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

Sem-G-RAG, combiner sémantique symbolique à base de graphes et LLM pour le RAG

Zhongjie LI[1, 2] Rim ABROUGUI[2] Guillaume LECHIEN[2] Benoît LAURENT[2]
Elisabeth SAVATIER[2] Patrick PAROUBEK[3]
(1) Inalco, 65 Rue des Grands Moulins, 75013 Paris, France
(2) Aday, 104 Bd du Montparnasse, 75014 Paris, France
(3) LISN, Campus Universitaire bât.507 - Rue du Belvédère, 91405 Orsay, France
lzj44010@gmail.com, innovations@aday.fr, pap@lisn.fr

RÉSUMÉ

Nous proposons une nouvelle méthodologie pour intégrer la sémantique explicite dans la phase de génération d'un système de génération augmentée par la recherche (RAG) à base de grand modèle de langue (LLM) afin d'améliorer la compréhension et la génération. Notre méthode repose sur les cadres sémantiques FrameNet et sur le format de graphe Penman pour les représenter. Nous analysons les apports de cette représentation sémantique explicite à base de graphe sur les données SQuAD (en anglais). Si l'ajout de ces informations n'améliore pas significativement les résultats dans nos expériences, notre analyse qualitative montre que cette sémantique explicite peut dans certains cas, améliorer la précision factuelle des réponses. Par ailleurs, si le remplacement du contexte textuel par le graphe FrameNet seul entraîne une dégradation importante de la qualité des réponses produites, nous n'avons pas observé de différence significative lorsque nous filtrons les graphes selon les entités présentes dans la questions, ce qui ouvre une perspective d'optimisation de notre méthodologie grace au filtrage sémantique du graphe.

ABSTRACT

Sem-G-RAG, combining graph-based semantic symbolism and LLM for RAG

We propose a new methodology to add an explicit semantic representation (based on Framenet and the Penman graph format) to the instruction given to an LLM during answer generation. We have analysed its benefits on the SQuad dataset. If adding these information did not significantly improve the performance in our experiments, our qualitative analysis shows that in some cases the factual precision of the answer can be better. We also observed that having only the semantics graph instead of the textual context induces a clear degradation of performance. On the other hand, we did not find a significative difference between having a complete Framenet graph and a filtered version according to the entities present in the associated text query, opening the way for optimizating of our methodology by semantic filtering of the graph.

MOTS-CLÉS : RAG, Sémantique, Graphe, FrameNet, Penman.

KEYWORDS: RAG, Semantic, Graph, FrameNet, Penman.

Actes des 33ème Conférence sur le Traitement Automatique des Langues Naturelles.
Volume 1 : articles scientifiques originaux, pages 544–563.
Cette œuvre est mise à disposition sous licence Attribution 4.0 International.

1 Introduction

Afin d'élargir les capacités des systèmes de génération augmentée par la recherche d'information (RAG), les graphes de connaissances ainsi que les représentations sémantiques ont été largement exploités dans des travaux récents. Les approches proposées dans (Han *et al.*, 2025) et (Guo *et al.*, 2025b) démontrent l'efficacité des graphes de connaissances pour structurer l'information et améliorer la pertinence de la récupération. Avec les avancées rapides en recherche d'information (IR) et en traitement automatique des langues (TAL), plusieurs techniques permettent d'accélérer et d'améliorer significativement le processus de recherche. Par exemple, l'utilisation d'encodeurs basés sur l'architecture Transformer améliore la pertinence des résultats en capturant la dimension sémantique et contextuelle des requêtes et des documents (Vaswani *et al.*, 2023), des méthodes d'indexation vectorielle comme FAISS permettent d'effectuer des recherches de similarité à grande échelle de manière efficace. Cependant, la majorité des travaux se concentre principalement sur l'amélioration du module de récupération, tandis que la composante génération demeure relativement moins explorée. En particulier, peu d'études s'intéressent explicitement à l'intégration de la dimension sémantique au sein du processus de génération lui-même. En conséquence, ce travail vise à explorer l'intégration explicite de la dimension sémantique dans le module de génération des architectures RAG, afin d'évaluer son impact sur la cohérence et la pertinence des réponses produites. Concrètement, nous proposons d'augmenter le contenu textuel du contexte récupéré lors de la phase de recherche d'information avec le graphe sémantique correspondant avant de passer à l'étape de génération. Nous présenterons successivement l'état de l'art, la méthodologie proposée, les expériences réalisées, puis l'analyse des résultats.

2 État de l'art

La sémantique contextuelle a démontré son efficacité dans de nombreux travaux portant sur le RAG. Dans SemRAG, les auteurs proposent de segmenter les textes selon la similarité cosinus entre les vecteurs de phrases, ce qui permet de regrouper davantage d'informations pertinentes au sein d'un même segment (Zhong *et al.*, 2025). D'autre part, le RAG multi-sémantique (MS-RAG) (You *et al.*, 2025) et le SPR-RAG (Wang *et al.*, 2025) exploitent la similarité cosinus en comparant les entités et les relations de la question avec celles issues de graphes de connaissances. Cette approche contribue à accélérer la recherche et améliore la pertinence du contenu récupéré.

Malgré ces avancées, ces approches reposent majoritairement sur la similarité vectorielle, qui capture principalement la proximité sémantique sans modéliser explicitement les relations conceptuelles entre les informations. Cette limitation affecte également les performances des LLMs. En effet, bien que ces grands modèles soient capables de comprendre efficacement la sémantique implicite grâce à ses paramètres pré-entraînés, ils n'ont pas accès au sens du texte d'entrée en termes de représentation sémantique explicite (Cuskley *et al.*, 2024). En outre, l'étape de l'IR réussit à fournir un contexte pertinent, mais la possibilité d'une erreur d'interprétation de ce contexte par le LLM subsiste.

En conséquence, nous souhaitons intégrer la dimension sémantique explicite dans le RAG, spécifiquement, à l'étape de génération. Pour cela, nous nous appuyons sur la théorie de « Frame semantics » (cadre sémantique). Cette dernière suggère que la meilleure façon de comprendre le sens des mots est de se référer aux « frames » (cadre) ou « schemas » (schéma) sémantiques et conceptuels dans

lesquels ils s'ancrent (Fillmore *et al.*, 2006). Partant de cette théorie, FrameNet [1] est un modèle de représentation sémantique qui décrit le sens des mots à travers des structures appelées cadres (Frame). Un cadre correspond à un type de situation, comme un événement, une relation ou un état, impliquant plusieurs participants jouant différents rôles. Ces participants sont appelés éléments du cadre (Frame Elements, FE). Le rôle de FrameNet est d'identifier et de définir ces cadres ainsi que leurs éléments, puis de décrire comment ces éléments s'organisent syntaxiquement dans une phrase autour de l'unité lexicale qui active le cadre.

