Détection des omissions dans les résumés médicaux générés par les grands modèles de langue

Achir Oukelmoun[1] Nasredine Semmar[2] Gaël de Chalendar[2]
Clément Cormi[4] Mariame Oukelmoun[3] Eric Vibert[4] Marc-Atoine Allard [4]
(1) Université Paris-Saclay, 3 Rue Joliot Curie, 91190 Gif-sur-Yvette, France
(2) Université Paris-Saclay, CEA, List, 2 Boulevard Thomas Gobert, 91120 Palaiseau, France
(3) Faculté de médecine et de pharmacie de Rabat, Impasse Souissi, Rabat 10100, Maroc
(4) Hôpital Paul-Brousse AP-HP, 12 Avenue Paul Vaillant Couturier, 94800 Villejuif, France
achir.oukelmoun@gmail.com, nasredine.semmar@cea.fr,
gael.de-chalendar@cea.fr, clement.cormi-ext@aphp.fr, oukelmoun@gmail.com,
eric.vibert@aphp.fr, marcantoine.allard@aphp.fr

RÉSUMÉ

Les grands modèles de langue (LLMs pour Large Language Models) sont de plus en plus utilisés pour résumer des textes médicaux, mais ils risquent d'omettre des informations critiques, compromettant ainsi la prise de décision. Contrairement aux hallucinations, les omissions concernent des faits essentiels absents. Cet article introduit un jeu de données validé en français pour détecter ces omissions et propose EmbedKDECheck, une approche frugale et sans référence. A l'opposé des méthodes basées sur les LLMs, cette approche utilise des plongements lexicaux issus d'un modèle de Traitement Automatique des Langues (TAL) léger combinant FastText et Word2Vec selon un algorithme précis couplé à un modèle non-supervisé fournissant un score d'anomalie. Cette approche permet d'identifier efficacement les omissions à faible coût computationnel. EmbedKDECheck a été évalué face aux frameworks de l'état de l'art (SelfCheckGPT, ChainPoll, G-Eval et GPTScore) et a montré de bonnes performances. Notre méthode renforce l'évaluation de la fiabilité des LLMs et contribue à une prise de décision médicale plus sûre.

ABSTRACT

Detecting Omissions in LLM-Generated Medical Summaries

Large Language Models (LLMs) are increasingly used in medical text summarization, but they risk omitting critical information, impacting decision-making. Unlike hallucinations, omissions involve missing essential facts. This paper introduces a validated French dataset for omission detection and proposes EmbedKDECheck, a frugal, reference-free method to assess LLM-generated summaries. Unlike LLM-based approaches, it uses embeddings from a lightweight Natural Language Processing (NLP) model combined with an anomaly scores. This enables effective omission detection with minimal computational cost. EmbedKDECheck was benchmarked against state-of-the-art black-box frameworks (SelfCheckGPT, ChainPoll, G-Eval and GPTScore) and demonstrated strong performance. Our method enhances LLM reliability assessment, contributing to safer medical decision-making.

MOTS-CLÉS : LLMs, détection d'omissions, évaluation, qualité des résumés médicaux, IA frugale.

KEYWORDS: LLMs, omission detection, evaluation, medical summary quality, frugal AI.

ARTICLE : **Accepté à** TALN 2025.

1 Introduction

L'avènement des grands modèles de langage (LLMs) comme GPT d'OpenAI (Achiam *et al.*, 2023) a révolutionné de nombreux domaines en permettant la génération de texte cohérent et contextuellement pertinent. Ces modèles sont utilisés dans des applications variées, allant des agents conversationnels à la création de résumés et de rapports détaillés. Dans le domaine médical, les LLMs ont montré un potentiel significatif pour générer des rapports complets, aidant ainsi les professionnels de santé à prendre des décisions éclairées (Alberts *et al.*, 2023).

Malgré ces avancées, un défi majeur reste non résolu : la détection des omissions dans les textes générés par les LLMs. Les ensembles de données et les cadres d'évaluation existants se concentrent principalement sur les hallucinations—c'est-à-dire les cas où le texte généré contient des informations incorrectes ou fabriquées (Li *et al.*, 2024). Bien que la détection des hallucinations soit cruciale, la question des omissions, où des informations essentielles de l'entrée originale sont absentes, représente un risque particulièrement grave, notamment dans le domaine médical. De telles omissions peuvent conduire à des résumés incomplets, compromettant ainsi la prise en charge et les traitements des patients. En effet, dans un contexte clinique, l'absence de mention de symptômes majeurs ou de contre-indications peut avoir des conséquences lourdes pour la sécurité des patients.

Par souci de clarté, voici les définitions adoptées :

— **Hallucination** : génération d'informations factuellement incorrectes, non vérifiables ou fabriquées, qui ne figurent pas dans le texte source ou qui contredisent des faits établis.
— **Omission** : absence ou suppression d'informations pertinentes et présentes dans le texte source, entraînant un résumé ou une réponse incomplète et susceptible de négliger des éléments critiques.

Dans ce travail, nous nous concentrons principalement sur la détection des omissions, une problématique encore peu explorée mais essentielle pour garantir la fiabilité des systèmes de génération automatique, en particulier dans des domaines sensibles comme la médecine.

