

Interprétabilité Mécaniste pour l'Extraction d'Information dans les systèmes de RAG

Marine DELVALLEZ

20 août 2026

Remerciements

Table des matières

1	Introduction	4
1.1	Environnement du stage	4
1.2	Outils utilisés durant le stage	5
1.3	Contexte et problématiques traitées dans ce mémoire	5
1.3.1	Contexte	5
1.3.2	Problématiques qui ont guidé ce travail	6
2	État de l’art	7
2.1	Systèmes de Génération et Grands Modèles de Langue Augmentés par Extraction (RAG et RALLM)	7
2.1.1	Besoin et Définition	7
2.1.2	L’architecture des modèles RA-LLM	7
2.1.3	Architectures aperçues au cours du stage	9
2.1.4	Architecture et Modèles utilisées au cours du stage	9
2.2	Interprétabilité pour les systèmes de RAG	10
2.2.1	Trustworthy AI	10
2.2.2	Explicabilité pour le RAG	10
2.2.3	Interprétabilité Mécaniste	10
2.2.4	Activation Patching	11
3	Travail préalable et préparation des expériences	14
3.1	Challenge RAG@EvalLLM	14
3.2	Construction d’un dataset adapté	14
3.3	Adaptation du modèle RAG4HN	15
3.4	Objectifs	17
3.4.1	Localisation de la langue de spécialité dans un modèle	17
3.4.2	Intégration des travaux dans l’explicabilité post-hoc des modèles d’extraction d’information .	17
4	Contribution méthodologique	18
4.1	Construction des perturbations	18
4.1.1	Identification des mots cibles	18
4.1.2	Perturbation replace par niveau	18
4.1.3	Perturbations de type full_replace	18
4.2	Estimation de la pertinence d’une perturbation	18
4.3	Interprétation des résultats de l’Activation Patching	19
4.3.1	Visualisation des résultats proposée par MechIR	19
4.3.2	Autres visualisations de la sensibilité du modèle à la perturbation	19
4.3.3	Critère d’identification des noeuds sensibles	19
4.3.4	Matrice des noeuds sensibles	19
5	Résultats expérimentaux	20
5.1	Résultats et Analyse	20
5.1.1	Application à la cartographie du vocabulaire de la défense français dans mE5	20
5.1.2	Préconisations d’utilisation	20
5.2	Perspectives d’évolution de la méthode proposée	20

6	Conclusions et Perspectives	21
A	Résultats bruts d'expériences	24
A.1	Question Comme Document	24
A.1.1	Représentation et similarité de la question paddées/pertubée	24
A.1.2	Différence des scores de pertinence pour question paddée et question perturbée	25
A.2	Séparer les données positives et négatives	26
A.2.1	Distribution des Scores	26
A.2.2	Scores pour les perturbations full_replace	27
A.3	Niveaux de perturbation	28
A.3.1	Distribution des scores en fonction du niveau de perturbation	28

Chapitre 1

Introduction

1.1 Environnement du stage

- Stage à lieu au LIFO, dans l'équipe CA, encadré par ...
- Stage de fin de master d'informatique Minerve GPEx
- Challenge@EvalLLM comme cas d'usage, participation envisagée mais timing pas OK
- différents événements et sorties organisées pendant le stage
 - TransIA : Tours, Présentations pluridisciplinaires sur l'intégration de l'IA (sous toutes ses formes) dans la société
 - TAL-I : journée du GDR TaL-I à Paris (Jussieu) sur les travaux portant sur l'interprétabilité et l'explicabilité dans le contexte des tâches de traitement de la langue naturelle
 - Frugal
 - 2nd Symposium on Mental Health and AI : deux jours de conférence (suivi en pointillé en distanciel) présentation des travaux à la croisée entre santé mentale et intelligence artificielle (impact, outils de détection, de prise en charge)
 - Place du Numérique : événement public de sensibilisation et de communication sur le Numérique organisé place du Martoi à Orléans (Des événements similaires organisés dans chaque département de la région CVL) J'ai participé à la tenue du stand de l'université, été ambassadrice de l'événement et ai joué le rôle d'un témoin dans le tribunal des générations futures portant sur "L'IA nous contrôle-t-elle?"
- Présentation aux journées d'équipe CA

Ce mémoire restitue les travaux et réflexions que j'ai porté à l'occasion de mon stage de fin de master ARIAS, parcours Minerve GPEx au sein de l'équipe Contraintes et Apprentissage du Laboratoire d'Informatique Fondamentale d'Orléans (LIFO) au cours duquel j'ai été encadré par Mme Halftermeyer. Au cours de ces six mois de stage et en parallèle de mon travail au sein du LIFO, j'ai pu suivre et participer aux événements scientifiques et de dissémination suivants :

Le colloque pluridisciplinaire **Sociétés en transition à l'ère de l'intelligence artificielle** (TransIA) portait sur les impacts de l'Intelligence Artificielle sur la société et les différents pans de la recherche. Il regroupait des présentations issues des différentes disciplines allant des Sciences de l'éducation au droit privé et public en passant par l'étude de l'impact social et environnemental de l'IA. Il m'a permis d'aborder la recherche au sujet de l'intelligence artificielle sous l'angle de son impact et d'avoir un regard plus ouvert sur les autres disciplines de recherche et leur application à l'IA.

La journée du Groupe de Recherche **TAL-I** regroupait des présentations et une session poster portant sur l'interprétabilité dans le contexte du Traitement Automatique des Langues Naturelles. Cette journée organisée à Paris m'a permis de compléter ma découverte du paysage de l'explicabilité pour le TALN effectuée durant mon immersion au semestre précédent et d'échanger avec d'autres chercheurs sur mon projet individuel ainsi qu'à propos de leur travaux.

J'ai aussi eu l'occasion de suivre en distanciel les présentations de la **Journée Frugalité en TAL et ML** qui portaient sur les enjeux et les impacts environnementaux et sociaux du développement du TAL et du ML et les solutions actuellement étudiées pour les prendre en compte et les réduire. Ces présentations m'ont permis de prendre

conscience de l'importance des ses enjeux et mieux comprendre quels sont les leviers pour les prendre en compte dans des travaux de recherche en IA.

Le **Second Symposium on Mental Health and AI** regroupait des présentations de travaux à la croisée entre l'intelligence artificielle, les sciences cognitives et la psychologie. J'ai pu suivre certaines présentations en lien avec le projet collaboratif que nous avons présenté avec Eva Troupillon au semestre précédent dans le cadre du module Défi Appel à Proposition de Projet.

À l'occasion des **journées de l'équipe CA**, j'ai pu découvrir les travaux en cours au sein de l'équipe. J'y ai moi-même présenté les premières contributions méthodologiques et les difficultés auxquelles je faisais face dans le cadre de mon stage. J'ai pu prendre conscience de l'importance des échanges et de la collaboration au sein de la recherche et de la capacité de ces échanges à faire prendre du recul sur ses travaux et permettre d'avancer.

[Évoquer dès maintenant le challenge ?](#)

[Ajouter les séminaires du lundi](#)

En dehors des événements internes au monde de la recherche et de l'informatique, j'ai aussi pu marrainer la **Place du Numérique** organisée à Orléans le 9 juin dernier. Cet événement vise à faire connaître et rendre accessible la diversité du monde du numérique dont la recherche en informatique fait partie. J'ai notamment joué le rôle de témoin dans le *tribunal des générations futures* questionnant l'influence de l'intelligence artificielle sur notre comportement.

1.2 Outils utilisés durant le stage

- Développement de l'architecture et réalisation des expérimentations en python dans 3 formats (script, implémentation d'un CLI, notebook)
- Principales librairies python utilisées : PyTorch, TransformerLens, HuggingFace, PyMuPDF4LLM,...
- Utilisation de plusieurs git pour assurer le suivi des versions des outils développés lors du stage
- Pour l'exécution de tâches demandantes en charge de calcul (Génération de la base de représentation des données, exécution des différentes architectures explorées et/ou implémentées) : utilisation des grappes de calcul accessibles aux chercheurs du LIFO CaSciModOT et Mirev
- utilisation d'outils à base d'IA très modérée et pour les tâches de l'ordre de reformulation (ChatGPT), traduction (DeepL), exploration de la littérature (LitMaps, Perplexity), rédaction de petites fonctions et outils python (ChatGPT)

1.3 Contexte et problématiques traitées dans ce mémoire

1.3.1 Contexte

- Intro RAG :
 - les modèles ont des soucis pour gérer la connaissance (hallucination, préemption des modèles)
 - on a couplé les modèles génératifs avec une base de connaissance entretenue et maîtrisée par concept
- Intro XAI
 - modèles opaques
 - besoin de confiance
 - méthode d'explications pour rendre comportement accessible et anticipable
- méthodes qui ouvrent la boîte, observent et donnent des explications
- On souhaite cartographier ces modèles
- On choisit un outil
- On découvre les difficultés et problématiques à l'usage
- On propose une méthodologie pour exploiter cet outil dans l'optique de cartographier la langue de spécialité

[à reprendre](#)

- Objectif naïf : Identifier les composants du modèle impliqués dans l'identification et la gestion de la langue de spécialité d'un dataset
- Objectif pratique : Identifier et proposer des préconisations d'usage de la librairie MechIR pour la localisation des composants impliqués dans la langue de spécialité

- Définition de langue de spécialité, réduction de l'objectif à l'identification du vocabulaire de spécialité

1.3.2 Problématiques qui ont guidé ce travail

TODO

Chapitre 2

État de l'art

2.1 Systèmes de Génération et Grands Modèles de Langue Augmentés par Extraction (RAG et RALLM)

2.1.1 Besoin et Définition

Les modèles d'apprentissage profond tels que les réseaux de neurones profond, réseaux convolutionnels ou les réseaux basés sur le mécanisme d'attention constituent leur connaissances à partir des données utilisées pour les entraîner. Cependant, leurs connaissances et leur capacité d'extrapolation sont limitées. Cela rend ces modèles vulnérables à l'obsolescence des résultats et aux hallucinations [Huang et al., 2025]. Pour traiter ces problématiques, [Lewis et al., 2020] propose d'associer une base de donnée et un modèle d'extraction des données de cette base à un modèle génératif. La base de donnée permet d'entretenir plus facilement les connaissances du modèle obtenu afin de limiter l'obsolescence des résultats et/ou agrandir la base de connaissance selon le besoin ou la tâche visée. Les systèmes combinant un modèle d'extraction (*retrieval* en anglais) et un modèle génératif sont appelées systèmes RAG (Retrieval Augmented by Generation). La question ou requête de l'utilisateur est fournie au modèle d'extraction qui récupère les documents ou extraits de documents pertinent au regard de la question. Ces documents ainsi que la question sont alors donnés au générateur qui produit le résultat final. Ce processus est schématisé dans le Figure 2.1

Les systèmes RAG peuvent être appliqués à de nombreuses tâches générant différents types de données (génération d'image, génération et/ou complétion de signaux, génération de texte, traduction,...). Lorsque le modèle génératif est un Grand Modèle de Langue (LLM), on parle de systèmes RA-LLM [Fan et al., 2024]. Ces modèles sont utilisées pour grande diversité de tâches du traitement automatique des langues. On retrouve notamment les différentes variantes de réponse aux questions (questions à forte intensité de raisonnement, question sur domaine ouvert,...), les tâches d'analyse de texte (vérification de fait, liage d'entité, détection d'entité nommées), des tâches de génération ou de complétion (traduction, prédiction de citation, dialogue, résumé de texte, complétion de texte à trous,...) ou encore les tâches d'extraction (extraction d'information, extraction d'argument, ...)