Un travail récent a également exploré l'efficacité de FrameNet dans les systèmes RAG (Madabushi, 2024). Cette approche part du constat que les représentations vectorielles contextuelles capturent principalement la similarité sémantique entre textes, sans nécessairement modéliser l'ensemble des relations logiques entre les informations. Par conséquent, le contenu récupéré peut être sémantiquement proche de la requête sans être suffisant pour soutenir un raisonnement multi-faits. Contrairement à notre approche, ce travail se concentre sur le composant de recherche d'information. Les auteurs proposent d'annoter des faits atomiques avec les cadres sémantiques principaux (Frames) qu'ils expriment. Lors de la phase de recherche, le système identifie d'abord le cadre principal de la question ainsi que les cadres qui lui sont associés. Cette étape permet de déterminer les cadres correspondant aux faits susceptibles de répondre à la question. La recherche d'information est ensuite guidée par ces cadres, dans le but d'améliorer la pertinence et l'efficacité de la récupération des informations. Dans cet article, les résultats expérimentaux montrent que cette approche surpasse les méthodes de recherche basées sur les mots-clés, tout en restant significativement moins performante que les approches reposant uniquement sur les plongements lexicaux (embeddings). Néanmoins, ce travail met en évidence l'intérêt de la sémantique explicite fournie par FrameNet. Étant donné que les LLMs sont entraînés sur du texte en langage naturel, il pourrait être plus pertinent d'intégrer cette représentation sémantique explicite dans la phase de génération.

Par ailleurs, une autre étude s'est focalisée sur la phase de génération (Jin *et al.*, 2024). Plutôt que de s'appuyer sur FrameNet, cette approche exploite une autre forme de sémantique, l'AMR (représentation sémantique abstraite). Dans le but d'évaluer l'impact de cette représentation sur les performances des LLMs, les auteurs fournissent simultanément au modèle le texte d'entrée et sa représentation AMR associée. Cependant, les expériences n'ont pas produit de résultats prometteurs. L'intégration de graphes AMR n'améliore globalement pas les performances des LLMs et entraîne même une légère dégradation sur plusieurs tâches, notamment la détection de paraphrases, la traduction automatique, l'extraction d'événements et la détection de sophismes logiques. Ces résultats suggèrent que l'AMR n'apporte pas d'avantage clair aux LLMs modernes. Pourtant, cette limitation peut s'expliquer par certaines caractéristiques de l'AMR. En effet, cette représentation se concentre principalement sur le prédicat et modélise essentiellement les relations entre celui-ci et ses arguments. De plus, ces relations sont généralement désignées par des étiquettes abstraites (ARG0, ARG1, ARG2, etc.), qui indiquent l'existence d'un lien structurel sans en expliciter la nature sémantique précise (Banarescu *et al.*, 2013). Ces limitations pourraient expliquer pourquoi l'AMR n'apporte pas de bénéfice significatif aux LLMs. À l'inverse, FrameNet propose des concepts sémantiques couvrant non seulement le prédicat, mais également d'autres unités du texte telles que le sujet, l'objet ou les prépositions, permettant ainsi de capturer une information sémantique plus riche. Par ailleurs, les concepts et les relations y sont exprimés en utilisant un vocabulaire concret enrichi d'exemples en langage naturel, ce qui facilite leur exploitation par les LLMs. Dans ce travail, nous adoptons une approche similaire consistant à combiner le texte avec son graphe sémantique correspondant, en substituant l'AMR par FrameNet. Afin de représenter les informations FrameNet, ce travail utilisera le format Penman. Ce format

1. https://framenet.icsi.berkeley.edu/WhatIsFrameNet

représente les graphes sous forme de structures enracinées, dirigées et étiquetées, et est conçu pour modéliser des structures sémantiques complexes dans une représentation textuelle lisible par l'humain (Goodman, 2020). Le format Penman encode la hiérarchie et l'arborescence des concepts à l'aide de parenthèses imbriquées, permettant ainsi de représenter la structure de dépendance d'un graphe sémantique. Certes, ce format est principalement connu pour encoder des graphes d'AMR ou bien des graphes pour d'autres tâches de NLU[2], comme dans les travaux de (Kang *et al.*, 2025; Bevilacqua *et al.*, 2021; Bonial *et al.*, 2020; Abrougui *et al.*, 2023). Néanmoins, la bibliothèque penman (outil Python)[3] n'est pas strictement limitée à l'AMR, et elle est également compatible avec les annotations FrameNet, dès lors que celles-ci respectent la structure et la grammaire du graphe. Il apparaît ainsi pertinent pour FrameNet, dont l'organisation en cadres (Frames) présente des similarités structurelles avec celle de l'AMR. Dans le cadre des annotations FrameNet exprimées en Penman, les nœuds correspondent aux concepts ou aux entités, tandis que les arêtes représentent les relations sémantiques entre ces éléments.

Le format Penman constitue une représentation intermédiaire entre structure arborescente et graphe sémantique complet. Il combine la lisibilité d'une notation textuelle linéaire avec les propriétés d'un graphe orienté, ce qui le rend adapté à la visualisation et au traitement automatique en TAL.

3 Méthodologie

Pour étudier les apports des graphes FrameNet dans la phase de génération du RAG, nous proposons de concaténer la question, le contexte et son graphe FrameNet correspondant dans l'instruction envoyée au LLM. Dans ce qui suit, nous expliquons comment les informations sémantiques FrameNet sont obtenues pour le contexte récupéré et comment sont elles intégrées dans notre architecture RAG.