Dans le contexte du secteur de la santé en France, l'utilisation de fournisseurs externes ou de solutions cloud publiques est souvent impraticable en raison de réglementations strictes en matière de confidentialité (AP-HP, 2024). Par conséquent, les implémentations des LLMs reposent généralement sur des modèles plus petits hébergés en interne ou sur des architectures hybrides utilisant des microservices API (Nabla, 2024). Ces contraintes peuvent aggraver le problème des omissions, car l'accès aux modèles LLMs les plus avancés est limité. Il est donc impératif de mettre en place des mécanismes de contrôle qualité efficaces dans ces environnements contraints en ressources.

Nous apportons, dans cet article, deux contributions principales. Premièrement, nous introduisons un nouvel ensemble de données spécifiquement conçu pour évaluer les méthodes de détection des omissions. Cet ensemble de données, rédigé en français, a été soigneusement élaboré et validé par des experts médicaux, dont des chirurgiens spécialisés en chirurgie hépatique et générale. Cette validation garantit que l'ensemble de données reflète fidèlement les situations rencontrées dans le secteur médical, le rendant ainsi particulièrement utile pour le développement et l'évaluation des algorithmes de détection des omissions (L'ensemble de données en question sera publié dans un dépôt Git et intégré à l'article dans sa version finale).

Deuxièmement, nous proposons une approche frugale et indépendante des modèles LLMs, appelée *EmbedKDECheck*, pour détecter les omissions dans les textes générés. Cette méthode exploite les plongements lexicaux en combinaison avec des modèles d'apprentissage automatique non-supervisé de détection d'anomalies pour identifier les informations manquantes. L'accent mis sur la frugalité est essentiel, car il permet à la méthode de fonctionner efficacement dans des environnements aux ressources limitées, sans exiger de lourdes capacités de calcul. Cet aspect est particulièrement important dans le domaine médical, où les ressources informatiques peuvent être restreintes et où la détection rapide des omissions est essentielle pour garantir l'exactitude et la fiabilité de la documentation médicale.

Plus spécifiquement, nous définissons une "omission" comme un cas où le texte généré ne contient pas des informations importantes ou ne reprend pas des détails attendus à partir de l'entrée. Cela peut entraîner des résumés incomplets ou trompeurs, ce qui peut être particulièrement préjudiciable dans les applications médicales.

Dans cette étude, nous présentons des algorithmes combinant des plongements lexicaux et des techniques de détection d'anomalies pour identifier les omissions. Nous évaluons rigoureusement ces algorithmes à l'aide de notre nouvel ensemble de données, nous permettant ainsi d'analyser l'efficacité de notre approche dans le domaine médical. En abordant le problème critique des omissions, ce travail contribue au développement de mécanismes de contrôle qualité robustes pour les textes générés par les LLMs, améliorant ainsi la sécurité et la fiabilité des systèmes d'aide à la décision médicale.

2 Travaux connexes

L'évaluation des résumés générés par les LLMs est un enjeu fondamental, en particulier pour détecter les omissions. Les métriques automatiques permettent d'évaluer la fluidité, la cohérence, la pertinence et la fidélité factuelle (van Schaik & Pugh, 2024). Parmi elles, les métriques sans référence comme BLANC (Vasilyev *et al.*, 2020) et SUPERT (Gao *et al.*, 2020) sont particulièrement intéressantes puisqu'elles évaluent les résumés sans nécessiter de textes de référence.

Les méthodes d'évaluation sans référence se divisent en trois catégories : les méthodes *boîte noire* analysent les sorties des modèles sans accès aux états internes ; par exemple, SelfCheckGPT (Manakul *et al.*, 2023) détecte les hallucinations en comparant différentes réponses échantillonnées. Les méthodes *boîte blanche* nécessitent un accès complet au modèle pour analyser ses poids et activations (Azaria & Mitchell, 2023). Enfin, les méthodes *boîte grise* offrent un accès partiel, comme l'analyse des probabilités au niveau des tokens. Notre approche, *EmbedKDECheck*, est une méthode *boîte noire* qui combine des plongements lexicaux d'un modèle tiers et la détection d'anomalies pour identifier les omissions.

Les avancées récentes en évaluation *boîte noire* apportent des perspectives intéressantes. La méthode *ChainPoll* (Friel & Sanyal, 2023) surpasse SelfCheckGPT (Manakul *et al.*, 2023) et GPTScore (Fu *et al.*, 2024) sur le benchmark *RealHall* (Friel & Sanyal, 2023), qui reflète fidèlement l'usage des LLMs. Pour améliorer l'évaluation des résumés, la méthode *G-Eval* (Liu *et al.*, 2023) intègre le raisonnement par chaîne de pensée (Wei *et al.*, 2022).

Concernant les ensembles de données (datasets) existants, ils ne ciblent pas les omissions. Nous pouvons citer : QAGS (Wang *et al.*, 2020) qui se concentre sur la fidélité factuelle sans traiter le contenu manquant, DROP (Dua *et al.*, 2019) qui met l'accent sur le raisonnement discret, SummEval

(Fabbri *et al.*, 2021) qui évalue la cohérence et la pertinence des résumés, mais ne prend pas en compte la détection des omissions, et *RealHall* (Friel & Sanyal, 2023) qui établit un benchmark pour la détection des hallucinations mais ne couvre pas les omissions. Ces ensembles de données ne permettent donc pas d'évaluer précisément la détection des omissions. Notre ensemble de données répond à ce besoin en fournissant un cadre spécifique pour cette tâche, adapté au domaine médical et validé par des experts chirurgiens spécialisés en chirurgie hépatique et générale.

