Actes de CORIA-TALN-RJCRI-RECITAL 2025
Marseille, France, 30 juin au 4 juillet 2025

Extraction de mots-clés à partir d'articles scientifiques: comparaison entre modèles traditionnels et modèles de langue

Motasem Alrahabi Nacef Ben Mansour Hamed Rahimi
Sorbonne Université, 75005, Paris, France
prenom-nom@sorbonne-universite.fr

RÉSUMÉ

L'extraction automatique des mots-clés est cruciale pour résumer le contenu des documents et affiner la recherche d'informations. Dans cette étude, nous comparons les performances de plusieurs modèles d'extraction et de génération de mots-clés appliqués aux résumés d'articles issus des archives HAL : des approches basées sur des statistiques et des modèles vectoriels, ainsi que des approches génératives modernes utilisant les LLMs. Les résultats montrent que les LLMs surpassent largement les méthodes traditionnelles en termes de précision et de pertinence, même en configuration zero-shot, et que l'inclusion des titres d'articles améliore significativement les scores F1. Nous introduisons également une nouvelle métrique pour évaluer les performances des LLMs en tenant compte des coûts de traitement, offrant ainsi une perspective équilibrée entre efficacité et coût.

ABSTRACT

Keyword extraction from scientific articles : Comparison between traditional models and language models.

Automatic keyword extraction is crucial for summarizing document content and refining information retrieval. In this study, we compare the performance of several keyword extraction and generation models applied to abstracts from the HAL archives : statistical and vector-based approaches, as well as generative approaches using LLMs. The results show that LLMs significantly outperform traditional methods in terms of precision and relevance, even in a zero-shot setting, and that including article titles significantly improves F1 scores. We also introduce a new metric to evaluate LLM performance while considering processing costs, providing a balanced perspective between efficiency and cost.

MOTS-CLÉS : Extraction de mots-clés, évaluation, modèles de langage, corpus HAL.

KEYWORDS: Keyword Extraction, Evaluation, Large Language Models, HAL Corpus.

ARTICLE : **Accepté à** AI & Scientific Discovery Workshop, NAACL 2025, May 3, 2025, lien.

1 Introduction

L'extraction de mots-clés est une étape cruciale pour identifier les termes les plus pertinents et représentatifs d'un document ou d'un corpus, facilitant la compréhension et l'exploitation de l'information contenue [1]. Cette technique possède un large éventail d'applications, allant de la condensation de

1. Dans cet article, nous employons le terme mot-clé "keyword" de manière générique pour englober aussi bien les mots-clés que les expressions-clés (keyphrase).

Actes des 32ème Conférence sur le Traitement Automatique des Langues Naturelles (TALN), volume 2 : traductions d'articles publiés, pages 15–27.

contenu et de l'optimisation des moteurs de recherche à l'extraction d'informations, au résumé automatique et à la détection des tendances émergentes. Les méthodes traditionnelles d'extraction, basées sur des règles ou des calculs quantitatifs, montrent cependant des limites dans la capture des subtilités sémantiques et contextuelles des mots-clés. L'avènement des modèles d'apprentissage profond a permis le développement de nouvelles approches qui allient la rigueur des méthodes statistiques à la capacité des LLMs à interpréter le contexte et la sémantique des mots (Song *et al.*, 2023). Ce travail propose, après un état des lieux des recherches récentes, de comparer les performances des approches traditionnelles et des techniques modernes dans le domaine de l'extraction de mots-clés.

2 Approches pour l'identification de mots-clés

L'évolution des approches pour l'extraction de mots-clés et de phrases-clés peut être divisée en plusieurs étapes, reflétant les progrès technologiques et méthodologiques dans le traitement du langage naturel :

- Règles linguistiques : Ces méthodes se basent sur l'analyse syntaxique, comme l'extraction de syntagmes nominaux ou l'analyse des n-grammes. Des algorithmes comme YAKE ! (Campos *et al.*, 2020) utilisent des caractéristiques linguistiques comme la place d'un mot dans la phrase pour extraire les mots-clés.
- Approches statistiques : Certaines méthodes comme TF-IDF attribuent un poids à chaque mot en fonction de sa fréquence d'apparition dans le texte et de sa distribution dans un corpus plus large (Salton & Buckley, 1988). L'algorithme RAKE, par exemple, identifie les mots-clés en fonction de leur co-occurrence dans les phrases et leur importance relative dans le texte. Les n-grammes permettent également d'identifier des séquences de mots fréquentes (Justeson et Katz, 1995). D'autres approches exploitent la position des mots dans les phrases pour identifier les termes clés (Turney, 2000) ou se basent sur des distributions statistiques plus complexes (Church & Gale, 1995).
- Apprentissage supervisé : Ces méthodes supervisées, comme les algorithmes de classification, exploitent des données annotées pour entraîner des modèles à extraire des mots-clés. KP-Miner (El-Beltagy & Rafea, 2009) et les approches supervisées de (Papagiannopoulou & Tsoumakas, 2020) en sont des exemples.
- Apprentissage non supervisé : Certaines méthodes non supervisées s'appuient sur l'utilisation de graphes. Des méthodes comme TextRank (Mihalcea & Tarau, 2004), SingleRank (Wan & Xiao, 2008), et MultipartiteRank (Boudin, 2018) utilisent des graphes de co-occurrence de mots pour extraire les mots-clés sans avoir besoin de données annotées. TopicRank (Bougouin *et al.*, 2013) et PositionRank (Florescu & Caragea, 2017) exploitent également des graphes pour évaluer l'importance des mots. Ces méthodes, bien que robustes et largement utilisées, peuvent manquer de contextualisation et de nuance dans l'extraction des mots-clés.
- Vectorisation : Certaines méthodes ont exploité des représentations vectorielles plus riches des mots ou des phrases pour l'extraction de mots-clés. Par exemple, EmbedRank utilise deux variantes : l'une basée sur Word2Vec (Mikolov, 2013) pour créer des embeddings des phrases candidates et du document, et l'autre s'appuyant sur Sent2Vec (Pagliardini *et al.*, 2017) pour générer des embeddings plus riches des phrases. Les phrases candidates sont ensuite extraites à l'aide de motifs syntaxiques, tels que les Part of Speech (PoS), et classées en fonction de leur similarité cosinus avec le document (Bennani-Smires *et al.*, 2018). Dans la continuité de ces travaux, des approches plus récentes, telles que PatternRank et KeyBERT,