2.1.2 L'architecture des modèles RA-LLM

Cette sous-section est un résumé de [Fan et al., 2024]

L'intuition de compléter un modèle génératif avec une base de donnée comme proposé par [Lewis et al., 2020]

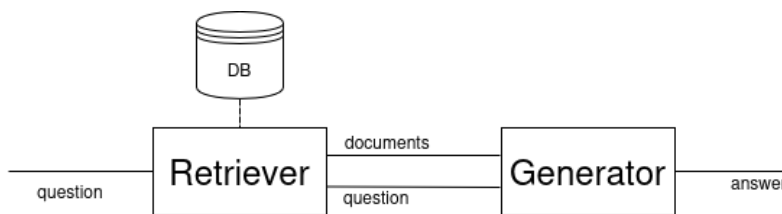


FIGURE 2.1 – Schema de l'architecture RAG proposée par [Lewis et al., 2020]

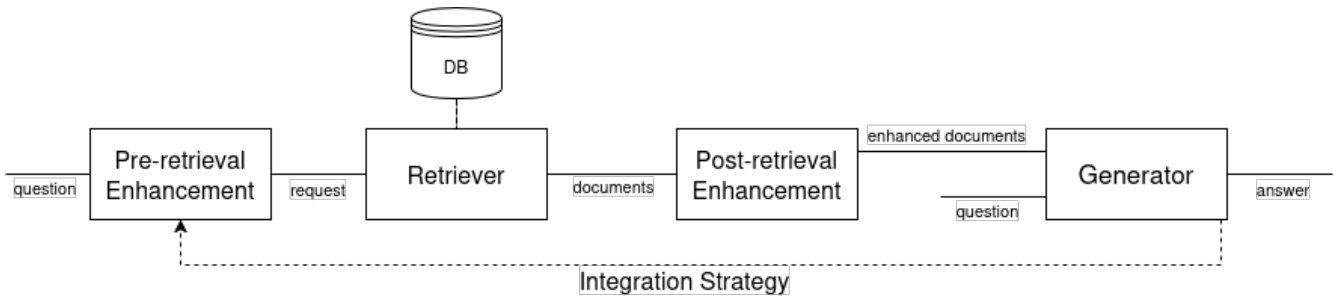


FIGURE 2.2 – Architecture généralisée des systèmes RA-LLM proposée par [Fan et al., 2024]

a été reprise à plusieurs reprises. Des stratégies et éléments complémentaires ont été ajoutés à ces systèmes pour faciliter la symbiose entre les deux composants principaux. [Fan et al., 2024] propose une nouvelle généralisation de cette famille de modèle illustré par la Figure 2.2

Extracteur On peut distinguer les extracteurs (aussi appelés retriever) en deux groupes. Les *Sparse retrievers* tels que TF-IDF ou BM25 [Sparck Jones, 1972, Lù, 2024] exploitent des statistiques sur les documents pour identifier les plus pertinents au regard de la question. Les *Dense retrievers* projettent les documents et les questions dans un espace latent au sein duquel les documents pertinents sont identifiés. On retrouve deux principaux fonctionnements pour ces extracteurs. Les *bi-encodeurs* projettent les documents et les questions séparément alors que les *cross-encodeurs* les projettent sous forme de paires (question, document) et fournissent directement le score de pertinence. Un hyper-paramètre important pour les extracteurs est la granularité d'extraction. Il correspond à la taille des extraits répertoriés dans la base de donnée. Les échelles les plus courantes sont celles du document, du passage et du token. On parle de *chunk retrieval* pour les deux premières et de *token retrieval* pour la troisième. On parle d'*entity retrieval* lorsque la base est construite sous l'angle de la connaissance et non celui du langage.

Augmentation pré-extraction Pour permettre une meilleure exploitation des informations disponibles dans la question, certains travaux proposent différentes stratégies de complétion et/ou reformulation de la question sous forme d'une requête. Le framework Rewrite-Retrieve-Read [Ma et al., 2023] propose de reformuler la question à l'aide d'un autre LLM pour la désambigüiser. L'approche de Query2Doc [Wang et al., 2023] consiste en générer des pseudo-documents qui pourraient correspondre aux documents pertinents à extraire et de les ajouter à la question avant de donner le tout à l'extracteur. L'objectif est de fournir des indices supplémentaires pour assurer une meilleure pertinence des documents extraits.

Augmentation post-extraction Pour que les documents extraits soient exploités le mieux possible, plusieurs travaux s'intéressent à des techniques d'augmentation de cet ensemble d'extraits. Cela peut passer par le classement des documents du plus pertinent eu moins pertinent, le ré-ajustement du classement lorsqu'il existe, la combinaison des résultats de plusieurs extracteurs, la reformulation et/ou la synthèse des documents extraits.

Générateur [Fan et al., 2024] distingue les générateurs en deux groupes en fonction de l'accessibilité des paramètres : *parameter accessible* pour les modèles locaux modifiables pour lesquels on dispose des paramètres et *parameter inaccessible* pour les modèles dont les paramètres ne sont pas publics, qui sont généralement accessibles par API.

Intégration Une fois les documents récupérés et augmentés, il faut les fournir au générateur. [Fan et al., 2024] distingue trois stratégies d'intégration. L'*intégration en entrée* consiste à combiner les documents et la question (généralement à l'aide d'une template de prompt) avant de la fournir au générateur. L'*intégration en sortie* consiste à prendre en compte le ou les documents à la fin du processus de génération à l'aide d'un processus de fusion. L'*intégration dans les couches intermédiaires* injecte les documents encodés dans les couches intermédiaires du générateur.

Un autre élément à prendre en compte dans l'intégration des documents au cours de la génération est la fréquence et donc le nombre d'extractions effectués pour une question. Le recours à la génération peut se faire avec une fréquence fixée (une seule fois, tout les n token,...) ou être déclenchée au cours de la génération (par un modèle annexe, par la génération d'un token spécifique, en fonction d'un logit,...).

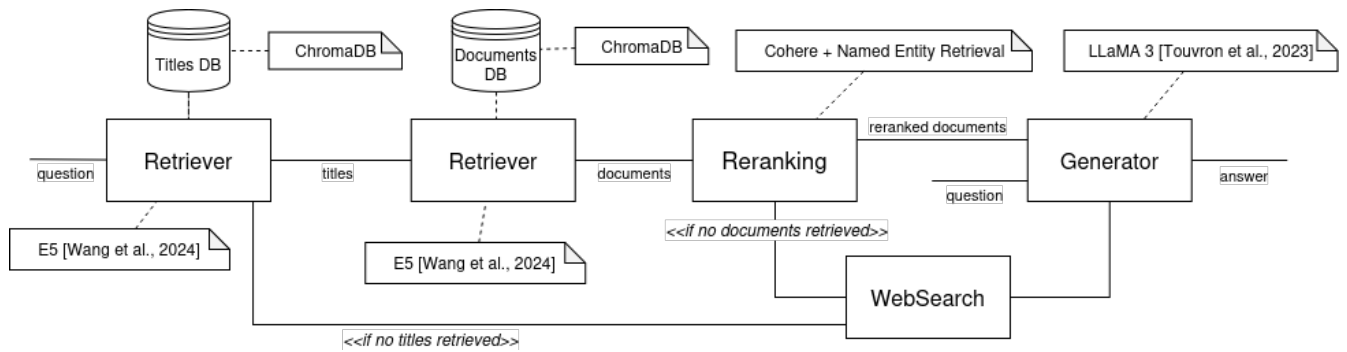


FIGURE 2.3 – Architecture de RAG4HistoricalNewspapers [Tran et al., 2024]

Parler de l'entraînement de ces architectures ?

2.1.3 Architectures aperçues au cours du stage

Au delà des tentatives de perfectionnement des systèmes RAG par la conception de nouvelles approches d'amélioration des entrées et des extraits et de nouvelles techniques d'intégration, cette architecture est aussi concernée par les problématiques de la gestion des données multimodales ou à flux continu, la résolution de tâches nécessitant un raisonnement long ou poussé et les enjeux d'interprétabilité et d'impact environnemental.

[Chen et al., 2025] propose un système RAG avec une base de données multi-modales où chaque entrée est un couple (image, texte). Les paires extraites à partir de la requête, elle aussi bi-modale, sont ré-ordonnées avant d'être fournies comme contexte au générateur.

Pour traiter les tâches à forte intensité de raisonnement (reasoning intensive tasks), [Xu et al., 2024] propose un système RAG combiné à un modèle de décomposition de raisonnement (Chain-of-Query). La requête de l'utilisateur est décomposée en sous-requêtes résolues une à une. Elles sont en suite vérifiées à l'aide des documents extraits. Le raisonnement est vérifié, complété et éventuellement repris en fonction du score de crédibilité obtenu. Une trace complète du raisonnement est restituée en sortie.

Les systèmes RAG sont aussi intéressants à intégrer à une architecture agentique en tant que source d'information. [Liu et al., 2025] présente un framework Hiérarchique Multi-agent Multi-modal composé d'un agent de décomposition des requêtes utilisateur, d'un agent RAG pour des données structurées, non-structurées et sous forme de graphes et d'un agent de prise de décision construisant la sortie finale.