En conclusion, l'évaluation des résumés générés par les LLMs est un défi complexe et les différentes méthodes semblent être complémentaire. Notre approche, relevant des métriques sans référence et *boîte noire*, se concentre sur l'évaluation de la fidélité factuelle et vise à pallier les limites des méthodes existantes en offrant une meilleure détection des omissions. La méthode *EmbedKDECheck*, basée sur les plongements lexicaux et la détection d'anomalies, garantit une évaluation robuste sans exiger de ressources computationnelles excessives. En s'appuyant sur un ensemble de données spécifiquement conçu pour le domaine médical et validé par des experts, notre approche constitue une solution efficace pour améliorer la qualité et la fiabilité de la documentation médicale générée par les LLMs.

3 Approche et Expérimentations

3.1 EmbedKDECheck : Une Méthode Frugale pour l'Évaluation des Omissions

La méthode EmbedKDECheck évalue la cohérence factuelle en détectant les omissions dans les résumés ou reformulations sans nécessiter de références ni d'accès aux états intermédiaires des LLMs. Cette méthode boîte noire, indépendante des références, fonctionne localement, la rendant indépendante de l'infrastructure et peu coûteuse en calcul.

Étant donné une entrée (ex. : un rapport) et une sortie (ex. : un résumé), EmbedKDECheck attribue :
- Un score global d'omission.
- Des indicateurs locaux de contenu critique manquant.

L'approche analyse les distributions des plongements lexicaux des segments de texte en utilisant l'Estimation de Densité par Noyaux (KDE - Kernel Density Estimator) (Węglarczyk, 2018). En modélisant les densités de probabilité, elle identifie le contenu de l'entrée non couvert par la sortie, signalant ainsi les omissions (Figure 1).

La KDE (Węglarczyk, 2018), une méthode non paramétrique d'estimation de densité (Parzen, 1962), attribue des probabilités aux distributions des plongements lexicaux :

$$\hat{f}_h(\mathbf{x}) = \frac{1}{n}\sum_{i=1}^{n} K_h(\mathbf{x} - \mathbf{x}_i)$$

où $K_h(\mathbf{x})$ est un noyau Gaussien et h le paramètre de bande passante. La KDE permet une estimation adaptative de la densité, facilitant la détection des omissions. Le paramètre h contrôle la finesse de l'estimation : un h plus grand génère une estimation plus lisse, tandis qu'un h plus petit accroît la sensibilité aux variations locales, améliorant ainsi la détection des omissions minimes, mais introduisant potentiellement du bruit.

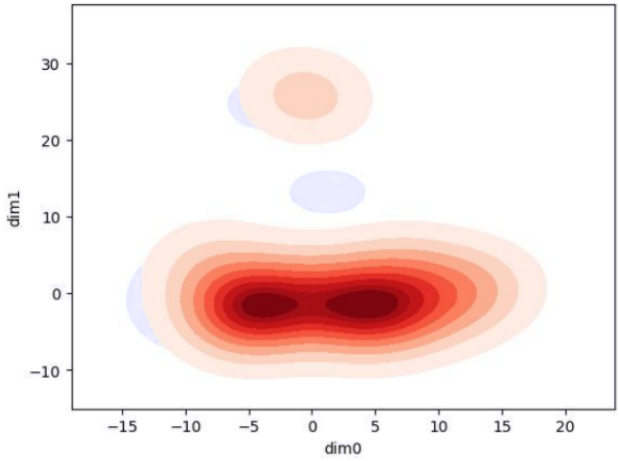

FIGURE 1 – Couverture des plongements lexicaux d'entrée (bleu) et de sortie (rouge) après une réduction de dimension par ACP (Analyse en Composantes Principales - PCA pour Principal Component Analysis). Les zones violettes représentent le contenu de l'entrée, tandis que les zones rouges correspondent à la sortie. Les zones bleues non couvertes indiquent des informations manquantes ou insuffisamment représentées, signalant ainsi des omissions.

La Figure 2 résume les principales étapes d'EmbedKDECheck :

— Segmentation du texte d'entrée et de sortie et extraction des plongements lexicaux.
— Construction des distributions KDE sur les plongements lexicaux.
— Calcul des scores d'omission via les ratios de probabilité :

$$\text{om}_{\text{score}}(X_j) = \frac{\text{KDE}(X_j)}{\min\limits_{w \in \bigcup_{i=1}^{M} Y_i} \text{KDE}(w)} \tag{1}$$

— Un score faible indique des omissions probables.

Pour une meilleure efficacité et frugalité, EmbedKDECheck exploite des plongements lexicaux FTW2V (Oukelmoun *et al.*, 2023), combinant FastText (Bojanowski *et al.*, 2017) et Word2Vec (Mikolov *et al.*, 2013). Ces plongements lexicaux ont été ajustés sur un corpus médical diversié, plus généraliste, extrait en vrac et indépendant de l'ensemble de données utilisé pour l'évaluation. L'ensemble de données d'affinement comporte 32 millions de mots et a été fourni par l'hôpital partenaire. L'ajustement fin a été exécuté sur CPU, nécessitant seulement 2,4 Go de RAM et un temps d'entraînement de 1 heure 35 minutes sur un processeur Intel i7-10750H.