Une fois que le système de recherche d'information obtient le contexte textuel pertinent pour la question reçue, un analyseur automatique extraira les cadres sémantiques (Frame). Cette annotation est réalisée par Frame Semantic Transformer (FST), fondé sur une version affinée du modèle T5 (Text-to-Text Transfer Transformer). Contrairement aux approches basées sur des réseaux de neurones récurrents (RNN) comme Open-SESAME (Swayamdipta *et al.*, 2017), FST reformule l'ensemble du processus d'annotation sémantique de FrameNet sous la forme d'une tâche séquence-à-séquence (Chanin, 2023). Concrètement, l'analyse sémantique de FrameNet est décomposée en trois sous-tâches réalisées en séquence :

1. Identification des déclencheurs (Triggers) : localiser les mots ou expressions qui activent un cadre sémantique
2. Classification des cadres : attribuer un élément du cadre (FE) au déclencheur identifié
3. Extraction des arguments : détecter les éléments du cadre et leurs valeurs dans la phrase

En général, FrameSemanticTransformer-base (la version basique pré-entraînée de FST) est un analyseur automatique qui est capable d'annoter du texte naturel sans dépendance externe. Nous nous concentrerons sur ce modèle dans le cadre de ce travail pour réaliser l'annotation de FrameNet sur le contexte récupéré de RAG. L'annotation générée sera par la suite convertie en une liste de triplets RDF, puis en graphe de format Penman. Voici un exemple de la transformation complète de notre phrase exemple en anglais *« Nanjing est un important nœud ferroviaire de l'est de la Chine. »* :

2. compréhension du langage naturel
3. https://penman.readthedocs.io/en/latest/

1. Annotation FrameNet par FST : Nanjing is an $_{\textbf{Importance}}$ IMPORTANT $_{\textbf{Roadways}}$ RAILWAY $_{\textbf{Political_locales}}$ HUB in $_{\textbf{Part_orientational}}$ EASTERN China.

2. Liste de triplets RDF (détail dans l'annexe A) :

```
[[("important", "evokes", "Frame:Importance"),
  ("Frame:Importance", "hasRole:Factor", "railway")], ...]]
```

3. Graphe Penman final :

```
(Sent1 / Sentence
    :frame (Im1 / Importance
        :text "important"
        :Factor "railway hub")
    :frame (Ro1 / Roadways
        :text "railway"
        :Roadway "railway")
    :frame (Po1 / Political_locales
        :text "hub"
        :Locale "Nanjing"
        :Descriptor "railway"
        :Relative_location "in eastern China")
    :frame (Pa1 / Part_orientational
        :text "eastern"
        :Whole "China))
```

Contrairement à l'approche proposée par (Jin *et al.*, 2024), qui consiste à insérer le graphe sémantique complet du texte après celui-ci, nous juxtaposons le graphe associé à chaque phrase immédiatement après celle-ci. Nous appelons cette représentation « HybridGraph » (voir Figure 1). Le graphe Penman

```
Nanjing is an important railway hub in eastern China.
(Sent1 / Sentence
    :frame (Im1 / Importance
        :text "important"
        :Factor "railway hub")
    ...)
It serves as rail junction for the Beijing-Shanghai (Jinghu).
(Sent2 / Sentence
    :frame (As1 / Assistance
        :text "serves"
        :Helper "It"
        :Goal "as rail junction for the Beijing-Shanghai (Jinghu)")
    :frame (Ro2 / Roadways
        :text "rail"
        :Roadway "rail"))
```

FIGURE 1 – Illustration de la représentation HybridGraph pour les phrases exemple en anglais *« Nanjing est un important nœud ferroviaire de l'est de la Chine. Elle sert de jonction ferroviaire pour la ligne Pékin-Shanghai (Jinghu). »*

pourrait être trop volumineux pour être intégré tel quel au contexte si la phrase correspondante est trop longue. Nous constatons que le graphe Penman contient généralement deux fois plus de tokens

que le texte initial. Afin de permettre l'inclusion d'un plus grand volume de contexte, donc plus d'informations récupérées, et sans dépasser la limite de tokens du LLM (par ex. 32 768 tokens pour Mistral 8×7B quantisé), nous appliquons un filtrage des annotations FrameNet dans les représentations contextuelles utilisées pour la génération, décrit dans la section suivante 3.1.

3.1 Filtrage d'annotations sémantiques (EN Filtrage)

Ce filtrage dépend de la question posée. Une extraction d'entités nommées est d'abord réalisée sur la question à l'aide du modèle SpaCy **en_core_web_trf**[4] basé sur **roberta-base**, afin d'obtenir une analyse robuste. Ensuite, parmi les annotations FrameNet du contexte récupéré, seules sont conservés les Frames contenant au moins l'une des entités nommées détectées. Le graphe obtenu est beaucoup plus concis. Les Frames contenant les entités nommées trouvées dans les questions posées encapsulent généralement une forte densité d'informations sémantiques pertinentes, ce qui permet de conserver les éléments les plus informatifs tout en réduisant le bruit contextuel. En général, lorsque le système de recherche d'information identifie et récupère le contexte pertinent pour la requête, ce contexte est ensuite enrichi de graphes FrameNet. La question est alors concaténée avec ce contexte augmenté, puis l'ensemble est transmis au LLM afin de générer la réponse. Un exemple détaillé du filtrage est détaillé dans l'annexe (voir B.1).

3.2 Méthode d'évaluation

Pour évaluer la qualité de génération des LLMs, nous avons utilisé les mesures suivantes : ROUGE-1, MNLI, BertScore, Exact Inclusion et LLM score. ROUGE-1 calcule le recouvrement des unités lexicales individuelles (unigrammes) entre le texte généré et le texte de référence, sans tenir compte de l'ordre des mots ni de la syntaxe (Lin, 2004). La deuxième mesure utilisée est MNLI (Multi-Genre Natural Language Inference) (Williams *et al.*, 2018) avec spécifiquement le modèle **facebook/bart-large-mnli** (Lewis *et al.*, 2019). Fondée sur la tâche d'inférence de relations logiques entre deux textes (Natural Language Inference), elle est utilisée pour quantifier la performance des modèles de langage sur la compréhension de relations sémantiques complexes entre deux phrases. L'usage de **facebook/bart-large-mnli** permet d'appliquer la métrique non seulement à des phrases isolées, mais également à des segments textuels légèrement plus longs, contrairement aux premiers modèles MNLI basés sur des architectures RNN. Cette métrique est ici particulièrement pertinente car elle évalue la cohérence sémantique entre la réponse générée et la référence, et permet de tester la capacité du modèle à raisonner, paraphraser et inférer. La troisième métrique utilisée est BertScore (Zhang *et al.*, 2020), qui évalue la similarité sémantique à l'aide de plongements de type BERT entre le texte généré et la référence. La métrique Exact Inclusion est une adaptation de la métrique Exact Match proposée dans SQuAD (Rajpurkar *et al.*, 2016), qui vérifie si le texte généré correspond exactement à la référence. Dans ce travail, nous adoptons une variante plus tolérante consistant à vérifier si la référence est entièrement contenue dans la réponse générée, plutôt que d'exiger une correspondance exacte. Enfin, nous utilisons une métrique de type LLM-as-a-judge, que nous appelons LLM Score, inspirée de (Wang *et al.*, 2023). Nous instruisons un LLM (ici Mistral 24B[5]) pour évaluer la qualité factuelle du texte généré par rapport à la référence. L'instruction utilisée est fournie en annexe (voir C.1).

