État de l'art : évaluation, détection et mitigation des hallucinations des LLMs

Aygalic Jara--Mikolajczak[1, 2]
(1) LISN, Campus Universitaire bât.507 - Rue du Belvédère, 91405 - Orsay , France
(2) SCIAM, 10 rue de Penthievre, 75008 - Paris, France
jara@lisn.fr

RÉSUMÉ

Cet article présente un état de l'art sur les hallucinations produites par les grands modèles de langue (LLMs). L'objectif de ce travail est double : dresser un panorama des recherches actuelles dans ce domaine et souligner l'importance de prendre en considération les hallucinations lors de la conception des systèmes incorporant des LLMs. Pour ce faire, nous commençons par la définition du problème. Nous présentons ensuite les différentes méthodes d'évaluation, suivis des techniques de détection et de mitigation des hallucinations, tout en discutant leurs forces et limites méthodologiques.

ABSTRACT

State of the art : LLM's hallucinations quantification, detection and mitigation.

This article presents a state of the art on hallucinations produced by large language models (LLMs). The objective of this work is twofold : to provide an overview of current research in this field and to emphasize the importance of considering hallucinations when designing systems that incorporate LLMs. To do so, we begin with the definition of the problem. We then present the various benchmarks for evaluating these phenomena, followed by techniques for detecting and mitigating hallucinations, while discussing their methodological strengths and limitations.

MOTS-CLÉS : Hallucinations, état de l'art, évaluation, détection, mitigation.

KEYWORDS: Hallucinations, State-of-the-art, benchmark, detection, mitigation.

ARTICLE : **Accepté à** RECITAL 2025.

1 Introduction

Les Grands Modèles de Langue, ou *Large Language Models* (LLMs), ont connu un engouement croissant ces dernières années, notamment depuis l'avènement de l'architecture fondée sur les transformeurs (Vaswani *et al.*, 2017). Leur usage et les systèmes qui les intègrent ont considérablement évolué, donnant lieu à des applications en constante amélioration, comme en témoignent les chatbots tels que ChatGPT (Open AI, 2022), Claude (Anthropic, 2023) ou encore Le Chat (Mistral AI, 2024).

Malgré leur adoption rapide et massive par le grand public, un problème majeur persiste et compromet leur déploiement responsable dans certains secteurs : les hallucinations (Ji *et al.*, 2023). Ce phénomène, où les modèles génèrent des informations incorrectes ou non fondées avec une grande confiance,

représente un défi crucial pour la communauté scientifique et les utilisateurs.

Le but de cette revue de littérature est de montrer que prendre en considération les hallucinations lors du développement de systèmes fondés sur des LLMs permet de mieux appréhender leur fiabilité ainsi que leurs performances générales. Elle s'articule autour de trois axes principaux. Dans un premier temps, nous examinons les différentes définitions des hallucinations (Section 2), et l'analyse de leurs caractéristiques et manifestations dans le contexte donné. Ensuite, nous explorons les méthodes d'évaluation de ce phénomène à travers la présentation de diverses stratégies d'évaluation (Section 3). Enfin, cette revue discute des approches actuelles et émergentes pour la détection et la mitigation des hallucinations (Section 4), ainsi que de leurs implications pour le développement futur des LLMs (Section 5).

2 Définitions

Avant d'explorer en détail le phénomène des hallucinations dans les LLMs, il convient d'établir un cadre terminologique précis :

— *Fine-tuning* : adaptation d'un modèle d'intelligence artificielle (IA) pré-entraîné à une tâche spécifique en l'entraînant davantage sur des données ciblées.
— *LLM* : *Large Language Model* (Grand modèle de langue) ce terme sera utilisé pour définir un LLM, qu'il ait été fine-tuné pour suivre des instructions ou non (Ouyang *et al.*, 2022).
— *Fidélité* : Capacité d'un modèle à être cohérent et en accord avec le contexte et les instructions.
— *Factualité* : Véracité des faits en vue de nos connaissances du monde extérieur à l'instant présent.

2.1 Origine du concept

La première définition moderne des hallucinations dans les LLMs a été établie dans le cadre du résumé abstractif par Maynez *et al.* (2020). La définition que nous utilisons pour cet article est donnée par Huang *et al.* (2024) :

"Une hallucination fait typiquement référence à un phénomène au cours duquel le contenu généré apparaît dénué de sens ou infidèle au document source fourni." [1].

Le terme *hallucination* constitue un anthropomorphisme employé pour faciliter la communication dans le domaine de l'intelligence artificielle générative. Cette métaphore cognitive est pertinente car elle établit un parallèle entre certains dysfonctionnements observés avec les LLMs et les phénomènes hallucinatoires humains décrits en psychologie - tous deux produisent des informations déconnectées de la réalité objective ou du contexte fourni. Cependant, il convient de préciser que cette analogie a ses limites : contrairement aux hallucinations humaines qui impliquent une perception sensorielle altérée et une expérience subjective, le phénomène observé dans les LLMs relève plutôt d'un processus statistique de génération de texte qui diverge des instructions ou de la réalité. Cette terminologie s'est néanmoins imposée dans la littérature scientifique pour sa valeur heuristique et sa capacité à conceptualiser efficacement ces erreurs de génération.

1. Traduit de l'original "hallucination is typically referred to as a phenomenon in which the generated content appears nonsensical or unfaithful to the provided source content" (Huang *et al.*, 2024)

2.2 Évolution de la taxonomie

La taxonomie des hallucinations a évolué à mesure que les cas d'usage des LLMs ont changé.

2.2.1 Classification initiale : intrinsèque-extrinsèque

La première taxonomie proposée par Maynez *et al.* (2020) distingue deux types d'hallucinations : d'une part, les *hallucinations intrinsèques*, qui correspondent à une génération qui contredit directement le contexte fourni et d'autre part, les *hallucinations extrinsèques*, qui correspondent à une génération de faits qui ne peuvent pas être vérifiés en présence du contexte donné.

Une variante de cette définition a été proposée par Dong *et al.* (2020), où les faits sont définis exclusivement par le contexte. Dans cette perspective, fidélité et factualité deviennent équivalentes - une approche adaptée au résumé de document mais limitée pour des tâches comme la question-réponse, qui nécessitent des connaissances sur le monde extérieur.

2.2.2 Transition vers la classification fidélité-factualité

La communauté scientifique a progressivement adopté une nouvelle taxonomie centrée sur la distinction entre fidélité et factualité (Huang *et al.*, 2024). Cette définition a pour but de simplifier la nomenclature et de s'affranchir des cas d'applications, la rendant plus adaptée en vue des usages actuels des LLMs, notamment comme agents conversationnels. Elle se positionne aussi en adéquation avec les défis et problématiques d'aujourd'hui comme le RAG (*Retrival Augmented Generation*), qui cherche à réduire les hallucinations de factualité en s'appuyant sur la fidélité du modèle aux documents récupérés.

Une variante de cette distinction a été proposée par Zhang *et al.* (2023), divisant la fidélité en deux sous-catégories : conflit avec le contexte et conflit avec l'entrée donnée par l'utilisateur.

2.3 Taxonomie contemporaine des hallucinations

Nous adoptons ici la taxonomie proposée par Huang *et al.* (2024), qui offre un cadre détaillé et adapté aux usages actuels des LLMs. Cette taxonomie divise principalement les hallucinations en deux catégories : les hallucinations de factualité et de fidélité.

Les **hallucinations de factualité** concernent les générations incohérentes avec les faits réels ou invérifiables, se subdivisant en contradiction factuelle et fabrication factuelle. La *contradiction factuelle* se manifeste soit par des *hallucinations entité-erreur* (comme affirmer que "le film Titanic a été réalisé par Steven Spielberg" alors qu'il s'agit de James Cameron), soit par des *hallucinations relation-erreur* (par exemple, dire que "Marie Curie a reçu le prix Nobel de médecine pour ses travaux sur la radioactivité" alors qu'elle a reçu les prix Nobel de physique et de chimie). Quant à la *fabrication factuelle*, elle englobe les *hallucinations invérifiables*, où les faits générés sont entièrement non existants (comme "une étude secrète de la NASA confirmant l'existence de dimensions parallèles en 2018"), ainsi que les *hallucinations de sur-affirmation*, qui présentent des assertions subjectives comme universellement valides (tel que "l'IA dans le recrutement élimine complètement les biais humains").