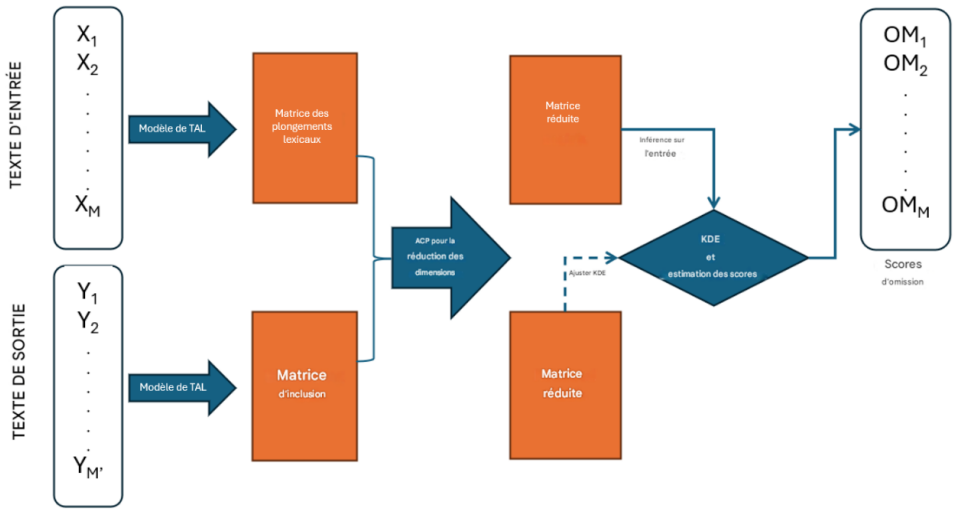

FIGURE 2 – Vue d'ensemble de la détection d'omission utilisant KDE.

3.2 Modèles et Métriques de Benchmarking

Cette section présente les modèles boîte noire et indépendants des références utilisés pour comparer EmbedKDECheck.

— **SelfCheckGPT** (Manakul *et al.*, 2023) : Évalue un résumé en le comparant à plusieurs variantes générées pour la même entrée. Pour ce benchmark, nous avons échantillonné huit résumés par entrée. Les plongements lexicaux ont été calculés via le modèle OpenAI *text-embedding-ada-002*, et le score final obtenu comme la moyenne de $1-$ similarité cosinus entre le résumé évalué et les échantillons. Cette méthode a été testée avec *GPT-3.5 Turbo* et *GPT-4*.
— **ChainPoll** (Friel & Sanyal, 2023) : Basé sur l'interrogation des modèles LLMs. Testé avec *GPT-3.5 Turbo* et *GPT-4*, ChainPoll demande au LLMs de suivre une chaîne de raisonnement avant de prédire si un résumé contient une omission.
— **G-Eval** (Liu *et al.*, 2023) : Utilise aussi le raisonnement Chain of Thought (COT), mais demande un score détaillé plutôt qu'une simple prédiction.
— **GPTScore** (Fu *et al.*, 2024) : Calcule un score d'omission basé sur la similarité cosinus entre les plongements lexicaux de l'entrée et de la sortie.

3.3 Jeu de données d'évaluation

3.3.1 Description

L'utilisation de données réelles ou de comptes rendus médicaux anonymisés en France n'était pas envisageable en raison des règlements stricts en matière de protection de la vie privée et des considérations éthiques. Les lois sur la protection des données, telles que le Règlement Général sur la Protection des Données (RGPD), imposent des restrictions importantes sur l'utilisation des

données personnelles, en particulier dans les domaines sensibles comme la santé. De plus, garantir une anonymisation totale des comptes rendus médicaux est particulièrement complexe en raison du niveau de détail des informations qu'ils contiennent (anonymization, 2024). Même avec des techniques d'anonymisation, le risque de ré-identification persiste, ce qui pourrait compromettre la confidentialité et la vie privée des patients (reinditification, 2024).

Les comptes rendus médicaux rédigés après des consultations ou des interventions chirurgicales dans les établissements de santé en France sont souvent manuscrits par le personnel médical, notamment les chirurgiens ou les assistants médicaux, ce qui peut entraîner des erreurs et des incohérences. Des variations dans le style d'écriture, le niveau de détail et l'adhérence à des modèles structurés sont courantes. Par conséquent, les rapports peuvent contenir des irrégularités et s'écarter du format standard, ce qui peut affecter la clarté et l'exhaustivité des informations ainsi que la performance de l'analyse linguistique qui leur est appliquée.

De plus, les jeux de données publics contenant des comptes rendus médicaux et des résumés utilisables pour la vérification de la cohérence factuelle sont rares et limités à quelques centaines d'exemples (Luo *et al.*, 2024). Le processus de qualification de ces jeux de données n'est pas suffisamment approfondi, et ils ne sont pas toujours adaptés à notre objectif de détection des omissions. Cela a motivé la création d'un nouveau jeu de données, joint à cette soumission et qui sera publié en open-source pour contribuer aux travaux de la communauté scientifique dans le domaine médical. De tels jeux de données en français sont encore plus rares, soulignant l'importance de cette contribution.

Pour répondre à ces défis, la première étape a consisté à anonymiser 50 rapports médicaux fournis par des experts médicaux. Un rapport médical typique comprend des sections telles que *MOTIF D'HOSPITALISATION*, *ANTÉCÉDENTS*, *HISTOIRE DE LA MALADIE*, *CLINIQUE* et *EVOLUTION DANS LE SERVICE*.