4. https://spacy.io/models/en#en_core_web_trf
5. https ://huggingface.co/dphn/Dolphin-Mistral-24B-Venice-Edition

4 Expérience

4.1 Corpus

Le corpus utilisé dans nos expériences est le SQuAD 1.1 (Rajpurkar *et al.*, 2016), un corpus de référence couramment utilisé pour l'évaluation des approches de type RAG, sur la tâche de question-réponse (Sawarkar *et al.*, 2025; Veturi *et al.*, 2024; Zhu, 2025; Zhang *et al.*, 2025; Finardi *et al.*, 2024; Guo *et al.*, 2025a).

Il s'agit d'un corpus de compréhension de lecture en anglais, conçu pour évaluer la capacité d'un système à lire un texte et répondre à des questions associées. Dans la version 1.1, le corpus contient plus de 100 000 paires question-réponse.

Dans le cadre de cette étude, nous sélectionnons aléatoirement 600 entrées issues du SQuAD 1.1, chacune comportant des champs tels que **question**, **contexte** et **réponses**. La réponse correspond toujours à un segment de texte extrait directement du paragraphe associé. Comme le SQuAD 1.1 couvre une large diversité thématique telle que la science, la culture, l'histoire, le divertissement, etc, un corpus de test ainsi formé permet de tester les systèmes de RAG dans divers domaines.

4.2 Protocole expérimental

Le RAG textuel mis en œuvre dans nos expériences est la version classique, composée d'une base de plongements lexicaux, d'un système de recherche d'information et d'un LLM. En effet, le système de recherche d'information est fondé sur la recherche vectorielle appliquée à la base de plongements lexicaux. Chaque question reçue est transformée en vecteur par un unique encodeur commun à tous les plongements de la base, dans le but de retrouver les textes pertinents les plus similaires à la question.

Pour construire cette base de plongements lexicaux, le contexte fourni dans le jeu de donnée SQuAD 1.1 est extrait et vectorisé par l'encodeur « E5-large-v2 », un encodeur bidirectionnel produisant des plongements lexicaux de haute qualité pour une large variété de tâches (Wang *et al.*, 2024). Aucun découpage du texte (chunking) n'a été effectué lors de la création de bases de données, car chaque contexte présente déjà une taille raisonnable (en moyenne 138 tokens par texte), suffisante pour être représentatif et en même temps compatible avec la capacité de l'encodeur. De plus, cela simplifie le traitement et évite la complexité liée à la segmentation des textes. Après chaque récupération, les trois textes les plus pertinents sont conservés comme contexte.

Afin de déterminer les différents impacts des composants de notre architecture HybridGraph, nous avons intégré différentes ressources de contexte dans nos expériences. Premièrement, nous supprimons complètement le contexte textuel et ne gardons que le graphe Penman. La deuxième ressource ne conserve qu'une version filtrée du graphe selon les entités nommées présentes dans la question. La troisième est HybridGraph Penman associant un contexte textuel et le graphe représentant sa sémantique. La quatrième ressource est identique à la précédente sauf que le graphe a été filtré en fonction des entités. Enfin, un contexte purement textuel est utilisé comme référence de comparaison par rapport à ces quatre types de ressources contextuelles.

Nous obtenons ainsi cinq types de prompts différents pour la génération :

1. Penman (voir Annexe D.1)
2. Penman EN filtrée (voir Figure 3.1)
3. HybridGraph Penman (voir Figure 1)
4. HybridGraph Penman EN filtrée (voir Figure 3.1)
5. Texte seul (ressource d'étalonnage)

Le format Penman permet d'évaluer les impacts du remplacement de contexte textuel par son graphe FrameNet. Le format HybridGraph Penman, dans lequel la phrase est concaténée à son graphe, permettra de déterminer si l'ajout d'un graphe FrameNet enrichit réellement le contexte textuel, et si remplacer entièrement le texte par un graphe entraîne une perte d'informations non représentées dans les graphes FrameNet. Enfin, les variantes avec filtrage consistent à étudier si une focalisation sur les entités nommées de la question pourrait donner une performance équivalente voire meilleure que celle sans filtrage.

Les différentes ressources de contexte sont concaténées avec la question avant d'être transmises au LLM pour générer la réponse. Dans les expériences, le LLM utilisé est toujours le Mistral 8×7B quantisé[6]. Nous avons calculé la significativité statistique (valeur $P < 0.05$) des résultats avec le test des rangs signés de Wilcoxon (Hae-Young, 2014). Les résultats non-significatifs sont affichés avec un fond gris dans le tableau 1.

Métriques %	Texte	Penman	Penman EN filtrées	HybridGraph Penman	HybridGraph Penman EN filtrée
ROUGE-1 Précision	17.3	14.7	9.8	**18.6**	16.9
ROUGE-1 Rappel	**86.0**	65.6	42.5	83.3	84.9
ROUGE-1 F1	26.3	21.8	14.5	**27.4**	25.7
MNLI Entailment	43.9	38.5	29.0	**47.3**	43.2
MNLI Contradiction	**6.8**	17.8	34.6	7.5	7.9
MNLI Neutral	49.3	43.8	36.5	45.3	49.0
BertScore Précision	82.9	82.8	82.4	**83.2**	82.8
BertScore Rappel	**87.6**	86.2	84.5	**87.6**	87.5
BertScore F1	85.2	84.4	83.4	**85.3**	85.1
Exact Inclusion	76.2	55.7	32.2	75.0	**76.7**
LLM Score	**84.9**	69.0	47.9	82.5	83.7

TABLE 1 – Avec 3 documents récupérés : comparaison Texte vs. **Graphe** et **HybridGraph**, les résultats significatifs sont indiqués avec un fond clair et les meilleures performances en gras.

5 Analyse de résultats

Nous avons mené des analyses quantitatives et qualitatives sur l'ensemble du corpus de test (600 lignes) pour étudier rigoureusement les résultats de génération. Selon le tableau 1, les résultats d'évaluation laissent à penser que dans notre système de RAG complet, le LLM avec un contexte uniquement composé de graphes FrameNet n'est pas capable de générer une réponse de qualité. De plus, le filtrage d'un tel graphe, pour ne retenir que les sous-graphes ayant les entités nommées présentes dans la requête, dégrade significativement les performances du LLM.