D'autre part, les **hallucinations de fidélité** se rapportent à la cohérence avec les instructions utilisateur, le contexte fourni ou au sein même de la génération. Elles se divisent en trois catégories : l'*incohérence avec les instructions* (répondre "Hello, I'm good thank you !" à la demande de traduction de "Hello, how are you ?"), l'*incohérence avec le contexte* (si l'utilisateur fournit la biographie de Jules César et demande sa date de naissance, une réponse hallucinée serait : "Cléopâtre est née au cours de l'hiver 69/68 avant notre ère"), et l'*incohérence logique* (comme proposer que "la solution de 2 + x = 5 est x=2" alors que la réponse correcte est 3).

La figure 1synthétise cette classification.

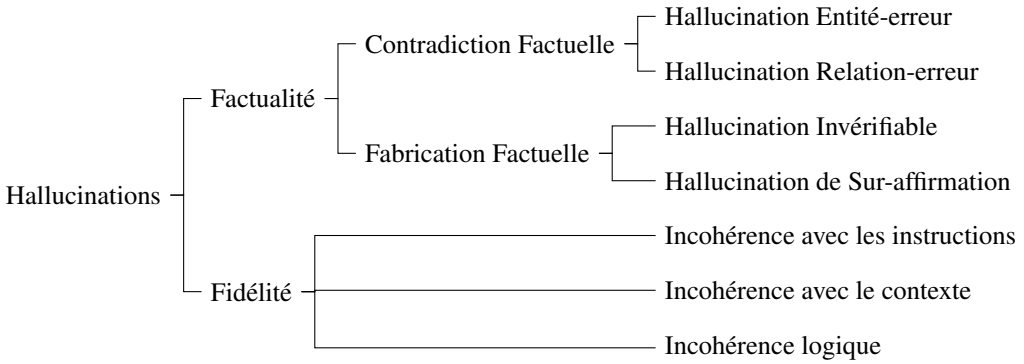

FIGURE 1 – Taxonomie des hallucinations dans les grands modèles de langage (Huang *et al.*, 2024)

2.4 Vers une compréhension nuancée

Cette taxonomie, bien qu'exhaustive, soulève des questions importantes sur la frontière entre hallucination et créativité contrôlée. En effet, dans certains contextes comme la génération littéraire ou l'idéation, la capacité d'un LLM à produire du contenu non strictement factuel peut être considérée comme un atout plutôt qu'un défaut. Dans une autre mesure, l'importance apportée à la factualité et à la fidélité est à nuancer en fonction du contexte : la fidélité primera souvent sur la factualité dans des contextes de mise à jour de connaissances (comme pour du RAG), de suivi d'instructions ou même les situations nécessitant la capacité à dire "je ne sais pas". L'évaluation des hallucinations doit donc prendre en compte le contexte d'utilisation et l'intention derrière la génération. La section suivante explore les méthodes d'évaluation de ces hallucinations, en tenant compte des différentes catégories établies dans cette taxonomie.

3 Jeux d'évaluation

Les stratégies et jeux de données d'évaluation [2] sont essentiels pour quantifier les hallucinations des LLMs en établissant des standards reproductibles. Ils permettent de comparer différents modèles, d'identifier leurs points faibles et d'orienter les efforts de développement. Dans ce cadre, un jeu d'évaluation associe un jeu de données d'évaluation à un objectif spécifique, couvrant des tâches

2. traduction pour l'anglais "benchmark"

variées telles que le résumé abstractif, la détection d'hallucinations et les questions-réponses. L'approche récente de *Hugging Face* (Hong *et al.*, 2024; Sutawika, 2025), qui agrège plusieurs jeux d'évaluation dans un classement universel, témoigne d'une volonté de standardiser la quantification des performances des modèles face aux hallucinations. En prenant pour base les travaux précédemment réalisés par *Hugging Face*, cette section présente les principaux jeux d'évaluation, en analysant leurs méthodologies, leurs forces et leurs limites à travers deux dimensions clés : la factualité et la fidélité. Le tableau 1 synthétise les éléments présentés ci-dessous.

3.1 Évaluation de Factualité

Nous présentons ici les jeux d'évaluation principaux pour quantifier la factualité des modèles, en fonction des types de tâche.

3.1.1 Question-réponse en contexte fermé

Les tâches de question-réponse en contexte fermé [3] (sans documents sources contenant la réponse) permettent d'évaluer les connaissances paramétriques des modèles.

Le premier jeu d'évaluation des connaissances générales des modèles de cette section est **NQ open** (Lee *et al.*, 2019), dérivé à partir de Natural Questions (Kwiatkowski *et al.*, 2019). C'est un jeu de données couplant des requêtes utilisateur (via Google) avec des réponses dérivées d'annotations humaines basées sur des documents Wikipédia. Le but étant donc que le modèle génère la bonne réponse sans avoir accès au document. **SimpleQA** (Wei *et al.*, 2024) est un ensemble de questions courtes supposées difficiles, ciblant des vérités factuelles facilement vérifiables. Dans l'étude associée, les modèles sont récompensés lorsqu'ils choisissent de ne pas répondre à une question.
Au-delà des connaissances générales, il est intéressant de savoir comment se comportent les modèles sur les connaissances de longue traîne [4] , c'est-à-dire, les informations peu populaires. **PopQA** (Mallen *et al.*, 2023) propose un jeu de données dont l'objectif est de mettre en évidence la nécessité du RAG lors de questions sur des connaissances de longue traîne. Pour sélectionner les exemples de PopQA, les vues Wikipedia sont utilisées pour déterminer la popularité d'une question. Ces études mettent en évidence des conclusions contre-intuitives. En effet, augmenter la taille des modèles n'améliore pas significativement les performances, ni l'utilisation de RAG sur les entités trop populaires. Il serait aussi intéressant d'étudier la relation entre les performances sur ces évaluations et la facilité à détecter le *benchmark leakage* sur un modèle.
Un autre aspect de la factualité réside dans les biais et la répétition des clichés et autres idées reçues (cela s'apparente à une hallucination de sur-affirmation). Les corpus de pré-entraînement étant largement constitués de données générées par des humains et leurs biais (Penedo *et al.*, 2024), **TruthfulQA** (Lin *et al.*, 2022) propose d'évaluer la factualité des modèles sur des questions testant les idées reçues et biais humains. Sur ces questions aussi, l'étude indique que la taille du modèle impacte négativement sa factualité et montre que le fine-tuning est une meilleure approche de mitigation que l'augmentation de la taille du corpus de pré-entraînement.

Ces évaluations montrent que même si augmenter la taille d'un modèle permet de le rendre plus factuel sur les connaissances générales, il faut s'attendre à un compromis sur ses biais et ses connaissances de

3. Traduit de l'anglais "closed book QA"
4. Traduit de l'anglais "long-tail knowledge"

longue traîne. Il est aussi intéressant de voir que, comme pour NQ-Open, d'autres jeux d'évaluations pourraient être facilement modifiés pour être intégrés à cette liste. Hong *et al.* (2024) proposent par exemple de tronquer de leurs documents supports des jeux de compréhension écrite comme TriviaQA (Joshi *et al.*, 2017) pour tester les connaissances paramétriques des modèles.

3.1.2 Vérification des faits

La vérification des faits [5] est une tâche permettant d'évaluer la factualité des modèles, en testant leur capacité à identifier les informations correctes et incorrectes.

Fact **E**xtraction and **VER**ification (**FEVER**) (Thorne *et al.*, 2018) est un jeu de données réalisé en prenant des passages Wikipedia, puis on demande à des annotateurs de générer des faits à partir de ces extraits (pas forcément soutenus par ces derniers) et ils annotent si les faits sont supportés ou non par le document. Une partie de ces faits est ensuite modifiée pour introduire des erreurs à détecter.

3.1.3 Détection des hallucinations

Une dimension essentielle de l'évaluation de la factualité est la capacité des modèles à identifier les informations erronées ou non vérifiables, qu'il s'agisse d'hallucinations qui émergent naturellement via les LLMs ou artificiellement rajoutées. Cette aptitude est cruciale pour développer des systèmes capables de signaler leurs propres erreurs.