[Zhu, 2025] s'intéresse au cas particulier des bases de données dynamiques. Lorsque le flux d'apport de donnée est important, le besoin de restructuration de la base de donnée est régulier or le coût calculatoire et énergétique associé est dissuasif. [Zhu, 2025] propose StreamingRAG, une architecture composée d'une base de données dynamique qui maintient un ensemble de prototypes compact représentatif du flux de données tout en évitant sa reconstruction permanente.

2.1.4 Architecture et Modèles utilisées au cours du stage

Dans la suite, nous utilisons une version simplifiée de l'architecture RAG4HistoricalNewspaper conçue par [Tran et al., 2024]. Cette architecture est conçue pour faciliter l'exploration d'archives de journaux historiques. Comme illustré dans le Figure 2.3, un premier extracteur récupère les articles pertinents au regard de leur titre. Ces informations sont utilisées pour orienter l'extraction des extraits articles. Dans le cas où aucun titre ou document ne sont extraits, des ressources sont récupérées via une recherche Web. La première extraction peut être vue comme une amélioration de la requête pour l'extraction des extraits d'articles. Les deux extracteurs utilisent le modèle mE5-small de [Wang et al., 2024] et ChromaDB pour la gestion des données. Après l'extraction, les articles obtenus sont reclassés. Le score utilisé est issu d'une combinaison linéaire entre le score de Cohere et le TF-IDF des titres concaténés avec les étiquettes des entités nommées de la question. Seuls les trois meilleurs documents au dessus d'un certain seuil sont conservés. Ce reclassement est une technique d'amélioration des ressources extraites. Pour finir, les documents et la question initiale sont fournis à travers une prompt template à LLaMa3 [Touvron et al., 2023] qui est utilisé comme générateur.

L'extracteur utilisé dans RAG4HistoricalNewspapers utilise l'encodeur mE5-small. La famille de modèles E5 proposée par [Wang et al., 2022] d'embedding de textes. Leur entraînement est fait par contrastive learning à partir

du dataset CCPairs conçu à l’occasion pour les versions anglophones et un échantillon de datasets multilingues pour les versions multilingues [Wang et al., 2024].

2.2 Interprétabilité pour les systèmes de RAG

Les systèmes RAG et RA-LLM comme une grande partie des modèles profonds actuels sont opaques. Il est difficile voire impossible de suivre leur raisonnement lors de leur utilisation. Dans certains contextes, cette opacité est problématique car les décisions prises peuvent avoir des conséquences importantes. Cette problématique illustre le besoin d’une Intelligence artificielle de confiance.

2.2.1 Trustworthy AI

La définition d’IA de confiance (Trustworthy AI) n’est pas fixée et change souvent en fonction des contextes d’utilisations de cette expression. Dans le cas du Machine Learning, on désigne l’IA de confiance comme l’ensemble des systèmes d’IA dans lesquels il est légitime (dans le sens humainement understandable) d’avoir confiance. [Ni et al., 2025] rappelle les composantes de l’IA de confiance proposés par Le NIST (National Institute of Standards and Technology (USA)) :

Fiabilité Le comportement d’un IA de confiance doit être systématiquement adapté, l’incertitude sur son comportement doit être quantifiable et quantifié et ses capacité de généralisation doivent être robustes.

Intimité Une IA de confiance doit prévenir des risques pour les données des utilisateurs ainsi que pour les données sensibles contenues dans le modèle (issues de l’apprentissage ou de la base de donnée)

Explicabilité Le processus de décision d’une IA de confiance doit être transparent et compréhensible pour l’utilisateur.

Justesse Une IA de confiance doit minimiser les biais.

Justifiabilité Une IA de confiance doit pouvoir justifier son bon comportement et notamment sa capacité à respecter les réglementations

Sécurité Une IA de confiance doit détecter et prévenir les usages malicieux et cherchant à nuire.

Dans la suite, on s’intéresse aux différentes stratégies pour améliorer l’explicabilité des systèmes RAG et RA-LLM.

2.2.2 Explicabilité pour le RAG

Je pensais initialement rédiger cette partie en calquant sur la partie [Ni et al., 2025] correspondante comme je l’ai fait pour la sous-section précédente. Certains éléments de sa version des fait ne me plaisent plus autant donc j’ai préféré une version plus gros grain avec une part de jugement personnel. Je ne sais pas si ce format est adapté pour un état de l’art.

L’explicabilité de l’intelligence artificielle, vise à rendre humainement compréhensible les raisons pour lesquelles un modèle ou système génère une certaine sortie. [Garouani et al., 2024, Bell et al., 2022].

Dans le contexte de l’explicabilité des systèmes RAG, [Ni et al., 2025] propose de distinguer l’explication du modèle d’extraction et l’explication du modèle génératif. De façon plus générale, une approche naïve d’explication des systèmes RAG est de construire des explications pour chaque composant du modèle (extraction, génération, améliorations, ...). La difficulté intervient au moment de mettre en perspective les explications fournies entre elles. [Ni et al., 2025] souligne l’absence de de méthode d’explication pour l’extraction d’information spécifique contexte du RAG ni d’étude de l’application des méthodes génériques à ce cas. La même question se pose pour l’explication des modèle génératifs : les méthodes d’explications classiques sont-elles capable de prendre en compte la présence des autres composants ?

Au delà de l’objectif de transparence de l’explicabilité, [Ni et al., 2025] pointe la possible utilisation d’explication d’un composant comme indication supplémentaire pour améliorer les performances d’un autre composant. [développer avec un exemple ?](#)

2.2.3 Interprétabilité Mécaniste

L’explicabilité de l’intelligence artificielle est constituée de deux branches. Certains modèles contiennent leur explication. On parle de modèle *explicable by-design*. C’est le cas des premiers modèles de machine learning tels que

les arbres de décision mais aussi de modèles plus récents comme les Sparse Auto-Encodeurs.

Les modèles explicables by-design s’opposent à des modèles plus opaques tels que les réseaux de neurones profonds. Le processus de décision de ces modèles n’est pas directement accessible. Les méthodes d’*explicabilité post-hoc* permettent de construire des explications à partir de l’étude de leur comportement.

[Somvanshi et al., 2026] présente un sous-paradigme de l’explicabilité : l’*interprétabilité mécaniste*. L’interprétabilité mécaniste vise à identifier les parties ou sous-parties d’un modèle responsables des différents comportements du modèle. Les méthodes d’interprétabilité mécaniste vont explorer la sensibilité des paramètres du modèle sur des entrées choisies afin de construire une cartographie du modèle faisant la correspondance entre ses compétences et connaissances et ses composants ou groupe de composants. on retrouve plusieurs approches au sein des méthodes d’interprétabilité mécaniste.

Les méthodes de **manual circuit tracing** cherchent à identifier les sous-graphes de modèles neuronaux responsables d’un certain comportement recherché. Pour cela, elles observent les paramètres et les flux de données lors de l’utilisation du modèle.

Les méthodes de **representation analysis** observent les représentations latentes des couches intermédiaires grâce à des outils d’unembedding. De cette manière, on peut localiser la détection de phénomènes présents dans l’entrée et la mise en place de notions dans la construction de la sortie.

Les techniques de **feature decomposition** cherchent à distinguer les différentes compétences traitées par chaque partie du modèle. Pour cela, elles intègrent des Sparse Auto-Encodeurs sur les parties du modèle visées puis déduisent, par l’observation de ces SAE, les compétences de chaque partie du modèle.

Les méthodes basées sur les **toy models and synthetic tasks** consistent à concevoir des environnements contrôlés favorisant la localisation de motifs dans le modèle associés un comportement visé. Ces méthodes supposent qu’une même compétence prendra la même forme dans tous les modèles disposant de cette compétence.

Les méthodes **intervention-based** observent l’impact de modifications du modèle sur son comportement pour en déduire la présence et la localisation des compétences du modèle. On retrouve trois approches parmi les techniques intervention-based : les techniques d’intervention par *ablation*, les techniques d’intervention par *activation patching* que nous détaillerons dans la prochaine sous partie et les techniques de *path patching*.

2.2.4 Activation Patching

Définition et fonctionnement de l’Activation Patching

L’activation patching est une méthode d’interprétabilité mécaniste intervention-based pour les modèles profonds. Elle identifie les composants d’un modèle sensibles à une famille de perturbations des entrées minimales pour changer la sortie [Geiger et al., 2021]. Pour cela, on effectue trois passes du dataset par perturbation : une sur les entrées non perturbées, une sur les entrées perturbées pour lesquelles on sauvegarde toutes les activations intermédiaires et une sur les entrées perturbées mais dans une version du modèle avec une activation remplacée par l’activation équivalente de la passe sur les données non perturbées. On s’intéresse alors à combien ce remplacement rapproche la sortie de celle attendue.

[Chen et al., 2024] propose une adaptation de cette méthode pour les modèles d’extraction d’information. Les entrées sont alors des paires (question, document) et la sortie, un score de pertinence. La perturbation ne doit plus nécessairement réduire la performance du modèle sur les entrées. (On peut penser à une perturbation qui duplique les mots importants de la question si ils apparaissent dans le document.) Par conséquent, la troisième passe ne se fait plus nécessairement sur les entrées non perturbées mais sur la version des entrées ayant la meilleure performance. La version de l’activation patching de [Chen et al., 2024] est formalisée dans l’encart 2.2.4.