Pour créer un jeu de données plus large, entièrement fictif mais toujours réaliste, le processus suivant a été mis en place : pour chaque rapport anonymisé, les noms, dates et lieux ont été modifiés, générant ainsi 50 rapports fictifs. Ces rapports, représentatifs des dossiers médicaux réels, ont ensuite servi de prompts pour GPT-4-32K. Chaque rapport a été utilisé comme prompt pour générer 15 rapports similaires en suivant des instructions spécifiques. Le modèle a été invité à conserver le format tout en modifiant de manière significative le contenu lié aux situations familiales, aux antécédents médicaux, aux symptômes, aux dates et aux complications. Après la génération des rapports synthétiques, ceux contenant moins de 200 mots ont été exclus, aboutissant à un total de 674 rapports d'une longueur moyenne de 353 mots. Cela a permis de constituer un jeu de données représentatif des comptes rendus médicaux couramment rencontrés en pratique. Pour garantir l'exactitude, la représentativité et la cohérence du jeu de données, une procédure de qualification a été menée, elle est détaillée dans la section suivante.

3.3.2 Évaluation de la qualité

La qualité du jeu de données synthétique a été évaluée selon trois critères : la similarité avec les rapports réels, la diversité lexicale et syntaxique, et l'exactitude du contenu. Deux experts médicaux ont mené une tâche de classification à l'aveugle sur 100 rapports (50 réels et 50 synthétiques) en les classant comme étant générés par un LLM ou réels. Les résultats montrent que l'Expert 1 avait une haute précision (93,8%) mais un rappel faible (30,0%), tandis que l'Expert 2 affichait des performances plus équilibrées. Les scores F1 des deux experts sont de 45,4% et 54,7%, proche de la

performance d'un classifieur aléatoire, suggérant que les rapports synthétiques ressemblent fortement aux données réelles, confirmant leur robustesse.

Pour évaluer la diversité linguistique, les plongements lexicaux CamemBERT (Martin *et al.*, 2020) des rapports synthétiques et réels ont été analysés par ACP (Analyse en Composantes Principales) (Figure 3). Cette analyse montre que les plongements lexicaux couvrent les mêmes zones avec les mêmes concentrations locales, confirmant que le jeu de données synthétique imite les caractéristiques linguistiques des données réelles.

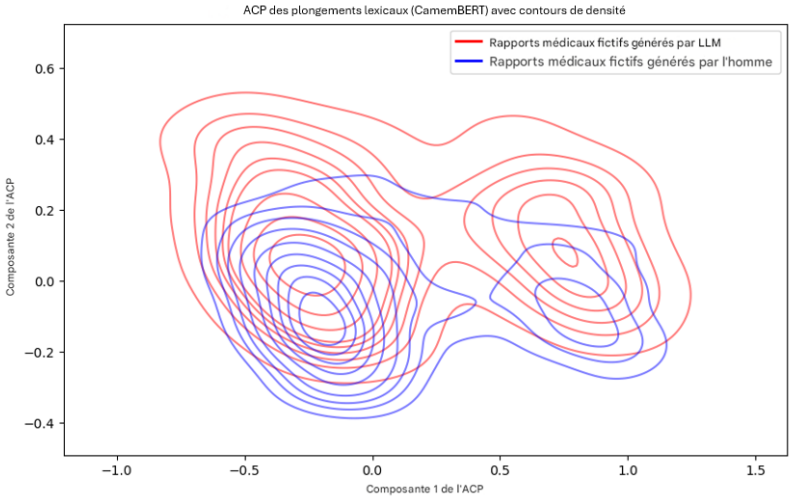

FIGURE 3 – Analyse ACP des plongements lexicaux CamemBERT avec contours de densité.

Enfin, 90 rapports ont été examinés par des experts médicaux pour évaluer la précision des résumés et des omissions. L'évaluation a confirmé que 93 % des résumés complets étaient de bonne qualité. Lors d'un test à l'aveugle ultérieur, les experts ont réussi à identifier les résumés avec omissions et ceux qui étaient complets, atteignant une précision de 95 %. Ce résultat confirme la fiabilité du jeu de données, les omissions identifiées par les experts étant en accord avec celles labellisées dans le jeu de données.

4 Résultats

Les performances du modèle *EmbedKDECheck* pour la détection des omissions dans les résumés générés par les LLM sont évaluées et présentées dans les tableaux 1 et 2. Le code source ainsi que le jeu de données validé par des experts médicaux sont librement accessibles sur GitHub à l'adresse suivante : `https://github.com/achok7893/EmbedKDECheck_OmissionsDection_Dataset_Fr_Healthcare`. Ces ressources sont mises à disposition sous une licence autorisant exclusivement leur utilisation à des fins académiques, avec obligation de citation de la publication associée. Tout autre usage, notamment commercial ou industriel, est strictement interdit sans accord préalable explicite et écrit.

4.1 Détection des omissions

EmbedKDECheck a été testé pour la détection des omissions dans les résumés médicaux en français et comparé à plusieurs algorithmes (Tableau 1). Il a obtenu le meilleur F1-Score de 0,91, avec un rappel de 0,88 et une précision de 0,93, démontrant ainsi sa robustesse dans l'identification des omissions. Bien que les modèles comme ChainPoll et G-Eval présentent un rappel presque parfait, leur tendance à prédire presque chaque instance comme une omission entraîne une précision et un F1-Score plus faibles. En revanche, EmbedKDECheck offre un meilleur équilibre, permettant une détection plus fiable des omissions. Cette méthode identifie également les thématiques omises, fournissant des indications sur la complétude du résumé. Son approche est généralisable à d'autres domaines, ce qui en fait un outil polyvalent pour la détection des omissions là où la frugalité est requise.