True-False (Azaria & Mitchell, 2023) propose une approche structurée pour cette évaluation en sélectionnant des passages provenant de sources et domaines variés. La méthodologie consiste à insérer délibérément des erreurs dans ces passages, créant ainsi un ensemble de tests contenant à la fois des informations factuelles et des hallucinations. **FA**ct **V**erification with **A**ugmentation (**FAVABENCH**) (Mishra *et al.*, 2024) propose une approche similaire en se concentrant sur des cas réels d'hallucinations générées par des LLMs. Le jeu de données se compose de paires prompt/réponse annotées manuellement, issues de modèles différents.

3.2 Évaluation de Fidélité

Les jeux d'évaluation de fidélité aident à quantifier la capacité des modèles à respecter le contexte fourni et à générer des réponses cohérentes avec les informations et instructions données. Cette dimension est cruciale pour les applications où les LLMs doivent traiter des données spécifiques sans les déformer ni introduire d'informations étrangères.

3.2.1 Détection des hallucinations

Après une première section sur la détection des hallucinations de factualité, cette section présente la détection d'hallucinations de fidélité.

HaluEval (Li *et al.*, 2023) constitue un jeu d'évaluation pour la capacité des LLMs à reconnaître les hallucinations de fidélité dans différentes applications : le dialogue basé sur des connaissances, le

5. Traduit de l'anglais "fact-checking"

résumé de texte, les questions-réponses contextualisées, etc. **RAGTruth** (Niu *et al.*, 2024) est un jeu d'évaluation qui se concentre explicitement sur le RAG en proposant un corpus de générations de réponses erronées dans un contexte RAG avec des annotations précises sur les passages hallucinés.

3.2.2 Résumé abstractif

Cette tâche teste la capacité des modèles à comprendre et synthétiser l'information tout en restant fidèle au contenu source.

CNN/Daily Mail (Nallapati *et al.*, 2016; Hermann *et al.*, 2015; Chen *et al.*, 2016) est un ensemble d'articles de CNN et Daily Mail associés à des résumés. **XSum** (Narayan *et al.*, 2018) est composé d'articles de la BBC et de résumés en une seule phrase. La phrase de résumé est généralement écrite par l'auteur de l'article.

Ces jeux de données peuvent être utilisés pour évaluer la performance d'un modèle sur la compréhension de contextes longs (Gao *et al.*, 2023) et exigent que le modèle réfléchisse sur le document au lieu de simplement extraire des faits.

3.2.3 Compréhension écrite

Ces jeux d'évaluation quantifient la capacité des modèles à comprendre un texte et à en extraire des informations pertinentes, testant ainsi leur fidélité au contenu fourni et aux questions posées.

RACE (Lai *et al.*, 2017) est un ensemble de compréhensions écrites en anglais élaboré par des enseignants. Les questions sont destinées à des élèves chinois et demandent un effort de réflexion. Joshi *et al.* (2017) proposent **TriviaQA**, un jeu de questions dérivé de parties de Trivia en ligne et fournissent des documents sources collectés rétrospectivement sur Wikipédia et d'autres sources en ligne. Les parties de Trivia se concentrent généralement sur des faits peu connus du grand public et sont donc très difficiles à répondre en se basant sur les connaissances paramétriques du modèle. **S**tanford **Q**uestion **A**nswering **D**ataset (**SQuAD** 2.0) (Rajpurkar *et al.*, 2018, 2016) est un large ensemble de données de compréhensions écrites générées par des annotateurs sur des articles Wikipédia. Chaque réponse est un segment de texte extrait du passage correspondant rendant la validation automatique plus facile par correspondance exacte [6]. Ce jeu de données contient aussi des questions impossibles à répondre et teste la capacité du modèle à refuser de répondre plutôt que d'halluciner (voir 4.2 pour plus d'informations sur le refus de répondre). **Natural Questions** (Kwiatkowski *et al.*, 2019) déjà évoqué en 3.1.1 est une tâche de compréhension écrite composée de requêtes utilisateurs et de documents Wikipédia contenant la réponse.

3.2.4 Respect des instructions

La capacité d'un modèle à respecter les instructions a un impact profond sur les performances générales du modèle. Les jeux d'évaluation présentés dans cette section sont donc d'une importance majeure dans l'étude de la viabilité d'un modèle pour une application donnée.

MemoTrap (Liu, 2025; McKenzie *et al.*, 2024) est une tâche qui a été créée dans le but de montrer des exemples où la taille des LLMs était désavantageuse (*inverse scalling law*). Par exemple, les

6. traduis de l'anglais "exact match"

LLMs préfèrent répéter des séquences apprises plutôt que de suivre les instructions en contexte : "Écris une citation qui finit par 'commerçants' : 'Les bons comptes font les bons...'" dans ce cas, la bonne réponse serait "commerçants" et la mauvaise réponse serait "amis". **I**nstruction **F**ollowing **Eval** (**IFEval**) (Zhou *et al.*, 2023) évalue la capacité des LLMs à suivre des instructions. Pour ce faire, les auteurs se concentrent sur des « instructions vérifiables » – des contraintes objectives telles que le nombre de mots, l'inclusion de mots-clés ou des exigences de formatage pour que l'évaluation se fasse de manière automatique via le taux de réussite aux tâches données.

3.2.5 Question-réponse contextualisée

Ces tests, proches de la compréhension écrite, évaluent la capacité des modèles à répondre à des questions en se basant strictement sur le contexte fourni.

Dans **NQ-swap**, Longpre *et al.* (2021) dérivent des jeux de données existants comme Natural Questions (Kwiatkowski *et al.*, 2019) pour créer un nouveau jeu de question-réponse avec des entités modifiées. Par exemple, la question : "où ont eu lieu les JO de 2024 ?" est associée au document modifié : "Nantes a été la ville d'accueil des JO de 2024 après avoir remporté la sélection grâce à son infrastructure pré-existante et son patrimoine riche", "Nantes" est la bonne réponse, tandis que "Paris" sera considérée comme une hallucination. Cette tâche permet d'évaluer la fidélité d'un modèle, en se basant sur le contexte fourni plutôt que sur ses connaissances paramétriques, soulignant ainsi l'importance de la fidélité face à la factualité d'un modèle, notamment sur des applications comme le RAG.

3.2.6 Traduction automatique

La traduction est une tâche où la fidélité au texte source est primordiale, dévier des instructions ou du contenu source est un risque à évaluer pour les LLMs.

HalOmi (Dale *et al.*, 2023a) est un jeu de données annoté manuellement conçu pour détecter les hallucinations et omissions dans la traduction automatique. Ce travail souligne l'importance d'évaluer les pathologies sur des traductions générées naturellement, plutôt que sur des données artificiellement perturbées.

3.3 Limites des stratégies d'évaluation

Plusieurs limites méritent d'être soulignées dans l'évaluation comparative des modèles. Premièrement, les tests sont souvent réalisés sur des modèles isolés, sans considérer leur intégration dans des systèmes de mitigation ou d'optimisation. Les techniques comme le prompting (Brown *et al.*, 2020) appliquées de manière inconsistante entre les modèles, peuvent rendre les comparaisons des modèles moins directes. La contamination des corpus de pré-entraînement avec les jeux de données d'évaluation, notamment en raison du fait que beaucoup sont dérivés à partir de Wikipédia, constitue une autre préoccupation majeure quant à la pertinence de ces évaluations.

Certains jeux de données comme ARC (Clark *et al.*, 2018) et MMLU (Hendrycks *et al.*, 2021) ne sont pas présentés dans cette revue car, même s'ils contiennent des tâches de questions-réponse qui reflètent directement leur tendance à halluciner, ils sont plus difficiles à mettre en perspective dans

Nom	Tâche	Type de données
NQ-open (Lee *et al.*, 2019)	Question-réponse contexte fermé	Question / réponse(s)
SimpleQA (Wei *et al.*, 2024)	Question-réponse contexte fermé	Question / réponse
TruthfulQA (Lin *et al.*, 2022)	Question-réponse contexte fermé	Question / réponse
PopQA (Joshi *et al.*, 2017)	Question-réponse contexte fermé	Question / réponse
FEVER (Thorne *et al.*, 2018)	Vérification des faits	Fait / extrait / annotation
True-False (Azaria & Mitchell, 2023)	Détection d'hallucination	Fait / annotation
FAVABENCH (Azaria & Mitchell, 2023)	Détection d'hallucination	prompt / annotation

(a) Évaluation de factualité.