Soit $Q \times D \subset \mathcal{Q} \times \mathcal{D}$ l'ensemble des paires question, document. Soit $Q \times \tilde{D}$ le même ensemble mais avec les documents perturbés à la place

1. Faire une passe pour toutes les paires $Q \times D$
 - noter o_n^e la sortie de chaque noeud $n, \forall e \in Q \times D$
 - noter p_D la performance du modèle sur les données non perturbées
2. Faire une passe pour toutes les paires $Q \times \tilde{D}$
 - noter $o_n^{\tilde{e}}$ la sortie de chaque noeud $n, \forall \tilde{e} \in Q \times \tilde{D}$
 - noter $p_{\tilde{D}}$ la performance du modèle sur les données perturbées
3. On renomme $D, e, \tilde{D}$ et $\tilde{e}$ respectivement
 - $\hat{D}, \hat{e}, \tilde{\tilde{D}}$ et $\tilde{\tilde{e}}$ si $p_D > p_{\tilde{D}}$
 - $\check{D}, \check{e}, \hat{\tilde{D}}$ et $\hat{\tilde{e}}$ sinon

Pour chaque noeud n
4. Faire une passe de $Q \times \tilde{D}$ en remplaçant $o_{i,j}^{\tilde{e}}$ par $o_n^{\tilde{e}}$ pour chaque $\tilde{e}$. On note la performance $\bar{p}_n$
5. $P_n = \frac{\bar{p}_n - p_{\hat{D}}}{p_{\check{D}} - p_{\hat{D}}}$ exprime l'impact de la perturbation sur la performance du modèle pour le noeud n

L'activation patching fourni trois performances par noeud avec lesquelles on peut estimer la sensibilité de chaque noeud à la perturbation. Si la sensibilité est élevée en valeur absolue, le noeud est sensible à la perturbation. Si la sensibilité est positive, le patch effectué atténue la perturbation. Si la sensibilité est négative, le patch accentue la perturbation. [Nouveauté à utiliser pour l'interprétation des matrices...](#)

Implémentation de l'Activation Patching

[Parry et al., 2025] propose MechIR, une implémentation en python des outils permettant d'appliquer les méthodes d'interprétabilité mécaniste à des modèles PyTorch et des modèles issus de HuggingFace. MechIR permet notamment de mettre en place l'activation patching comme proposé par [Chen et al., 2024].

MechIR permet à l'heure actuelle d'appliquer trois formes de perturbation sur les documents :

- **append** : Ajoute un mot fourni en paramètre à la fin des documents,
- **prepend** : Ajoute un mot fourni en paramètre au début des documents,
- **replace** : Remplace chaque occurrence d'un mot par un autre. Les deux mots sont fourni en paramètre.

Pour qu'une perturbation soit pertinente, elle doit changer significativement la sortie du modèle (ie. perturbe significativement le score) sans dénaturer excessivement l'entrée. MechIR propose une visualisation préalable de la distribution des scores de pertinence attribué par le modèle sur les données et les données perturbées (voir Figure 2.4) pour estimer la capacité de la perturbation à impacter les scores. Une perturbation impacte significativement les sorties des paires question, document si les distributions des scores de pertinence pour les documents perturbés et non-perturbés sont significativement différentes.

Par exemple, la Figure 2.4 visualise les distributions des scores de pertinence pour trois perturbations (ajouter *microwave* au début du document, ajouter *microwave* à la fin du document et remplacer *microwave* par *toaster*). Le dataset utilisé est une partie du dataset Vaswani issu de la librairie irdataset [Hou et al., 2025] et le modèle analysé est TAS-B [Hofstätter et al., 2021].

L'axe des abscisses correspond à la valeur prise par les scores et l'axe des ordonnées à leur densité. Plus la densité est élevée, plus il y a de scores proche de la valeur en ordonnée. La distribution des scores de pertinence des documents vis à vis des question est représentée en jaune et celle des scores pour les documents perturbés est en vert. La distribution des scores pour les documents non perturbés est identique pour les trois perturbation étant donné qu'on manipule le même dataset. Les perturbations de type **append** et **replace** impactent un peu les distribution des scores mais elles continuent à suivre la même tendance. La perturbation de type **prepend** impacte plus la distribution des scores de pertinence : les densité sont élevées pour les scores différents et le profil de la courbe est très différent pour les scores de documents perturbés et non perturbés.

Une fois la perturbation choisie, on peut appliquer l'activation patching pour chaque noeud du modèle choisi et visualiser les sensibilité à la perturbation de chacun d'entre eux (voir Figure 2.5). Chaque noeud du modèle est identifié à l'aide ses coordonnées (numéro de couche, position dans la couche). Le score du noeud correspond à la moyenne des sensibilité obtenu pour chaque document. Plus le noeuds à un score élevé en valeur absolue (ie. plus la couleur est foncée), plus la perturbation impacte son comportement. Cela signifie que la modification porte sur une

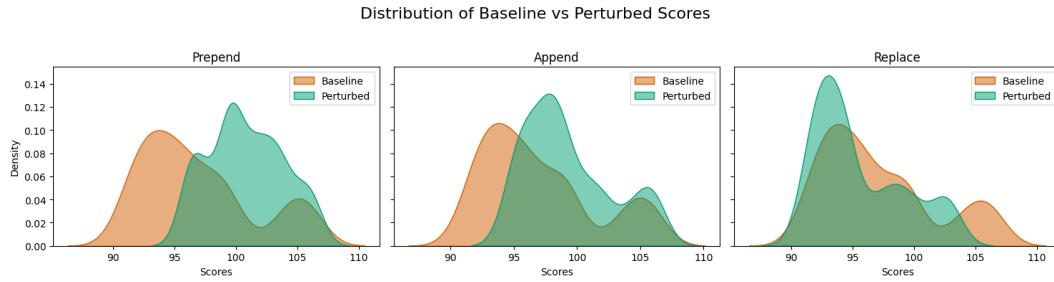


FIGURE 2.4 – Distribution des scores de pertinences pour trois perturbation sur les dataset Vaswani et le modèle TAS-B

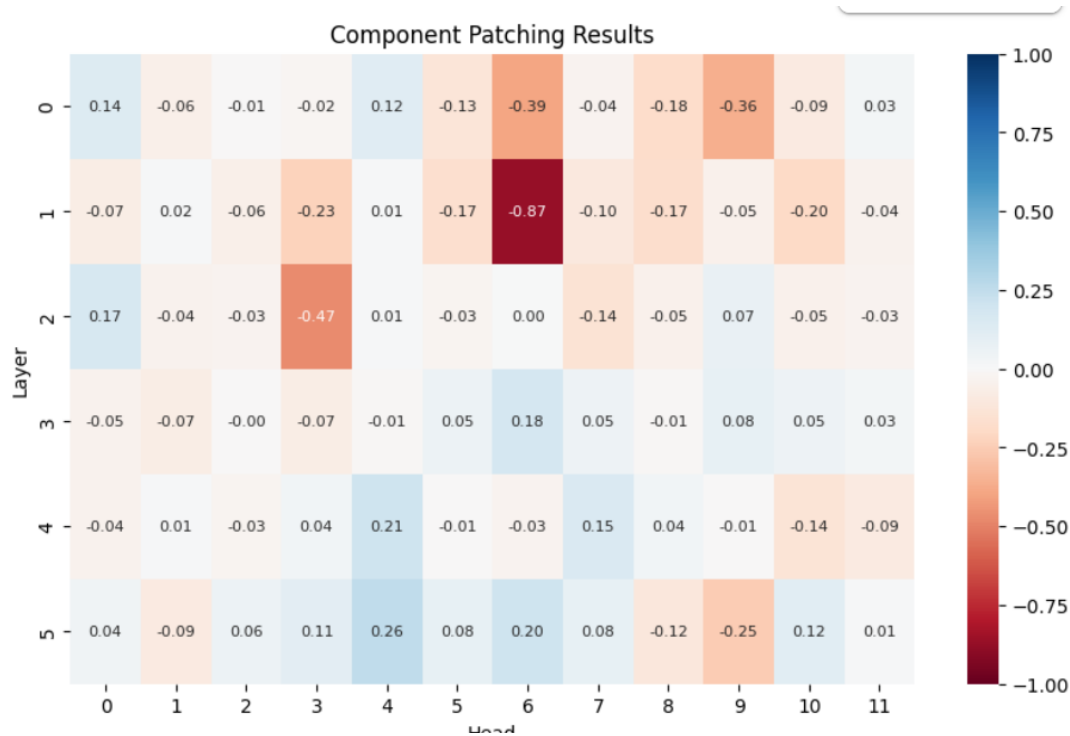


FIGURE 2.5 – Matrices de sensibilité des noeuds du modèle TAS-B pour la première perturbation de l'exemple

information utilisée par le noeud dans le modèle. Si le score est positif (resp. négatif), le patching atténue (resp. accentue) la perturbation en moyenne.

Dans la Figure 2.5, on remarque principalement le noeud (1,6). Une interprétation possible est que le patching du noeud diminue fortement l'estimation de pertinence effectuée par le modèle sur les documents non perturbé. [pas satisfaisant](#)

RQ : ajouter la distinction entre QA (et autres tâches de RAG) et IR et leur lien

Chapitre 3

Travail préalable et préparation des expériences

3.1 Challenge RAG@EvalLLM

- Le sujet de l'extraction d'information et du RAG est d'actualité
- Présentation du Challenge RAG@EvalLLM
 - La tâche
 - Identification des documents pertinents pour une question (phrase parfois complexe ou suite de mots clés)
 - questions fournies au dernier moment
 - pas de gold
 - Les données
 - pdf à traiter sans structure normée du contenu
 - majoritairement en français sauf quelques uns en anglais
 - peuvent contenir des images

Le sujet de l'extraction d'information et du RAG est d'actualité au sein de la recherche en machine learning à l'international. Son application au français et son évaluation anime les recherches en France. À l'occasion de l'atelier EvalLLM2026 qui portait sur l'évaluation des modèles génératifs et du RAG, ont été organisé deux challenges. L'un d'entre eux portait sur le RAG. La première tâche était une tâche d'extraction d'information et la seconde tâche portait sur l'attribution de sources (à partir d'une question, d'extraits de documents et d'une réponse, associer à chaque fragment de la réponse un document d'où l'information est issue). Ce challenge mettant à disposition une grande quantité de documents en français spécifiques à un domaine (la défense) et un protocole d'évaluation des résultats, je me suis inscrite à la première tâche pour donner un cadre à la construction du modèle et du protocole d'amélioration.

La tâche d'extraction d'information demandait de répondre à des requêtes sous la forme d'une ou plusieurs questions complexes ou d'une suite de mots clés sans séparateur à l'aide d'une base de documents et fournir les pages des documents utilisés pour répondre ainsi que leur rang de pertinence. La base de documents contenait en grande partie des PDF contenant parfois des images. Les documents avaient diverses origines, ne suivaient pas structure commune particulière et étaient en majorité en français (quelques documents en anglais ou disponibles dans les deux langues). Le format des entrées et des sorties était fourni à l'avance. Les documents étaient accessibles dès l'inscription mais les questions n'étaient communiquées que durant les trois jours de test (à l'exception des cinq questions présentes dans l'exemple de format des données). Aucun dataset étiqueté n'était fourni.

