Actes de CORIA-TALN 2026
Nantes, France, 29 juin au 3 juillet 2026

Résumé juridique automatique : sélection de connaissances fondée sur un graphe de connaissances

Yannis CHUPIN[1] Monique Rimbert[1]
(1) Nantes Université, École Centrale Nantes, CNRS, LS2N, UMR 6004, F-44000 Nantes, France
yannis.chupin@etu.univ-nantes.fr, monique.rimbert@etu.univ-nantes.fr

RÉSUMÉ

Les grands modèles de langage (LLMs) peinent à résumer des décisions judiciaires longues en raison de leur contexte limité et de leur tendance à générer des informations non pertinentes. Nous explorons l'utilisation de graphes de connaissances (GC) pour structurer l'information et proposons cinq stratégies de sélection de sous-graphes pertinents. Nos expériences, menées sur 114 décisions de la Cour suprême américaine, révèlent que la sélection reposant sur le syllabus et la composante connexe principale améliore la précision des résumés tout en réduisant significativement le coût énergétique et temporel. Cependant, la phase d'indexation des GC reste coûteuse, limitant les gains pour des tâches ponctuelles. Nos travaux ouvrent la voie à des applications multi-tâches où le GC pourrait être réutilisé.

ABSTRACT

Addressing relevant knowledge selection with a knowledge graph for legal summarisation

Large Language Models (LLMs) struggle to summarize lengthy legal decisions due to their limited context window and propensity to generate irrelevant information. We explore the use of knowledge graphs (KGs) to structure information and propose five strategies for selecting relevant subgraphs. Our experiments, conducted on 114 U.S. Supreme Court decisions, demonstrate that selection based on the syllabus and the main connected component enhances summary accuracy while significantly reducing computational and temporal costs. However, the KG indexing phase remains resource-intensive, limiting efficiency gains for one-off tasks. Our work paves the way for multi-task applications where the KG can be reused.

MOTS-CLÉS : Résumé automatique, Recherche d'information, Graphes de connaissances, Droit.

KEYWORDS: Automatic summarization, Information retrieval, Knowledge graphs, Legal domain.

1 Introduction

De nombreux pays, dont les États-Unis, fonctionnent selon le *common law*. Dans ce système juridique, les juges fondent leurs décisions sur la jurisprudence plutôt que des codes écrits. Le *case brief*, un résumé d'une affaire, permet aux experts, clients ou étudiants de comprendre ces affaires, d'identifier les jurisprudences étayant leur défense, ou encore d'anticiper l'application de ces décisions à posteriori (Mentzingen *et al.*, 2025). Cependant, rédiger ces résumés est long, cher et fastidieux, alors que le volume de *décisions de justice* à traiter ne cesse d'augmenter (Farzindar & Lapalme, 2004). D'où l'importance de l'automatisation pour les nombreuses affaires qui manquent de résumé (Jain *et al.*,

Actes des 19e Rencontres Jeunes Chercheurs en RI (RJCRI) et 28ème Rencontre des Étudiants Chercheurs en Informatique pour le Traitement Automatique des Langues (RECITAL), pages 163–185.

2021; Akter *et al.*, 2025).

Les documents judiciaires sont généralement longs et utilisent une terminologie légale très précise, les résumer automatiquement n'est donc pas trivial. Leur structure ne suit aucune convention, et n'ont en commun que leur nature de jurisprudence propre au cadre juridique. En effet, chaque fait décrit y est accompagné d'analyses et de références à de précédentes affaires afin de justifier la conclusion des juges Dans cet article, nous aborderons donc la tâche de résumé en se concentrant sur l'identification des faits d'une affaire, une section d'un *case brief*.

Pour produire un résumé fluide les approches abstractives sont les plus utilisées. Mais le fait qu'elles reposent souvent sur des grands modèles de langage (LLM) induit des difficultés à respecter la terminologie judiciaire (Akter *et al.*, 2025) et à s'adapter aux trop longs contextes en accordant une importance excessive aux extrémités du contexte au détriment des éléments situés au centre (Bishop *et al.*, 2024). Récemment, les approches *GraphRAG* (Edge *et al.*, 2025) ont montré un potentiel pour faire face à ces limites. La combinaison de la génération à enrichissement contextuel (RAG) avec des graphes de connaissances montre de bonnes performances pour le *résumé ciblé par requête* (RCR). Mais le coût en calcul du GraphRAG à l'indexation demeure la barrière principale. En effet, cette phase repose sur la génération de textes résumant des groupements d'idées fortement liées appelés *communautés*, par un LLM. Or, ce processus multiplie les appels au LLM, engendrant ainsi une consommation importante de temps et d'énergie.

Pour éviter le calcul des résumés communautaires, nous étudions cinq stratégies de recherche d'informations dans le graphe de connaissances (GC), pour générer des *résumés de faits* en sélectionnant : (1) tout le GC en contexte, (2) seulement sa composante principale, (3) les entités possédant des relations, (4) les communautés de Leiden les plus pertinentes, (5) le contenu lié à celui du syllabus. Nous évaluons ces stratégies sur des décisions de justices de la *Cours Suprême des États Unis d'Amérique* (Scotus). Nous montrons aussi que les LLMs comprennent le contenu des graphes de connaissances. À contrario, nous observons que les LLMs ont des difficultés à capturer ce que constitue un fait de l'affaire puisqu'ils ajoutent des informations non pertinentes pour le résumé des faits, mais valides pour d'autres sections d'un résumé d'affaire.

2 Travaux connexes

Les RCR mélangent les notions de Question-Réponse (Q&R) et la tâche de résumé de textes. Les Q&R répondent à des demandes spécifiques, alors que les résumés automatiques raccourcissent les documents. De fait, les RCR, répondent à des questions à la fois ouvertes et complexes sans viser d'entité en particulier. C'est-à-dire que les RCR nécessitent du raisonnement à la manière des Q&R non factuelles (Deng *et al.*, 2024) tout en allant plus loin que les résumés génériques, puisqu'ils ciblent les propos du résumé autour d'un type d'informations (Kanapala *et al.*, 2019; Liu *et al.*, 2024). Des exemples sont disponibles dans la table 1.

Tâche	**Exemple de requête**
Question-réponse (Q&R)	Monsieur X a-t-il tué Madame Y ?
Q&R non factuelle	La SCOTUS se saisira-t-elle de l'affaire ?
Résumé générique	Résume les documents : (*pas de requête spécifique*)
RCR	Quels sont les faits de l'affaire mentionnée dans les documents ?

TABLE 1 – Exemples de requêtes pour différent types de tâches.

Résumer des affaires est un sous-domaine du résumé de textes en général et a récemment gagné en popularité (Kanapala *et al.*, 2019). Cette tâche fait face à des défis tels que la longueur et la complexité, notamment langagière — par exemple le mot anglais « *sentence* » a deux sens : une phrase ou une peine — des documents judiciaires à traiter (Farzindar & Lapalme, 2004; Kanapala *et al.*, 2019; Shukla *et al.*, 2022). Les méthodes abstractives pâtissent du contexte limité des LLM et de biais d'entraînement issus de corpus journalistiques, souvent imprécis juridiquement (ex. confusion entre « article 49-3 » et « article 49 al. 3 »). De plus, l'évaluation automatique reste complexe, tant par la pluralité des résumés valides, que par l'exigence de rigueur sémantique, décourageant ainsi l'usage de synonymes approximatifs.

La génération augmentée par récupération — connu aussi sous *Retrieval Augmented Generation* (RAG) — a montré une capacité de réduction de génération d'informations non factuelles (Lewis *et al.*, 2020). Le but du RAG est de donner au LLM un contexte plus court et adéquat à l'aide d'un extracteur, qui récupère des fragments de données dans une base souvent sémantique. Cette approche donne de bons résultats en Q&R, mais pas en RCR. Face à cela, l'approche *GraphRAG*, proposée par (Edge *et al.*, 2025), intègre la connaissance des documents au sein d'un graphe de connaissances et semble plus pertinente par sa « *capacité à produire une compréhension globale à partir d'un vaste corpus de textes* ». En effet, le graphe de connaissances améliore le contexte via les chemins qui peuvent en être extraits et qui représentent des entités et leurs liens (Peng *et al.*, 2026). GraphRAG a aussi été mis en œuvre avec succès dans le domaine médical qui présente des enjeux de terminologie similaire (Wu *et al.*, 2025). Avant GraphRAG (Edge *et al.*, 2025), de nombreuses approches ont été proposées reposant sur une base de connaissances sous forme de graphe (Peng *et al.*, 2026) . Tout comme le RAG, celles-ci se composent de trois étapes : l'*indexation*, la *récupération* et la *génération*, avec la particularité de GraphRAG qui introduit une phrase de récupération à multiples générations en générant des résumés intermédiaires (voir Figure 1).