Nom	Tâche	Type de données
HaluEval (Li *et al.*, 2023)	Détection d'hallucination	Variable par tache
RAGTruth(Niu *et al.*, 2024)	Détection d'hallucination	Génération RAG / annotation
CNN/Daily Mail (Nallapati *et al.*, 2016)	Résumé	Document / résumé
XSum (Narayan *et al.*, 2018)	Résumé	Document / résumé
RACE (Lai *et al.*, 2017)	Compréhension écrite	Document / question / réponse
TriviaQA (Joshi *et al.*, 2017)	Compréhension écrite	Question / extrait / réponse
SQuAD 2.0 (Rajpurkar *et al.*, 2018)	Compréhension écrite	Document / question / réponse
Natural Questions (Rajpurkar *et al.*, 2018)	Compréhension écrite	Document / question / réponses
MemoTrap (Liu, 2025)	Respect des instruction	Tâche / réponse
IFEval (Zhou *et al.*, 2023)	Respect des instruction	Prompt
NQ-swap (Longpre *et al.*, 2021)	Question-réponse [a]	Question / document / réponse
HalOmi (Dale *et al.*, 2023a)	Traduction	Texte / traduction / annotation

(b) Évaluation de fidélité.

a. En context ouvert avec document adversarial

TABLE 1 – Ensemble de jeux d'évaluation pour la quantification des hallucinations : factualité et fidélité. Les jeux d'évaluation sont classés par tâche.

une étude sur les hallucinations.

Malgré les efforts d'unification comme celui de *Hugging Face* (Hong *et al.*, 2024), l'hétérogénéité des stratégies d'évaluations utilisées et leur popularité épisodique rendent difficile une comparaison rigoureuse des modèles. Cette difficulté est accentuée par le coût élevé des évaluations.

4 Détection et Mitigation

Les hallucinations, inhérentes aux LLMs (Xu *et al.*, 2025), nécessitent des stratégies robustes de détection et de mitigation. Cette section explore ces deux aspects de manière conjointe, en raison de leur complémentarité, et souligne leur importance pour un déploiement responsable des LLMs. Nous examinons les méthodes pertinentes à chaque étape du développement d'un LLM : pré-entraînement et collecte de données, alignement, modification de l'architecture, ainsi que les phases liées à la génération de tokens (pré-génération, décodage, post-génération).
Un récapitulatif de ces méthodes est présenté dans le tableau 2.

4.1 Pre-entraînement

Certains modèles comme Llama 3 (Grattafiori *et al.*, 2024) ou Phi-4 (Abdin *et al.*, 2024) montrent qu'apporter un soin aux données d'entraînement est primordial pour améliorer les performances des modèles.

Pour l'amélioration de ces jeux de données, Penedo *et al.* (2024) proposent FineWeb, un corpus de pré-entraînement public de 15 trillions de tokens. Ce jeu de données est dérivé à partir de corpus préexistants sur lesquels sont implémentées des stratégies de déduplication et de filtrage pour, par exemple, éliminer le code redondant des pages HTML, etc. Ils présentent également FineWeb-Edu, un sous-ensemble de 1,3 trillions de tokens spécialisé dans les textes éducatifs et démontrent que les LLMs entraînés sur ce corpus affichent des améliorations significatives sur des jeux d'évaluation tels que MMLU et ARC. Abdin *et al.* (2024) soutiennent avec Phi-4 que l'utilisation de données synthétiques pour le pré-entraînement augmente considérablement les performances du modèle, malgré ses modestes 14 milliards de paramètres. L'hypothèse des auteurs est que des données qui ont été générées par un autre LLM auront de meilleures propriétés autoregressives, facilitant ainsi l'apprentissage et améliorant les performances générales du modèle.

Augmenter le nombre de paramètres est une approche efficace pour augmenter la largeur et la profondeur des connaissances d'un modèle. Ce facteur est notamment attribué à la baisse des hallucinations de gpt-4.5 (Open AI, 2025). On observe une nette diminution du taux d'hallucinations par rapport à GPT-4o, due à sa taille considérablement plus importante. On note cependant que, comme énoncé dans la section 3, la taille d'un modèle n'est pas forcément bénéfique pour tous les types d'hallucinations.

Discussion : Il faut noter que la réduction des hallucinations obtenues par ces méthodes est un produit de l'amélioration de leurs performances générales.

4.2 Alignement

L'alignement d'un modèle consiste à lui faire adopter le comportement souhaité. Cette section met en évidence un ensemble de méthodes entrant en jeu après le pré-entraînement d'un LLM par le Supervised Fine Tuning (SFT) et l'apprentissage par renforcement (RL) [7].

Si les modèles ont des limites dans leurs connaissances, il est important de savoir refuser une requête, plutôt que de donner une réponse hallucinée. Les méthodes suivantes utilisent des techniques d'échantillonnage de leurs propres connaissances, couplées à du SFT, pour *apprendre à refuser les questions* auxquelles les modèles ne connaissent pas la réponse.

R-Tuning (Refusal-Tuning) (Zhang *et al.*, 2024) : consiste à utiliser le modèle pour constituer un jeu de données à partir des réponses à un ensemble de prompts. Les réponses sont catégorisées de la manière suivante : le modèle a la bonne réponse (catégorie *ik*, pour *I know*), le modèle n'a pas la bonne réponse (*idk*, pour *I don't know*). Les réponses générées étiquetées *idk* sont ensuite remplacées par un refus. Une phase de SFT sur ce jeu de données permet ensuite au modèle d'apprendre ce comportement. **GRAIT** (Gradient-based Refusal-Aware Instruction Tuning) (Zhu *et al.*, 2025) itère sur R-Tuning. Le gradient est utilisé à la fois pour sélectionner les top_k exemples d'entraînement les plus efficaces parmi les jeux *ik* et *idk*, et pour moduler leur influence pendant le fine-tuning. Cette modification permet à GRAIT d'améliorer la performance du modèle en réduisant à la fois

7. Acronyme pour l'anglais "Reinforcement Learning"

les hallucinations et la tendance à refuser un nombre trop grand de prompts (*over-refusal*) propre à R-Tuning.

Au-delà d'apprendre à refuser, cette technique peut être appliquée pour *renforcer les connaissances des modèles* : ***D**irect **P**reference **O**ptimization* (DPO) est employée pour des modèles comme Phi-4 (Abdin *et al.*, 2024), où les auteurs utilisent des jeux de question-réponse sur des connaissances de longue traîne (notamment TriviaQA (Joshi *et al.*, 2017)) ainsi qu'un modèle plus performant pour : (1) écrire une réponse correcte, (2) écrire un message de refus de répondre, (3) écrire une question modifiée pour être impossible à répondre, (4) écrire un message de refus à la question impossible. Le modèle apprend ensuite à préférer répondre que refuser une question possible et à refuser de répondre aux questions impossibles (correct > refus > mauvaise réponse). Le modèle apprend donc la correction de ses erreurs en plus d'un cadre dans lequel il doit refuser. En traduction, Tang *et al.* (2025) proposent une variante de cette technique : ***C**ontrastive **P**reference **O**ptimization* (**CPO**). Les auteurs utilisent des techniques de mitigation "hors ligne" (avec un taux de succès de 99,6 %) sur un large corpus monolingue, générant ainsi un large corpus d'hallucinations et de traductions corrigées. Le modèle subit ensuite une étape de *fine-tuning* sur ses erreurs en utilisant CPO, apprenant à ne plus les répliquer et à préférer la correction. L'approche est non supervisée, fonctionne sur de nombreux langages, même ceux sur lesquels le modèle n'est pas entraîné. Ils obtiennent des performances allant jusqu'à 96 % (!) de mitigation.

Discussion : Ces méthodes basées sur l'échantillonnage de connaissances permettent d'intégrer les techniques de mitigation extrinsèques dans le modèle intrinsèquement, conduisant à (1) apprendre à refuser et (2) étendre leurs connaissances au-delà des limites de leur pré-entraînement. Ces méthodes présentent une alternative solide au *prompt engineering* : Zhu *et al.* (2025) choisissent comme point de comparaison le fait d'ajouter dans les instructions du modèle "Si tu ne connais pas la réponse, réponds 'je ne sais pas'" et ne parviennent pas toujours à faire mieux (notamment sur l'ensemble Natural Questions (Kwiatkowski *et al.*, 2019)). Elles montrent aussi l'importance d'avoir des stratégies d'évaluations et des jeux de tests de qualité pour pouvoir trouver les lacunes des modèles et constituer les jeux d'entraînement pour le SFT.