ont intégré des embeddings contextuels et des modèles de langage avancés, tels que SBERT ou BERT, pour améliorer l'évaluation des mots-clés. Ces méthodes combinent également des motifs syntaxiques, comme les PoS, afin d'identifier les termes clés et évaluent leur proximité contextuelle avec le texte source (Schopf *et al.*, 2022). Ces méthodes ont marqué une transition importante vers des approches plus contextuelles, tout en restant en deçà des capacités des LLMs.

— Les approches génératives reposent sur les réseaux neuronaux profonds et les LLMs tels que les familles BERT, GPT et LLaMA. Ces modèles, grâce à leur capacité à saisir des relations contextuelles complexes entre les termes, permettent une extraction précise et pertinente des mots-clés via des approches zero-shot, few-shot ou par fine-tuning. Ils utilisent des réseaux de neurones basés sur une architectures novatrice, les Transformers (Vaswani, 2017), pour comprendre le contexte et générer des mots-clés pertinents basés sur une compréhension plus profonde du texte. Ces modèles peuvent être utilisés directement pour extraire des mots-clés, ou bien être fine-tunés sur des tâches spécifiques d'extraction de mots-clés avec des données annotées dans un domaine particulier. Contrairement aux méthodes extractives, qui se limitent à sélectionner des mots-clés directement présents dans le texte, les approches génératives modernes peuvent créer ou reformuler des mots-clés, même s'ils n'apparaissent pas explicitement dans le texte d'origine (Song *et al.*, 2023).

D'autres tendances émergentes incluent l'intégration d'approches hybrides, l'évolution de l'apprentissage automatique non supervisé, l'adaptation de modèles pré-entraînés pour des domaines spécifiques ou des langues moins courantes (Song *et al.*, 2023), ainsi que l'extraction thématique de mots-clés à l'aide de techniques comme BERTopic pour regrouper les documents selon leurs thèmes centraux (Grootendorst, 2020).

Soulignons enfin que des bibliothèques Python comme NLTK (`https://www.nltk.org/`), Scikit-learn (`https://scikit-learn.org/`), et Spacy (`https://spacy.io/`) fournissent des modules pour l'extraction de mots-clés, avec des outils tels que TF-IDF et des méthodes basées sur l'analyse linguistique. Gensim (`https://github.com/piskvorky/gensim`) est également couramment utilisée pour la modélisation de sujets et l'extraction de termes clés. Par ailleurs, la bibliothèque PKE (`https://github.com/boudinfl/pke`) implémente plusieurs algorithmes avancés, offrant une large couverture de techniques basées sur des graphes et des statistiques.

3 Méthodologie

La méthodologie que nous avons employée pour comparer différentes méthodes d'extraction des mots-clés comprend plusieurs étapes :

— Préparation des données : Extraction des résumés et des mots-clés d'auteurs à partir d'articles scientifiques de la base de données HAL (`https://hal.science/`). Les expériences ont été menées en deux configurations : une avec les résumés et les titres combinés, et une autre avec uniquement les résumés, sans les titres.

— Mise en œuvre des approches : Nous avons évalué différentes méthodes d'extraction de mots-clés, en les répartissant en deux catégories : les méthodes génératives modernes et les autres approches, qualifiées de traditionnelles. Pour l'utilisation des LLMs, cette étude a adopté une approche zero-shot, choisie pour sa simplicité d'implémentation et son efficacité, tout en reflétant les performances de base représentatives de l'état de l'art pour ces modèles.

— Évaluation : Les mots-clés extraits sont évalués en fonction de leur pertinence et de leur précision par rapport aux mots-clés fournis par les auteurs dans leurs articles. Les critères d'évaluation incluent la précision, le rappel et la mesure F1.
— Analyse des résultats : Comparaison des performances des méthodes traditionnelles et modernes à l'aide de statistiques descriptives et d'analyses qualitatives.

Ces étapes permettent d'assurer une comparaison robuste pour observer les divergences de performance entre approches traditionnelles et modernes.

