# Étude de l'utilisation des modèles de langages pour l'interrogation en langue naturelle des graphes de connaissances

Alexandra Épiphanie Padonou, Peggy Cellier, Sébastien Ferré

Univ Rennes, INSA, CNRS, Inria, IRISA - UMR 6074
Campus de Beaulieu, 35042 Rennes cedex, France
alexandra-epiphanie.padonou@etudiant.univ-rennes.fr - ferre@irisa.fr - cellier@irisa.fr

**Résumé.** Dans cet article, nous présentons les résultats d'une étude approfondie des performances des grands modèles de langage (LLM) dans le contexte de l'interrogation des graphes de connaissances en langue naturelle (KGQA). La méthodologie de l'expérimentation a été structurée en deux approches distinctes : la génération de requêtes SPARQL et l'interrogation directe. Les résultats sur le benchmark QALD-10 ont révélé des performances très faibles dans la première approche et des performances correctes dans la deuxième, avec des variations importantes selon le type des questions-réponses.

## 1 Introduction

L'émergence des grands modèles de langages (LLM) (Vaswani et al., 2017), tels que ChatGPT, a considérablement transformé le domaine du traitement automatique de la langue, ouvrant de nouvelles perspectives pour toutes sortes de tâches. La tâche à laquelle nous nous intéressons dans cette étude est l'interrogation en langue naturelle des graphes de connaissances (ex., Wikidata), appelée en anglais *Question-Answering over Knowledge Graphs* (KGQA) (Diefenbach et al., 2017). Cette tâche est cruciale pour l'accès à ces sources de connaissances et particulièrement difficile car elle nécessite de faire le pont entre la langue naturelle et les langages formels du Web sémantique : RDF, et SPARQL. Le benchmark QALD (*Question Answering over Linked Data*) (Usbeck et al., 2023) est couramment utilisé pour comparer différentes approches. Dans sa 10e et dernière édition (2022), la meilleure approche a obtenu un score F1 de 45.4%. Depuis l'arrivée des LLM et de ChatGPT, peu de travaux ont à ce jour étudié les performances des LLM pour la tâche KGQA (Klager et Polleres, 2023; Faria et al., 2023). Ces travaux utilisent entre autres les questions de QALD et ils comparent deux formes d'utilisation d'un LLM : (a) par génération d'une requête SPARQL puis évaluation de cette requête sur le graphe de connaissance et (b) par interrogation directe du LLM, sans passer par le graphe de connaissances.

Comparé à ces travaux, notre étude porte sur les 393 questions test de la dernière version de QALD, plutôt que sur d'anciennes versions et/ou de petits échantillons de questions. Pour cela, nous automatisons la comparaison des réponses obtenues et des réponses attendues en

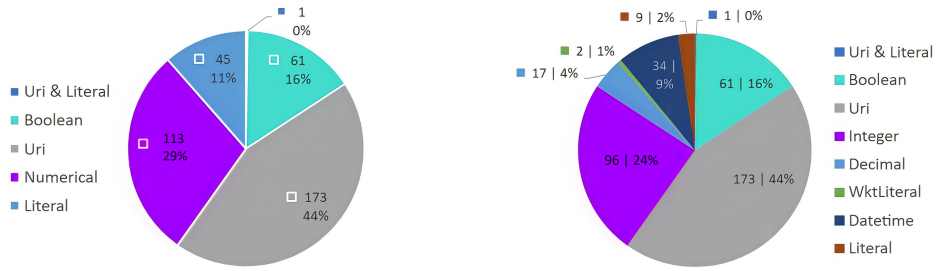


FIG. 1 – *Répartition des questions QALD-10 par types et sous-types de réponses*

TAB. 1 – *Quatre questions de types différents et leurs réponses*

| Type | Question | Réponse attendue |
|---|---|---|
| Booléen | Is Python a kind of Programming languages ? | `true` |
| URI | At which school was Yayoi Kusama educated at ? | `Q749884` |
| Nombre | How many ancient civilizations are there ? | `26` |
| Autre litéral | When was Penicillin discovered ? | `1928-09-28` |

prenant en compte les ambiguités (ex., Titanic comme bateau ou film) et les imprécisions. Nous effectuons également une analyse par type de réponse (booléen, URI, nombre ou autre littéral). Enfin, en plus du modèle GPT-3.5, nous considérons également un LLM léger, GPT4All, pour ses qualités éthiques (protection des données, frugalité). [1]

# 2 Méthodologie

Cette section expose en détail la méthodologie employée dans notre étude. Nous présentons tout d'abord le jeu de questions-réponses utilisé comme benchmark, ainsi que les modèles de langage retenus pour l'étude. Puis nous décrivons les deux approches que nous avons étudiées : la génération de requêtes SPARQL et l'interrogation directe. Pour chaque approche, nous discutons les avantages et inconvénients et nous donnons les prompts employés.