6. https://huggingface.co/casperhansen/mixtral-instruct-awq

Cependant, le contexte que nous appelons HybridGraph, associant le contexte textuel issu de la recherche d'information avec des graphes FrameNet complets ou dans leur version filtrée permet au LLM d'obtenir des performances globalement proches de celles obtenues avec un contexte purement textuel, voire supérieures selon certaines mesures. Malheureusement, cette légère amélioration de performances observée dans nos expériences n'est pas statistiquement significative. Néanmoins, d'après notre analyse qualitative, nous observons que HybridGraph contribue à améliorer la factualité des réponses générées par le LLM. Parmi les vingt-deux questions auxquelles le LLM a moins bien répondu avec le contexte textuel qu'avec HybridGraph, les types d'imprécision sont :

1. Omission : le modèle omet un élément important de la réponse attendue.
   Exemple :
   — Pour la référence « les anciennes lignes ferroviaires de Jinghu et Huning », l'adjectif « anciennes » est absent dans la réponse du LLM avec le contexte textuel.
2. Confusion sémantique : le modèle remplace une relation sémantique précise par une relation proche mais incorrecte.
   Exemples :
   — Pour la question « À quelle époque l'Antarctique était-il encore relié à l'Australie ? », le LLM confond la borne finale et l'intervalle temporel en répondant « pendant le Crétacé » au lieu de « à la fin du Crétacé ».
   — Le LLM confond l'auteur et le scénariste. Dans le contexte, « The BFG » est attribué à Roald Dahl, tandis que le scénario du film adapté est écrit par Melissa Mathison. Le modèle attribue le conte à Melissa Mathison au lieu de Roald Dahl.
   Extrait du contexte : « Spielberg a tourné et travaille actuellement à la post-production d'une adaptation du célèbre conte de Roald Dahl, Le Bon Gros Géant ... Le scénario du film a été écrit par Melissa Mathison... ».
3. Sur-généralisation : le modèle propose plusieurs réponses possibles.
   Exemple :
   — Le LLM répond « fin du XVIIIe siècle ou début du XIXe siècle » alors que la réponse exacte est « début du XIXe siècle ».
4. Confusion contextuelle : le modèle répond un élément proche dans le contexte mais incorrect.
   Exemple :
   — Dans la phrase mentionnant « recommandations de Bâle II », le LLM choisit « législateurs européens », qui apparaît à proximité mais n'est pas la bonne réponse.
   Extrait du contexte : « ...Les législateurs européens ont poussé de manière quelque peu dogmatique à l'adoption des recommandations de Bâle II... ».

En revanche, l'insertion de graphes FrameNet entre les phrases semble aider à accentuer les informations essentielles, telles que les rôles des personnes (par exemple, l'inventeur). Nous avons également identifié un cas dans lequel le LLM n'a pas su distinguer le scénariste de l'auteur de l'œuvre originale. Par ailleurs, avec un contexte purement textuel, le LLM a tendance à s'appuyer sur ses propres connaissances lorsque le contexte ne fournit pas d'informations pertinentes, malgré une consigne explicite dans l'instruction lui demandant de ne pas le faire. En outre avec les HybridGraphs, la différence de performance entre la version filtrée et la version non filtrée n'est pas statistiquement significative selon trois des cinq mesures. Cela montre ainsi l'apport du filtrage des entités en termes de diminution de la taille du contexte nécessaire à la génération lorsque l'on associe un contexte textuel à une représentation sémantique explicite. Notre analyse qualitative montre également que le LLM commet des erreurs sur des questions en grande partie similaires avec ou sans filtrage d'entité.

Notre évaluation humaine relève également plusieurs observations qui pourraient expliquer pourquoi

HybridGraph n'est pas nettement supérieur à l'utilisation du seul contexte textuel. Bien que les graphes sémantiques issus de FrameNet encodent de nombreuses informations avec les Frames et les Frame Elements, ils omettent certaines informations présentes dans la phrase originale. Par exemple, la concession « But » (mais) et la négation « didn't » (n'a pas) n'apparaissent ni explicitement ni implicitement dans les graphes. Cette absence résulte des limites du modèle FrameSemanticTransformer-base, le corpus d'entraînement comprend peu de données avec une négation. D'une manière générale, certains types d'informations linguistiques ne sont que partiellement représentés dans FrameNet. La modalité (probabilité, obligation, possibilité, etc.) n'est annotée que lorsqu'elle est lexicalisée par un mot déclenchant un Frame modal. Le modèle automatique d'annotation (ici basé sur T5) est donc limité par son corpus d'entraînement. Par ailleurs, FrameNet n'encode pas les auxiliaires grammaticaux tels que « will » (aller, futur proche) ou « have » (avoir, passé composé) lorsqu'ils servent uniquement à marquer le temps, l'aspect ou d'autres fonctions syntaxiques indépendantes de sens lexical autonome. Dans notre corpus de test, 239 contextes sur 600 présentent une forme de modalité. Parmi ceux-ci, seules 58 instances sont effectivement annotées par le modèle automatique. L'analyse des formes de surface montre que les auxiliaires modaux sont la catégorie la plus fréquente, représentant 33,44% des occurrences de modalité détectées au moyen d'une expression régulière.

Ainsi, représenter un texte uniquement par son graphe sémantique FrameNet conduit probablement à ignorer des informations essentielles pour une interprétation complète et dégrade également la compréhension linéaire du texte original. L'ajout d'un filtrage basé sur les entités nommées dans le graphe accentue encore cette perte d'information. Pour ces raisons, il nous paraît pertinent d'explorer la combinaison du texte et de son graphe FrameNet dans le cadre du RAG, ce qui correspond bien à nos expériences HybridGraph.

En nous concentrant sur les questions auxquelles le LLM a moins bien répondu avec un contexte d'HybridGraph, nous observons également plusieurs erreurs typiques :

1. Sur-utilisation du graphe : le LLM tend à privilégier les informations issues du graphe, même lorsque la réponse est explicitement présente dans les phrases du contexte.
   Exemples :
   — Pour une question portant sur l'année d'implication de Tito dans les processus de paix israélo-arabes, le modèle conclut à une absence d'information dans le graphe, alors que la réponse (1967) est présente dans le texte
   — De même, pour la question « Combien d'équipes sportives universitaires masculines l'université Northwestern aligne-t-elle ? », le modèle échoue à exploiter les phrases du contexte et se limite au graphe.
2. Rupture de continuité dans le contexte : l'insertion de graphes au sein du texte peut perturber la lecture séquentielle et nuire à la capacité du modèle à relier des informations complémentaires.
   Exemple :
   — Dans l'extrait du contexte : « La British Architectural Library, parfois appelée bibliothèque du **RIBA**, a été fondée en 1834 ... Avec désormais **plus de quatre millions d'ouvrages**, c'est l'une des trois plus grandes bibliothèques d'architecture. », les deux informations clés (en gras) apparaissent dans des phrases distinctes et sont séparées par le graphe inséré. Cette dissociation pourrait être la raison pour laquelle le LLM ne parvient pas à les combiner pour répondre correctement à la question portant sur le nombre d'ouvrages du RIBA.