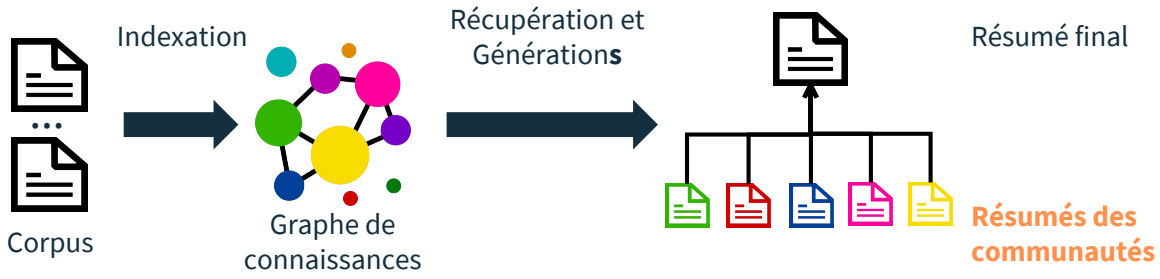

FIGURE 1 – Fonctionnement de GraphRAG pour la tâche du résumé

L'indexation du graphe diffère selon la classe du graphe : des *arbres* représentent les similarités entre des documents ; les *graphes de documents* forment des relations hiérarchiques entre segments de texte et résumés ; les GC — dans lequel s'inscrit GraphRAG — révèlent les relations entre les différentes entités d'un corpus. Afin de construire manuellement la base de connaissances, les approches à graphes de connaissances font appel notamment à des techniques de reconnaissance d'entités nommées et d'extraction de relations à partir de grands modèles de langages, d'enrichissement textuel des arcs et sommets, ainsi que des méthodes de détection de communautés. GraphRAG se base notamment sur ces communautés afin de pré-générer des résumés à différentes échelles afin d'enrichir le graphe de connaissances.

La phase de récupération des connaissances consiste à sélectionner le sous-graphe candidat se rapprochant au mieux de la requête demandée. Peng *et al.* (2026) définissent au moins deux types de

modèles et deux paradigmes de récupération. Les modèles *non paramétriques* emploient simplement des heuristiques sur les graphes, contrairement aux *modèles de langage* qui permettent de considérer les données textuelles ou d'exploiter la structure du graphe. Cette récupération peut s'effectuer itérativement ou en un seul traitement. Dans GraphRAG, la connaissance est récupérée à partir des résumés de communautés. De manière itérative, chaque résumé est sélectionné en fonction de la taille du contexte restant disponible ainsi que la pertinence de la communauté correspondante.

Le partitionnement des graphes de connaissances est pertinent dans la mesure où leur structure peut être étudiée pour comprendre la connaissance qu'ils détiennent. En effet, calculer les composantes connexes du graphe permet de détecter les parties isolées du graphe (Tarjan, 1972). Nous appelons *principale composante connexe* la composante connexe contenant le plus de sommets. Si un graphe de connaissances s'apparentait à une histoire, sa principale composante connexe serait le cœur de son narratif. En parallèle, la détection de communauté permet de comprendre la structuration et les interactions au sein du graphe en agglomérant les sommets en fonction de leur contenu (Li *et al.*, 2024). *L'algorithme de Leiden* (LDA) (Traag *et al.*, 2019) se base sur la modularité pour détecter les communautés, il décide du partitionnement en comparant la densité des arrêtes au sein de la communauté candidate. Contrairement à Louvain, LDA garanti que les communautés ont une forte densité interne et sont connectées (Blondel *et al.*, 2008).

3 Extraire la connaissance dans un graphe pour résumer les faits

Dans cette section, nous allons expliquer comment parvenir au résumé final. Traditionnellement, le résumé d'affaire (*case brief*) contient les sections suivantes (Northridge, 2021; University of Wisconsin-Madison, 2025) :

> (1) Faits de l'affaire : « *the history of the dispute, including the events that led to the lawsuit, the legal claims and defences of each party, and what happened in the trial court* »
> (2) Question(s) juridique(s) : « *questions of law that the court must answer in order to decide which party should win* »
> (3) Conclusion : « *the answer to the questions and the reasoning behind this decision* »

Le *résumé des faits* — c'est-à-dire la section des faits de l'affaire (1) — est le résumé final que nous voulons générer. Nous détaillons donc comment nous passons de documents judiciaires à un résumé des faits. Notre objectif est d'extraire les connaissances des documents vers un graphe, puis de les exploiter pour la rédaction finale (voir fig. 2).

3.1 Documents écrits de l'affaire

Les documents à partir desquels nous générons ces résumés sont des décisions d'affaires de la Cour Suprême des États-Unis, disponibles sur la plateforme Justia [1]. De manière générale, ces décisions sont composées de deux parties : un syllabus, suivi d'une ou plusieurs opinions. Le *syllabus* est une section écrite par le *Reporter of Decisions* [2]. Les *opinions* décrivent les raisons pour lesquelles les juges de la cour suprême sont en accord ou non avec la conclusion d'une affaire (Justia, 2026).

1. https://supreme.justia.com/cases/federal/us/
2. Un officiel ayant la responsabilité de publier les décisions de la cour.

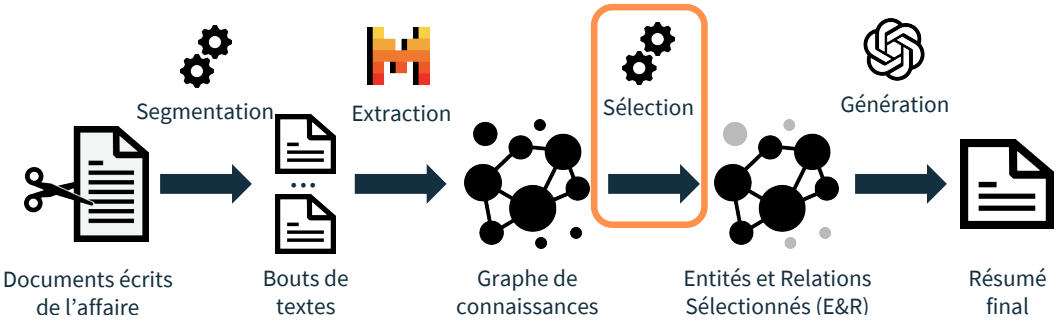

FIGURE 2 – Processus global de génération de résumés de faits. Entourée en orange, la phase de sélection (ou récupération) dans laquelle s'installent nos cinq stratégies.

3.2 Préparation des données et segmentation

Un nettoyage des documents légaux est nécessaire pour résumer les affaires. En particulier, nous effaçons la pagination, supprimons les éléments dupliqués (documents et textes) tout en réparant l'encodage de sorte à retrouver les sauts de lignes. Par ailleurs, les segments de textes constituent la base depuis laquelle les entités et relations sont extraites. Ils proviennent du découpage des documents légaux et l'objectif est d'agglutiner les paragraphes jusqu'à arriver à la fenêtre de contexte choisie. Nous évaluons la taille de cette fenêtre en comptant les tokens. Si les paragraphes sont indétectables ou trop grands, nous agglomérons les phrases à la place. De cette manière, nous pouvons contrôler la taille de contexte pour les LLMs qu'importe la taille de l'affaire.

3.3 Extraction des entités et leurs relations

Avec un LLM, nous extrayons ensuite les *entités* pertinentes de chaque segment ainsi que les *relations* entre ces dernières. Les deux extractions sont effectuées au sein de la même requête (Annexe G). Notre requête demande au modèle qu'il ne prélève des relations que s'il estime que les entités sont significativement reliées. Les entités sont définies par un nom, un type et une description, tandis que les relations ont un nom source, un nom cible, un descripteur (p. ex. `has_fact`) et une description (une phrase expliquant la relation). Une description est demandée avec le descripteur pour que les LLMs justifient les relations. Nos tests ont montré que n'exiger qu'un descripteur conduisait le LLM à constituer des relations insensées ou à choisir de mauvais descripteurs. Parfois, les LLMs forment des relations pour lesquelles il manque une entité source ou cible valide. Nous décidons de les conserver malgré tout, car même s'il manque un des deux liens, l'information qu'apportent ces relations partiellement orphelines reste pertinente. Les relations sans cible et source valide sont jetées.