4 Les données

Les données utilisées pour cette étude proviennent de la plateforme HAL, une archive ouverte dédiée à la diffusion des travaux de recherche scientifique. Des travaux récents, tels que HALvest (Kulumba *et al.*, 2024), démontrent le potentiel sous-exploité de la base HAL pour l'exploration et l'analyse des publications scientifiques.

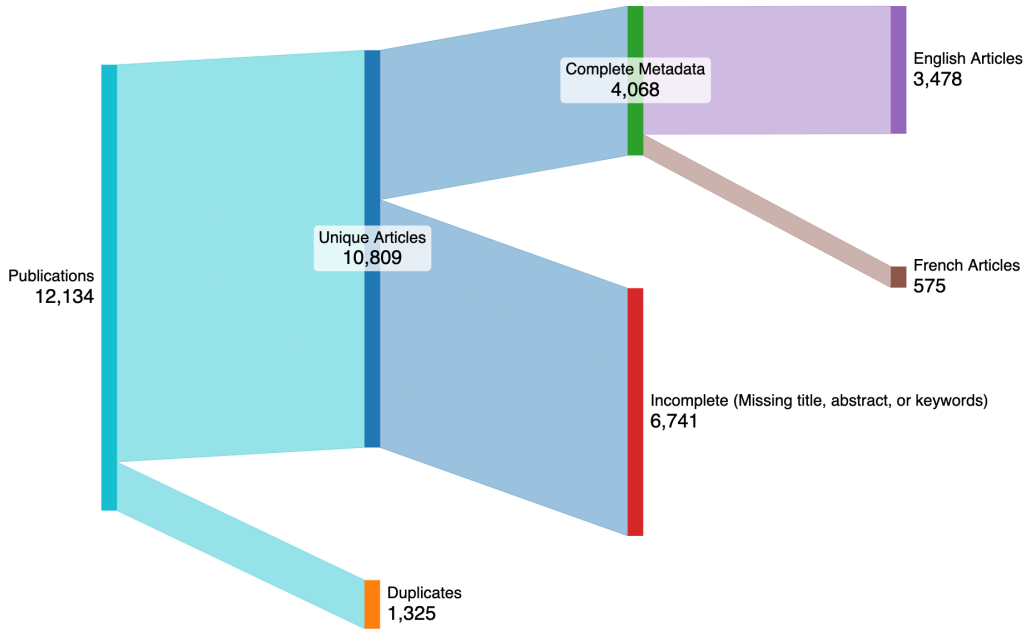

FIGURE 1 – **Répartition des articles selon la langue**

Les articles sélectionnés pour les chercheurs de SU couvrent divers domaines scientifiques. Chaque article est accompagné de résumés et de mots-clés fournis par les auteurs, qui serviront de référence pour évaluer la qualité des méthodes d'extraction. Pour une première exploration, notre jeu de données contient les articles d'une centaine de chercheurs affiliés au Sorbonne Center for Artificial Intelligence (SCAI). Ce jeu de données a été constitué à l'aide d'un script exploitant l'API de HAL. Les données collectées comptaient environ 12 000 articles. Un premier tri a permis d'éliminer 1 300 doublons, tandis qu'environ 6 000 autres articles ont été exclus en raison de l'absence de mots-clés ou de résumés. Après ce filtrage, le corpus final se compose de 4 700 articles exploitables pour notre analyse, représentant environ 30 % des données initiales.

Une première observation révèle une répartition linguistique marquée avec 85 % des articles en

anglais, contre 15 % en français. Concernant les articles en anglais, la moyenne du nombre de mots-clés par article est de 5,35 avec une longueur moyenne des mots-clés de 2,14 mots. En comparaison, pour les articles en français, le nombre moyen de mots-clés est légèrement supérieur à 6,32, avec une longueur moyenne de 2,23 mots.

La distribution des domaines scientifiques varie également selon la langue, comme illustré dans la figure 2. Sans surprise, le domaine de l'informatique reste majoritaire pour les deux langues. Les sciences humaines arrivent en deuxième position en français, tandis que les sciences de la vie prennent cette place en anglais. Les sciences humaines, bien représentées en français, sont moins présentes en anglais.