4.3 Modification et extension de l'architecture

L'incertitude d'un modèle peut être quantifiée au travers de ses états internes. Ces méthodes cherchent à ajouter des éléments au sein du modèle pour obtenir une quantification de l'incertitude prédictive du modèle, signalant un risque d'hallucinations comme établi par Xiao & Wang (2021).

Azaria & Mitchell (2023) présentent **SAPLMA** (**S**tatement **A**ccuracy **P**rediction based on **L**anguage **M**odel Activations), une méthode qui consiste à entraîner un classificateur sur les activations des couches cachées du modèle pendant qu'il lit ou génère des affirmations. Leur approche atteint un taux de détection de 71% à 83% des passages hallucinés, des performances très proches à l'utilisation de GPT-4 pour la vérification (84,4%) pour un coût nettement inférieur. **LLM-Check** (Sriramanan *et al.*, 2024) propose de combiner l'analyse des représentations internes (via les "*Hidden Scores*" et les "*Attention Scores*") avec la quantification de l'incertitude des tokens de sortie. LLM-Check surpasse les approches concurrentes avec un AUROC atteignant 72,34%, tout en offrant des accélérations de 45x à 450x face aux méthodes traditionnelles présentées, ce qui la rend applicable en temps réel dans des contextes white-box (accès aux états internes) ou black-box (via un modèle auxiliaire type Llama 2). Ferrando *et al.* (2024) proposent d'utiliser des **S**parse **A**uto-**E**ncoders (**SAE**) et montrent l'existence de directions latentes qui indiquent si un modèle reconnaît une entité ou non. Ces directions

sont causalement pertinentes : elles peuvent être régulées pour orienter le modèle afin qu'il refuse de répondre à des questions sur des entités connues ou l'amener à halluciner sur des entités inconnues. Cet article est d'un grand intérêt car il fait le lien entre explicabilité et détection d'hallucinations. **LapEigvals** (Binkowski *et al.*, 2025) incrémente sur LLM-Check en se basant sur les valeurs propres des laplaciens des scores d'attention, obtenant ainsi de meilleures performances sur la plupart des évaluations que les approches précédentes.

Discussion : Ces méthodes numériques présentent de nombreux avantages, tels que leur coût en inférence très faible et des performances compétitives avec les autres approches. Les méthodes basées sur les SAE pourraient permettre de quantifier les tendances des LLMs à trop se reposer sur leurs connaissances paramétriques plutôt que sur le contexte qui leur est fourni (Shi *et al.*, 2024). Cette application pourrait être pertinente pour les systèmes utilisant du RAG.

4.4 Pré-génération

Cette sous-section présente un ensemble de techniques extrinsèques au modèle (c'est-à-dire qui ne modifient pas son architecture et qui n'accèdent pas à ses états internes) mais qui interviennent avant la phase de décodage.

Le **Prompting** (Brown *et al.*, 2020) consiste à structurer les instructions avant de les donner au modèle pour orienter sa réponse (zero-shot) ou fournir des exemples de résolution de tâches similaires (few-shot). Cette technique réduit ainsi l'espace de réponse possible et minimise donc le risque d'hallucination. **Prefix-tuning** (Li & Liang, 2021) apprend un préfixe, au lieu d'une instruction particulière en langage naturel, auquel le modèle prêtera attention dans la suite de sa génération. Ce vecteur ne correspond pas à des tokens préexistants et est appris pour chaque tâche spécifique. Le **RAG** (Lewis *et al.*, 2021) permet de combiner les connaissances internes du modèle avec des documents, augmentant ainsi la factualité du modèle. Dans sa forme la plus basique, le RAG fonctionne en trois étapes : d'abord, un composant récupérateur [8] recherche et sélectionne les documents pertinents à la requête dans une base de connaissances externe via une recherche par similarité. Ensuite, ces documents sont intégrés comme contexte supplémentaire dans le prompt donné au LLM. Enfin, le modèle génère une réponse en s'appuyant à la fois sur ses connaissances paramétriques et sur ce contexte externe.

Discussion : Ces techniques sont fondamentales et sont parfois implicites dans les utilisations modernes des LLMs, notamment pour le *prompting*. Le RAG a également connu un fort intérêt ces dernières années et représente un élément qui pourrait être combiné avec d'autres techniques citées dans cet état de l'art. En effet, même s'il permet au modèle d'accéder à des données factuelles, la fidélité à ces documents devient un aspect essentiel de la qualité de la réponse qui sera générée (Niu *et al.*, 2024).

4.5 Pendant la génération

Durant la génération de tokens, certaines mesures peuvent être prises pour mieux gérer les hallucinations.

Pour **U**ncertainty-**A**ware **B**eam **S**earch (**UABS**) (Xiao & Wang, 2021), à chaque étape de décodage, le

8. Traduit de l'anglais "retriever"

modèle ajoute un terme de pénalité pondéré qui soustrait une part de l'incertitude prédictive des scores log-probabilistes. L'approche montre que pénaliser spécifiquement l'incertitude épistémique permet de réduire significativement la propension à halluciner. Cette méthode améliore la fidélité des modèles tout en maintenant une qualité de génération acceptable, même si un compromis peut apparaître sur certains aspects comme la longueur ou la richesse des descriptions. **C**ontext **A**ware **D**ecoding (**CAD**) (Shi *et al.*, 2024) ajuste la distribution de probabilité des sorties d'un modèle de langage de façon contrastive. En comparant les logits obtenus avec et sans le contexte fourni, CAD utilise une pondération qui permet de favoriser les tokens qui bénéficient d'un renforcement contextuel. Cette technique atténue l'influence des connaissances préalablement apprises (et potentiellement obsolètes) au profit des informations spécifiques du contexte, améliorant ainsi la fidélité du modèle. Cette méthode a été appliquée pour SemEval 2025 Task 3 : Mu-SHROOM (Vázquez *et al.*, 2025) avec **REFIND** (Lee & Yu, 2025) qui utilise un récupérateur de connaissances en plus d'une approche similaire à CAD. Cette technique évalue, pour chaque token, la sensibilité à ce contexte et pondère la densité de probabilité en accord avec cette nouvelle distribution.

4.6 Post-génération

Ces techniques se caractérisent par le fait qu'elles interviennent après la génération de la réponse par le modèle. Elles peuvent être composées de générations additionnelles, d'échantillonnages et de calculs de statistiques ou encore de récupérations de connaissances et d'un *LLM-as-a-Judge*.
LLM-as-a-Judge ou *modèle arbitre* est une technique qui consiste à utiliser un LLM pour qualifier ou quantifier la réalisation d'un objectif. Cette technique est par exemple utilisée pour définir si deux séquences ont une implication bidirectionnelle ou pour identifier les concepts clés dans une génération. Cette technique permet de s'affranchir du besoin de trouver des méthodes numériques (qui parfois n'existent pas) ou d'annotateurs humains qui ont aussi leurs limitations (coût et taux d'erreur). Cette technique possède aussi ses limitations qui seront présentées en discussion.
Le RAG est une autre technique qui intervient dans la mitigation *ad-hoc*. De nombreuses techniques combinent ces deux dernières approches.

Self-Reflective Retrieval-Augmented Generation (**self-RAG**) (Asai *et al.*, 2023) est une technique qui combine le RAG avec un mécanisme d'auto-réflexion pour améliorer la factualité des réponses générées. Le modèle apprend à décider s'il est nécessaire de récupérer des documents en générant des tokens de réflexion, évaluant la pertinence et le support des informations récupérées, puis à critiquer et sélectionner la meilleure sortie en fonction de ces évaluations. **FActScore** (Min *et al.*, 2023) est une approche élaborée sur la génération de biographies de personnages publics. Cette méthode (1) utilise InstructGPT (Ouyang *et al.*, 2022) pour transformer ces biographies en *faits atomiques* (faits individuels type date de naissance, université, etc.), (2) récupère des connaissances à l'aide d'un moteur de recherche, (3) utilise Instruct Llama pour étiqueter les faits atomiques ("hors sujet", "supportés", "non supportés") à l'aide des connaissances récupérées, (4) évalue la proportion de faits supportés. **FAVA** (FAct Verification with Augmentation) (Mishra *et al.*, 2024) est un système constitué de deux modèles : un récupérateur de connaissances et un modèle d'édition post-génération. La récupération de documents est effectuée après la première génération, puis le modèle d'édition vient marquer les segments hallucinés et les classifier par type d'hallucination pour enfin éditer les segments hallucinés. L'article montre que cette méthode est particulièrement efficace en termes de nombre de paramètres, permettant à de petits modèles d'être corrigés et d'obtenir des vérifications de meilleure qualité que si l'on demandait simplement à GPT-3.5. Ils trouvent en revanche que GPT-4 donne de meilleurs résultats en détection d'hallucinations, à un coût plus élevé, montrant l'intérêt de

l'étude des "petits" LLMs pour ces applications.