## 2.1 Jeu de questions-réponses

Pour cette étude nous avons utilisé le jeu de questions-réponses QALD (*Question Answering over Linked Data*) (Usbeck et al., 2023) comme benchmark. QALD est une série d'évaluations annuelles organisées dans le domaine du Web sémantique, qui vise à comparer les systèmes d'interrogation en langue naturelle des graphes de connaissances. Nous avons choisi de travailler avec la campagne d'évaluation la plus récente : QALD-10 (2022). Contrairement aux campagnes d'évaluation précédentes qui s'appuyaient sur DBpedia, QALD-10 porte sur Wikidata, un graphe de connaissances plus complet et mieux actualisé mais aussi plus complexe. QALD-10 contient 393 questions. Chaque question QALD comprend un identifiant unique, un booléen qui indique si la question implique une opération d'agrégation, la question elle-même

1. Les réponses de ChatGPT sont accessibles à `https://www.irisa.fr/LIS/ferre/pub/egc2024/`.

en quatre langues dont l'anglais, la ou les réponse(s) attendue(s) à la question et la requête SPARQL permettant d'obtenir les réponses depuis Wikidata. Il existe trois types de réponses, et donc de questions : *booléen* pour des réponses vrai/faux, *URI* pour des entités et *littéral* pour des littéraux au sens de RDF. Parmi les littéraux, nous distinguons les *nombres* (types "integer" et "decimal") des *autres littéraux* (types "date/time", "wkt" et "string") car ils s'évaluent différemment. La figure 1 montre la distribution des 393 questions de QALD-10 selon les types (et sous-types) de réponses attendues. L'unique question mixant URI et littéraux est ignorée dans la suite. La table 1 donne un exemple de question-réponse pour chacun des quatre types.

## 2.2 Modèles de langage

Pour mener à bien notre étude, nous avons choisi d'utiliser deux modèles de langage spécifiques, GPT-3.5 et ggml-vicuna-13b, qui sont respectivement des versions de ChatGPT [2] et GPT4All [3]. Nous avons choisi GPT-3.5 comme étant le meilleur modèle librement accessible au moment de notre étude. Il est un des plus grands modèles avec 175 milliards de paramètres. Il est largement reconnu pour sa capacité à générer un texte naturel et cohérent. Il a été développé par OpenAI et est largement utilisé pour des tâches de génération de texte.

Plutôt que de comparer GPT-3.5 avec les nombreux modèles concurrents (ex., BARD ou LaMDA), nous avons choisi un modèle léger pouvant s'exécuter localement sur un PC standard. Un modèle léger est a priori moins performant mais il présente comme avantages importants d'être plus sûr puisque les données d'interaction ne sont pas diffusées et d'être moins gourmand en énergie. **ggml-vicuna-13b** est le résultat de la compression (*quantization*) par GPT4All du modèle Vicuna-13B (Chiang et al., 2023) qui comporte 13 milliards de paramètres et a montré des performances proches de ChatGPT malgré sa taille plus petite. Un modèle GPT4All se présente sous la forme d'un fichier de 3 à 8 Go, téléchargeable et utilisable sur un PC standard avec le logiciel de l'écosystème open source GPT4All.

## 2.3 Approche par "Génération de requêtes SPARQL"

La première approche d'utilisation d'un LLM, schématisée dans la figure 2, consiste à demander au LLM de traduire la question en une requête SPARQL pour Wikidata. En effet, le mode d'accès privilégié à Wikidata passe par des requêtes SPARQL et les LLM sont connus pour leurs capacités de traduction des langues naturelles vers d'autres langues naturelles (traduction) mais aussi vers des langages informatiques, par exemple des langages de programmation. De plus, même si on ne connait pas bien le corpus d'apprentissage de certains LLM tels que GPT, on sait qu'il couvre une grande partie du web qui contient de nombreux exemples de requêtes SPARQL associées à des textes expliquant ce qu'elles font (ex., tutoriels, forums de discussion). Le prompt employé est le suivant.