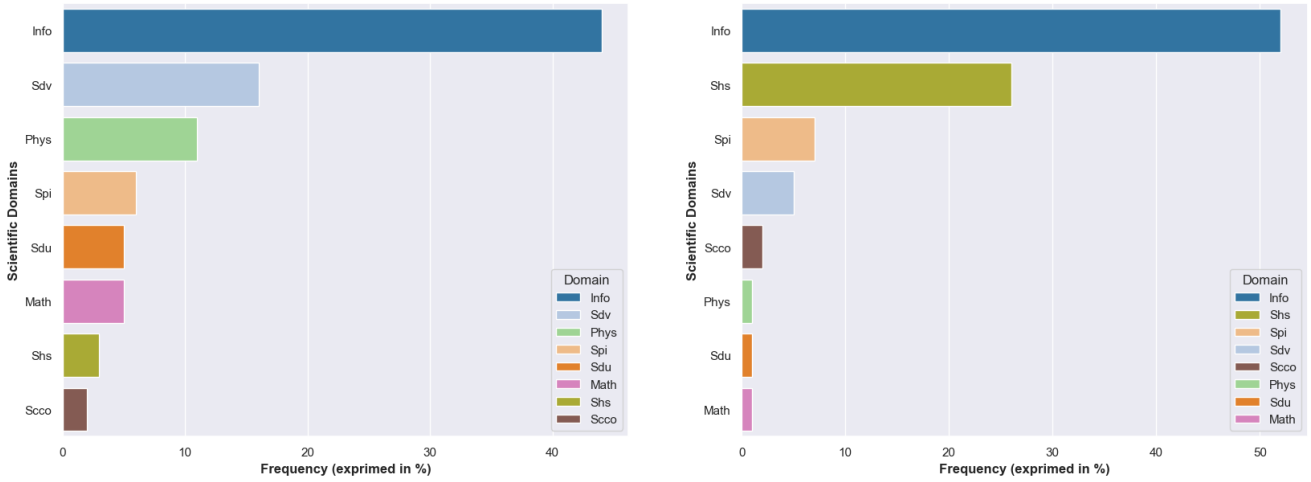

FIGURE 2 – **Distribution des domaines selon la langue**

Pour la suite de l'analyse, il est important de préciser que tous les titres, mots-clés et résumés ont été convertis en minuscules afin d'assurer des résultats homogènes et fiables.

5 Evaluation

Une évaluation manuelle consisterait à comparer les mots-clés générés à ceux d'une référence (*Gold Standard*). Bien que souvent efficace, cette méthode présente des inconvénients : elle est chronophage, énergivore et limite le traitement de gros volumes de données. En outre, elle peut introduire des biais en raison des perspectives et interprétations subjectives des évaluateurs humains.

Dans notre étude, nous avons opté pour une évaluation automatique, en comparant les mots-clés extraits automatiquement aux mots-clés de référence fournis par les auteurs sur HAL. Cette comparaison s'est effectuée en mode exact matching, où seuls les termes identiques étaient considérés comme des correspondances, afin d'assurer une évaluation précise et cohérente des résultats. Pour chaque article, les mots-clés les plus pertinents ont été extraits des résumés à l'aide de toutes les méthodes évaluées. Nous avons utilisé le F1-Score, une métrique couramment employée pour évaluer la performance des modèles d'extraction de mots-clés. Le F1-Score est la moyenne harmonique entre la précision, qui est le rapport entre le nombre de mots-clés correctement extraits et le nombre total de mots-clés extraits, et le rappel, qui mesure la proportion de mots-clés pertinents extraits par rapport au total des mots-clés pertinents dans le texte. Dans le contexte de l'extraction de mots-clés, un F1-Score

élevé indique que le modèle parvient à extraire une proportion importante de mots-clés pertinents (haut rappel) tout en limitant l'extraction de mots-clés non pertinents (haute précision).

Pour chaque article et méthode d'extraction dans cette étude, un F1-Score est calculé. En moyennant ces scores sur l'ensemble du corpus, nous obtenons une évaluation comparative des performances des méthodes classiques et modernes [2].

6 Résultats

Les résultats de l'évaluation sont présentés dans un tableau avec la précision, le rappel et le F1 score, classés par performance décroissante.

6.1 Les approches traditionnelles

Cette section présente les performances des modèles fondés sur des approches graphiques et statistiques, évalués selon les critères de précision, rappel et F1-Score. Les résultats sont également segmentés en fonction de l'inclusion ou non du titre dans l'évaluation.

Les résultats montrent une performance disparate (voir Table 1), avec des écarts allant jusqu'à 60% entre les modèles les plus faibles (TextRank) et les plus performants (PositionRank, suivi de MultipartiteRank).

En ce qui concerne les modèles basés sur les embeddings, nous avons évalué le système KeyBERT. Dans ce système, il existe deux paramètres qui régissent la sélection des mots-clés : MMR (Maximal Marginal Relevance) et MSum (Maximum Similarity to Unselected). MMR équilibre la pertinence des mots-clés avec le texte source et leur diversité, en réduisant la redondance entre les termes sélectionnés. En revanche, MSum se concentre uniquement sur la maximisation de la similarité entre les mots-clés non encore sélectionnés et le texte source, favorisant une extraction plus pertinente mais potentiellement moins variée.

6.2 Les approches génératives

L'évaluation des approches basées sur les LLMs a été réalisée en testant des modèles génératifs open-source et des modèles propriétaires :

- Les modèles open-source sont : LLaMA-3 70B et LLaMA-3 8B (Meta) ; Mixtral 8*7B et Mistral 7B (Mistral) ; Gemma 7B (Google).
- Les modèles propriétaires sont : GPT-3.5 Turbo et GPT-4o (OpenAI) ; Claude 3 Haiku et Claude 3 Instant (Anthropic).

Les résultats pour les LLMs (Table 1) montrent une variabilité importante, avec des performances allant du simple au triple. L'inclusion des titres améliore les performances d'environ 10 %.

