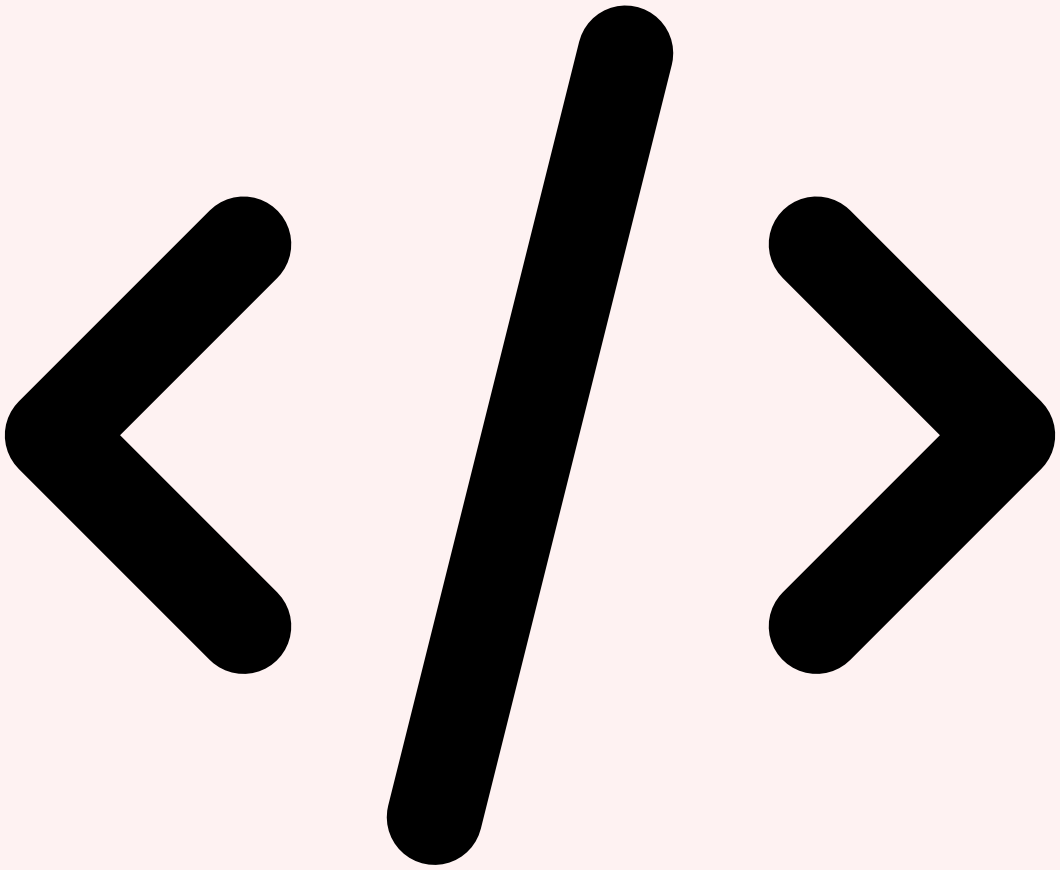


Feature Visualization Audit de Modèles Outils



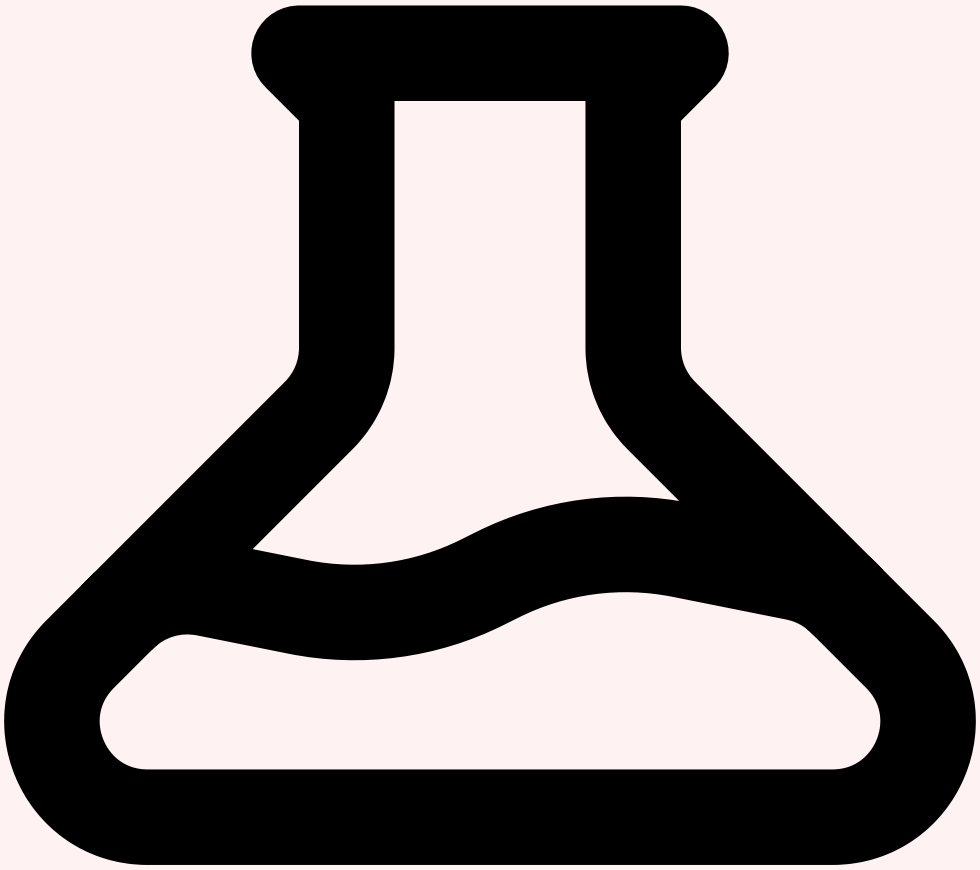
7 Outils : TransformerLens, SAELens

L'écosystème d'outils pour l'interprétabilité mécanistique s'est considérablement structuré en 2025-2026, avec des bibliothèques Python matures qui rendent ces techniques accessibles aux praticiens de la sécurité IA.



TransformerLens : le scalpel du mécaniste

TransformerLens est la bibliothèque Python de référence pour l'interprétabilité mécanistique des transformers. Développée par Neel Nanda (DeepMind) et maintenue par la communauté, elle fournit une implémentation propre et instrumentée des architectures transformer standard (GPT-2, GPT-Neo, Llama, Mistral, Gemma, Pythia) avec des **hooks d'accès à toutes les activations intermédiaires**. TransformerLens permet d'accéder au residual stream, aux activations pré/post attention, aux patterns d'attention, aux sorties des couches MLP, et à tout point intermédiaire du forward pass. Les fonctionnalités clés incluent l'**activation patching** (remplacement des activations à un point du réseau par celles d'un autre input pour identifier les composants causaux), le **logit lens** (projection des résidus intermédiaires vers l'espace de vocabulaire), et le **direct logit attribution** (décomposition de la contribution de chaque composant à la prédiction finale). La bibliothèque s'intègre nativement avec PyTorch et supporte l'exécution sur GPU pour les modèles de grande taille.