Enfin, nous avons réalisé des expériences complémentaires pour nous assurer de la robustesse de nos résultats expérimentaux. Il est connu que les LLMs sont parfois sensibles aux petites variations de forme des requêtes qu'ils reçoivent (Pecher *et al.*, 2026). Étant donné une éventuelle dépendance

au format de représentation sémantique des informations de contexte utilisées pour la génération, nous avons d'abord conduit les mêmes expériences, mais en représentant systématiquement les informations Penman en listes de triplets RDF plutôt qu'au format Penman. Les résultats obtenus ne présentent pas de différence significative par rapport à ceux obtenus avec Penman. Puis dans une seconde expérience, une variante de HybridGraph a été évaluée en concaténant chaque phrase son graphe associé représenté sous forme d'une structure de clé-valeur de type JSON (la phrase servant de clé et la liste de triplets correspondante servant de valeur). Là aussi les performances du LLM demeurent quasiment équivalentes à celles observées avec notre format HybridGraph Penman.

6 Limitation

Bien que SQuAD 1.1 soit un jeu de données largement utilisé pour l'évaluation des systèmes de RAG, il date de 2016. Le LLM utilisé dans nos expériences, Mistral 8×7B quantisé, est publié en 2023. Il y a donc un risque de contamination des données de notre étude. Par ailleurs, les questions dans SQuAD 1.1 ne sont peut être pas complètement adaptées pour évaluer notre méthodologie car elles sont uniquement de type « extraction », ce qui demande peu de raisonnement et une compréhension assez superficielle. De plus nous avons utilisé pour nos mesures les données d'entraînement et non les données de test, ce qui peut introduire un biais.

7 Conclusion

Ce travail propose une approche appelée HybridGraph, qui consiste à enrichir les informations de contexte issues de la phase de recherche d'information avant leur intégration dans la phase de génération du système RAG-LLM. Dans cette approche, nous associons un contenu textuel avec son graphe sémantique associé à base de cadres FrameNet. Ce format nous permet d'apprécier l'apport de l'explicitation de la sémantique du contexte textuel lorsque celle-ci est fournie au LLM en charge de la génération de la réponse. Bien que les performances du LLM avec un tel contexte ne soient pas statistiquement différentes dans nos expériences de celles obtenues avec un contexte purement textuel, notre analyse qualitative montre que cette sémantique explicite peut dans certains cas, améliorer la précision factuelle des réponses générées. Des expériences complémentaires sont nécessaires pour confirmer ou infirmer cet apport de manière générale. En revanche, le remplacement complet du contexte textuel par le graphe FrameNet entraîne une dégradation importante de la qualité des réponses produites par les LLMs. Par contre, dans le cas des expériences HybridGraph associant un contexte textuel et un graphe FrameNet filtré selon les entités présentes dans la question, les performances de la version filtrée ne montrent pas de différence statistiquement significative par rapport à une approche sans filtrage, prouvant ainsi l'apport du filtrage des entités dans l'approche HybridGraph pour ce qui concerne la diminution de la taille de la représentation sémantique explicite.

Afin d'approfondir l'étude des impacts de HybridGraph dans le cadre du RAG, il serait pertinent d'évaluer cette approche sur des corpus comportant des questions de types variés, notamment des tâches de raisonnement multi-sauts ou de résumé contextualisé. Un autre axe de recherche prometteur consiste à affiner l'annotateur FrameNet, FST, afin d'obtenir des annotations plus précises et plus cohérentes. Enfin, il serait également envisageable d'introduire le chaînage d'entités et la résolution de coréférence dans les graphes, dans l'objectif de produire des graphes plus compacts, plus cohérents et moins redondants.

Références

ABROUGUI R., DAMNATI G., HEINECKE J. & BÉCHET F. (2023). Abstract representation for multi-intent spoken language understanding. In *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, p. 1–5. DOI : 10.1109/ICASSP49357.2023.10095062.

BANARESCU L., BONIAL C., CAI S., GEORGESCU M., GRIFFITT K., HERMJAKOB U., KNIGHT K., KOEHN P., PALMER M. & SCHNEIDER N. (2013). Abstract meaning representation for sembanking. In *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*, p. 178–186 : Association for Computational Linguistics.

BEVILACQUA M., BLLOSHMI R. & NAVIGLI R. (2021). One spring to rule them both : Symmetric amr semantic parsing and generation without a complex pipeline. *Proceedings of the AAAI Conference on Artificial Intelligence*, **35**(14), 12564–12573. DOI : 10.1609/aaai.v35i14.17489.

BONIAL C., DONATELLI L., ABRAMS M., LUKIN S. M., TRATZ S., MARGE M., ARTSTEIN R., TRAUM D. & VOSS C. (2020). Dialogue-AMR : Abstract Meaning Representation for dialogue. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, p. 684–695, Marseille, France : European Language Resources Association.

CHANIN D. (2023). Open-source frame semantic parsing. *arXiv preprint arXiv :2303.12788*.

CUSKLEY C., WOODS R. & FLAHERTY M. (2024). The limitations of large language models for understanding human language and cognition. *Open Mind*, **8**, 1058–1083. DOI : 10.1162/opmi_a_00160.

FILLMORE C. J. *et al.* (2006). Frame semantics. *Cognitive linguistics : Basic readings*, **34**, 373–400.

FINARDI P., AVILA L., CASTALDONI R., GENGO P., LARCHER C., PIAU M., COSTA P. & CARIDÁ V. (2024). The chronicles of rag : The retriever, the chunk and the generator.

GOODMAN M. W. (2020). Penman : An open-source library and tool for AMR graphs. In A. CELIKYILMAZ & T.-H. WEN, Éds., *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics : System Demonstrations*, p. 312–319, Online : Association for Computational Linguistics. DOI : 10.18653/v1/2020.acl-demos.35.

GUO S., ZHANG S. & REN Z. (2025a). Enhancing rag efficiency with adaptive context compression.

GUO Z., XIA L., YU Y., AO T. & HUANG C. (2025b). Lightrag : Simple and fast retrieval-augmented generation.