3.2 Construction d'un dataset adapté

À REPRENDRE

Base de données:

- Paires
- liste des champs et justification

Choix pour datasets

- Dataset avec des paires question-document (questions différentes et différents pdf, taille des documents, répartition positif/négatif)
 - Dataset avec une seule question (construction à partir de 3 pdf)
- [description du dataset dans les expés et les specs]
- motivation (reprendre l'expé présentée par MechIR, faciliter la création)
 - limites (biais, diversité ~> généralisabilité)
- => Placer le nom des outils utilisés (PyMuPDF4LLM et langchain.text_splitter.MarkdownSplitter)

- Le format de dataset visé
 - tous les dérouler V0, V1 et format V2 (à chaque fois: format ce qui manque, les modifications proposées)

Les données fournies par le challenge étant brutes, un premier traitement a été effectué. Les PDF ont été converti au format Markdown à l'aide de PyMuPDF4LLM [Artifex Software, 2026] pour les stocker dans base de donnée. Ce format est lisible pour l'humain et est souvent utilisé en entrée et sortie des grands modèles de langue. Les images n'ont pas été conservées car nous ne nous intéressons pas au traitement de données multi-modales. Ce pourrait être une perspective d'évolution dans la conception du modèle. [hors sujet ?](#) Les textes récupérés sont découpés en page et en chunk plus petits pour les pages contenant plus de 500 tokens.

La construction de la base de donnée a nécessité plusieurs itérations. Les différentes versions sont toutes au format CSV avec des champs d'identification de la question (identifiant) et de l'extrait (nom du fichier, identifiant, numéro de page, numéro de chunk dans le document), la question et l'extrait du document. Pour la base de donnée du modèle et son utilisation, les questions et les documents sont fournis dans deux fichiers CSV distincts pour toutes les versions. La base est mise sous forme de paires (produit cartésien) pour l'utilisation de MechIR et l'application de l'activation patching.

La première base de donnée est une base naïve avec toutes les questions et tous les extraits des documents fournis. Pour effectuer les manipulations avec MechIR, il faut manipuler le produit cartésien des questions et des extraits, soit une base de plus de 27 million d'entrées, surdimensionnée pour l'utilisation. De plus, du fait du produit cartésien, la majorité des documents ne sont pas pertinents pour la question. La base de donnée n'est pas équilibrée. Cela rend l'utilisation de MechIR et de l'activation patching moins accessible car plus difficile à interpréter. Dans un premier temps, la base de donnée a été échantillonnée pour ne garder qu'une petite partie mais la base de donnée obtenue n'est toujours pas équilibrée.

Pour résoudre le problème de l'équilibre, la seconde base de donnée est construite artificiellement à partir des extraits et des questions précédentes. Pour chacune des 26 questions, un extrait pertinent et un extrait non pertinent ont été choisis. La base de donnée obtenue est équilibrée, étiquetée et de taille très raisonnable. Pour autant, la diversité des mots utilisés et des sujets abordés rend difficile la construction d'une perturbation qui impacte la sortie sans dénaturer les documents.

La troisième base de donnée a été construite à partir d'une seule question. Ainsi, les extraits pertinents partagent un plus grand nombre de mots sur lesquels il est possible de s'appuyer pour construire des perturbations. Pour maintenir l'équilibre, les extraits sont pour moitié non pertinents. Le choix des extraits est fait à partir de trois PDF : un pour les extraits pertinents et deux pour les extraits non pertinents.

3.3 Adaptation du modèle RAG4HN

- Choix de ce modèle : traite des données en FR
- Suppression de la partie WebSearch (hors sujet)
- Suppression de la première extraction sur la base des titres (incompatible avec le format des données du challenge)
- Montée en version des librairies pour la compatibilité avec les outils
- Reprise du reranking (mettre au propre la photo du schema sur l'ardoise)

Pour répondre au challenge, nous avons besoin d'une architecture RAG complète fonctionnelle avec un extracteur sur lequel appliquer l'activation patching. Le stage ne portant pas sur la conception d'une architecture,

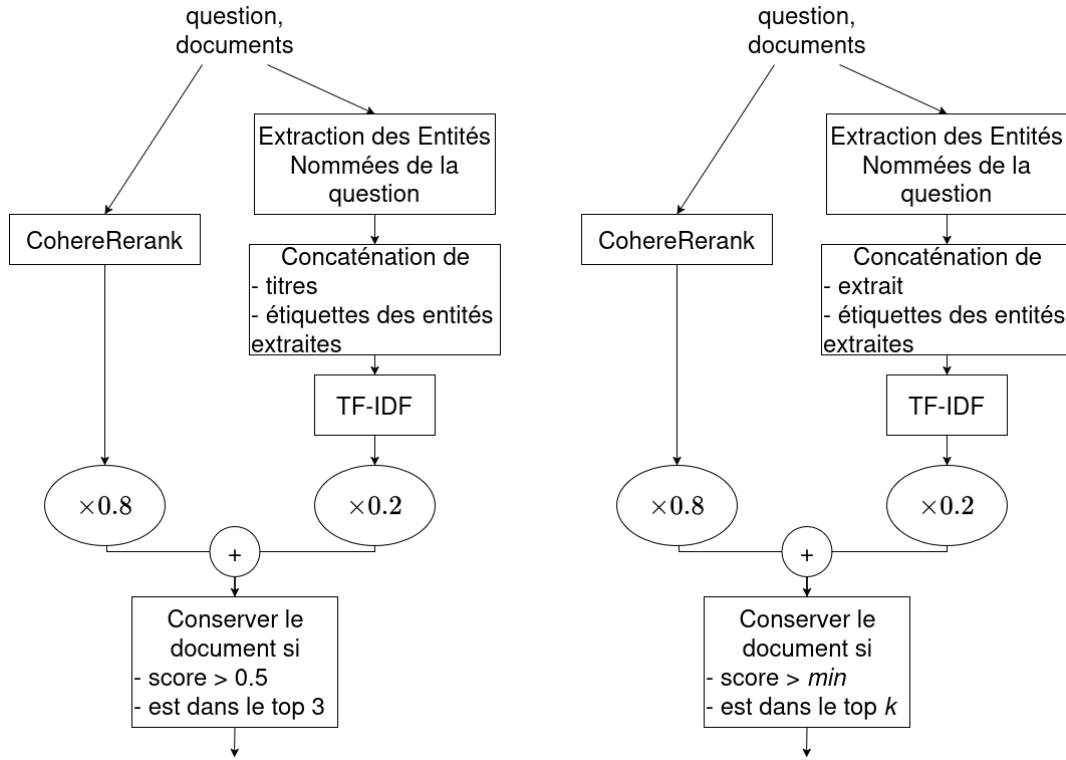


FIGURE 3.1 – Calcul du score de pertinence pour le reclassement des extraits de RAG4historicalNewspapers à gauche et de notre version simplifiée à droite

nous avons repris et simplifié l’architecture proposée par [Tran et al., 2024] pour l’adapter à la première tâche du challenge RAG. Le choix de cette architecture est motivé par la disponibilité du code source et le fait qu’elle est déjà adapté aux documents en français.

Les données du challenge n’étant pas au format (titre, document), la première extraction a été supprimée. Pour la même raison, le reclassement des textes extraits a été revu : la composante du TF-IDF est appliquée sur les extraits au lieu des titres (voir Figure 3.1). Le module de récupération de document sur le web n’étant pas compatible avec le challenge, il a aussi été écarté.

L’architecture obtenue est illustrée par la Figure 3.2.

Est-il pertinent de mentionner la montée en version des librairies dans cette partie ?

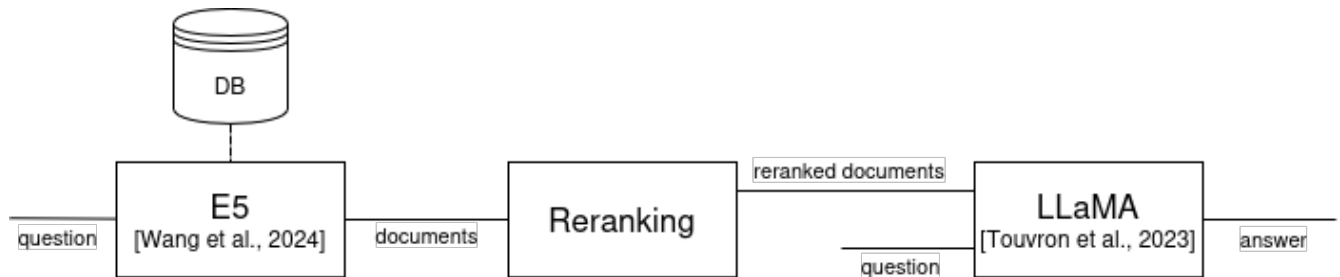


FIGURE 3.2 – Version simplifiée de RAG4HistoricalNewspapers utilisée pour le challenge

3.4 Objectifs

3.4.1 Localisation de la langue de spécialité dans un modèle

- Définition de langue de spécialité
- Formalisation de l'objectif

3.4.2 Intégration des travaux dans l'explicabilité post-hoc des modèles d'extraction d'information

- Cartographie de modèle d'extraction d'information
- Analyse des éléments identifiés par le modèle lors du passage d'un document

Chapitre 4

Contribution méthodologique

À REMETTRE EN FORME

4.1 Construction des perturbations

- on s'intéresse aux perturbations de type replace pour travailler sur l'impact des mots importants (car liés au vocabulaire de spécialité) en les remplaçant
- La conception de perturbations nécessite de bien choisir les mots à remplacer (traité dans la sous partie 3)
- La sous partie 3 propose un nouveau type de perturbations

4.1.1 Identification des mots cibles

- Le choix des remplaçant est fait par jugement humain
 - Le choix des mots à remplacer se fait à l'aide de
 - fréquence dans l'échantillon de données sélectionné
 - IDF
 - appartenance au vocabulaire de spécialité (jugement humain)
 - présence du mot dans la question étudiée (cas du dataset à 1! question)
- la bonne couverture des documents est importante (commentaire sur NaN) pour avoir des résultats lisibles