3.4 Stratégies de sélection du contexte

Le graphe de connaissances résulte naturellement de l'association des entités au travers des relations relevées. Plus précisément, toutes les entités du même nom sont rassemblées dans une même entité au sein du graphe, les types et descriptions deviennent donc des listes. Pour ce qui est des relations semi-orphelines, nous les associons à leur parent au travers d'une relation artificielle, comme si nous ajoutions un commentaire à l'entité (exemples en Annexe D). Désormais, il reste à sélectionner

les entités et relations à inclure dans le contexte du LLM pour écrire le résumé final, avec une des stratégies suivantes :

(1) **Naïf** est le nom de notre première stratégie de sélection. Elle nécessite de donner l'entièreté du graphe de connaissances au modèle dans son contexte. Cependant, cette approche peut se révéler assez coûteuse, peu robuste à la taille de l'affaire, d'où notre volonté de la raffiner.
(2) **CompPrinc** est une stratégie qui repose sur l'exploitation de la topologie du graphe pour en extraire la composante connexe principale. L'intuition ici est que l'« histoire » d'une affaire est plutôt continue, donc les éléments qui ne se rattachent pas à ce fil (la composante connexe principale) sont soit des erreurs du LLM soit des éléments insignifiants. Les méthodes suivantes s'appuient toutes sur celle-ci pour effectuer leur sélection.
(3) **Syllabus** sélectionne uniquement les entités du syllabus présentes dans la composante connexe principale. Les syllabus ont le désavantage d'être parfois absents des documents (p. ex. Annexe H).
(4) **EntLiées** quant à elle s'appuie sur l'orientation des relations et exclut toute entité ne conduisant à aucune entité de la même composante par une relation, et n'ayant aucun fait attaché. L'intuition ici est de conserver les entités rattachées au plus d'information.
(5) **Rerank**, pour finir, utilise les communautés de Leiden détectées et les classes en fonction d'un score. Puis, nous prenons les n premières au classement en fonction de la place disponible dans le contexte. Le score évalue le niveau de connectivité de la communauté avec les autres communautés ainsi que la taille de la communauté. L'intuition ici est que l'on cherche à trouver une idée ou un thème central et important de l'affaire.

3.5 LLM et stratégies de requêtes génératives

Une fois le contexte rassemblé avec l'une des cinq approches, celui-ci est ajouté au sein de la requête générative (Annexe F) destinée à la rédaction finale du résumé des faits. Notre méthode utilise les LLMs à deux étapes du processus : l'extraction de connaissances et la rédaction du résumé final. Pour mesurer quel modèle fonctionne le mieux à chaque étape, nous testons notre méthode intégrale sur des résumés cibles — au lieu des décisions de justice — associés aux affaires. Les affaires concernées par le test sont isolées du reste du jeu de données. L'idée est d'essayer de réassembler le résumé cible à partir de son graphe de connaissances. De cette manière, nous vérifions la compréhension du graphe de connaissances par le LLM. Ce test évalue aussi le formatage des connaissances extraites et d'évaluer différentes stratégies de requêtes génératives (voir Table 2).

	BLEU	R1	R2	RL	METEOR	BARTScore	BERTScore	UNIEval
Baseline	8.371	0.403	0.163	0.232	0.372	-2.954	0.858	0.659
gpt-oss :20b	30.70	0.68	0.49	0.54	0.54	-1.69	0.92	0.69
llama3.1 :8b	29.39	0.63	0.45	0.47	0.53	-1.98	0.91	0.64
mistral :7b	32.72	0.66	0.46	0.49	0.54	-1.74	0.92	***0.68***
qwen3 :8b	***31.01***	***0.67***	0.49	***0.53***	0.54	***-1.62***	0.92	***0.68***

TABLE 2 – Évaluation des résumés générés par notre approche à partir des résumés cible avec plusieurs LLMs pour évaluer leur compréhension des graphes de connaissances. Les meilleurs scores sont soulignés et les deuxièmes meilleurs ***en italique gras***.

4 Protocole expérimental

Jeu de données	Partie	Moy.	Std.	Min.	Med.	Max.
Entraînement (2756)	Décision	12 977	10 276	61	10 581	147 983
	Faits	194	95	3	183	791
	Conclusions	184	113	3	148	950
Échantillon (114)	Décision	14 230	12 696	149	10 361	84 750
	Faits	202	89	31	192	484
	Conclusions	192	128	6	163	950

TABLE 3 – Statistiques sur le nombre de mots des jeux de données utilisées calculé avec en_core_web_sm@3.8.0 (Honnibal *et al.*, 2020)

Notre expérience repose sur un sous-ensemble de 114 documents issus du jeu de données d'entraînement (Zakaria, 2025; Srun *et al.*, 2024) comprenant à l'origine 2756 décisions de justice (voir Table 3 et Annexe B). Le jeu d'origine contient aussi les résumés cibles respectifs [3] qui incluent les sections faits, question rhétorique et conclusion. Ces résumés sont écrits par des membres de l'équipe Justia. Les documents qui constituent l'échantillon ont été choisis aléatoirement, mais de sorte à couvrir de manière plus ou moins uniforme la période allant de 1911 à 2024 (Annexe A). Toutefois, nous y incluons davantage d'affaires récentes dans l'objectif de limiter les effets de contamination des LLMs sur la phase de génération. Nos expériences ont été menées localement sur un serveur équipé d'une carte graphique NVIDIA L4. Les entités et relations sont extraites avec `Mistral:7b` [4] et `GPT-OSS:20b` [5] permet de générer les résumés finaux.

4.1 Métriques d'évaluation

Afin d'évaluer la qualité des résumés générés, nous utilisons les métriques suivantes :

(1) **SacreBLEU** (Post, 2018; Papineni *et al.*, 2002) est à l'origine destinée à la tâche de traduction. Les métriques BLEU mesurent le recouvrement du contenu et sont orientées précision. SacreBLEU fixe un ensemble de paramètres de BLEU permettant ainsi la comparaison entre différents papiers.
(2) **ROUGE-N et L** (Lin, 2004) ont été conçues pour évaluer la tâche de résumé. Celles-ci mesurent les recouvrements entre un candidat et un ou plusieurs résumés cible et sont orientées rappel.
(3) **METEOR** (Banerjee & Lavie, 2005) fait correspondre les unigrames de différentes façons (synonimes, formes racinisées...) pour être moins sensible à la paraphrase que BLEU. METEOR considère à la fois précision, rappel de la concordance d'uni-grammmes, ainsi que leur ordre d'apparition.
(4) **UniEVAL - Consistency** (Zhong *et al.*, 2022) repose sur un système de question-réponses booléenne avec un LLM. Elle permet de mesurer la cohérence — s'il n'y a pas contradiction — entre un propos à évaluer et un document source.
(5) **BARTScore** (Yuan *et al.*, 2021) calcule la log-probabilité qu'une paire de séquences se génèrent l'une l'autre. Elle permet de capturer les chevauchements et la couverture sémantique entre le résumé cible et le résumé généré.
(6) **BERTScore** (Zhang *et al.*, 2020) évalue la similarité sémantique entre différentes séquences à l'aide de prolongements.

3. Les résumés cibles ont été récupérés via l'API Oyez.
4. https://ollama.com/library/mistral
5. https://ollama.com/library/gpt-oss:20b

Afin d'évaluer le coût d'exécution , nous utilisions la librairie codecarbon (Courty *et al.*, 2024) :

(1) **Temps** : évalue en seconde le temps d'exécution moyen pour une décision de justice.
(2) **Émissions** : estime en gramme la quantité moyenne de CO_2 émis pour une décision de justice.
(3) **Puissance GPU** : estime en Watt la puissance utilisée par le GPU pour une décision de justice.

4.2 Approche de référence

Nous définissons également une méthode `Baseline` qui produit des résumés de faits directement à partir de LLM. Nous l'appliquons sur les mêmes données et lui fournissons l'entièreté de la décision de justice à l'intérieur du contexte du modèle. Afin de sélectionner le modèle et la requête générative (requête en Annexe E) de cette approche, nous avons mené une expérimentation sur un échantillon isolé de celui présenté dans la section 4, en se basant sur les mêmes métriques présentées plus tôt.

Nous testons aussi une implémentation `GraphRAG` de Microsoft (Edge *et al.*, 2025) sur un sous-échantillon de notre ensemble (quinze affaires). Par souci de comparabilité, nous utilisons `Mistral:7b` pour la première phase d'indexation (en passant les étapes liées aux résumés communautaires). Puis, nous utilisons `GPT-OSS:20b` pour générer les rapports communautaires et les résumés.