| Prompt |
| --- |
| Your task is to translate the following question to the SPARQL query which gives the real answer(s) on Wikidata. QUESTION : question. Give the correct SPARQL query as simple as possible without any explanation. |

2. https://openai.com/blog/chatgpt
3. https://gpt4all.io/index.html

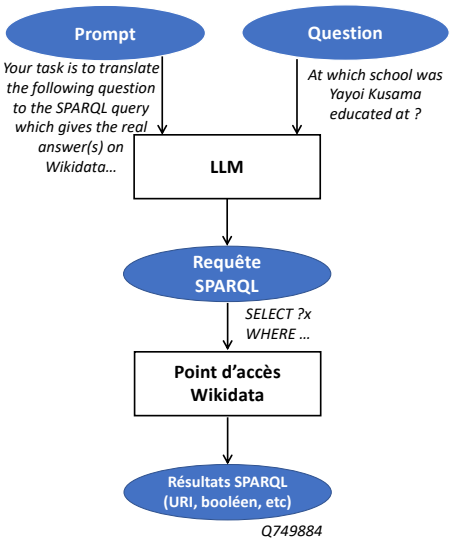


FIG. 2 – *Génération de requêtes SPARQL*

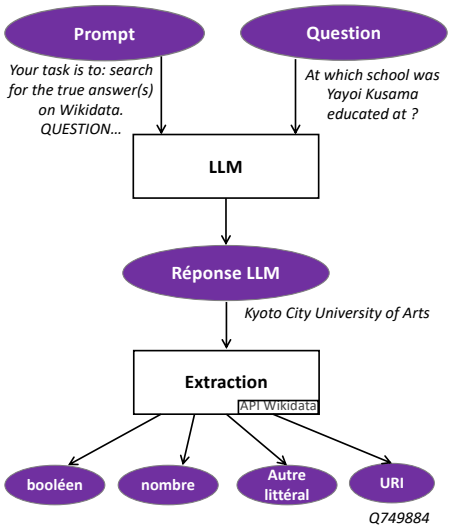


FIG. 3 – *Interrogation directe*

La requête SPARQL obtenue est ensuite exécutée sur le point d'accès de Wikidata [4], et les réponses obtenues peuvent être directement comparées avec les réponses fournies par QALD-10. Cependant, étant donné que la base de connaissances de Wikidata est continuellement mise à jour, nous avons réexécuté les requêtes SPARQL fournies par QALD-10 pour que les réponses attendues soient synchrones avec celles obtenues via la requête produite par le LLM.

## 2.4 Approche par "Interrogation directe"

La deuxième approche d'utilisation d'un LLM, schématisée dans la figure 3, consiste à demander au LLM de retourner directement les réponses à la question, sans passer par une requête SPARQL. Bien que les LLM n'aient pas accès à Wikidata dans cette approche, ils ont la possibilité, au moins en principe, de pouvoir répondre aux questions de QALD. En effet, ces questions portent sur des connaissances générales, qui se trouvent dans les grands corpus de textes utilisés pour l'entraînement des LLM, notamment Wikipedia.

Il est important de définir soigneusement les prompts qui seront soumis aux LLM afin d'obtenir des réponses claires et concises. D'une part, cette concision facilite la comparaison entre les réponses attendues et celles produites par les modèles de langage. D'autre part, elle réduit la longueur des textes générés et donc le temps et le coût des expérimentations. La table 2 montre les prompts utilisés en fonction du type des réponses.

L'évaluation des réponses du LLM est plus difficile que dans l'approche par génération de requêtes SPARQL à cause du format libre des réponses des LLM, par rapport aux résultats structurés d'une requête SPARQL. Les réponses de type booléen, numérique ou autre littéral sont directement comparables mais nécessitent une étape d'extraction. Par exemple, dans le cas des réponses numériques, malgré l'utilisation des prompts retenus précédemment, plusieurs