4.1.2 Perturbation replace par niveau

- On propose plusieurs candidats au remplacement classé en 4 niveaux ...
- Rq : il faudrait fonder l'estimation du niveau des perturbations

4.1.3 Perturbations de type full_replace

- pour augmenter le nombre de document perturbé et limiter l'impact d'un mot seulement dans la cartographie (avoir une apperçu plus global), on propose un nouveau type de perturbation : full_replace
- on étend les niveaux à ce nouveau type de perturbations

4.2 Estimation de la pertinence d'une perturbation

SI PAS DÉJÀ PRÉSENTÉ DANS PARTIE DE L'ÉTAT DE L'ART SUR MECHIR

- On visualise des scores de similarité entre question et document
- graphique d'exemple + texte d'accompagnement à la lecture
- Une perturbation peut être considérée comme pertinente lorsqu'elle modifie significativement (jugement humain) la distribution des cores pour l'échantillon du dataset étudié

- on propose d'autres visualisations et confrontation de la distribution des scores entre perturbation :
 - observation de la distribution des scores pour les documents pertinents et non-pertinents (sont-ils tr
 - observation de la distribution des scores d'un niveau de perturbation à l'autre pour un mot cible (que

4.3 Interprétation des résultats de l'Activation Patching

Un fois la ou les perturbations choisies, on peut appliquer l'activation patching à l'aide de l'outil MechIR

4.3.1 Visualisation des résultats proposée par MechIR

SI PAS DÉJÀ FAITE DANS SOTA sur MechIR

- graphique + accompagnement à la lecture (on obtient une sensibilité de chaque noeuds pour chaque paire (Question, document), on visualise la moyenne)

4.3.2 Autres visualisations de la sensibilité du modèle à la perturbation

à placer : confronter les matrices d'une perturbation à l'autre

- étant donné la construction de ces valeurs, on s'intéresse aussi aux écarts-types
- pour faciliter la visualisation et la comparaison en cas de sensibilité variable d'une perturbation à l'autre on propose de rescaler les poids des matrices

4.3.3 Critère d'identification des noeuds sensibles

- L'identification des noeuds sensibles doit reposer sur un critère commun. On propose un critère de sensibilité ...

4.3.4 Matrice des noeuds sensibles

- Pour faciliter le besoin de mise en perspective des matrice de sensibilité d'un perturbation à l'autres
- Visualisation des noeuds sensibles ****commun à plusieurs perturbation**** (matrice des noeuds sensibles)
 - > croiser les différentes caractéristiques (mot remplacé, type, niveau)
 - > critique des noeuds sensibles de la dernière couche

Chapitre 5

Résultats expérimentaux

À REPRENDRE

5.1 Résultats et Analyse

5.1.1 Application à la cartographie du vocabulaire de la défense français dans mE5

- Grand nombre de noeuds de la dernière couche du modèle
- pas de noeuds clairement associé au vocabulaire
- ...

5.1.2 Préconisations d'utilisation

- des perturbations qui couvrent tous ou très grande partie des documents étudiés
- utiliser la question comme document témoin

5.2 Perspectives d'évolution de la méthode proposée

- échelle pour situer le niveau d'un mot candidat remplaçant en fonction du mot remplacé
- critère d'identification du vocabulaire de spécialité
- intégration d'autres informations au critère de sensibilité

Chapitre 6

Conclusions et Perspectives

TODO

Bibliographie

- [Artifex Software, 2026] Artifex Software (2026). PyMuPDF4LLM : PDF Markdown Extraction for LLM workflows.
- [Bell et al., 2022] Bell, A., Solano-Kamaiko, I., Nov, O., and Stoyanovich, J. (2022). It’s Just Not That Simple : An Empirical Study of the Accuracy-Explainability Trade-off in Machine Learning for Public Policy. In *2022 ACM Conference on Fairness Accountability and Transparency*, pages 248–266, Seoul Republic of Korea. ACM.
- [Chen et al., 2025] Chen, B., Khare, A., Kumar, G., Akula, A., and Narayana, P. (2025). Seeing Beyond : Enhancing Visual Question Answering with Multi-Modal Retrieval. In Rambow, O., Wanner, L., Apidianaki, M., Al-Khalifa, H., Eugenio, B. D., Schockaert, S., Darwish, K., and Agarwal, A., editors, *Proceedings of the 31st International Conference on Computational Linguistics : Industry Track*, pages 410–421, Abu Dhabi, UAE. Association for Computational Linguistics.
- [Chen et al., 2024] Chen, C., Merullo, J., and Eickhoff, C. (2024). Axiomatic Causal Interventions for Reverse Engineering Relevance Computation in Neural Retrieval Models. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1401–1410, Washington DC USA. ACM.
- [Fan et al., 2024] Fan, W., Ding, Y., Ning, L., Wang, S., Li, H., Yin, D., Chua, T.-S., and Li, Q. (2024). A Survey on RAG Meeting LLMs : Towards Retrieval-Augmented Large Language Models. arXiv :2405.06211.
- [Garouani et al., 2024] Garouani, M., Mothe, J., Barhrhouj, A., and Aligon, J. (2024). Investigating the Duality of Interpretability and Explainability in Machine Learning. In *2024 IEEE 36th International Conference on Tools with Artificial Intelligence (ICTAI)*, pages 861–867. arXiv :2503.21356 [cs].
- [Geiger et al., 2021] Geiger, A., Lu, H., Icard, T., and Potts, C. (2021). Causal Abstractions of Neural Networks. arXiv :2106.02997 [cs.AI].
- [Hofstätter et al., 2021] Hofstätter, S., Lin, S.-C., Yang, J.-H., Lin, J., and Hanbury, A. (2021). Efficiently Teaching an Effective Dense Retriever with Balanced Topic Aware Sampling. arXiv :2104.06967 [cs.IR].
- [Hou et al., 2025] Hou, J., Cosma, G., and Finke, A. (2025). Advancing continual lifelong learning in neural information retrieval : Definition, dataset, framework, and empirical evaluation. *Information Sciences*, 687 :121368.
- [Huang et al., 2025] Huang, L., Yu, W., Ma, W., Zhong, W., Feng, Z., Wang, H., Chen, Q., Peng, W., Feng, X., Qin, B., and Liu, T. (2025). A Survey on Hallucination in Large Language Models : Principles, Taxonomy, Challenges, and Open Questions. *ACM Transactions on Information Systems*, 43(2) :1–55.
- [Lewis et al., 2020] Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W.-t., Rocktäschel, T., Riedel, S., and Kiela, D. (2020). Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. *arXiv : Computation and Language*.
- [Liu et al., 2025] Liu, P., Liu, X., Yao, R., Liu, J., Meng, S., Wang, D., and Ma, J. (2025). HM-RAG : Hierarchical Multi-Agent Multimodal Retrieval Augmented Generation. *arXiv.org*.
- [Lù, 2024] Lù, X. H. (2024). BM25S : Orders of magnitude faster lexical search via eager sparse scoring. arXiv :2407.03618.
- [Ma et al., 2023] Ma, X., Gong, Y., He, P., Zhao, H., and Duan, N. (2023). Query Rewriting for Retrieval-Augmented Large Language Models. arXiv :2305.14283 [cs.CL].
- [Ni et al., 2025] Ni, B., Liu, Z., Wang, L., Lei, Y., Zhao, Y., Cheng, X., Zeng, Q., Dong, L., Xia, Y., Kenthapadi, K., Rossi, R., Dernoncourt, F., Tanjim, M. M., Ahmed, N., Liu, X., Fan, W., Blasch, E., Wang, Y., Jiang, M., and Derr, T. (2025). Towards Trustworthy Retrieval Augmented Generation for Large Language Models : A Survey. arXiv :2502.06872.
- [Parry et al., 2025] Parry, A., Chen, C., Eickhoff, C., and MacAvaney, S. (2025). MechIR : A Mechanistic Interpretability Framework for Information Retrieval. In *Advances in Information Retrieval - 47th European Conference*

on Information Retrieval, *ECIR 2025, Lucca, Italy, April 6-10, 2025, Proceedings, Part V*, volume 15576 of *Lecture Notes in Computer Science*, pages 89–95. Springer.