On observe que les trois meilleurs modèles sont LLaMA3 70B, Claude 3 Haiku, et LLaMA3 8B. En revanche, Gemma 7B affiche de faibles performances, principalement en raison du non-respect des

2. Le code complet des évaluations est disponible ici : https://github.com/obtic-sorbonne/keywords/tree/main

prompts dans le format des résultats, ce qui pénalise fortement l'évaluation en exact matching.

Modèle	Abstract + Titre			Abstract		
	Precision	**Recall**	**F1**	**Precision**	**Recall**	**F1**
Approche basée sur LLM						
LLaMA 3.1 70b	0.132	0.245	**0.163**	0.120	0.224	**0.148**
Claude 3 Haiku	0.130	0.218	0.154	0.120	0.204	0.143
LLaMA 3.1 8b	0.147	0.181	0.151	0.136	0.172	0.142
GPT 4o	0.075	0.222	0.108	0.071	0.206	0.101
Claude Instant 1.2	0.073	0.183	0.097	0.066	0.171	0.088
GPT 3.5 Turbo	0.089	0.094	0.087	0.086	0.089	0.083
Mixtral 8x7b	0.057	0.188	0.083	0.047	0.176	0.070
Mistral 7b	0.050	0.199	0.077	0.048	0.156	0.069
Gemma 7b	0.051	0.079	0.059	0.052	0.081	0.060
Approche basée sur Embedding						
KeyBERT Default	0.058	0.081	0.067	0.056	0.078	0.065
KeyBERT with MMR and MSum	0.052	0.073	0.061	0.050	0.070	0.058
Approches traditionnelles						
PositionRank	0.062	0.115	0.080	0.056	0.103	0.072
MultipartiteRank	0.062	0.113	0.079	0.056	0.103	0.072
TopicRank	0.059	0.108	0.076	0.053	0.096	0.068
SingleRank	0.053	0.098	0.068	0.052	0.096	0.067
YAKE	0.053	0.098	0.068	0.045	0.083	0.058
TextRank	0.039	0.072	0.050	0.036	0.066	0.046

TABLE 1 – **Résultat de l'évaluation en Exact Matching**

7 Correspondance flexible : distance de Levenshtein

Jusqu'à présent, nous avons utilisé l'approche Exact Matching, une méthode de comparaison stricte des termes, sans tolérance pour les variations telles que les formes plurielles, l'usage des tirets ou éventuellement les erreurs typographiques. Cette approche, bien que précise, est rigide et réduit la capacité à capturer toutes les correspondances pertinentes.

Nous avons donc expérimenté le Fuzzy Matching, qui permet de comparer les mots-clés générés avec ceux de référence en prenant en compte les variations formelles. Plusieurs métriques peuvent attribuer un "score de proximité" entre deux chaînes de caractères, telles que Levenshtein, Jaro-Winkler, ainsi que différents modèles d'embeddings (Alqahtani *et al.*, 2021). Dans cette étude, nous avons adopté la distance de Levenshtein, également appelée distance d'édition. Elle quantifie le nombre minimal d'opérations nécessaires pour transformer une chaîne de caractères en une autre, les opérations possibles étant l'insertion, la suppression ou la substitution de caractères. Les résultats sont présentés sous forme de graphiques pour illustrer l'évolution du F1-Score à mesure que l'on augmente la flexibilité de la distance de Levenshtein (de 0 à 4).

7.1 Les approches traditionnelles

Ces modèles, bien qu'anciens, offrent une bonne robustesse dans des contextes où la complexité des termes est modérée et où la structure du texte joue un rôle important dans l'extraction des mots-clés.

Les résultats pour les approches traditionnelles confirment l'importance de la flexibilité offerte par la distance de Levenshtein pour mieux capturer les variations linguistiques (Table 2). Il est à noter que l'ajout des titres améliore la performance des modèles traditionnelles, même si l'impact reste modéré.

L'outil KeyBERT tire parti des représentations contextuelles fournies par les modèles d'embeddings, offrant ainsi une approche plus nuancée pour l'extraction de mots-clés, notamment lorsque les termes sont répartis de manière moins homogène dans le texte. L'utilisation de la distance de Levenshtein avec KeyBERT montre des gains intéressants en termes de précision, mais au détriment de la rapidité d'exécution. Avec les titres, on observe également une légère augmentation de la performance.

7.2 Les approches génératives

Les LLMs, grâce à leur compréhension approfondie du contexte et à leur capacité à traiter de grandes quantités de données, surpassent souvent les autres approches, notamment dans des tâches complexes ou pour des textes aux structures variées. Ils semblent bénéficier grandement de la souplesse offerte par la distance de Levenshtein. L'inclusion des titres dans les modèles LLMs continue d'améliorer les scores, mais avec un retour décroissant au-delà d'un certain seuil.