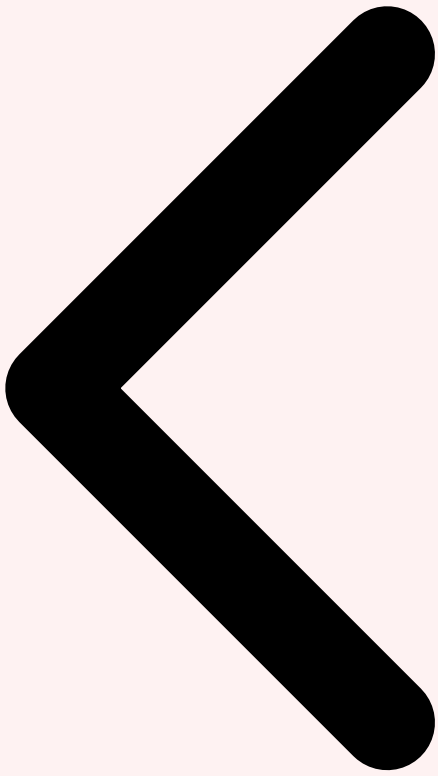


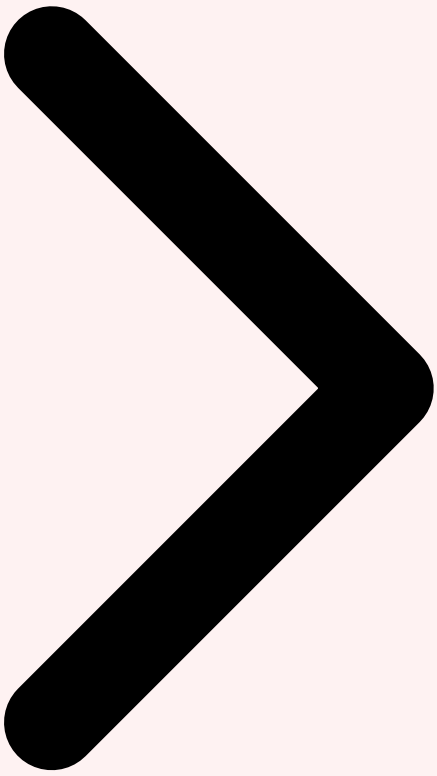
SAELens : entraînement et analyse de sparse autoencoders

SAELens (anciennement mats_sae_training) est la bibliothèque spécialisée dans l'entraînement, l'évaluation et l'analyse de sparse autoencoders pour les LLM. Développée par Joseph Bloom et la communauté d'interprétabilité, elle s'intègre directement avec TransformerLens et fournit des implémentations optimisées des principales architectures de SAE : **vanilla SAE** (ReLU + pénalité L1), **TopK SAE** (activation des K features les plus actives), et **Gated SAE** (avec un mécanisme de gating appris). SAELens inclut des métriques d'évaluation standardisées pour la qualité des SAE : la **loss de reconstruction** (fidélité de la reconstruction des activations), le **L0** (nombre moyen de features actives par input), la **feature density** (distribution de la fréquence d'activation des features), et les **dead features** (features qui ne s'activent jamais). La bibliothèque fournit également des outils de visualisation et d'exploration des features entraînées, facilitant l'identification de features sémantiquement significatives parmi des millions de candidates.