5 Résultats

Tout d'abord, nous constatons de meilleurs résultats avec nos approches qu'avec `GraphRAG`. Alors que `Baseline` obtient des scores encore meilleurs pour résumer les faits. Cependant, en les lisant, nous avons remarqué que celle-ci inclue des informations qui n'apparaissent pas dans le document source. Cela semble surtout affecter les jugements rendus *Per Curiam*[6].

Au-delà des métriques, le coût de `GraphRAG` se révèle parfaitement inadapté à la tâche, tant il est supérieur aux autres méthodes. Du reste, `Naïf` mis à part, générer les résumés à partir d'entités et relations est moins coûteux en temps et en énergie que la `Baseline` (voir Table 4). Cette observation est corrélée avec la taille des informations fournies en entrée à la phase de génération. Toutes les stratégies avancées de sélection d'entités et relation raccourcissent le contexte de moitié. L'ensemble de mots est ainsi réduit de 9949 à 678 mots pour représenter les entités et relations avec la stratégie `Syllabus` (plus de résultats, en Annexe C). Soit une réduction de 87 % par rapport au nombre de mots moyen d'un document, qui est de 11 786 mots. Notons par ailleurs que toutes ces méthodes sont à nuancer par la phase d'indexation qui dure plus de dix minutes.

	Baseline	GraphRAG	Naïf	CompPrinc	Syllabus	EntLiées	ReRank	Indexation
Durée (s)	29.70	5759.07	34.09	28.64	13.80	25.48	19.07	670.69
Émissions (g)	0.39	31.09	0.51	0.42	0.20	0.37	0.28	8.73
Puissance GPU (W)	71.46	70.75	71.34	70.53	68.39	70.66	69.98	71.89

TABLE 4 – Durée, émissions et puissance moyennes nécessaires à la génération et indexation d'une décision de justice avec une NVIDIA L4. GraphRAG combine indexation et génération.

6. Opinion dont le juge auteur est inconnu.

Ces résultats sont à nuancer puisque la taille du contexte en entrée n'a montré aucun impact direct sur la qualité des résumés de la `Baseline` et que le temps d'exécution de la phase d'indexation est importante. Approximativement dix minutes sont nécessaires afin d'extraire les entités et relations de tous les segments d'une décision. Parmi toutes les stratégies de sélection d'entités et relations, `Syllabus` obtient les meilleurs scores sur les métriques utilisant les n-gram, BARTScore et BERTScore (voir Table 5 ou Annexe I). Donner l'ensemble des entités et relations, en moyenne, présente les deuxièmes meilleurs scores, surtout avec les métriques utilisant la similarité sémantique.

	BLEU	R1	R2	RL	METEOR	BARTScore	BERTScore	UNIEval
Baseline	8.371	0.403	0.163	0.232	0.372	-2.954	0.858	0.659
GraphRAG (15)	1.808	0.160	0.033	0.098	0.188	-4.014	0.821	0.491
Naïf	***5.333***	0.301	***0.119***	0.167	0.325	***-3.410***	***0.843***	0.548
CompPrinc	5.103	0.305	0.113	0.166	***0.315***	-3.470	0.839	0.515
Syllabus	7.110	0.387	0.136	0.206	0.305	-3.276	0.850	0.481
EntLiées	4.832	0.303	0.111	0.166	0.313	-3.481	0.841	***0.517***
ReRank	4.346	***0.323***	0.093	***0.171***	0.273	-3.582	0.833	0.440

TABLE 5 – Évaluation des résumés générés en considérant les faits en tant que résumé cible. Les meilleurs scores sont soulignés et les deuxièmes meilleurs ***en italique gras***.

Afin de mieux comprendre le contenu des résumés générés, nous les évaluons en considérant, en plus des faits, la section conclusion du résumé cible. À l'exception de METEOR, nous observons une amélioration des métriques utilisant les n-gram pour la `Baseline`, `Syllabus` et `ReRank` (voir Table 6 ou Annexe J). Nous notons que cette croissance et moins importante pour la `Baseline` que pour `Naïf`. Parmi les stratégies de sélection d'entités, `Naïf` obtient les meilleurs résultats, suivi par `CompPrinc` et `EntLiées` qui montrent des résultats similaires.

	BLEU	R1	R2	RL	METEOR	BARTScore	BERTScore	UNIEval
Baseline	9.156	0.456	0.166	0.228	0.313	-2.884	0.849	0.550
Naïf	8.248	0.424	0.163	0.206	0.338	***-3.214***	0.846	0.531
CompPrinc	***7.709***	***0.420***	***0.154***	0.199	***0.321***	-3.298	0.842	0.500
Syllabus	6.511	0.405	0.144	***0.202***	0.248	-3.164	***0.845***	0.450
EntLiées	7.473	0.419	0.153	0.201	0.318	-3.313	0.844	***0.505***
ReRank	5.239	0.385	0.116	0.185	0.251	-3.465	0.834	0.439

TABLE 6 – Évaluation des résumés générés en considérant les faits *et* la conclusion en tant que résumé cible. Les meilleurs scores sont soulignés et les deuxièmes meilleurs ***en italique gras***.

6 Discussion

Au final, nos stratégies produisent des résumés moins coûteux et de meilleure qualité que ceux de `Graphrag`, puisque nous évitons la génération d'en moyenne 116,6 résumés communautaires par affaire indexée (médiane 92,5), qui constituent autant de requêtes à `GPT-OSS:20B`. Cependant, nos cinq méthodes ont tous des métriques inférieures à celle de `Baseline` dont le coût est à nuancer. En effet, la phase de génération à elle-seule est plus coûteuse que pour nos méthodes. Or, l'indexation n'est à faire qu'une fois pour nos méthodes et le graphe produit peut être réinterrogé. De ce fait, le coût amorti de nos méthodes est inférieur à celui de `Baseline`. En examinant les résumés ayant des

résultats aberrants, nous avons constaté des hallucinations pour `Baseline` que nous n'avons pas retrouvé avec nos méthodes (Annexe H).

L'analyse de nos résultats révèle que nos résumés de faits incluent parfois des éléments appartenant à d'autres sections du résumé d'affaire, comme la conclusion. En effet, recalculer nos métriques en ajoutant la section conclusion au résumé cible conduit à une augmentation généralisée des scores. Nous attribuons ce phénomène à la nature des documents judiciaires, qui sont rédigés par les juges pour justifier une décision finale. Bien que ces documents contiennent bel et bien des faits marquants, l'orientation argumentative de ces sources semble influencer la génération. Par ailleurs, la requête générative aussi semble jouer un rôle prépondérant. En effet, cette augmentation des métriques est plus forte pour nos cinq méthodes que pour `Baseline`, qui utilise une requête générative différente. Pour finir, on ne peut exclure que les modèles aient été influencés par la structure des résumés d'affaire qu'ils auraient pu rencontrer à leur entraînement.

6.1 Limites

Dans la section 3.5, nous avons utilisé les résumés cibles pour élaborer les requêtes génératives d'*extraction* et de *génération* pour la tâche de résumés de faits. Cette expérience atteste de la compréhension des GC par le modèle. Mais, elle ne vérifie pas que la connaissance ou les documents fournis en contexte portent l'information nécessaire et suffisante à la construction d'un résumé des faits. Elle n'atteste pas non plus de la capacité des modèles à sélectionner les informations pertinentes pour le résumé à générer.

Par ailleurs, notre jeu de données contient les transcriptions des plaidoiries, mais leur format actuel ne permet pas de distinguer les orateurs. En effet, leur incorporation aurait permis de connaître des faits qui ne sont pas mentionnés par les juges dans les documents écrits, mais qui sont mentionnés dans les résumés cibles.

Enfin, nos expériences avec `GraphRAG` se sont déroulées que sur un sous échantillon des 114 affaires en raison du coût que cette implémentation classique engendrait à l'écriture des rapports.

7 Conclusion

Ce travail propose cinq stratégies de sélection au sein de graphes de connaissances (GC) visant à réduire le coût computationnel associé aux méthodes de type GraphRAG lors de la rédaction de résumés d'affaires. Nos expérimentations indiquent que la sélection ciblée des entités et relations permet de réduire le nombre de requêtes aux modèles de langage (LLM) tout en conservant, voire en améliorant, les scores de qualité par rapport aux approches standards.