Algorithme	**Rappel**	**Précision**	**F1-Score**
SelfCheck-gpt3-turbo	0,79	0,77	0,78
SelfCheck-gpt4	0,86	0,84	0,85
ChainPoll-gpt3-turbo	0,97	0,67	0,51
ChainPoll-gpt4	**0,99**	0,68	0,80
G-Eval-gpt3-turbo	0,98	0,46	0,63
G-Eval-gpt4	0,99	0,68	0,80
GPTScore	0,80	0,79	0,80
EmbedKDECheck	0,88	**0,93**	**0,91**

TABLE 1 – Scores de détection des omissions.

4.2 Evaluation de la frugalité

La frugalité a été évaluée en utilisant les FLOPS estimés (Floating Point Operations Per Second) pour chaque modèle (Tableau 2). Pour les modèles basés sur GPT, les FLOPS ont été calculés en multipliant le nombre de tokens par la taille du modèle. Pour EmbedKDECheck, les FLOPS ont été dérivés du produit du temps d'exécution et des FLOPS maximaux du CPU utilisé. EmbedKDECheck consomme environ 80 000 fois moins que SelfCheck-gpt4.

Modèle	**FLOPS estimés (Tera)**
SelfCheck-gpt4	8804,03
ChainPoll-gpt4	4404,16
G-Eval-gpt4	2936,04
SelfCheck-gpt3-turbo	1545,40
ChainPoll-gpt3-turbo	770,73
G-Eval-gpt3-turbo	513,81
GPTScore	2,14
EmbedKDECheck	**0,11**

TABLE 2 – FLOPS estimés pour les différents modèles.

Ces résultats soulignent l'efficacité et la frugalité d'EmbedKDECheck, en faisant un candidat solide

pour les tâches de résumé nécessitant à la fois performance et efficacité computationnelle.

5 Conclusion et perspectives

L'évaluation du modèle EmbedKDECheck en termes de détection des omissions et d'efficacité computationnelle, présentée dans les tableaux 1 et 2, met en évidence ses points forts. EmbedKDECheck a obtenu le score F1 le plus élevé de 0,91, démontrant une performance robuste avec un rappel de 0,88 et une précision de 0,93. Cela indique son efficacité pour identifier les sujets omis, ce qui est crucial pour l'intégralité des résumés de rapports médicaux, tout en minimisant les faux positifs. De plus, EmbedKDECheck est particulièrement efficace, avec des FLOPS estimés bien plus faibles que ceux d'autres modèles, ce qui en fait une solution rentable, notamment dans des environnements à ressources limitées. Au-delà de ses solides performances, la polyvalence du modèle le rend applicable à divers domaines nécessitant la détection des omissions, tels que les documents juridiques et financiers. Cette capacité à se généraliser accroît l'impact potentiel d'EmbedKDECheck pour garantir des résumés complets et précis dans différents domaines. En outre, l'approche EmbedKDECheck semble prometteuse pour l'identification d'autres types d'erreurs dans les résumés générés par les LLMs. Par exemple, le modèle pourrait être adapté pour détecter la génération de faits fictifs, garantissant que les résumés sont factuellement exacts et fiables. De plus, EmbedKDECheck pourrait être affiné pour repérer les présentations biaisées de l'information, où certains éléments sont disproportionnellement mis en avant, ce qui pourrait fausser l'interprétation du résumé. En étendant ses capacités, EmbedKDECheck pourrait offrir une solution plus complète pour améliorer la qualité et la fiabilité des résumés générés par des LLMs.

Les travaux futurs pourraient se concentrer sur l'optimisation supplémentaire du modèle et son adaptation à d'autres domaines. Cela inclut l'expansion de son évaluation sur plusieurs langues, l'affinement des modèles fournissant les plongmenets lexicaux, et l'intégration de mécanismes de vérification supplémentaires pour détecter des erreurs au-delà des omissions. Ces améliorations contribueront à faire d'EmbedKDECheck un outil plus robuste et fiable pour garantir l'intégrité et l'exactitude des résumés générés par des LLMs. Les considérations éthiques ont également été prises en compte, car les rapports médicaux réels ou anonymisés ne pouvaient pas être utilisés en raison de préoccupations relatives à la confidentialité. À la place, un jeu de données synthétique a été généré à l'aide de GPT-4-32K pour créer des rapports fictifs similaires, garantissant ainsi la conformité aux régulations sur la confidentialité telles que le RGPD.

En résumé, EmbedKDECheck présente une solution puissante et computationnellement efficace pour la détection des omissions tout en offrant des perspectives prometteuses pour des applications plus larges dans l'amélioration de la qualité du contenu généré par l'IA. Son développement et son affinage continus amélioreront encore son utilité et son impact dans divers domaines.

Considérations éthiques

En raison des règlementations RGPD et des enjeux éthiques, aucun rapport médical réel n'a été utilisé. Un jeu de données synthétique a été généré à partir de rapports anonymisés en utilisant GPT-4-32K, garantissant le respect des normes de confidentialité.

Remerciements

Nous tenons à exprimer notre profonde gratitude aux chirurgiens et chirurgiennes spécialisés en chirurgie hépatique et générale pour leur précieuse contribution à l'annotation et à la validation du jeu de données utilisé dans cette étude. Leur expertise clinique a été essentielle pour garantir la qualité et la pertinence médicale des annotations, rendant notre jeu de données particulièrement robuste pour l'évaluation de la détection des omissions.