- ▹**TransformerLens :** accès aux activations intermédiaires, activation patching, logit lens — le couteau suisse du mécaniste
- ▹**SAELens :** entraînement et évaluation de SAE (vanilla, TopK, Gated) avec métriques standardisées

- ▹**Neuronpedia :** plateforme interactive d'exploration des features SAE avec dashboards et descriptions auto-générées
- ▹**CircuitsVis :** bibliothèque de visualisation des circuits d'attention et des patterns d'activation
- ▹**Baukit / pyvene :** framework de manipulation causale des activations pour l'analyse d'intervention





8 Conclusion et perspectives

L'**interprétabilité mécanistique** a franchi un cap décisif en 2025-2026. Les sparse autoencoders ont transformé ce qui était un domaine de recherche fondamentale en un ensemble d'outils pratiques pour l'audit et la gouvernance des LLM. La capacité à extraire des features monosémantiques, à identifier des circuits causaux et à visualiser les représentations internes des modèles ouvre des perspectives considérables pour la sécurité des systèmes d'IA.

Les applications pratiques sont déjà tangibles : détection de backdoors dans les modèles fine-tunés, audit de biais pour la conformité AI Act, détection d'hallucinations par inspection des représentations internes, et vérification de la robustesse des guardrails de sécurité. La maturité de l'écosystème d'outils — TransformerLens, SAELens, Neuronpedia — rend ces techniques accessibles aux praticiens de la sécurité IA sans nécessiter une expertise de recherche en machine learning.

Les défis restent néanmoins significatifs. La **scalabilité** aux modèles les plus grands (>100B paramètres) reste coûteuse en ressources computationnelles. La **complétude** de l'interprétation est loin d'être garantie : même avec des millions de features SAE, nous ne comprenons qu'une fraction des mécanismes internes des LLM. Et la **fidélité** des SAE — leur capacité à capturer véritablement les features pertinentes plutôt que des artefacts statistiques — nécessite des recherches méthodologiques continues. Néanmoins, la trajectoire est claire : l'interprétabilité mécanistique est en passe de devenir un composant standard de la boîte à outils d'audit et de gouvernance de tout système d'IA déployé en production. Les organisations qui investissent dès maintenant dans ces compétences seront les mieux préparées pour répondre aux exigences réglementaires et sécuritaires de demain.

Besoin d'un accompagnement expert ?

Nos consultants en cybersécurité et IA vous accompagnent dans vos projets d'audit et d'interprétabilité des modèles. Devis personnalisé sous 24h. Pour approfondir, consultez Speculative Decoding et Inférence Accélérée : Techniques 2026.

Références et ressources externes

- OWASP LLM Top 10 — Les 10 risques majeurs pour les applications LLM
- MITRE ATLAS — Framework de menaces pour les systèmes d'intelligence artificielle
- NIST AI RMF — AI Risk Management Framework du NIST
- arXiv — Archive ouverte de publications scientifiques en IA
- HuggingFace Docs — Documentation de référence pour les modèles de ML

Pour approfondir ce sujet, consultez notre outil open-source ai-threat-detection qui facilite la détection de menaces basée sur l'IA.

Sources et références : ArXiv IA · Hugging Face Papers

FAQ

Qu'est-ce que Sparse Autoencoders et Interprétabilité Mécanistique ?

Le concept de Sparse Autoencoders et Interprétabilité Mécanistique est détaillé dans les premières sections de cet article, qui couvrent les fondamentaux, les enjeux et le contexte opérationnel. Pour un accompagnement sur ce sujet, contactez nos experts.

Pourquoi Sparse Autoencoders et Interprétabilité Mécanistique est-il important en cybersécurité ?

La compréhension de Sparse Autoencoders et Interprétabilité Mécanistique permet aux équipes de sécurité d'améliorer leur posture défensive. Les sections « Table des Matières » et « 1 Introduction : Pourquoi ouvrir la boîte noire des LLM » détaillent les raisons de cette importance. Pour un accompagnement sur ce sujet, contactez nos experts.

Comment mettre en œuvre les recommandations de cet article ?

Les recommandations pratiques sont détaillées tout au long de l'article, avec des commandes, des outils et des méthodologies éprouvées. La section « Conclusion » fournit une synthèse actionnable. Pour un accompagnement sur ce sujet, contactez nos experts.

Conclusion

Cet article a couvert les aspects essentiels de Table des Matières, 1 Introduction : Pourquoi ouvrir la boîte noire des LLM, 2 Qu'est-ce que l'interprétabilité mécanistique. La mise en pratique de ces recommandations permet de renforcer significativement la posture de securite de votre organisation.

Ayi NEDJIMI Consultants — Expert cybersécurité offensive & intelligence artificielle

ayinedjimi-consultants.fr · ayi@ayinedjimi-consultants.fr