4. `https://query.wikidata.org/sparql`

TAB. 2 – *Prompts utilisés pour générer chaque type de réponse.*

| Type des réponses | Prompt ($Q désigne la question) |
|---|---|
| Booléen | Your task is to : search for the true answer(s) on Wikidata.<br>QUESTION : $Q. Just give the answer as a boolean. |
| Numérique | Your task is to : search for the true answer(s) on Wikidata.<br>QUESTION : $Q. Just give the number very briefly. |
| URI, autre litéral | Your task is to : search for the true answer(s) on Wikidata.<br>QUESTION : $Q. Just give the real answer without any explanation. |

réponses retournées par les LLM étaient excessivement verbeuses. Par conséquent, il était nécessaire d'extraire les valeurs numériques des blocs de texte. Grâce à l'utilisation conjointe du module `Word2number` et d'expressions régulières, nous avons pu corriger partiellement ce problème. En revanche, les réponses de type URI ne sont pas directement comparables car les réponses attendues sont des identifiants Wikidata (ex., Q1088 pour la couleur bleu) alors que les réponses des LLM sont textuelles (ex., "blue" ou "the color blue"). Nous avons choisi de convertir les réponses textuelles en URI, plutôt que l'inverse, car il est plus fiable de comparer des URI que des textes. Pour ce faire, nous avons soumis au service `wbSearchEntities` [5] de l'API de Wikidata chaque réponse issue des LLM, par exemple le mot "blue", pour récupérer une liste ordonnée (*ranking*) d'URI candidates correspondant à ce texte.

# 3 Évaluation

Dans cette section, nous présentons tout d'abord les mesures de succès utilisées pour comparer les réponses obtenues aux réponses attendues ; puis nous donnons et discutons les résultats obtenus dans les deux approches présentées à la section précédente.

## 3.1 Mesures de succès

Pour chaque question de QALD-10 il peut y avoir plusieurs réponses attendues et on se retrouve donc à devoir comparer des ensembles de réponses dans le cas général, bien que de nombreuses questions attendent une seule réponse. Nous utilisons donc les indicateurs usuels de précision, rappel et F1-score pour évaluer chaque question et considérons la moyenne de ces indicateurs sur des ensembles de questions. Il reste donc à définir dans quelles conditions une réponse attendue est présente dans les réponses obtenues (vrai positif) ou non (faux négatif). Cette définition dépend du type des réponses attendues.

Pour les réponses de type URI, une réponse obtenue est : soit un ensemble d'URI dans l'approche par génération de requête, i.e., toutes les URI retournées par la requête SPARQL ; soit un ensemble de rankings d'URI candidates dans l'approche par interrogation directe. Pour rappel, dans le cas d'une interrogation directe, le LLM renvoie une ou des réponses textuelles (e.g., "Paris") ; et à chaque réponse on associe, via l'API Wikidata, un ranking d'URI possibles pour cette réponse (e.g., {Q90, Q8330149, Q3181341}). Notons que comme la première forme (ensemble d'URI) est un cas particulier de la seconde forme (ensemble de rankings),

5. `https://www.wikidata.org/w/api.php?action=help&modules=wbsearchentities`

TAB. 3 – *Comparaison des deux approches pour les différents types de réponses, avec ChatGPT et les plus petits seuils. Toutes les mesures sont en pourcentage.*

| | Génération de requête | | | | Interrogation directe | | |
|---|---|---|---|---|---|---|---|
| type | résultats | précision | rappel | F1-score | précision | rappel | F1-score |
| Booléen | 11.47 | 0.00 | 0.00 | 0.00 | 57.37 | 57.37 | **57.37** |
| URI | 15.60 | 0.75 | 0.75 | 0.75 | 44.28 | 46.89 | **45.55** |
| Nombre | 85.08 | 13.15 | 13.15 | 13.15 | 23.00 | 23.00 | **23.00** |
| A. littéral | 40.00 | 10.00 | 10.00 | 10.00 | 33.39 | 40.74 | **40.05** |
| tous | 37.79 | 5.27 | 5.27 | 5.27 | 38.93 | 40.93 | **40.26** |

on ne considérera que la seconde. On définit le *rang* $R$ d'une URI attendue comme le rang minimal d'apparition de cette URI dans un des rankings, s'il existe, et $+\infty$ sinon. On considère différents seuils Hit@N pour lesquels une réponse attendue est considérée comme un vrai positif si $R \leq N$ et comme un faux négatif sinon. Tout ranking, donc toute réponse obtenue, qui n'a fourni aucun vrai positif est considéré comme un faux positif. Nous appliquons trois seuils usuels : 1, 3 et 10. Pour Hit@1, la réponse attendue doit être en première position.