Model	Abstract + Titre				Abstract			
	$d \leq 1$	$d \leq 2$	$d \leq 3$	$d \leq 4$	$d \leq 1$	$d \leq 2$	$d \leq 3$	$d \leq 4$
Approches basée sur LLM								
LLaMA 3.1 70b	**0.19**	**0.197**	**0.21**	**0.228**	**0.174**	**0.18**	**0.193**	**0.212**
Claude 3 Haiku	0.179	0.185	0.195	0.21	0.168	0.173	0.183	0.198
LLaMA 3.1 8b	0.175	0.183	0.198	0.223	0.165	0.172	0.187	0.21
GPT 4o	0.127	0.132	0.141	0.155	0.12	0.124	0.134	0.148
Claude Instant 1.2	0.116	0.13	0.147	0.176	0.105	0.118	0.135	0.163
GPT 3.5 Turbo	0.101	0.106	0.118	0.137	0.096	0.102	0.114	0.13
Mixtral 8x7b	0.1	0.107	0.118	0.133	0.084	0.095	0.107	0.123
Mistral 7b	0.092	0.097	0.107	0.123	0.085	0.09	0.099	0.116
Gemma 7b	0.069	0.072	0.076	0.084	0.071	0.073	0.078	0.086
Approches basée sur Embedding								
KeyBERT Default	0.084	0.095	0.12	0.158	0.081	0.092	0.116	0.154
KeyBERT with MMR and MSum	0.072	0.08	0.101	0.137	0.07	0.078	0.098	0.135
Approches Traditionnelles								
PositionRank	0.097	0.101	0.108	0.123	0.087	0.091	0.099	0.114
MultipartiteRank	0.095	0.099	0.113	0.139	0.087	0.091	0.105	0.13
TopicRank	0.089	0.094	0.108	0.135	0.08	0.084	0.099	0.125
SingleRank	0.083	0.087	0.092	0.102	0.072	0.075	0.081	0.091
YAKE	0.081	0.085	0.094	0.113	0.079	0.082	0.091	0.11
TextRank	0.062	0.065	0.068	0.075	0.058	0.06	0.064	0.071

TABLE 2 – **Résultat de l'évaluation en Fuzzy Matching (F1 Scores)**

8 LLMs et coût par token

L'utilisation de LLMs requiert des ressources matérielles importantes, ce qui peut limiter leur applicabilité pour des institutions avec des moyens techniques restreints. Cela pose également des enjeux environnementaux en raison de la consommation énergétique élevée. Le coût de traitement par token constitue donc un facteur déterminant. À F1 Score similaire, certains modèles peuvent être de 10 à 100 fois plus coûteux à exécuter. Dans le cadre de traitements de données massives, il devient essentiel d'intégrer cette dimension dans notre évaluation. Pour synthétiser cette approche, nous introduisons une nouvelle métrique dédiée aux LLMs qui combine la performance (F1 Score) et le coût ($/million de tokens) : le Token Efficiency Score (TES). Cette métrique intègre les performances et les coûts, en appliquant une pénalisation plus forte sur les coûts, tout en privilégiant la performance. Nous utilisons une moyenne harmonique pondérée, définie par la formule suivante :

$$\text{TES} = \frac{(1+\alpha) \times F_1 \times \text{Cost}}{\alpha \times \text{Cost} + F_1} \quad (\alpha = 10), \tag{1}$$

Le calcul montre que les modèles les plus performants sont également parmi les moins coûteux, notamment Llama-3 70B, Llama-3 8B, et Claude 3 Haiku.

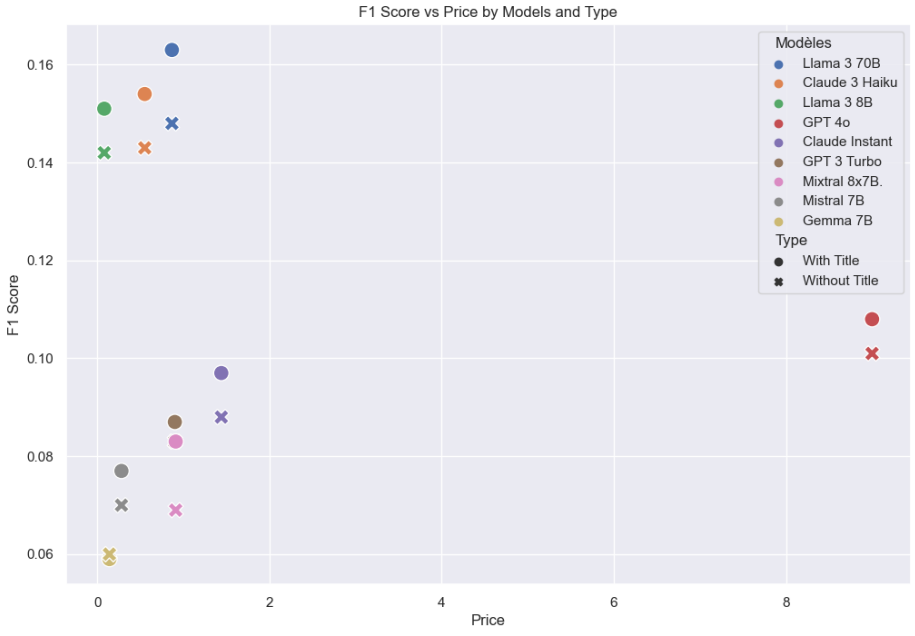

FIGURE 3 – Performance du F1 Score en fonction du prix.

Dans la figure 4 qui suit, on ordonne les modèles génératifs selon leur score TES du plus efficient au moins efficient. On retrouve comme prévu les trois modèles en tête : Llama 3 70B - Claude 3 Haiku - Llama 3 8B en tête et Gemma 7B de Google en dernière position.