Nous remercions également les équipes de recherche du **CEA List** pour leur encadrement scientifique, leur soutien méthodologique et les discussions techniques enrichissantes qui ont largement contribué à la maturation de ce travail.

Enfin, nous souhaitons remercier la **Chaire BOPA (Bloc Opératoire Augmenté)** pour son appui institutionnel, son accompagnement stratégique et son engagement à promouvoir des recherches interdisciplinaires à l'interface de la médecine et de l'intelligence artificielle.

Ce travail n'aurait pas été possible sans l'investissement de l'ensemble des partenaires mentionnés.

Références

ACHIAM J., ADLER S., AGARWAL S., AHMAD L., AKKAYA I., ALEMAN F. L., ALMEIDA D., ALTENSCHMIDT J., ALTMAN S., ANADKAT S. *et al.* (2023). Gpt-4 technical report. *arXiv preprint arXiv :2303.08774*.

ALBERTS I. L., MERCOLLI L., PYKA T., PRENOSIL G., SHI K., ROMINGER A. & AFSHAR-OROMIEH A. (2023). Large language models (llm) and chatgpt : what will the impact on nuclear medicine be ? *European journal of nuclear medicine and molecular imaging*, **50**(6), 1549–1552.

ANONYMIZATION C. (2024). L'anonymisation de données personnelles. https://www.cnil.fr/fr/technologies/lanonymisation-de-donnees-personnelles.

AP-HP E. (2024). L'Entrepôt de Données de Santé de l'AP-HP. https://www.aphp.fr/connaitre-lap-hp/recherche-innovation/lentrepot-de-donnees-de-sante-de-lap-hp.

AZARIA A. & MITCHELL T. (2023). The internal state of an llm knows when it's lying. In H. BOUAMOR, J. PINO & K. BALI, Éds., *Findings of the Association for Computational Linguistics : EMNLP 2023*.

BENAMARA F., HATOUT N., MULLER P. & OZDOWSKA S., Éds. (2007). *Actes de TALN 2007 (Traitement automatique des langues naturelles)*, Toulouse. ATALA, IRIT.

BOJANOWSKI P., GRAVE E., JOULIN A. & MIKOLOV T. (2017). Enriching word vectors with subword information. *Transactions of the association for computational linguistics*, **5**, 135–146.

DIAS G., Éd. (2015). *Actes de TALN 2015 (Traitement automatique des langues naturelles)*, Caen. ATALA, HULTECH.

DUA D., WANG Y., DASIGI P., STANOVSKY G., SINGH S. & GARDNER M. (2019). Drop : A reading comprehension benchmark requiring discrete reasoning over paragraphs. *arXiv preprint arXiv :1903.00161*.

FABBRI A. R., KRYŚCIŃSKI W., MCCANN B., XIONG C., SOCHER R. & RADEV D. (2021). Summeval : Re-evaluating summarization evaluation. *Transactions of the Association for Computational Linguistics*, **9**, 391–409.

FRIEL R. & SANYAL A. (2023). Chainpoll : A high efficacy method for llm hallucination detection. *arXiv preprint arXiv :2310.18344*.

FU J., NG S.-K., JIANG Z. & LIU P. (2024). GPTScore : Evaluate as You Desire. In K. DUH, H. GOMEZ & S. BETHARD, Éds., *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies (Volume 1 : Long Papers)*, p. 6556–6576, Mexico City, Mexico : Association for Computational Linguistics. DOI : 10.18653/v1/2024.naacl-long.365.

GAO Y., ZHAO W. & EGER S. (2020). Supert : Towards new frontiers in unsupervised evaluation metrics for multi-document summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, p. 1347–1354.

LAIGNELET M. & RIOULT F. (2009). Repérer automatiquement les segments obsolescents à l'aide d'indices sémantiques et discursifs. In A. NAZARENKO & T. POIBEAU, Éds., *Actes de TALN 2009 (Traitement automatique des langues naturelles)*, Senlis : ATALA LIPN.

LANGLAIS P. & PATRY A. (2007). Enrichissement d'un lexique bilingue par analogie. In (Benamara *et al.*, 2007), p. 101–110.

LI H., WANG H., SUN X., HE H. & FENG J. (2024). Prompt-guided generation of structured chest x-ray report using a pre-trained llm. *arXiv e-prints*, p. arXiv–2404.

LIU Y., ITER D., XU Y., WANG S., XU R. & ZHU C. (2023). G-eval : Nlg evaluation using gpt-4 with better human alignment. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, p. 2511–2522.

LUO Z., XIE Q. & ANANIADOU S. (2024). Factual consistency evaluation of summarisation in the era of large language models. *Expert Systems with Applications*, p. 124456.

MANAKUL P., LIUSIE A. & GALES M. (2023). Selfcheckgpt : Zero-resource black-box hallucination detection for generative large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, p. 9004–9017.

MARTIN L., MULLER B., SUAREZ P. O., DUPONT Y., ROMARY L., DE LA CLERGERIE É. V., SEDDAH D. & SAGOT B. (2020). Camembert : a tasty french language model. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, p. 7203–7219.

MIKOLOV T., SUTSKEVER I., CHEN K., CORRADO G. S. & DEAN J. (2013). Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems*, **26**.

NABLA (2024). All you need to know about Nabla's privacy and security features. https://www.nabla.com/blog/privacy-security/.