HAE-YOUNG K. (2014). Statistical notes for clinical researchers : Nonparametric statistical methods : 1. nonparametric methods for comparing two groups. *Restor Dent Endod*, **39**(3), 235–239. DOI : 10.5395/rde.2014.39.3.235.

HAN H., WANG Y., SHOMER H., GUO K., DING J., LEI Y., HALAPPANAVAR M., ROSSI R. A., MUKHERJEE S., TANG X., HE Q., HUA Z., LONG B., ZHAO T., SHAH N., JAVARI A., XIA Y. & TANG J. (2025). Retrieval-augmented generation with graphs (graphrag).

JIN Z., CHEN Y., GONZALEZ F., LIU J., ZHANG J., MICHAEL J., SCHÖLKOPF B. & DIAB M. (2024). Analyzing the role of semantic representations in the era of large language models.

KANG J., COAVOUX M., LOPEZ C. & SCHWAB D. (2025). Reassessing graph linearization for sequence-to-sequence amr parsing : On the advantages and limitations of triple-based encoding.

LEWIS M., LIU Y., GOYAL N., GHAZVININEJAD M., MOHAMED A., LEVY O., STOYANOV V. & ZETTLEMOYER L. (2019). Bart : Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension.

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.

MADABUSHI H. T. (2024). Fs-rag : A frame semantics based approach for improved factual accuracy in large language models.

PECHER B., SPIEGEL M., BELANEC R. & CEGIN J. (2026). Revisiting prompt sensitivity in large language models for text classification : The role of prompt underspecification.

RAJPURKAR P., ZHANG J., LOPYREV K. & LIANG P. (2016). SQuAD : 100,000+ questions for machine comprehension of text. In J. SU, K. DUH & X. CARRERAS, Éds., *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, p. 2383–2392, Austin, Texas : Association for Computational Linguistics. DOI : 10.18653/v1/D16-1264.

SAWARKAR K., SOLANKI S. R. & MANGAL A. (2025). Metagen blended rag : Unlocking zero-shot precision for specialized domain question-answering.

SWAYAMDIPTA S., THOMSON S., DYER C. & SMITH N. A. (2017). Frame-Semantic Parsing with Softmax-Margin Segmental RNNs and a Syntactic Scaffold. *arXiv preprint arXiv :1706.09528*.

VASWANI A., SHAZEER N., PARMAR N., USZKOREIT J., JONES L., GOMEZ A. N., KAISER L. & POLOSUKHIN I. (2023). Attention is all you need.

VETURI S., VAICHAL S., JAGADHEESH R. L., TRIPTO N. I. & YAN N. (2024). Rag based question-answering for contextual response prediction system.

WANG J., LIANG Y., MENG F., SUN Z., SHI H., LI Z., XU J., QU J. & ZHOU J. (2023). Is chatgpt a good nlg evaluator ? a preliminary study.

WANG L., YANG N., HUANG X., JIAO B., YANG L., JIANG D., MAJUMDER R. & WEI F. (2024). Text embeddings by weakly-supervised contrastive pre-training.

WANG Y., XU T. & ZHU Y. (2025). Spr-rag : Semantic parsing retriever-enhanced question answering for power policy. *Algorithms*, **18**(12). DOI : 10.3390/a18120802.

WILLIAMS A., NANGIA N. & BOWMAN S. (2018). A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies, Volume 1 (Long Papers)*, p. 1112–1122, New Orleans, Louisiana : Association for Computational Linguistics. DOI : 10.18653/v1/N18-1101.

YOU X., LUO Y. & GU L. (2025). MS-RAG : Simple and effective multi-semantic retrieval-augmented generation. In C. CHRISTODOULOPOULOS, T. CHAKRABORTY, C. ROSE & V. PENG, Éds., *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, p. 22609–22625, Suzhou, China : Association for Computational Linguistics. DOI : 10.18653/v1/2025.emnlp-main.1151.

ZHANG N., ZHANG C., TAN Z., YANG X., DENG W. & WANG W. (2025). Credible plan-driven rag method for multi-hop question answering.

ZHANG T., KISHORE V., WU F., WEINBERGER K. Q. & ARTZI Y. (2020). Bertscore : Evaluating text generation with bert.

ZHONG K., SULEIMAN B., ERRADI A. & CHEN S. (2025). Semrag : Semantic knowledge-augmented rag for improved question-answering.

ZHU Y. (2025). From static to dynamic : A streaming rag approach to real-time knowledge base.

A Liste de triplets RDF

```
[[("important", "evokes", "Frame:Importance"),
  ("Frame:Importance", "hasRole:Factor", "railway")],
  [("railway", "evokes", "Frame:Roadways"),
  ("Frame:Roadways", "hasRole:Roadway", "railway")],
  [("hub", "evokes", "Frame:Political_locales"),
  ("Frame:Political_locales", "hasRole:Locale", "Nanjing"),
  ("Frame:Political_locales", "hasRole:Descriptor", "railway"),
  ("Frame:Political_locales", "hasRole:Relative_location",
  "in eastern China")],
  [("eastern", "evokes", "Frame:Part_orientational"),
  ("Frame:Part_orientational", "hasRole:Whole", "China")]]
```

B Filtrage d'annotations sémantiques (EN Filtrage)

B.1 Exemple du filtrage de l'HybridGraph

B.1.1 Contexte

Nanjing is an important railway hub in eastern China. It serves as rail junction for the Beijing-Shanghai (Jinghu) (which is itself composed of the old Jinpu and Huning Railways), Nanjing-Tongling Railway (Ningtong), Nanjing-Qidong (Ningqi), and the Nanjing-Xian (Ningxi) which encompasses the Hefei-Nanjing Railway. Nanjing is connected to the national high-speed railway network by Beijing-Shanghai High-Speed Railway and Shanghai-Wuhan-Chengdu Passenger Dedicated Line, with several more high-speed rail lines under construction.