Nous démontrons que les LLM sont capables d'exploiter les connaissances structurées des GC et que nos méthodes de sélection permettent de stabiliser la taille du contexte génératif face à la longueur variable des documents sources. Cette maîtrise du contexte limite le coût de génération et rend l'approche plus robuste que les méthodes fondées uniquement sur une fenêtre de contexte. Bien que la phase d'indexation initiale demeure coûteuse, la structure en graphe permet d'amortir cet investissement sur le long terme. Contrairement aux approches par fenêtres glissantes qui saturent avec l'accumulation de documents, l'utilisation de GC offre une base d'information pérenne et flexible, capable d'agréger des volumes de données croissants sans nécessiter un retraitement systématique.

En somme, cette approche permet d'orienter le résumé juridique vers une exploitation plus sélective et efficiente des connaissances. Les travaux futurs pourront explorer l'intégration d'une dimension sémantique aux stratégies de sélection et l'extension du modèle aux différentes sections constitutives d'un résumé d'affaire.

Remerciements

Nous remercions Nicolas Hernandez pour nous avoir proposé ce sujet, ainsi que nos encadrants de stage Florian Boudin et Richard Dufour, pour leur accompagnement tout au long de nos travaux ainsi que Quentin Vantrepotte et Cyril Camachon pour leur soutien en amont de la conférence. Nous remercions également l'équipe TALN du laboratoire LS2N pour nous avoir accueillis, ainsi que les relecteurs anonymes pour leurs retours constructifs.

Références

AKTER M., ÇANO E., WEBER E., DOBLER D. & HABERNAL I. (2025). A Comprehensive Survey on Legal Summarization : Challenges and Future Directions. DOI : 10.48550/arXiv.2501.17830.

BANERJEE S. & LAVIE A. (2005). METEOR : An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments. In J. GOLDSTEIN, A. LAVIE, C.-Y. LIN & C. VOSS, Éds., *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, p. 65–72, Ann Arbor, Michigan : Association for Computational Linguistics.

BISHOP J. A., ANANIADOU S. & XIE Q. (2024). LongDocFACTScore : Evaluating the Factuality of Long Document Abstractive Summarisation. In N. CALZOLARI, M.-Y. KAN, V. HOSTE, A. LENCI, S. SAKTI & N. XUE, Éds., *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, p. 10777–10789, Torino, Italia : ELRA and ICCL.

BLONDEL V. D., GUILLAUME J.-L., LAMBIOTTE R. & LEFEBVRE E. (2008). Fast unfolding of communities in large networks. *Journal of Statistical Mechanics : Theory and Experiment*, **2008**(10), P10008. DOI : 10.1088/1742-5468/2008/10/P10008.

COURTY B., SCHMIDT V., LUCCIONI S., GOYAL-KAMAL, MARIONCOUTAREL, FELD B., LECOURT J., LIAMCONNELL, SABONI A., INIMAZ, SUPATOMIC, LÉVAL M., BLANCHE L., CRUVEILLER A., OUMINASARA, ZHAO F., JOSHI A., BOGROFF A., DE LAVOREILLE H., LASKARIS N., ABATI E., BLANK D., WANG Z., CATOVIC A., ALENCON M., MICHAŁ STĘCHŁY, BAUER C., DE ARAÚJO L. O. N., JPW & MINERVABOOKS (2024). mlco2/codecarbon : v2.4.1. DOI : 10.5281/zenodo.11171501.

DENG Y., ZHANG W., XU W., SHEN Y. & LAM W. (2024). Nonfactoid Question Answering as Query-Focused Summarization With Graph-Enhanced Multihop Inference. *IEEE Transactions on Neural Networks and Learning Systems*, **35**(8), 11231–11245. DOI : 10.1109/TNNLS.2023.3258413.

EDGE D., TRINH H., CHENG N., BRADLEY J., CHAO A., MODY A., TRUITT S., METROPOLITANSKY D., NESS R. O. & LARSON J. (2025). From Local to Global : A Graph RAG Approach to Query-Focused Summarization. DOI : 10.48550/arXiv.2404.16130.

FARZINDAR A. & LAPALME G. (2004). Legal Text Summarization by Exploration of the Thematic Structure and Argumentative Roles. In *Text Summarization Branches Out*, p. 27–34, Barcelona, Spain : Association for Computational Linguistics.

HONNIBAL M., MONTANI I., VAN LANDEGHEM S. & BOYD A. (2020). spaCy : Industrial-strength Natural Language Processing in Python. DOI : 10.5281/zenodo.1212303.

JAIN D., BORAH M. D. & BISWAS A. (2021). Summarization of legal documents : Where are we now and the way forward. *Computer Science Review*, **40**, 100388. DOI : 10.1016/j.cosrev.2021.100388.

JUSTIA (2026). Reading a Supreme Court Decision.

KANAPALA A., PAL S. & PAMULA R. (2019). Text summarization from legal documents : A survey. *Artificial Intelligence Review*, **51**(3), 371–402. DOI : 10.1007/s10462-017-9566-2.

LEWIS P., PEREZ E., PIKTUS A., PETRONI F., KARPUKHIN V., GOYAL N., KÜTTLER H., LEWIS M., YIH W.-T., ROCKTÄSCHEL T., RIEDEL S. & KIELA D. (2020). Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. In *Advances in Neural Information Processing Systems*, volume 33, p. 9459–9474 : Curran Associates, Inc.

LI J., LAI S., SHUAI Z., TAN Y., JIA Y., YU M., SONG Z., PENG X., XU Z., NI Y., QIU H., YANG J., LIU Y. & LU Y. (2024). A comprehensive review of community detection in graphs. *Neurocomputing*, **600**, 128169. DOI : 10.1016/j.neucom.2024.128169.

LIN C.-Y. (2004). ROUGE : A Package for Automatic Evaluation of Summaries. In *Text Summarization Branches Out*, p. 74–81, Barcelona, Spain : Association for Computational Linguistics.

LIU Y., WANG Z. & YUAN R. (2024). QuerySum : A Multi-Document Query-Focused Summarization Dataset Augmented with Similar Query Clusters. *Proceedings of the AAAI Conference on Artificial Intelligence*, **38**(17), 18725–18732. DOI : 10.1609/aaai.v38i17.29836.

MENTZINGEN H., ANTÓNIO N. & BACAO F. (2025). Effectiveness in retrieving legal precedents : Exploring text summarization and cutting-edge language models toward a cost-efficient approach. *Artificial Intelligence and Law*. DOI : 10.1007/s10506-025-09440-2.

NORTHRIDGE C. S. U. (2021). HOW TO BRIEF A CASE_Saunders.

PAPINENI K., ROUKOS S., WARD T. & ZHU W.-J. (2002). Bleu : A Method for Automatic Evaluation of Machine Translation. In P. ISABELLE, E. CHARNIAK & D. LIN, Éds., *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, p. 311–318, Philadelphia, Pennsylvania, USA : Association for Computational Linguistics. DOI : 10.3115/1073083.1073135.

PENG B., ZHU Y., LIU Y., BO X., SHI H., HONG C., ZHANG Y. & TANG S. (2026). Graph retrieval-augmented generation : a survey. *ACM Transactions on Information Systems*, **44**(2), 1–52. DOI : 10.1145/3777378.

POST M. (2018). A Call for Clarity in Reporting BLEU Scores. In *Proceedings of the Third Conference on Machine Translation : Research Papers*, p. 186–191, Belgium, Brussels : Association for Computational Linguistics. DOI : 10.18653/v1/W18-6319.

SHUKLA A., BHATTACHARYA P., PODDAR S., MUKHERJEE R., GHOSH K., GOYAL P. & GHOSH S. (2022). Legal Case Document Summarization : Extractive and Abstractive Methods and their Evaluation. In Y. HE, H. JI, S. LI, Y. LIU & C.-H. CHANG, Éds., *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 1 : Long Papers)*, p. 1048–1064, Online only : Association for Computational Linguistics. DOI : 10.18653/v1/2022.aacl-main.77.

SRUN N., HERNANDEZ N. & JACQUIN C. (2024). *Automated Generation Argumentative Role from Legal Opinions with Pertinent Information Recognition*. Rapport interne, Ls2N.

TARJAN R. (1972). Depth-First Search and Linear Graph Algorithms. *SIAM Journal on Computing*. DOI : 10.1137/0201010.

TRAAG V. A., WALTMAN L. & VAN ECK N. J. (2019). From Louvain to Leiden : Guaranteeing well-connected communities. *Scientific Reports*, **9**(1), 5233. DOI : 10.1038/s41598-019-41695-z.

UNIVERSITY OF WISCONSIN-MADISON (2025). A Guide to Case Briefing.