Le TES permet d'identifier clairement les modèles les plus performants tout en prenant en compte le facteur coût, crucial dans des scénarios à grande échelle.

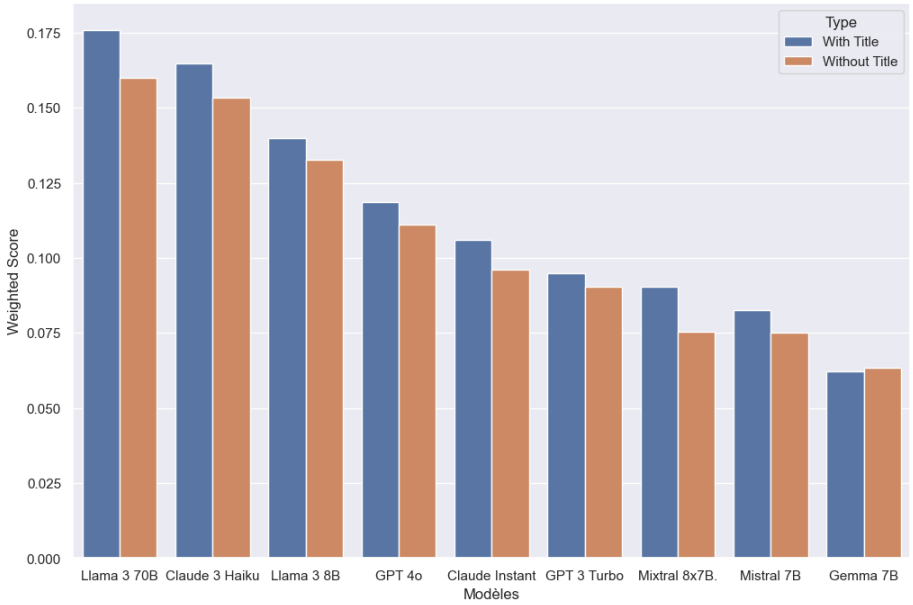

FIGURE 4 – Résultats TES pour les modèles basés sur les LLMs

9 Limites

Les LLMs apportent des avancées significatives dans l'extraction de mots-clés, mais leur utilisation présente certaines limites. Tout d'abord, leur formation sur des corpus génériques limite leur précision dans des domaines scientifiques spécifiques, ce qui nécessite souvent un fine-tuning sur des données annotées pour capturer les termes techniques propres à chaque discipline. De plus, les LLMs fonctionnent comme des "boîtes noires" difficiles à interpréter, rendant complexe l'explication des choix de mots-clés générés et limitant ainsi leur adoption dans des contextes exigeant une traçabilité des méthodes.

La sensibilité des LLMs aux prompts et leur tendance à produire des réponses variables introduisent également une certaine instabilité dans les résultats (stochasticity). Cela impose des ajustements fréquents des prompts pour obtenir des mots-clés cohérents, et oblige à répéter les évaluations pour lisser les fluctuations des F1-Scores. Par ailleurs, l'utilisation de prompts détaillés et conversationnels peut alourdir les coûts de traitement sans garantir une amélioration de la pertinence des mots-clés extraits, bien que cela permette d'affiner la structure des mots-clés générés.

De plus, une extraction fondée sur une correspondance exacte des mots-clés manque de souplesse. Cette approche ne prend pas en compte les synonymes ni les variations grammaticales, ce qui peut conduire à l'exclusion de termes équivalents sur le plan sémantique.

Enfin, le corpus HAL lui-même comporte des limites pour l'extraction de mots-clés. De nombreux mots-clés fournis par les auteurs ne figurent ni dans les résumés ni dans les titres des articles, ce qui pose un défi pour les modèles extractifs et introduit un biais lors de l'évaluation de leur performance.

10 Conclusion et perspectives

Les résultats de la comparaison montrent que les approches génératives basées sur les LLMs surpassent les méthodes traditionnelles en termes de précision et de pertinence des mots-clés extraits. Ces modèles modernes, même en configuration zero-shot learning, parviennent à capturer des nuances contextuelles et des relations sémantiques que les méthodes statistiques traditionnelles ne peuvent pas toujours saisir, rendant l'extraction de mots-clés plus riche et plus représentative des contenus scientifiques. L'introduction de la nouvelle métrique TES, que nous avons conçue, a révélé que les modèles offrant le meilleur compromis entre qualité et coût sont également les moins onéreux, confirmant leur efficacité économique. De plus, l'intégration des titres, riches en information, améliore significativement les scores F1 sans augmenter notablement la charge en tokens.

Plusieurs pistes restent à explorer pour améliorer l'extraction automatique de mots-clés. En priorité, le calibrage des prompts pourrait affiner les résultats en fournissant aux LLMs des indications spécifiques, comme le nombre ou la longueur des mots-clés souhaités, tout en veillant à ne pas alourdir les requêtes API. De plus, structurer les prompts sous forme d'une conversation simulée entre l'utilisateur et le modèle pourrait améliorer la consistance des réponses en réduisant les variations dans le format des mots-clés générés, avec un impact direct sur le F1-Score, en particulier pour des modèles comme Gemma de Google.