B.1.2 Avant le filtrage

```
Nanjing is an important railway hub in eastern China.
(Sent1 / Sentence
    :frame (Im1 / Importance
        :text "important"
        :Factor "railway hub")
    :frame (Ro1 / Roadways
        :text "railway"
        :Roadway "railway")
    :frame (Po1 / Political_locales
        :text "hub"
        :Locale "Nanjing"
        :Descriptor "railway"
        :Relative_location "in eastern China")
    :frame (Pa1 / Part_orientational
        :text "eastern"
        :Whole "China"))
It serves as rail junction for the Beijing-Shanghai (Jinghu) (which
    is itself composed of the old Jinpu and Huning Railways),
    Nanjing-Tongling Railway (Ningtong), Nanjing-Qidong (Ningqi), and
    the Nanjing-Xian (Ningxi) which encompasses the Hefei-Nanjing
    Railway.
(Sent2 / Sentence
    :frame (As1 / Assistance
        :text "serves"
        :Helper "It"
        :Goal "as rail junction for the Beijing-Shanghai (Jinghu)
            (which is itself composed of the old Jinpu and Huning
            Railways ), Nanjing-Tongling Railway (Ningtong ),
            Nanjing-Qidong (Ningqi ), and the Nanjing-Xian (Ningxi )
            which encompasses the Hefei-Nanjing Railway")
    :frame (Ro1 / Roadways
        :text "rail"
        :Roadway "rail")
    :frame (Ag1 / Age
        :text "old"
        :Entity "Jinpu and Huning Railways")
    :frame (Ro2 / Roadways
        :text "Railways"
        :Name "Jinpu and Huning"
        :Roadway "Railways")
    :frame (Ro3 / Roadways
        :text "Railway"
        :Name "Nanjing-Tongling"
        :Roadway "Railway")
    :frame (Ro4 / Roadways
        :text "Railway"
        :Name "Hefei-Nanjing"
        :Roadway "Railway"))
Nanjing is connected to the national high-speed railway network by
```

```
    Beijing-Shanghai High-Speed Railway and Shanghai-Wuhan-Chengdu
    Passenger Dedicated Line, with several more high-speed rail lines
    under construction.
(Sent3 / Sentence
    :frame (At1 / Attaching
        :text "connected"
        :Item "Nanjing"
        :Goal "to the national high-speed railway network"
        :Connector "by Beijing-Shanghai High-Speed Railway and
            Shanghai-Wuhan-Chengdu Passenger Dedicated Line")
    :frame (Po1 / Political_locales
        :text "national"
        :Locale "national")
    :frame (Ro1 / Roadways
        :text "railway"
        :Roadway "railway")
    :frame (Ne1 / Network
        :text "network"
        :Possessor "national"
        :Nodes "high-speed railway"
        :Network "network")
    :frame (Ro2 / Roadways
        :text "Railway"
        :Name "Beijing-Shanghai High-Speed"
        :Roadway "Railway")
    :frame (Ro3 / Roadways
        :text "Line"
        :Characteristic "Dedicated"
        :Roadway "Line")
    :frame (Ro4 / Roadways
        :text "rail"
        :Roadway "rail")
    :frame (Bu1 / Building
        :text "construction"
        :Created_entity "several more high-speed rail lines"))
```

Listing 1 – Exemple de HybridGraph avant le filtrage EN

B.1.3 Après le filtrage

```
Nanjing is an important railway hub in eastern China.
It serves as rail junction for the Beijing-Shanghai (Jinghu) (which
    is itself composed of the old Jinpu and Huning Railways),
    Nanjing-Tongling Railway (Ningtong), Nanjing-Qidong (Ningqi), and
    the Nanjing-Xian (Ningxi) which encompasses the Hefei-Nanjing
    Railway.
(S2 / Sent
    :frame (As1 / "Assistance"
        :text "serves"
        :Helper "It"
        :Goal "as rail junction for the Beijing-Shanghai (Jinghu)
            (which is itself composed of the old Jinpu and Huning
            Railways ), Nanjing-Tongling Railway (Ningtong ),
            Nanjing-Qidong (Ningqi ), and the Nanjing-Xian (Ningxi )
            which encompasses the Hefei-Nanjing Railway"))
Nanjing is connected to the national high-speed railway network by
    Beijing-Shanghai High-Speed Railway and Shanghai-Wuhan-Chengdu
    Passenger Dedicated Line, with several more high-speed rail lines
    under construction.
(S3 / Sent
    :frame (At1 / "Attaching"
        :text "connected"
        :Item "Nanjing"
        :Goal "to the national high-speed railway network"
        :Connector "by Beijing-Shanghai High-Speed Railway and
            Shanghai-Wuhan-Chengdu Passenger Dedicated Line")
    :frame (Ro1 / "Roadways"
        :text "Railway"
        :Name "Beijing-Shanghai High-Speed"
        :Roadway "Railway"))
```

Listing 2 – Exemple de HybridGraph après le filtrage EN

C Instruction (Prompts)

C.1 Métrique LLM Score

```
[INST]You are an expert evaluator. Your task is to judge how well the
    Actual Answer responds to the Question, using the Reference Answer as
    the ground truth.

Guidelines:
- Return only a single integer score from 0 to 10.
- 0 = completely incorrect (does not answer the question or contradicts
    the reference).
- 10 = fully correct, answers the question, and matches the reference
    meaning.
- Intermediate scores = partially correct, incomplete, or somewhat
    aligned with the reference.
- The Actual Answer may be a full sentence, while the Reference Answer
    may be a phrase or segment of the correct information.
- Penalize if the Actual Answer introduces irrelevant or incorrect
    information.

Question: {entry['question']}

Actual Answer: {entry['generated_answer']}

Reference Answer: {entry['answer']}

Score (0-10):[/INST]
```

D Exemples de ressources de contexte

D.1 Penman

(Root_1 / Root) représente la racine du graphe. Celle-ci regroupe l'ensemble des informations FrameNet extraites des phrases du texte annoté. La racine est reliée à autant de nœuds *Sentence* qu'il y a de phrases dans le texte. Chaque nœud de phrase est instancié sous la forme *Sent#*, où le numéro correspond à son indice dans le texte (à partir de 1). Chaque nœud de phrase est ensuite relié à l'ensemble des Frames détectés, par une relation portant le nom du Frame. Chaque Frame est lui-même connecté à ses Frame Elements. La relation *:text* introduit la valeur des Frames, et la valeur des Frame Elements est directement annotée juste après. Voici un exemple de graphe Penman :

```
(R1 / Root
    :Sentence (Sent1 / Sentence
        :frame (Po1 / Possession
            :text "have"
            :Owner "I"
            :Possession "an apple")))
```

FIGURE 2 – Exemple de graphe Penman

D.2 HybridGraph Penman

Ici, chaque phrase est suivie directement par son graphe Penman. Il faut noter que le graphe de chaque phrase commence directement par le nœud *Sentence*. La variable des nœuds *Sentence* est toujours nommé comme *Sent#* mais il n'y a plus de nœud de racine *(Root_1 / Root)* comme dans le graphe Penman.

```
I have an apple.
(Sent1 / Sentence
    :frame (Po1 / Possession
        :text "have"
        :Owner "I"
        :Possession "an apple"))
```

FIGURE 3 – Exemple de HybridGraph Penman