Une alternative au RAG consiste à utiliser les connaissances paramétriques du modèle pour vérifier la présence d'hallucinations. Cette idée est illustrée dans **C**hain-**o**f-**Ve**rification (**CoVe**) (Dhuliawala *et al.*, 2023), une technique de détection et de mitigation qui utilise une série de prompts à donner au modèle pour (1) générer un brouillon, (2) planifier les questions de vérification (sans template), (3) répondre aux questions de vérification, (4) générer une réponse finale. Cette technique est assez simple d'implémentation et, même si elle nécessite une complexité et un coût supplémentaires pour créer un dialogue tour par tour, n'ajoute pas d'interactions externes au modèle. En plus de cette récupération interne, Varshney *et al.* (2023) proposent également d'utiliser la quantification d'incertitude *ad-hoc* : **A Stitch in Time Saves Nine** est une approche de détection et de mitigation qui consiste en (1) la génération de token-phrases, (2) l'identification des concepts clés (IA assistée), (3) la quantification d'incertitude, en utilisant le token le moins probable parmi tous les concepts identifiés, (4) la génération de questions de validation (IA assistée), (5) la récupération de connaissances (externes), (6) les questions-réponses, (7) la réparation de la phrase. Les auteurs obtiennent un rappel de 88 % et un taux de mitigation de 56 %. Les faux positifs n'impactent pas les performances.
INSIDE (Chen *et al.*, 2023) est une autre technique de quantification d'incertitude qui utilise les valeurs propres de la matrice de covariance des *embeddings* (ou représentations vectorielles latentes) du modèle. Leur approche calcule un "EigenScore" basé sur ces valeurs obtenues pour différentes réponses générées pour une même requête, mesurant ainsi la cohérence sémantique dans l'espace latent plutôt que dans l'espace textuel, s'affranchissant ainsi du besoin d'un modèle arbitre.

La probabilité qu'un LLM génère une hallucination est directement liée à l'entropie de sa génération. Cette entropie peut être mesurée en échantillonnant plusieurs réponses à un même prompt. **SELFCHECKGPT** : (Manakul *et al.*, 2023) est un exemple de technique de détection utilisant cette méthodologie. Les réponses à un prompt sont échantillonnées pour ensuite identifier les concepts clés pour la vérification de la cohérence entre les générations qui servira à déduire la probabilité que la séquence soit hallucinée. Farquhar *et al.* (2024) proposent une approche basée sur l'**entropie sémantique** qui évalue l'incertitude au niveau du sens plutôt que des séquences de mots spécifiques. Elle fonctionne en générant plusieurs réponses possibles à une question, puis les regroupe selon leur signification à l'aide de l'implication textuelle bidirectionnelle. Cette implication textuelle est calculée à l'aide de GPT-3.

Pour la tâche de traduction, **Sentence Similarity** (Dale *et al.*, 2023b) est une des premières études à explorer les limites de ce qui est possible en utilisant uniquement les représentations internes du modèle. Ils construisent d'abord un pipeline de détection/mitigation en se basant sur la contribution de la source à la traduction (basée sur les scores d'attention) et explorent ensuite l'utilisation de modèle arbitre externe pour calculer la *sentence similarity* améliorant ainsi leurs résultats de manière significative.

Discussion : Il est important de prendre en considération le coût de ces méthodes à l'inférence : l'utilisation de méthodes d'arbitrage ou d'échantillonnage entraîne une génération de tokens supplémentaires.

Étape	Référence	Data	Détection	Mitigation
Pré-entraînement	Filtrage et déduplication (Penedo *et al.*, 2024)	✓	✗	✓
	Synthétisation de la donnée (Abdin *et al.*, 2024)	✓	✗	✓
Alignement	R-Tunning (Zhang *et al.*, 2024)	✗	✗	✓
	GRAIT (Zhu *et al.*, 2025)	✗	✗	✓
	DPO	✗	✗	✓
	CPO (Tang *et al.*, 2025)	✗	✗	✓
Modification de l'architecture	SAPLMA (Azaria & Mitchell, 2023)	✓	✓	✗
	LLM-CHECK (Sriramanan *et al.*, 2024)	✗	✓	✗
	SAE (Ferrando *et al.*, 2024)	✗	✓	✓
	LapEigvals (Binkowski *et al.*, 2025)	✗	✓	✗
Pré-génération	Prompting (Brown *et al.*, 2020)	✗	✗	✓
	Prefix-Tuning (Li & Liang, 2021)	✗	✗	✓
	RAG (Lewis *et al.*, 2021)	✗	✗	✓
Décodage	UABS (Xiao & Wang, 2021)	✗	✗	✓
	CAD (Shi *et al.*, 2024)	✗	✗	✓
	REFIND (Lee & Yu, 2025)	✗	✗	✓
Post-génération	FAVA (Mishra *et al.*, 2024)	✓	✓	✓
	INSIDE (Chen *et al.*, 2023)	✗	✓	✗
	SelfRAG (Asai *et al.*, 2023)	✓	✗	✓
	FActScore (Min *et al.*, 2023)	✗	✓	✗
	Sentence Similarity (Dale *et al.*, 2023b)	✗	✓	✓
	SELFCHECKGPT (Manakul *et al.*, 2023)	✓	✓	✗
	Entropie Sémantique (Farquhar *et al.*, 2024)	✗	✓	✗
	CoVe (Dhuliawala *et al.*, 2023)	✗	✓	✓
	A Stitch in Time Saves Nine (Varshney *et al.*, 2023)	✗	✓	✓

TABLE 2 – Technique de détection et mitigation en fonction de leur place dans le cycle de vie d'un système LLM. Les colonnes indiquent si un jeu de donnée est présent avec l'étude, si il s'agit d'une stratégie détection et/ou de mitigation des hallucinations.

5 Discussion et perspectives futures

Les différentes techniques d'évaluation, de détection et de mitigation présentées dans cet état de l'art illustrent la richesse et la complexité des approches actuelles pour fiabiliser l'intégration et la transparence des LLMs dans des systèmes applicatifs. Chaque étape du développement – du pré-entraînement à la post-génération – offre des leviers d'amélioration qui contribuent à la robustesse des modèles.

D'une part, les méthodes et données d'évaluation jouent un rôle central en fournissant des repères quantitatifs indispensables. Leur diversité permet de mesurer avec finesse l'impact des différentes stratégies sur la factualité et la fidélité des réponses générées. En revanche, ces évaluations peinent à se démocratiser de manière uniforme et il est difficile de trouver des données exhaustives sur l'évaluation de la plupart des modèles.

D'autre part, les avancées dans l'alignement et la modification architecturale des modèles montrent que la mitigation des hallucinations ne se limite pas à des ajustements en post-traitement, mais qu'elle peut être intégrée dans l'entraînement du modèle. Les techniques telles que le R-Tuning, la

quantification de l'incertitude via SAEs ou modèles auxiliaires ouvrent la voie à des systèmes capables non seulement de détecter leurs propres erreurs, mais aussi de les corriger de manière proactive et ce, à un coût inférieur à celui des stratégies de prompting classiques. Ces approches redonnent donc de l'importance à des modèles auxiliaires moins coûteux, proposant une alternative plus efficace et ciblée à la simple croissance des architectures classiques.

En perspective, la recherche future devrait s'orienter vers l'intégration de plusieurs approches dans des systèmes plus complexes. Leurs interactions pourraient ouvrir la voie à une progression vers des systèmes véritablement résilients face aux hallucinations. Par ailleurs, une large partie des techniques présentées dans cet état de l'art repose sur des LLMs qui, à mesure que les modèles sous-jacents s'améliorent, devraient bénéficier d'améliorations significatives dans leur efficacité en termes d'évaluation, de détection ou de mitigation.