Pour les réponses de type *Nombre*, une réponse attendue $r^*$ est considérée comme valide s'il existe une réponse obtenue $r$ telle que la distance relative entre les deux est inférieure à un seuil $\varepsilon$, c-à-d. $\frac{|r-r^*|}{|r^*|} \leq \varepsilon$. Nous appliquons deux seuils, 10% et 50%. Quant aux réponses de type *Autre littéral*, assimilables à des chaines de caractères, nous avons opté pour une distance d'édition. Plus précisément, nous utilisons une variante qui mesure la distance minimale entre la réponse attendue et n'importe quelle sous-chaine de la réponse obtenue (*substring distance*). En effet, les réponses produites par les LLM ont tendance à être plus verbeuses que nécessaire. Le littéral attendu $r^*$ est considéré comme valide s'il existe un littéral obtenu $r$ tel que

$$\frac{\text{substr\_dist}(r^*, r)}{2 \times |r^*|} \leq \varepsilon$$

pour des seuils de distance de 5% et 10%. Enfin, pour les réponses de type booléen, on suppose que les réponses obtenues ont été normalisées et donc que des tests d'égalités suffisent.

## 3.2 Résultats et interprétation

La table 3 compare nos deux approches d'utilisation d'un LLM, par génération de requête et par interrogation directe, pour chaque type de réponse attendue et pour tous types confondus. Seuls les résultats obtenus avec ChatGPT sont rapportés ici car l'objectif est de comparer les deux approches et les résultats avec GPT4All sont inférieurs. Les deux LLM sont comparés ci-dessous pour l'approche par interrogation directe. Dans l'approche par interrogation directe, les seuils minimaux sont utilisés (i.e., Hit@1 pour les URI, 10% pour les réponses numériques et 5% pour les autres littéraux), les performances affichées sont donc minorées. Néanmoins, il est frappant de voir à quel point l'approche directe est plus performante que l'approche par génération de requêtes. Cela s'explique par le fait que la plupart des requêtes générées sont mal formées ou bien ne retournent pas de résultat comme le montre la colonne "résultats" du tableau. Cette colonne donne le pourcentage de requêtes retournant effectivement un résultat,

TAB. 4 – *Comparaison des deux LLM, GPT4All et ChatGPT, pour les différents types de réponses et les différents seuils, avec l'approche par interrogation directe.*

| | | GPT4All | | | ChatGPT | | |
|---|---|---|---|---|---|---|---|
| type | seuil | précision | rappel | F1-score | précision | rappel | F1-score |
| Booléen | | 44.26 | 44.26 | 44.26 | 57.37 | 57.37 | **57.37** |
| URI | Hit@1 | 35.38 | 30.56 | 32.79 | 44.28 | 46.89 | **45.55** |
| | Hit@3 | 37.52 | 32.58 | 34.88 | 49.86 | 52.75 | **51.26** |
| | Hit@10 | 38.89 | 33.94 | 36.25 | 50.64 | 53.53 | **52.04** |
| Nombre | 10% | 16.81 | 16.81 | 16.81 | 23.00 | 23.00 | **23.00** |
| | 50% | 33.62 | 33.62 | 33.62 | 41.59 | 41.59 | **41.59** |
| A. littéral | 5% | 33.33 | 29.62 | 31.37 | 33.39 | 40.74 | **40.05** |
| | 10% | 44.44 | 31.48 | 36.85 | 39.39 | 40.74 | **40.05** |

par exemple seules 15% des requêtes de type URI retournent des résultats. Une autre explication est que les requêtes sur Wikidata nécessitent des URI à base d'identifiants numériques (ex., Q1088, P169) pour lesquels les LLM sont mal adaptés.

La table 4 détaille les résultats pour l'approche par interrogation directe, en comparant cette fois les deux LLM d'une part (colonnes), et les différents seuils d'autre part (lignes). Pour les types *Booléen* et *Nombre*, les trois indicateurs sont égaux car ces types de questions ont une seule réponse. On remarque que ChatGPT est toujours meilleur que GPT4All, comme on pouvait s'y attendre. Néanmoins, pour un modèle léger, ses performances sont remarquables, atteignant au moins les 2/3 des performances de ChatGPT. Un autre point remarquable est que, pour les deux modèles, la précision et le rappel sont équilibrés, ce qui favorise le F1-score. ChatGPT a tendance à avoir un meilleur rappel tandis que GPT4All tend à avoir une meilleure précision. Comme attendu, les scores augmentent avec les seuils, notamment sur les types URI et *Nombre*. Pour les questions de type *Nombre*, cela indique que les LLM sont souvent capables de donner un ordre de grandeur mais peinent à donner des valeurs précises. Pour les questions de type URI, cela révèle des ambiguités sur l'identité des réponses obtenues avec pour effet de repousser la réponse attendue à des rangs plus élevés dans les rankings d'URI.