OUKELMOUN A., SEMMAR N., DE CHALENDAR G., HABRAN E., VIBERT E., GOBLET E., OUKELMOUN M. & ALLARD M.-A. (2023). A study on the relevance of generic word embeddings for sentence classification in hepatic surgery. In *2023 20th ACS/IEEE International Conference on Computer Systems and Applications (AICCSA)*, p. 1–8 : IEEE.

PARZEN E. (1962). On estimation of a probability density function and mode. *The annals of mathematical statistics*, **33**(3), 1065–1076.

REINDITIFICATION C. (2024). L'anonymisation de données personnelles. https://www.cnil.fr/fr/technologies/lanonymisation-de-donnees-personnelles.

SERETAN V. & WEHRLI E. (2007). Collocation translation based on sentence alignment and parsing. In (Benamara *et al.*, 2007), p. 401–410.

VAN SCHAIK T. A. & PUGH B. (2024). A field guide to automatic evaluation of llm-generated summaries. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, p. 2832–2836.

VASILYEV O., DHARNIDHARKA V. & BOHANNON J. (2020). Fill in the blanc : Human-free quality estimation of document summaries. In *Proceedings of the First Workshop on Evaluation and Comparison of NLP Systems*, p. 11–20.

WANG A., CHO K. & LEWIS M. (2020). Asking and answering questions to evaluate the factual consistency of summaries. *arXiv preprint arXiv :2004.04228*.

WĘGLARCZYK S. (2018). Kernel density estimation and its application. In *ITM web of conferences*, volume 23, p. 00037 : EDP Sciences.

WEI J., WANG X., SCHUURMANS D., BOSMA M., ICHTER B., XIA F., CHI E. H., LE Q. V. & ZHOU D. (2022). Chain-of-thought prompting elicits reasoning in large language models. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, p. 24824–24837, Red Hook, NY, USA : Curran Associates Inc.

A G-Eval prompt - prompt en français

Vous recevrez un **compte rendu médical** et un **résumé** de ce compte rendu. Votre tâche est d'évaluer le résumé sur la base de sa **complétude** et de sa capacité à inclure toutes les informations critiques issues du compte rendu.

Veuillez suivre attentivement les instructions ci-dessous et vous y référer tout au long de l'évaluation.

Critères d'évaluation

Complétude et pertinence (1-3)

- **Score 1 (Insuffisant)** :
  - Le résumé manque plusieurs informations critiques essentielles à la compréhension du cas.
  - Les omissions pourraient avoir un impact significatif sur la prise de décision médicale ou les soins du patient.
- **Score 2 (Moyen)** :
  - Le résumé inclut certaines informations clés, mais omet une ou deux informations importantes.
  - Bien que les omissions soient notables, elles ne compromettent pas totalement la compréhension du cas.
- **Score 3 (Excellent)** :
  - Le résumé est complet et inclut toutes les informations critiques issues du compte rendu médical.
  - Aucune omission significative n'est présente, et le résumé permet une compréhension totale du cas.

Étapes d'évaluation

1. **Étape 1 :** Lisez attentivement le compte rendu médical et identifiez les détails principaux (diagnostics, traitements, résultats de tests, antécédents, etc.).
2. **Étape 2 :** Comparez le résumé au compte rendu médical. Identifiez les informations manquantes ou incorrectes.
3. **Étape 3 (raisonnement en chaîne) :**
   - Analysez étape par étape si le résumé correspond au compte rendu.
   - Soulignez les divergences ou omissions et évaluez leur importance.
   - Expliquez clairement votre raisonnement pour le score attribué.
4. **Étape 4 :** Attribuez un score de complétude entre 1 et 3, en suivant les critères ci-dessus.

Compte rendu médical : [INSÉREZ ICI LE COMPTE RENDU MÉDICAL]

Résumé fourni : [INSÉREZ ICI LE RÉSUMÉ À ANALYSER]

Formulaire d'évaluation (scores UNIQUEMENT) : - Complétude :

B ChainPoll prompt - prompt en français

Vous êtes un assistant médical spécialisé dans l'analyse de comptes rendus médicaux. Vous recevrez : 1. Un **compte rendu médical** détaillé. 2. Un **résumé** de ce compte rendu.

Votre tâche : 1. Identifier et expliquer si le résumé omet des informations médicales importantes qui figurent dans le compte rendu. 2. Indiquer s'il existe des omissions importantes avec une valeur binaire : - **0** : Pas d'omissions importantes. - **1** : Des omissions importantes sont présentes.

IMPORTANT : À la fin de votre analyse, incluez une ligne au format clair : `OMISSION_RESULT = [0 ou 1]`

Étapes

1. **Étape 1 :** Analysez le compte rendu médical en identifiant les informations essentielles (diagnostics, traitements, antécédents, résultats de tests, etc.).
2. **Étape 2 :** Comparez le résumé fourni au compte rendu original. Relevez les informations importantes manquantes, le cas échéant.
3. **Étape 3 :** Justifiez votre décision en listant les éléments omis ou confirmant qu'aucune omission significative n'est présente.
4. **Étape 4 :** Fournissez le résultat binaire au format clair.

Compte rendu médical : [INSÉREZ ICI LE COMPTE RENDU MÉDICAL]

Résumé fourni : [INSÉREZ ICI LE RÉSUMÉ À ANALYSER]

C SelfCheckGPT System Prompt - prompt en français

Le prompt système utilisé dans le modèle est le suivant :

> *Vous êtes un assistant. Je vais vous fournir un compte-rendu médical synthétique et vous allez devoir me fournir un résumé complet.*