WU J., ZHU J., QI Y., CHEN J., XU M., MENOLASCINA F., JIN Y. & GRAU V. (2025). Medical Graph RAG : Evidence-based Medical Large Language Model via Graph Retrieval-Augmented Generation. In W. CHE, J. NABENDE, E. SHUTOVA & M. T. PILEHVAR, Éds., *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*, p. 28443–28467, Vienna, Austria : Association for Computational Linguistics. DOI : 10.18653/v1/2025.acl-long.1381.

YUAN W., NEUBIG G. & LIU P. (2021). BARTScore : Evaluating Generated Text as Text Generation. DOI : 10.48550/ARXIV.2106.11520.

ZAKARIA B. (2025). *Generating rhetorically coherent summaries of long legal documents*. Thèse de doctorat, Faculty of Sciences Semlalia Marrakech.

ZHANG T., KISHORE V., WU F., WEINBERGER K. Q. & ARTZI Y. (2020). BERTScore : Evaluating Text Generation with BERT. DOI : 10.48550/arXiv.1904.09675.

ZHONG M., LIU Y., YIN D., MAO Y., JIAO Y., LIU P., ZHU C., JI H. & HAN J. (2022). Towards a Unified Multi-Dimensional Evaluator for Text Generation. DOI : 10.48550/arXiv.2210.07197.

Annexes

A Détails sur le jeu de données

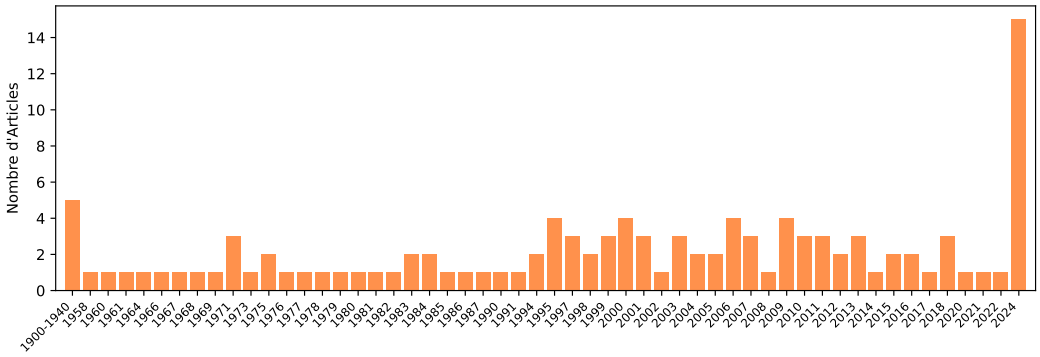

FIGURE A – Distribution sur les années des affaires de notre échantillon.

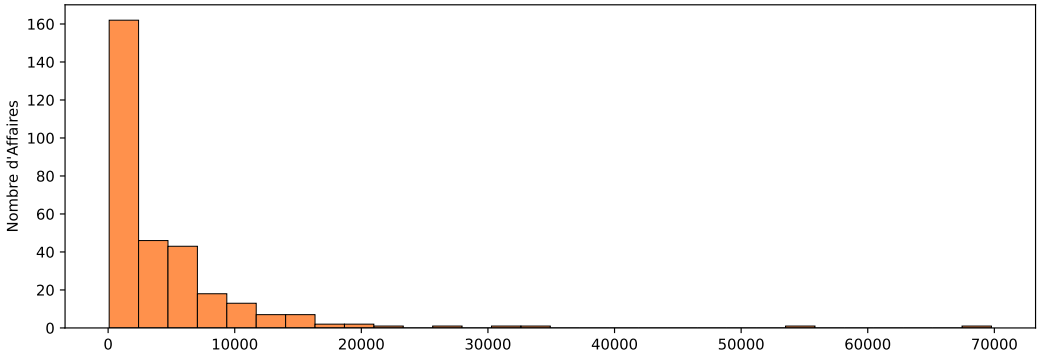

FIGURE B – Distribution des documents judiciaires par nombre de mots

B Contenu des graphes

	Ent.		Rel.		Rel. orph.		Total	
	#	#mots	#	#mots	#	#mots	#	#mots
Baseline								14 230
Naïf	199	6150	127	2455	60	1344	386	9949
CompPrinc	79	3212	105	2003	28	597	212	5812
Syllabus	7	468	5	107	5	104	17	678
EntLiées	53	2553	105	2003	28	597	186	5153
Rerank	16	716	20	367	6	123	42	1206

TABLE C – Nombre d'entités, relations et relations orphelines, ainsi que le nombre de mots nécessaires en moyenne afin de les représenter, obtenus selon la stratégie de sélection en comparaison avec le nombre moyen de mots d'une décision de justice fourni à la méthode de référence.

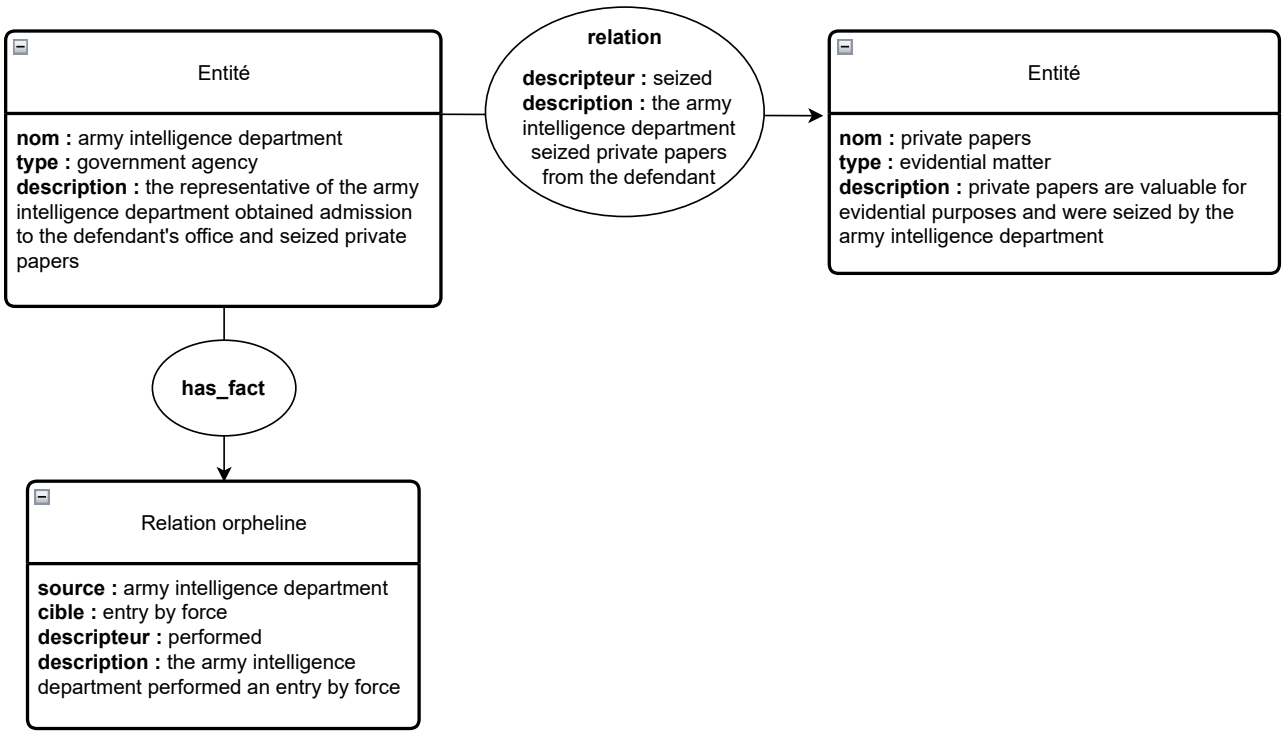

FIGURE D – Schéma de construction des relations du graphe de connaissances. Les entités du même nom sont agglomérées ensemble.