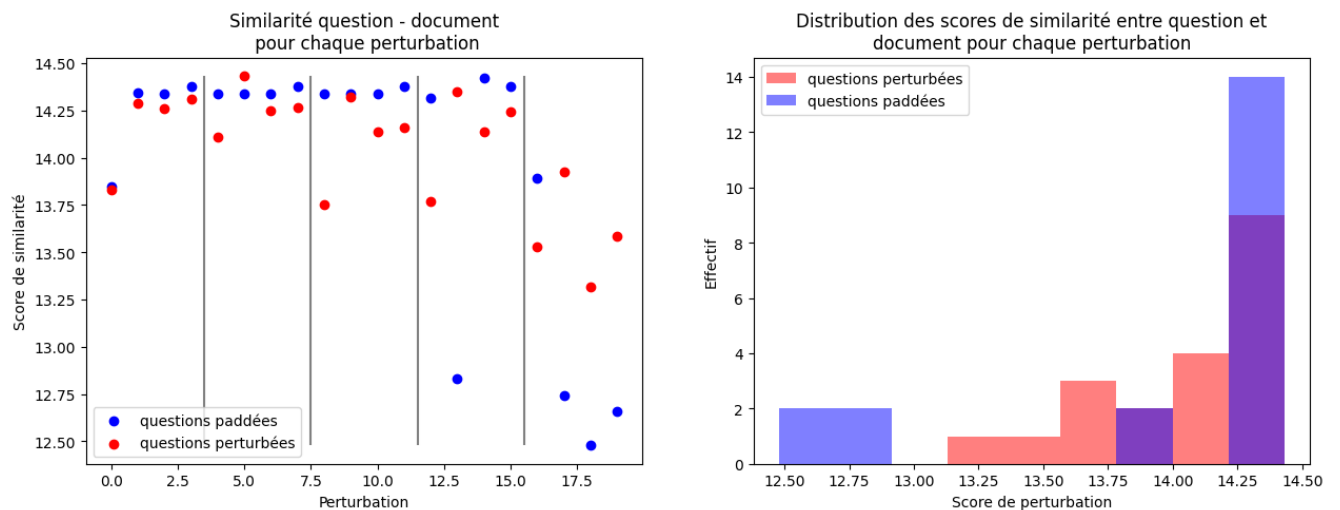
- [Somvanshi et al., 2026] Somvanshi, S., Islam, M. M., Rafe, A., Tusti, A. G., Chakraborty, A., Baitullah, A., Chowdhury, T. I., Alnawmasi, N., Dutta, A., and Das, S. (2026). Bridging the Black Box : A Survey on Mechanistic Interpretability in AI. *ACM Computing Surveys*, 58(8) :1–35.
- [Sparck Jones, 1972] Sparck Jones, K. (1972). A STATISTICAL INTERPRETATION OF TERM SPECIFICITY AND ITS APPLICATION IN RETRIEVAL. *Journal of Documentation*, 28(1) :11–21.
- [Touvron et al., 2023] Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.-A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., Rodriguez, A., Joulin, A., Grave, E., and Lample, G. (2023). LLaMA : Open and Efficient Foundation Language Models. arXiv :2302.13971.
- [Tran et al., 2024] Tran, T. T., González-Gallardo, C.-E., and Doucet, A. (2024). Retrieval Augmented Generation for Historical Newspapers. In *Proceedings of the 24th ACM/IEEE Joint Conference on Digital Libraries*, pages 1–5, Hong Kong China. ACM.
- [Wang et al., 2022] Wang, L., Yang, N., Huang, X., Jiao, B., Yang, L., Jiang, D., Majumder, R., and Wei, F. (2022). Text Embeddings by Weakly-Supervised Contrastive Pre-training.
- [Wang et al., 2024] Wang, L., Yang, N., Huang, X., Yang, L., Majumder, R., and Wei, F. (2024). Multilingual E5 Text Embeddings : A Technical Report.
- [Wang et al., 2023] Wang, L., Yang, N., and Wei, F. (2023). Query2doc : Query Expansion with Large Language Models. In Bouamor, H., Pino, J., and Bali, K., editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9414–9423, Singapore. Association for Computational Linguistics.
- [Xu et al., 2024] Xu, S., Pang, L., Shen, H., Cheng, X., and Chua, T.-S. (2024). Search-in-the-Chain : Interactively Enhancing Large Language Models with Search for Knowledge-intensive Tasks. arXiv :2304.14732.
- [Zhu, 2025] Zhu, Y. (2025). From Static to Dynamic : A Streaming RAG Approach to Real-time Knowledge Base. arXiv :2508.05662.

Annexe A

Résultats bruts d'expériences

A.1 Question Comme Document

A.1.1 Représentation et similarité de la question paddées/pertubée



Dataset manipulé Une paire (question, question) où la question est exploitée comme question mais aussi comme document

Perturbations appliquées

drone -> engin autonome
limitation -> contrainte
système -> dispositif
technique -> spécialisé
drone -> vecteur
limitation -> bornage
système -> ensemble
technique -> difficile
drone->gazon
limitation->animation
système->panneau
technique->magique
drone->ver de verre
limitation->plastique anarchique
système->esperluette en vent

```

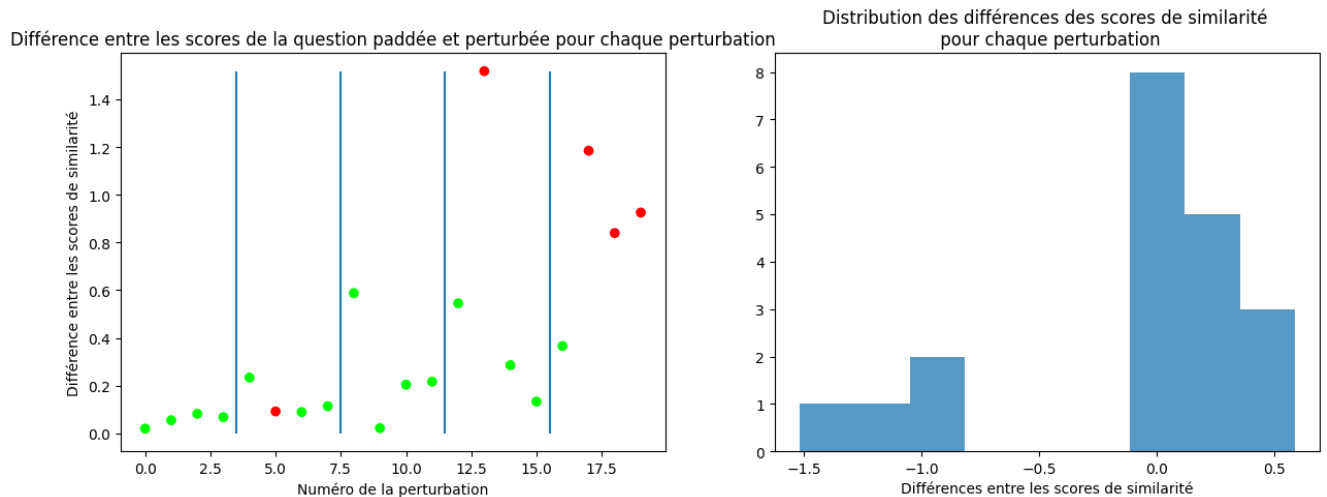
technique->feuille entubée
Full-replace Synonymes 1
    drone -> engin autonome
    limitation -> contrainte
    système -> dispositif
    technique -> spécialisé
Full-replace Polysémie 1
    drone -> vecteur
    limitation -> bornage
    système -> ensemble
    technique -> difficile
Full-replace Elephants 1
    drone -> gazon
    limitation -> animation
    système -> panneau
    technique -> magique
Full-replace Elephants Roses 1
    drone -> ver de verre
    limitation -> plastique anarchique
    système -> esperluette en vent
    technique -> feuille entubée

```

Données représentées Le graphique de gauche représente les scores de pertinence de la question comme document pour chaque perturbation. Les questions non modifiées (représentées en bleu) sont juste paddées pour faire correspondre la taille du document non modifié avec celui perturbé (ici de la question perturbée, représentées en rouge). Les scores de pertinence pour chaque perturbation sont dans l'ordre des perturbations citées. Les lignes verticales séparent les perturbations par niveau (de 1 à 4 en allant de gauche à droite) puis regroupent les perturbation de type full_replace (le plus à droite, de niveau 1 à 4 en allant de gauche à droite) Le graphique de droite est un histogramme des scores (en bleu pour la question paddée et en rouge pour la question perturbée)

Commentaires On observe bien la différence de représentation entre les questions et les question perturbées. On constate aussi que le système de padding is en place par [Parry et al., 2025] provoque une perturbation des documents. Certaines perturbations améliorent la similarité de la question avec elle même. Cela est surprenant

A.1.2 Différence des scores de pertinence pour question paddée et question perturbée



Données et perturbation Question comme document
Perturbation présentées précédemment

Lecture des graphiques Le graphique de gauche associe à chaque perturbation (dans l'ordre cité) la différence entre les scores de similarité à la question de la question perturbée et la question paddée pour la perturbation. Les points en rouge correspondent à aux différences négatives et les points en vert aux différences positives. On attend des scores positifs car perturber un objet (en l'occurrence une question) réduit sa similarité à lui même. Les points sont séparés pour identifier les perturbations de type replace de différent niveau (1 à 4 de gauche à droite). Le cinquième (dernier) groupe correspond aux perturbations de type full_replace (niveau 1 à 4 de gauche à droite) Le graphique de droite représente la distribution de ces différences.

Analyse des graphiques Pour le premier graphique, on constate une augmentation de la différence pour les niveaux les plus élevés concernant les perturbations de type replace. Les perturbations de type full_replace ont un effet de perturbation moins important que le padding induit par le modèle. On peut lier ce phénomène au fait que les modèles récents sont capables d'associer les expressions de même sens et de faire abstraction du contenu incohérent au sein d'un texte.

A.2 Séparer les données positives et négatives

Données Question : Quelles sont les trois principales catégories de limitations techniques affectant les systèmes de drones, et comment chacune influence-t-elle l'efficacité opérationnelle de ces vecteurs ?

Documents : 3 pdf portant sur 3 sujets distincts (limitations matérielles du drone, code de la défense - matériel de guerre, fausses infos pdt les JO/JP) dont sont extraits 28 textes (50% de documents pertinents)

Sujet du PDF	Documents Positifs	Documents Négatifs
Limitations techniques de drones	14 (100%)	0
Code de la Défense (Chapitre sur le matériel de guerre)	0	6 (43%)
Desinformation pendant les Jeux Olympiques	0	8 (57%)