6 Limites

Compte tenu de la rapidité d'évolution du domaine, cet état de l'art a été construit en s'appuyant sur des synthèses existantes et une analyse des publications de l'année passée parues sur des sources variées (archives, conférences, etc.) ainsi que sur les références bibliographiques de ces dernières. Cette méthodologie, non systématique, implique que l'état de l'art n'est pas exhaustif.

7 Conclusion

Ce travail montre que la quantification, la détection et la mitigation des hallucinations dans les LLMs constituent des domaines en pleine évolution, avec des avancées considérables ces dernières années. La diversité et l'efficacité des stratégies présentées montrent qu'une prise en compte rigoureuse des hallucinations dans l'architecture des systèmes LLMs peut drastiquement améliorer leurs performances et leur fiabilité.

Les techniques actuelles, qu'elles interviennent au niveau du pré-entraînement, de l'alignement, de la modification architecturale ou en post-génération, ouvrent la voie à des approches permettant non seulement de détecter les erreurs, mais également de les corriger de manière proactive. En parallèle, l'essor de méthodes peu coûteuses et efficaces, ainsi que des stratégies d'évaluation de plus en plus sophistiquées, laisse entrevoir la possibilité d'un déploiement de LLMs plus responsables, performants et transparents quant à leurs forces et leurs limites.

8 Remerciements

Je tiens à remercier d'une part Sophie Rosset, Thomas Lavergne, Christophe Servan, Sahar Ghannay ainsi que toute l'équipe encadrante au sein du LISN pour m'avoir permis de commencer ces recherches aussi efficacement, tout en bénéficiant de leur riche expertise sur le sujet ainsi que d'un environnement de travail idéal. Je souhaite d'autre part remercier Christophe Séjourné et plus généralement SCIAM pour m'avoir donné l'opportunité de me consacrer à ces travaux tout en me permettant de les valoriser.

Références

ABDIN M. *et al.* (2024). Phi-4 Technical Report. DOI : 10.48550/arXiv.2412.08905.

ANTHROPIC (2023). Introducing Claude. https ://www.anthropic.com/news/introducing-claude.

ASAI A., WU Z., WANG Y., SIL A. & HAJISHIRZI H. (2023). Self-RAG : Learning to Retrieve, Generate, and Critique through Self-Reflection. DOI : 10.48550/arXiv.2310.11511.

AZARIA A. & MITCHELL T. (2023). The Internal State of an LLM Knows When It's Lying. DOI : 10.48550/arXiv.2304.13734.

BINKOWSKI J., JANIAK D., SAWCZYN A., GABRYS B. & KAJDANOWICZ T. (2025). Hallucination Detection in LLMs Using Spectral Features of Attention Maps. DOI : 10.48550/arXiv.2502.17598.

BROWN T. B. *et al.* (2020). Language Models are Few-Shot Learners. DOI : 10.48550/arXiv.2005.14165.

CHEN C., LIU K., CHEN Z., GU Y., WU Y., TAO M., FU Z. & YE J. (2023). INSIDE : LLMs' Internal States Retain the Power of Hallucination Detection. In *The Twelfth International Conference on Learning Representations*.

CHEN D., BOLTON J. & MANNING C. D. (2016). A Thorough Examination of the CNN/Daily Mail Reading Comprehension Task. DOI : 10.48550/arXiv.1606.02858.

CLARK P., COWHEY I., ETZIONI O., KHOT T., SABHARWAL A., SCHOENICK C. & TAFJORD O. (2018). Think you have Solved Question Answering ? Try ARC, the AI2 Reasoning Challenge. DOI : 10.48550/arXiv.1803.05457.

DALE D. *et al.* (2023a). HalOmi : A Manually Annotated Benchmark for Multilingual Hallucination and Omission Detection in Machine Translation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, p. 638–653, Singapore : Association for Computational Linguistics. DOI : 10.18653/v1/2023.emnlp-main.42.

DALE D., VOITA E., BARRAULT L. & COSTA-JUSSÀ M. R. (2023b). Detecting and Mitigating Hallucinations in Machine Translation : Model Internal Workings Alone Do Well, Sentence Similarity Even Better. In A. ROGERS, J. BOYD-GRABER & N. OKAZAKI, Éds., *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*, p. 36–50, Toronto, Canada : Association for Computational Linguistics. DOI : 10.18653/v1/2023.acl-long.3.

DHULIAWALA S., KOMEILI M., XU J., RAILEANU R., LI X., CELIKYILMAZ A. & WESTON J. (2023). Chain-of-Verification Reduces Hallucination in Large Language Models. DOI : 10.48550/arXiv.2309.11495.

DONG Y., WANG S., GAN Z., CHENG Y., CHEUNG J. C. K. & LIU J. (2020). Multi-Fact Correction in Abstractive Text Summarization. In B. WEBBER, T. COHN, Y. HE & Y. LIU, Éds., *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, p. 9320–9331, Online : Association for Computational Linguistics. DOI : 10.18653/v1/2020.emnlp-main.749.

FARQUHAR S., KOSSEN J., KUHN L. & GAL Y. (2024). Detecting hallucinations in large language models using semantic entropy. *Nature*, **630**(8017), 625–630. DOI : 10.1038/s41586-024-07421-0.

FERRANDO J., OBESO O. B., RAJAMANOHARAN S. & NANDA N. (2024). Do I Know This Entity ? Knowledge Awareness and Hallucinations in Language Models. In *The Thirteenth International Conference on Learning Representations*.

GAO Y., WANG L., FANG J., HU L. & CHENG J. (2023). Empower Your Model with Longer and Better Context Comprehension. DOI : 10.48550/arXiv.2307.13365.

GRATTAFIORI A. *et al.* (2024). The Llama 3 Herd of Models. DOI : 10.48550/arXiv.2407.21783.

HENDRYCKS D., BURNS C., BASART S., ZOU A., MAZEIKA M., SONG D. & STEINHARDT J. (2021). Measuring Massive Multitask Language Understanding. DOI : 10.48550/arXiv.2009.03300.

HERMANN K. M., KOCISKY T., GREFENSTETTE E., ESPEHOLT L., KAY W., SULEYMAN M. & BLUNSOM P. (2015). Teaching Machines to Read and Comprehend. In *Advances in Neural Information Processing Systems*, volume 28 : Curran Associates, Inc.

HONG G. *et al.* (2024). The Hallucinations Leaderboard – An Open Effort to Measure Hallucinations in Large Language Models. DOI : 10.48550/arXiv.2404.05904.

HUANG L. *et al.* (2024). A Survey on Hallucination in Large Language Models : Principles, Taxonomy, Challenges, and Open Questions. *ACM Transactions on Information Systems*, p. 3703155. DOI : 10.1145/3703155.

JI Z., LEE N., FRIESKE R., YU T., SU D., XU Y., ISHII E., BANG Y. J., MADOTTO A. & FUNG P. (2023). Survey of Hallucination in Natural Language Generation. *ACM Comput. Surv.*, **55**(12), 248 :1–248 :38. DOI : 10.1145/3571730.

JOSHI M., CHOI E., WELD D. S. & ZETTLEMOYER L. (2017). TriviaQA : A Large Scale Distantly Supervised Challenge Dataset for Reading Comprehension. DOI : 10.48550/arXiv.1705.03551.

KWIATKOWSKI T. *et al.* (2019). Natural Questions : A Benchmark for Question Answering Research. *Transactions of the Association for Computational Linguistics*, **7**, 453–466. DOI : 10.1162/tacl_a_00276.

LAI G., XIE Q., LIU H., YANG Y. & HOVY E. (2017). RACE : Large-scale ReAding Comprehension Dataset From Examinations. DOI : 10.48550/arXiv.1704.04683.

LEE D. & YU H. (2025). REFIND : Retrieval-Augmented Factuality Hallucination Detection in Large Language Models. DOI : 10.48550/arXiv.2502.13622.

LEE K., CHANG M.-W. & TOUTANOVA K. (2019). Latent Retrieval for Weakly Supervised Open Domain Question Answering. In A. KORHONEN, D. TRAUM & L. MÀRQUEZ, Éds., *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, p. 6086–6096, Florence, Italy : Association for Computational Linguistics. DOI : 10.18653/v1/P19-1612.

LEWIS P. *et al.* (2021). Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. DOI : 10.48550/arXiv.2005.11401.