C Requêtes génératives

```
    ---Instructions--

      Write a single, coherent paragraph summarizing ONLY the FACTS of the case.
      Do not include the issue, procedural posture beyond what is necessary to explain
the facts, the holding, reasoning, legal rules, or any opinions.
      FACTS are the who, what, when, where, and why of the dispute.
      Include the key actors by name, the events leading to the lawsuit, the conduct of
each party, the claims and defenses raised, and what occurred in the trial court if
described.
      Do not use bullet points, headings, numbering, or lists. Produce one narrative
paragraph.
      Do not copy sentences from the case verbatim.
      Include only facts that were significant to the court's reasoning: a fact is
significant if the court repeats it, relies on it, or if the decision might have
changed without it.

    ---Example--

      Ramon Nelson was riding his bike when he suffered a lethal blow to the back of
his head with a baseball bat. After two eyewitnesses identified Lawrence Owens from an
array of photos and then a lineup, he was tried and convicted for Nelson’s death.
Because Nelson was carrying cocaine and crack cocaine potentially for distribution, the
judge at Owens’ bench trial ruled that Owens was probably also a drug dealer and was
trying to “knock [Nelson] off.” Owens was found guilty of first-degree murder and
sentenced to 25 years in prison. Owens filed a petition for a writ of habeas corpus on
the grounds that his constitutional right to due process was violated during the trial.
He argued that the eyewitness identification should have been inadmissible based on
unreliability and that the judge impermissibly inferred a motive when a motive was not
an element of the offense. The district court denied the writ of habeas corpus, and
Owens appealed. The U.S. Court of Appeals for the Seventh Circuit reversed the denial
and held that the trial judge’s inference about Owens’s motive violated his right to
have his guilt adjudicated solely based on the evidence presented at trial.

    ---Case--

    {{decision}}
```

FIGURE E – Requête générative de la méthode de référence

```
### Instruction ###

You are a **legal automatic summarizer** specializing in synthesizing structured legal
data into **coherent, consistent, and fluent plain-text narratives**. Your task is to
transform provided entities, relationships, and facts into a **clear, logically
structured, and legally precise** text.

#### **Requirements**
- **Output Format**: Plain text only. Avoid using Markdown, lists, or any other syntax.
- **Language**: English only.
- **Insufficient Information**: If the list of entities is empty, the output MUST BE
only **N/A**.
- **Factual Accuracy**: Only include information directly supported by the provided
data. You will be penalized if you infer, speculate, or add unsupported details.
- **Relevancy**: Ensure the output use only the provided entities, relationship and
facts. You will be penalized if you add implicit details or details missing from the
given information.
- **Legal Precision**: Refrain from altering the vocabulary used.
- **Structure**: Organize the narrative to reflect legal relevance and logical flow.

#### **Input Data Structure**
- **Entities**: Each entity is defined by its name, type, and description.
- **Relationships**: Each relationship connects entities with a verb or phrase and
includes a supporting statement.
- **Facts**: Each fact is a statement about an entity or relationship.

###Your task###
Given the following input data, produce a plain-text narrative that adheres to the
above requirements.

**Input:**
- **Entities**:
{{entites}}

- **Relationships**:
{{relations}}

- **Facts**:
{{relations_orphelines}}

**Output:**
```

FIGURE F – Requête générative pour la phase de génération de notre approche

```
##INSTRUCTIONS##
You are an excellent legal assistant. Your task is to extract entities and
relationships related to **legal facts** from a given U.S. Supreme Court case chunk.
#### Types of legal facts
 - Material facts directly affect the outcome of a legal issue or case
 - Immaterial facts do not significantly impact the legal analysis or outcome
 - Disputed facts are contested by opposing parties and require resolution
 - Undisputed facts are agreed upon by all parties involved
 - Direct evidence directly proves a fact without inference (ex: eyewitness testimony)
 - Circumstantial evidence requires inference to connect it to a conclusion (ex:
fingerprints at a crime scene)
#### Steps
1. Identify all entities. For each identified entity, extract the following
information:
    - entity name: Name of the entity, capitalized
    - entity type: A type describing the status, role, object, location, event, norm or
process type of the entity, capitalized
    - entity description: Comprehensive description of the entity's attributes and
activities
2. From the entities identified in step 1, identify all pairs of (source entity, target
entity) that are *clearly related* to each other. For each pair of related entities,
extract the following information:
    - source entity: name of the source entity, as identified in step 1
    - target entity: name of the target entity, as identified in step 1
    - relationship descriptor: a short predicate (i.e. verb phrase, without subject and
object) that would vaguely describe how the source is related to the target
    - relationship description: a complete sentence that would describe how the source
is related to the target
##### Output format
Your answer MUST be returned in English as a list of xml object:
- You MUST declare entities using:
    <entity name="{{entity name}}" type="{{entity type}}">{{entity
description}}</entity>
- Otherwise, you MUST declare relationships using:
    <relationship>
        <source name="{{source entity}}"></source>
        <target name="{{target entity}}"></target>
        <description descriptor="{{relationship descriptor}}">{{relationship
description}}</description>
    </relationship>
###Example###
 - Example case:
{{ "In 1925, in Burk-Waggoner Oil Assn. v. Hopkins, the Court articulated that
fundamental principle. 269 U.S. 110. The case involved a tax on the income of an entity
that state law treated as a partnership. Under state law, the partnership's property
was considered the property of the partners. For that reason, the partnership argued
that Congress had to tax the partners on the income and could not tax the
partnership.\\
This Court rejected that argument. The Court stated: "Neither the conception of
unincorporated associations prevailing under the local law, nor the relation under that
law of the association to its shareholders, nor their relation to each other and to
```

```
outsiders, is of legal significance as bearing upon the power of Congress to determine
how and at what rate the income of the joint enterprise shall be taxed.” Id., at 114.
In other words, Congress could tax the income as it chose, taxing either the
partnership or the partners on the partnership’s undistributed income. So the tax on
the partnership was proper.\\
”}}
 - Example output:
<entity name="BURK-WAGGONER OIL ASSN. V. HOPKINS" type="CASE">Burk-Waggoner Oil Assn.
v. Hopkins involved a tax on the income an entity treated as a partnership</entity>
<entity name="COURT" type="COURT">The court articulated a fundamental principal in
1925</entity>
<entity name="PARTNERSHIP" type="NON-LEGAL PARTICIPANT">A  partnership is an entity
that can be taxed</entity>
<entity name="PARTNERS" type="NON-LEGAL PARTICIPANT">Partners own the property of the
partnership</entity>
<entity name="CONGRESS" type="GOVERNING BODY">Congress can tax the income as it
chooses</entity>
<entity name="UNINCORPORATED ASSOCIATIONS" type="NON-LEGAL PARTICIPANT">Unincorporated
associations prevail under the local law</entity>
<entity name="SHAREHOLDERS" type="NON-LEGAL PARTICIPANT">Shareholders are related to
unincorporated association under the local law</entity>
<relationship>
    <source name="UNINCORPORATED ASSOCIATIONS"></source>
    <target name="SHAREHOLDERS"></target>
    <description descriptor="related to">UNINCORPORATED ASSOCIATIONS can be related to
SHAREHOLDERS</description>
</relationship>
<relationship>
    <source name="COURT"></source>
    <target name="BURK-WAGGONER OIL ASSN. V. HOPKINS"></target>
    <description descriptor="heard">The COURT heard BURK-WAGGONER OIL ASSN. V.
HOPKINS</description>
</relationship>
<relationship>
    <source name="CONGRESS"></source>
    <target name="PARTNERSHIP"></target>
    <description descriptor="taxes">CONGRESS can tax the income as it chose for
PARTNERSHIP</description>
</relationship>
##Query##
- Case:
 {{{chunk}}}
- Output:
```

FIGURE G – Requête générative pour la phase d’indexation de notre approche

D Résultats

Stratégie	Résumé
Baseline	Stephen Duncan served as the warden of a federal prison located in a state that falls under the Seventh Circuit's jurisdiction. The dispute began when Lawrence Owens, an inmate at that facility, contested a decision made by the warden. Owens alleged that the warden's administrative action—whether it involved a disciplinary measure, a transfer, or another custodial decision—had violated his rights, and he filed a lawsuit against the warden to challenge that action. The district court adjudicated the matter and rendered a decision that the warden found unfavorable. Duncan appealed that ruling to the United States Court of Appeals for the Seventh Circuit, seeking review of the lower court's determination. After the Seventh Circuit considered the appeal, Duncan petitioned the Supreme Court for a writ of certiorari, asking the Court to review the appellate decision. The Supreme Court dismissed the petition as improvidently granted.
Naïf	Stephen Duncan, warden is the petitioner in a case against Lawrence Owens, who is the respondent. The United States Court of Appeals for the Seventh Circuit was the lower court that made the original decision in this case, and the Seventh Circuit Court of Appeals subsequently issued its decision regarding the dispute between the petitioner and the respondent. A writ of certiorari was then issued in the matter, but the Supreme Court of the United States dismissed that writ as improvidently granted.
CompPrinc	Stephen Duncan, warden is the petitioner in a case against Lawrence Owens, who is the respondent.
Syllabus	N/A
EntLiees	Stephen Duncan, a warden, is the petitioner in the case. He filed a writ of certiorari. In the same proceeding, he is the opponent against Lawrence Owens.
ReRank	Stephen Duncan, a warden, is the petitioner in the case. Lawrence Owens is the respondent in the case. Stephen Duncan, the opponent in the case against Lawrence Owens, filed a writ of certiorari. A writ of certiorari has been issued.