Globalement, que peut-on dire des performances des LLM pour l'interrogation d'un grand graphe de connaissances générales tel que Wikidata ? D'un côté, ces performances sont plutôt bonnes pour des modèles qui n'ont pas accès au graphe de connaissances et qui n'ont pas été entraînés spécifiquement pour cette tâche. Par exemple, le F1-score des questions de type URI dépasse les 50% avec le seuil Hit@3, ce qui se rapproche du meilleur système évalué sur QALD-10, SPARQL-QA (Borroto et Ricca, 2023) avec un F1-score à 45.4%. D'un autre côté, on peut supposer que les meilleures performances de l'approche directe reflètent simplement le fait que les connaissances de Wikidata sont cohérentes avec les connaissances présentes sur le Web sous forme textuelle. Cela n'est pas vraiment satisfaisant si l'objectif est d'interroger le contenu d'un graphe de connaissances plutôt que des connaissances en général. En particulier, cela ne fonctionnera pas sur des graphes de connaissances spécialisés ou privés (ex., base de connaissances d'une entreprise). De plus, l'approche par interrogation directe a l'inconvénient de fonctionner comme une boîte noire alors que l'approche par génération de requête a l'avantage de permettre l'explicabilité et la traçabilité des résultats.

# 4 Conclusion et perspectives

Notre étude conforte les études précédentes en montrant que les LLM peuvent assez souvent répondre directement aux questions de connaissance générale. Cependant, ils échouent le plus souvent à les traduire en requêtes SPARQL effectives, alors que c'est à priori le seul moyen d'interroger le graphe de connaissances lui-même. L'utilisation des LLM pour la tâche de KGQA offre de nombreuses perspectives. Dans l'approche directe, les questions de type booléen ou numérique pourraient être reformulées pour permettre de justifier la réponse. Par exemple au lieu de demander le nombre de films on peut demander la liste des films et en déduire le nombre. Dans l'approche par génération de requêtes, il est possibe d'aider le LLM. Par exemple, Lehmann et al. (2023) obtiennent un F1-score de 36% en ciblant un langage naturel contrôlé comme intermédiaire entre la langue naturelle et SPARQL.

# Références

Borroto, M. A. et F. Ricca (2023). SPARQL-QA-v2 system for knowledge base question answering. *Expert Systems with Applications 229*.

Chiang, W.-L., Z. Li, Z. Lin, et al. (2023). Vicuna : An open-source chatbot impressing GPT-4 with 90%* ChatGPT quality. https ://lmsys.org/blog/2023-03-30-vicuna/.

Diefenbach, D., V. Lopez, K. Singh, et P. Maret (2017). Core techniques of question answering systems over knowledge bases : a survey. In *Knowledge and Information Systems (KAIS)*.

Faria, B., D. Perdigão, et H. Gonçalo Oliveira (2023). Question Answering over Linked Data with GPT-3. In A. Simões, M. M. Berón, et F. Portela (Eds.), *Symposium on Languages, Applications and Technologies (SLATE)*, OASIcs 113. Schloss Dagstuhl.

Klager, G. G. et A. Polleres (2023). Is GPT fit for KGQA ? – preliminary results. *WU Wien - Vienna University of Economics and Business*.

Lehmann, J., P. Gattogi, D. R. Bhandiwad, S. Ferré, et S. Vahdati (2023). Language models as controlled natural language semantic parsers for knowledge graph question answering. In *European Conf. Artificial Intelligence*, Volume 372. IOS Press.

Usbeck, R. et al. (2023). QALD-10 — the 10th challenge on question answering over linked data. *Semantic Web*. In press.

Vaswani, A., N. Shazeer, N. Parmar, et al. (2017). Attention is all you need. *Advances in neural information processing systems 30*.

# Summary

In this article, we present the results of an in-depth study of the performance of language models (LLMs) in the context of question-answering over knowledge graphs (KGQA). The experimental methodology was structured around two different approaches: generation of SPARQL queries and direct question-answering. The results on the QALD-10 benchmark showed very poor results in the first approach and fair results in the second approach, with important variations between the different types of questions and answers.