LI J., CHENG X., ZHAO W. X., NIE J.-Y. & WEN J.-R. (2023). HaluEval : A Large-Scale Hallucination Evaluation Benchmark for Large Language Models. DOI : 10.48550/arXiv.2305.11747.

LI X. L. & LIANG P. (2021). Prefix-Tuning : Optimizing Continuous Prompts for Generation. DOI : 10.48550/arXiv.2101.00190.

LIN S., HILTON J. & EVANS O. (2022). TruthfulQA : Measuring How Models Mimic Human Falsehoods. DOI : 10.48550/arXiv.2109.07958.

LIU J. (2025). Liujch1998/memo-trap.

LONGPRE S., PERISETLA K., CHEN A., RAMESH N., DUBOIS C. & SINGH S. (2021). Entity-Based Knowledge Conflicts in Question Answering. In M.-F. MOENS, X. HUANG, L. SPECIA & S. W.-T. YIH, Éds., *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, p. 7052–7063, Online and Punta Cana, Dominican Republic : Association for Computational Linguistics. DOI : 10.18653/v1/2021.emnlp-main.565.

MALLEN A., ASAI A., ZHONG V., DAS R., KHASHABI D. & HAJISHIRZI H. (2023). When Not to Trust Language Models : Investigating Effectiveness of Parametric and Non-Parametric Memories. DOI : 10.48550/arXiv.2212.10511.

MANAKUL P., LIUSIE A. & GALES M. J. F. (2023). SelfCheckGPT : Zero-Resource Black-Box Hallucination Detection for Generative Large Language Models. DOI : 10.48550/arXiv.2303.08896.

MAYNEZ J., NARAYAN S., BOHNET B. & MCDONALD R. (2020). On Faithfulness and Factuality in Abstractive Summarization. In D. JURAFSKY, J. CHAI, N. SCHLUTER & J. TETREAULT, Éds., *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, p. 1906–1919, Online : Association for Computational Linguistics. DOI : 10.18653/v1/2020.acl-main.173.

MCKENZIE I. R. *et al.* (2024). Inverse Scaling : When Bigger Isn't Better. DOI : 10.48550/arXiv.2306.09479.

MIN S., KRISHNA K., LYU X., LEWIS M., YIH W.-T., KOH P. W., IYYER M., ZETTLEMOYER L. & HAJISHIRZI H. (2023). FActScore : Fine-grained Atomic Evaluation of Factual Precision in Long Form Text Generation. DOI : 10.48550/arXiv.2305.14251.

MISHRA A., ASAI A., BALACHANDRAN V., WANG Y., NEUBIG G., TSVETKOV Y. & HAJISHIRZI H. (2024). Fine-grained Hallucination Detection and Editing for Language Models. DOI : 10.48550/arXiv.2401.06855.

MISTRAL AI (2024). Le Chat | Mistral AI. https ://mistral.ai/en/news/le-chat-mistral.

NALLAPATI R., ZHOU B., DOS SANTOS C. N., GULCEHRE C. & XIANG B. (2016). Abstractive Text Summarization Using Sequence-to-Sequence RNNs and Beyond. DOI : 10.48550/arXiv.1602.06023.

NARAYAN S., COHEN S. B. & LAPATA M. (2018). Don't Give Me the Details, Just the Summary ! Topic-Aware Convolutional Neural Networks for Extreme Summarization. DOI : 10.48550/arXiv.1808.08745.

NIU C. *et al.* (2024). RAGTruth : A Hallucination Corpus for Developing Trustworthy Retrieval-Augmented Language Models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*, p. 10862–10878, Bangkok, Thailand : Association for Computational Linguistics. DOI : 10.18653/v1/2024.acl-long.585.

OPEN AI (2022). Introducing ChatGPT. https ://openai.com/index/chatgpt/.

OPEN AI (2025). GPT-4.5 system card. https ://openai.com/index/gpt-4-5-system-card/.

OUYANG L. *et al.* (2022). Training language models to follow instructions with human feedback. DOI : 10.48550/arXiv.2203.02155.

PENEDO G., KYDLÍČEK H., ALLAL L. B., LOZHKOV A., MITCHELL M., RAFFEL C., WERRA L. V. & WOLF T. (2024). The FineWeb Datasets : Decanting the Web for the Finest Text Data at Scale. DOI : 10.48550/arXiv.2406.17557.

RAJPURKAR P., JIA R. & LIANG P. (2018). Know What You Don't Know : Unanswerable Questions for SQuAD. DOI : 10.48550/arXiv.1806.03822.

RAJPURKAR P., ZHANG J., LOPYREV K. & LIANG P. (2016). SQuAD : 100,000+ Questions for Machine Comprehension of Text. DOI : 10.48550/arXiv.1606.05250.

SHI W., HAN X., LEWIS M., TSVETKOV Y., ZETTLEMOYER L. & YIH W.-T. (2024). Trusting Your Evidence : Hallucinate Less with Context-aware Decoding. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies (Volume 2 : Short Papers)*, p. 783–791, Mexico City, Mexico : Association for Computational Linguistics. DOI : 10.18653/v1/2024.naacl-short.69.

SRIRAMANAN G., BHARTI S., SADASIVAN V. S., SAHA S., KATTAKINDA P. & FEIZI S. (2024). LLM-Check : Investigating Detection of Hallucinations in Large Language Models.

SUTAWIKA L. (2025). EleutherAI/lm-evaluation-harness : V0.4.8. Zenodo. DOI : 10.5281/zenodo.14970487.

TANG Z., CHATTERJEE R. & GARG S. (2025). Mitigating Hallucinated Translations in Large Language Models with Hallucination-focused Preference Optimization. DOI : 10.48550/arXiv.2501.17295.

THORNE J., VLACHOS A., CHRISTODOULOPOULOS C. & MITTAL A. (2018). FEVER : A large-scale dataset for Fact Extraction and VERification. DOI : 10.48550/arXiv.1803.05355.

VARSHNEY N., YAO W., ZHANG H., CHEN J. & YU D. (2023). A Stitch in Time Saves Nine : Detecting and Mitigating Hallucinations of LLMs by Validating Low-Confidence Generation. DOI : 10.48550/arXiv.2307.03987.

VASWANI A., SHAZEER N., PARMAR N., USZKOREIT J., JONES L., GOMEZ A. N., KAISER L. & POLOSUKHIN I. (2017). Attention Is All You Need. DOI : 10.48550/arXiv.1706.03762.

VÁZQUEZ R. *et al.* (2025). Welcome to SemEval-2025 Task-3 — Mu-SHROOM, the Multilingual Shared-task on Hallucinations and Related Observable Overgeneration Mistakes. https ://helsinki-nlp.github.io/shroom/.

WEI J., KARINA N., CHUNG H. W., JIAO Y. J., PAPAY S., GLAESE A., SCHULMAN J. & FEDUS W. (2024). Measuring short-form factuality in large language models. DOI : 10.48550/arXiv.2411.04368.

XIAO Y. & WANG W. Y. (2021). On Hallucination and Predictive Uncertainty in Conditional Language Generation. In P. MERLO, J. TIEDEMANN & R. TSARFATY, Éds., *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics : Main Volume*, p. 2734–2744, Online : Association for Computational Linguistics. DOI : 10.18653/v1/2021.eacl-main.236.

XU Z., JAIN S. & KANKANHALLI M. (2025). Hallucination is Inevitable : An Innate Limitation of Large Language Models. DOI : 10.48550/arXiv.2401.11817.

ZHANG H., DIAO S., LIN Y., FUNG Y. R., LIAN Q., WANG X., CHEN Y., JI H. & ZHANG T. (2024). R-Tuning : Instructing Large Language Models to Say 'I Don't Know'. DOI : 10.48550/arXiv.2311.09677.

ZHANG Y. *et al.* (2023). Siren's Song in the AI Ocean : A Survey on Hallucination in Large Language Models. DOI : 10.48550/arXiv.2309.01219.

ZHOU J., LU T., MISHRA S., BRAHMA S., BASU S., LUAN Y., ZHOU D. & HOU L. (2023). Instruction-Following Evaluation for Large Language Models. DOI : 10.48550/arXiv.2311.07911.

ZHU R., JIANG Z., WU J., MA Z., SONG J., BAI F., LIN D., WU L. & HE C. (2025). GRAIT : Gradient-Driven Refusal-Aware Instruction Tuning for Effective Hallucination Mitigation. DOI : 10.48550/arXiv.2502.05911.