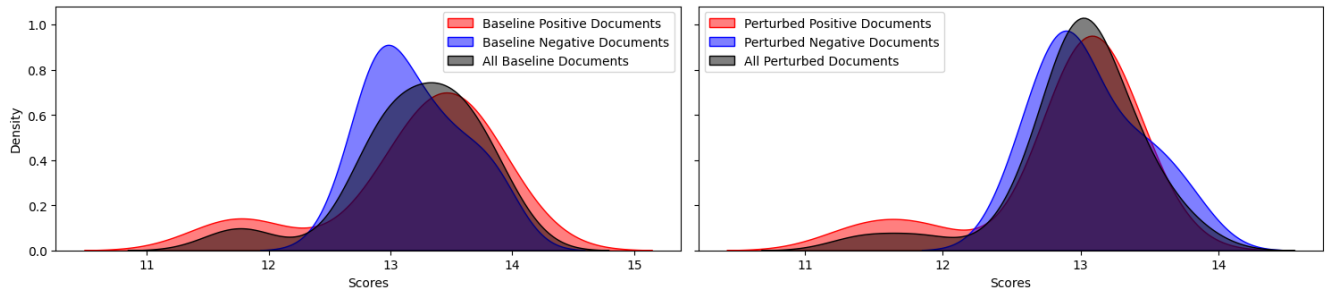
Perturbations

TODO

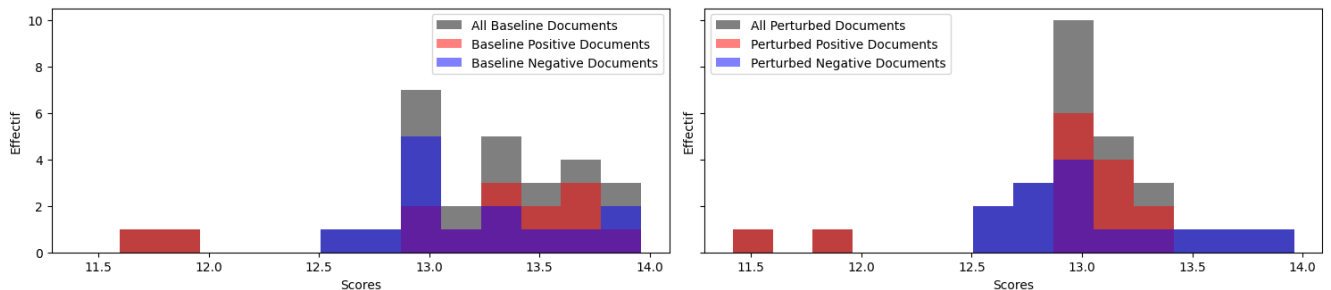
Full-replace

A.2.1 Distribution des Scores

Distribution des scores de similarité des documents par rapport à la question



Histogramme des scores de similarité entre le document et la question

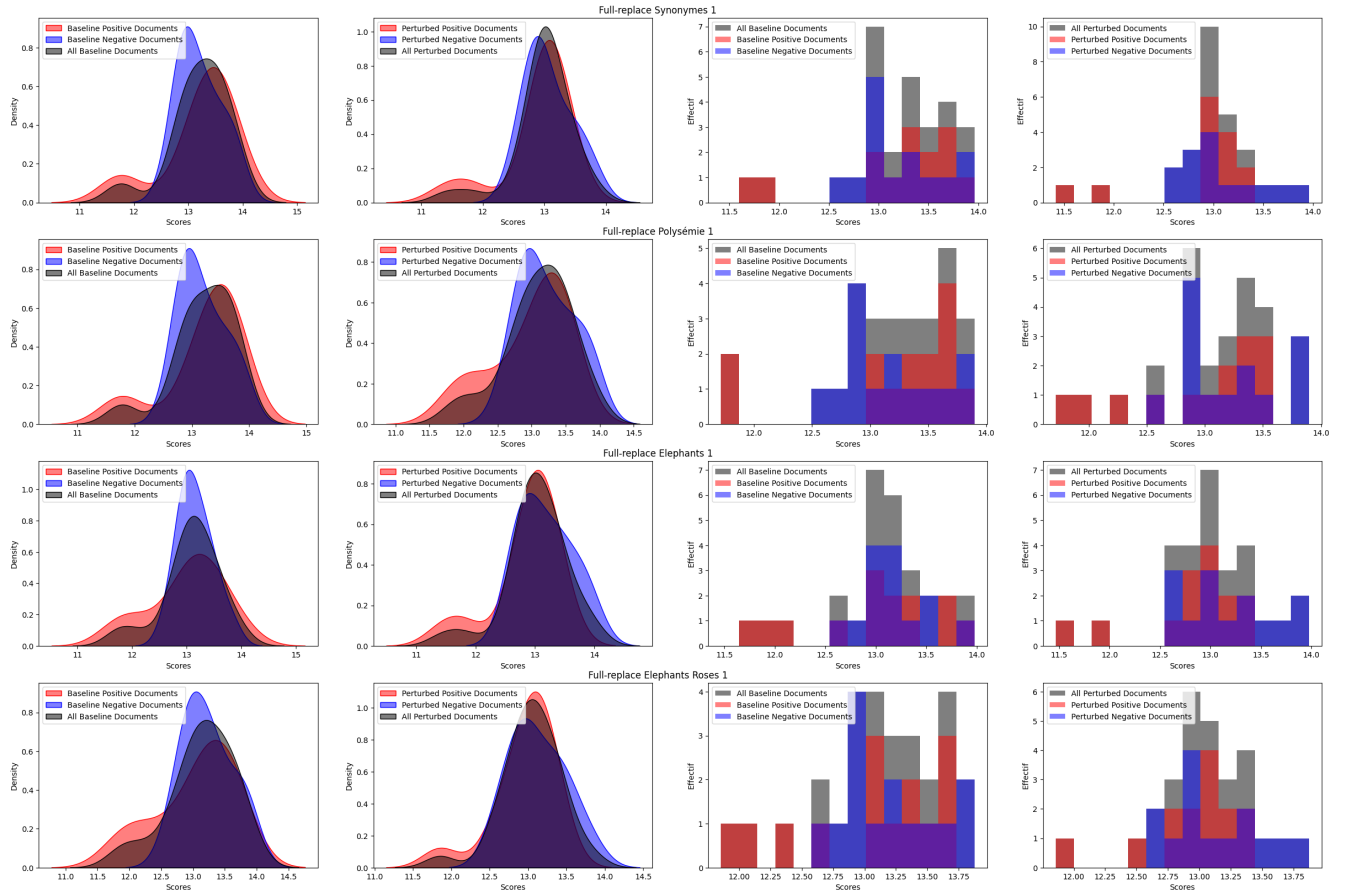


Données visualisées Ces graphiques représentent la répartition des scores de similarité entre la question et chaque document pour la perturbation **Full-replace Synonyme 1**. En rouge sont représentées les documents désignés pertinents vis à vis de la question, en bleu les documents non pertinents et en noire l'ensemble des documents. Le graphique de gauche correspond au cas des documents paddés et les graphiques de droite correspond aux documents perturbés.

Analyse

A.2.2 Scores pour les perturbations full_replace

Distribution et Histogramme des scores de similarité des documents par rapport à la question



Données Visualisées Chaque ligne de graphique traite une perturbation. Les graphiques représentent la distribution/ l'histogramme des scores de similarité à la question de chaque document. En rouge, on considère les documents désignés comme pertinents, en bleu les documents considérés comme non-pertinents et en noir l'ensemble des documents. Pour chaque paire de graphique, celui de gauche concerne les documents paddés (ramenés à la même longueur que les documents perturbés) et celui de droite concerne les documents perturbés par la perturbation indiquée dans le titre de la ligne de graphique.

Analyse Dans un premier temps, concernant les documents paddés, on retrouve une répartition des documents pertinents et non-pertinents cohérente : les documents pertinents ont un score de similarité à la question plus élevé que les documents non-pertinents. On notera que cette variation de distribution reste assez faible.

Finalement, on ne fait que vérifier le comportement des scores vis-à-vis de l'étiquette des documents. La visualisation des scores des documents perturbés n'est pas exploitée. J'ai l'impression que les documents positifs voient leur score diminuer et que les documents négatifs voient leur score augmenter. Ce comportement (1) est

suffisamment faible pour n'être qu'une impression, (2) n'a pas de sens étant donné la nature de la perturbation (en tout cas je ne l'explique pas). Peut-être que la visualisation des scores pour les documents non perturbés suffit.

A.3 Niveaux de perturbation

Intégrer dans l'analyse de la proposition de perturbation par niveau

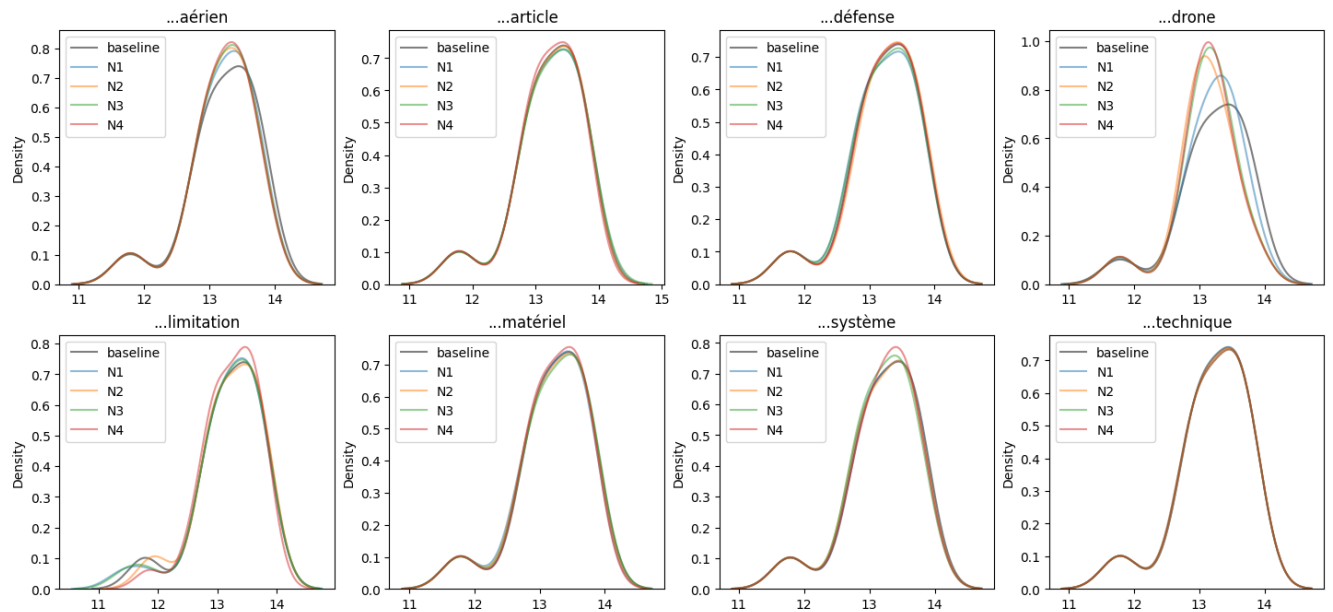
Dataset Idem que précédemment

Perturbations

TODO

A.3.1 Distribution des scores en fonction du niveau de perturbation

Distribution des scores pour chaque type de perturbation appliquée au mot ...



Données visualisées On visualise la distribution des scores de similarité à la question des documents. Les distributions sont regroupées par mot remplacé (appelé mot cible dans la suite du paragraphe). Pour chaque mot on observe la distribution des scores de perturbation pour les documents non modifiés (paddés pour la perturbation de niveau 1) en noir, avec le mot cible remplacé par un synonyme (perturbation de niveau 1, en bleu), avec le mot cible remplacé par le synonyme d'un autre de ses sens que celui utilisé dans le document (perturbation de niveau 2, en orange), avec le mot cible remplacé par un mot ne partageant aucun sens avec le mot cible mais respectant sa nature (perturbation de niveau 3, en vert), et avec le mot cible remplacé par un groupe de mot absurde (perturbation de niveau 4, en rouge).