Une autre piste prometteuse est de fine-tuner un modèle LLM, ce qui permettrait d'intégrer des approches modernes tout en améliorant les performances des méthodes extractives. À moyen terme, élargir le traitement aux textes complets des articles, plutôt qu'aux seuls résumés, offrirait des opportunités pour extraire des mots-clés encore plus représentatifs du contenu global, notamment dans des contextes scientifiques (Teufel & Moens, 2002).

Enfin, pour valider nos résultats, il est essentiel d'élargir notre approche à des jeux de données de référence (gold standard) tels que SemEval (Kim *et al.*, 2010), Inspec (Hulth, 2003) et OpenKP (Xiong *et al.*, 2019). Ces corpus annotés permettent une évaluation objective, garantissant la comparabilité avec l'état de l'art ainsi qu'une robustesse sur divers domaines scientifiques.

Références

ALQAHTANI A., ALHAKAMI H., ALSUBAIT T. & BAZ A. (2021). A survey of text matching techniques. *Engineering, Technology & Applied Science Research*, **11**(1), 6656–6661.

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

BENNANI-SMIRES K., MUSAT C., HOSSMANN A., BAERISWYL M. & JAGGI M. (2018). Simple unsupervised keyphrase extraction using sentence embeddings. *arXiv preprint arXiv :1801.04470*.

BOUDIN F. (2018). Unsupervised keyphrase extraction with multipartite graphs. *arXiv preprint arXiv :1803.08721*.

BOUGOUIN A., BOUDIN F. & DAILLE B. (2013). Topicrank : Graph-based topic ranking for keyphrase extraction. In *International joint conference on natural language processing (IJCNLP)*, p. 543–551.

CAMPOS R., MANGARAVITE V., PASQUALI A., JORGE A., NUNES C. & JATOWT A. (2020). Yake ! keyword extraction from single documents using multiple local features. *Information Sciences*, **509**, 257–289.

CHURCH K. W. & GALE W. A. (1995). Poisson mixtures. *Natural Language Engineering*, **1**(2), 163–190. DOI : 10.1017/S1351324900000135.

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

EL-BELTAGY S. R. & RAFEA A. (2009). Kp-miner : A keyphrase extraction system for english and arabic documents. *Information systems*, **34**(1), 132–144.

FLORESCU C. & CARAGEA C. (2017). Positionrank : An unsupervised approach to keyphrase extraction from scholarly documents. In *Proceedings of the 55th annual meeting of the association for computational linguistics (volume 1 : long papers)*, p. 1105–1115.

GROOTENDORST M. (2020). Bertopic : Leveraging bert and c-tf-idf to create easily interpretable topics. *Zenodo, Version v0*, **9**(10.5281).

HULTH A. (2003). Improved automatic keyword extraction given more linguistic knowledge. In *Proceedings of the 2003 conference on Empirical methods in natural language processing*, p. 216–223.

KIM S. N., BALDWIN T., KAN M.-Y. & KUMIKO T.-I. (2010). Semeval-2010 task 5 : Automatic keyphrase extraction from scientific articles. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, p. 21–26 : Association for Computational Linguistics.

KULUMBA F., ANTOUN W., VIMONT G. & ROMARY L. (2024). Harvesting textual and structured data from the hal publication repository. *arXiv preprint arXiv :2407.20595*.

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.

MIHALCEA R. & TARAU P. (2004). Textrank : Bringing order into text. In *Proceedings of the 2004 conference on empirical methods in natural language processing*, p. 404–411.

MIKOLOV T. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv :1301.3781*, **3781**.

PAGLIARDINI M., GUPTA P. & JAGGI M. (2017). Unsupervised learning of sentence embeddings using compositional n-gram features. *arXiv preprint arXiv :1703.02507*.

PAPAGIANNOPOULOU E. & TSOUMAKAS G. (2020). A review of keyphrase extraction. *Wiley Interdisciplinary Reviews : Data Mining and Knowledge Discovery*, **10**(2), e1339.

SALTON G. & BUCKLEY C. (1988). Term-weighting approaches in automatic text retrieval. *Information processing & management*, **24**(5), 513–523.

SCHOPF T., KLIMEK S. & MATTHES F. (2022). Patternrank : Leveraging pretrained language models and part of speech for unsupervised keyphrase extraction. *arXiv preprint arXiv :2210.05245*.

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

SONG M., GENG X., YAO S., LU S., FENG Y. & JING L. (2023). Large language models as zero-shot keyphrase extractors : A preliminary empirical study. *arXiv preprint arXiv :2312.15156*.

TEUFEL S. & MOENS M. (2002). Summarizing scientific articles : experiments with relevance and rhetorical status. *Computational linguistics*, **28**(4), 409–445.

TURNEY P. D. (2000). Learning algorithms for keyphrase extraction. *Information retrieval*, **2**, 303–336.

VASWANI A. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*.

WAN X. & XIAO J. (2008). Single document keyphrase extraction using neighborhood knowledge. In *AAAI*, volume 8, p. 855–860.

XIONG C., GUPTA N., CHEN Z., ZHONG V., WANG F. & SOCHER R. (2019). Openkp : A large-scale keyphrase extraction benchmark. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing (EMNLP)* : Association for Computational Linguistics.