FIGURE H – Exemples de résumés des faits d'une affaire disponible via le lien suivant : https://supreme.justia.com/cases/federal/us/577/189/

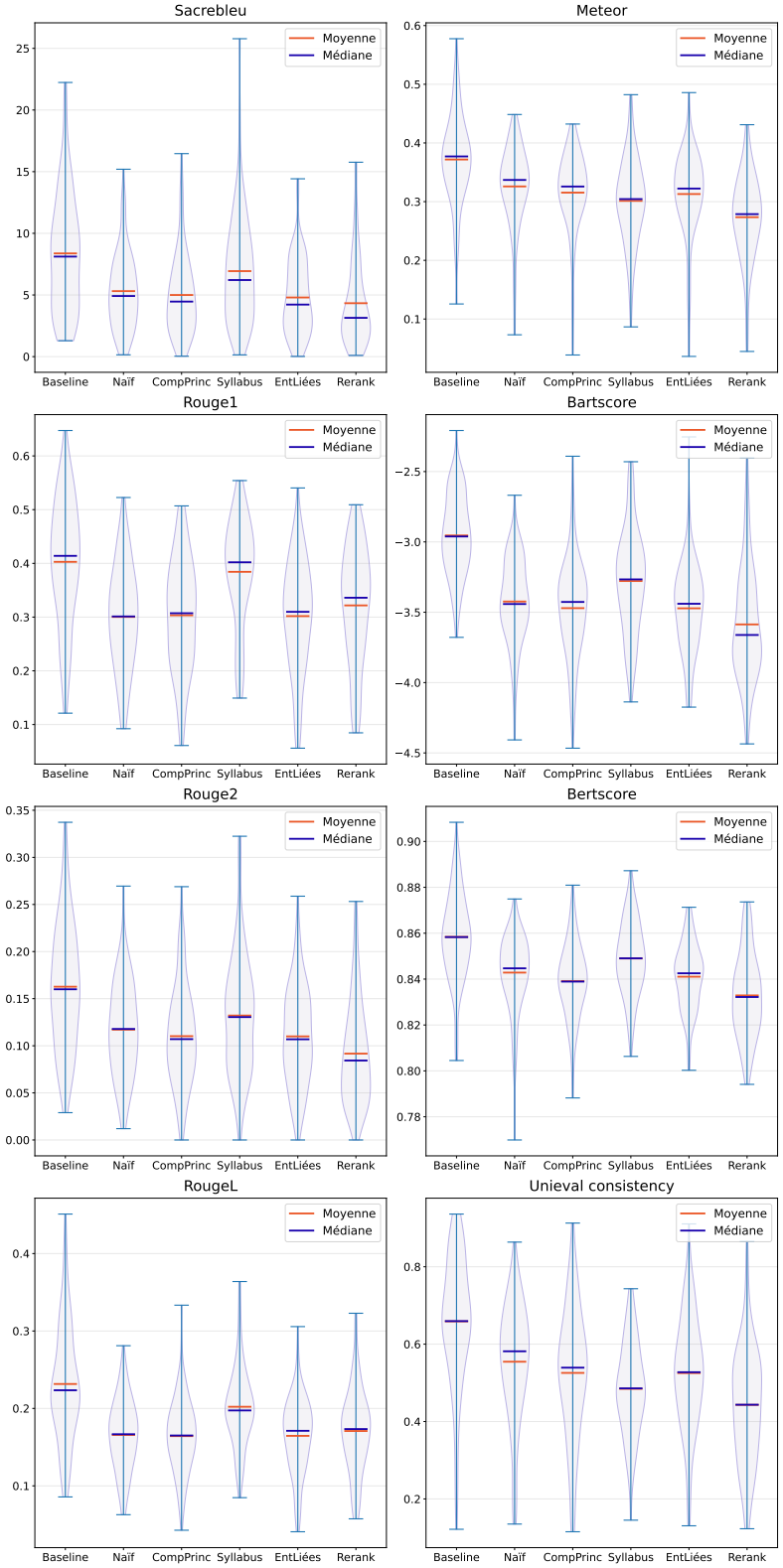

FIGURE I – Distribution, moyenne et médiane des scores pour les résumés obtenus avec chaque stratégie en utilisant les faits comme résumé cible.

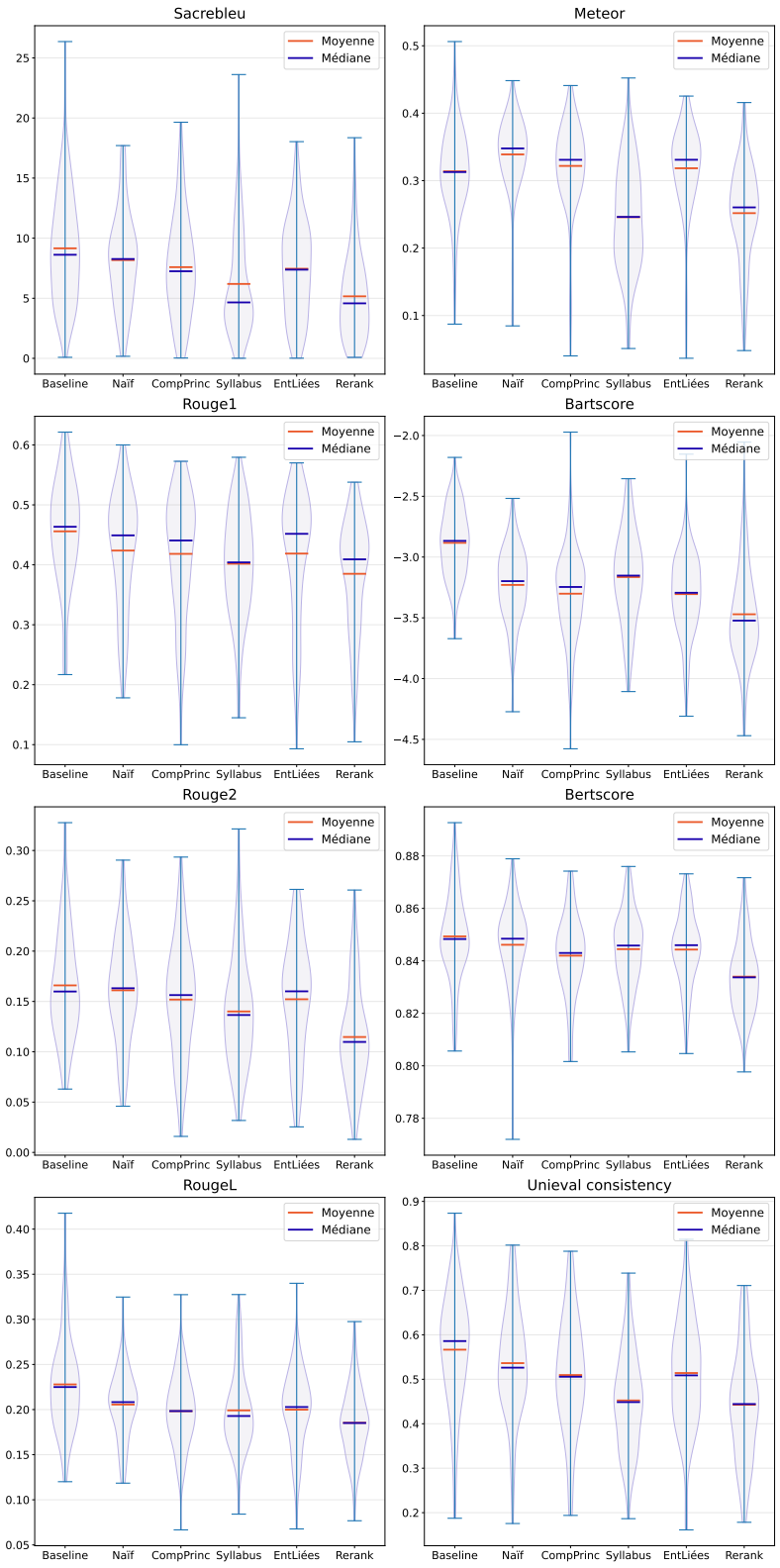

FIGURE J – Distribution, moyenne et médiane des scores pour les résumés obtenus avec chaque stratégie en utilisant les faits et la conclusion comme résumé cible.