



Passage de la langue naturelle à une requête SPARQL

Chemam Chaouki ^{a *}, Touati Zineb ^a, Zarzour Hafed ^b

^a Département d'Informatique — Université Chadli Bendjedid El-Tarf, BP 73, (36000) — Algérie.

^b Universités M ohammed-Chérif Messaadia, souk ahras, Algérie .

Mots clés :

Web, Web sémantique
 RDF, SPARQL,
 Linked data, NLP, http,
 URI

*Correspondance :

Che_chaouki@yahoo.fr

Avec la quantité croissante de données sémantiques disponibles sur le Web, il existe un fort besoin de systèmes permettant aux utilisateurs Web courants d'accéder à ce corpus de connaissances. Les systèmes de réponse aux questions ont particulièrement retenu l'attention, car ils permettent aux utilisateurs d'exprimer des besoins d'informations arbitrairement complexes de simple et intuitive. Le principal défi consiste à traduire les besoins d'information des utilisateurs sous une forme telle qu'ils puissent être évalués à l'aide de techniques standard de traitement et d'inférence de requêtes sur le Web sémantique. Au cours des dernières années, une série d'approches ont été développées pour relever ce défi, montrant des avancées significatives vers la réponse aux questions de langage naturel par rapport à de grands ensembles hétérogènes de données structurées. Cependant, seuls quelques systèmes traitent encore le fait que les données structurées disponibles de nos jours sont réparties entre une grande collection d'ensembles de données interconnectés et que les réponses aux questions ne peuvent souvent être fournies que si les informations provenant de plusieurs sources sont combinées. En outre, de nombreuses informations ne sont toujours disponibles que sous forme textuelle, à la fois sur le Web et sous forme d'étiquettes et de résumés dans des sources de données liées. [1]

Informations	Résumé
1. Introduction Une tâche difficile dans la réponse aux questions en langage naturel (Q / A pour faire court) sur le graphe de connaissances RDF est de savoir comment combler l'écart entre les questions de langage naturel non structurées (NLQ) et les données RDF structurées en graphe (G). L'un des outils effectives est le «modèle», qui est souvent utilisé dans de nombreux systèmes RDF Q / A existants. Cependant, peu d'entre eux étudient comment générer automatiquement des modèles. À notre connaissance, nous sommes les premiers à proposer une approche de jointure pour la génération de modèles. Étant donné une charge de travail D de requêtes SPARQL et un	ensemble N de questions en langage naturel, l'objectif est de trouver des paires q, n, pour q ∈ D A n G N, où la requête SPARQL q est la meilleure correspondance pour la question n en langage naturel n . Ces paires fournissent des conseils prometteurs pour la génération automatique de modèles. En raison de l'ambiguïté des langues naturelles, nous modélisons le problème ci-dessus comme une tâche de jointure de graphe incertaine. Nous proposons plusieurs techniques d'élague structurel et probabiliste pour accélérer l'assemblage. Des expériences approfondies sur des ensembles de données de référence RDF Q / A réels confirment à la fois l'efficacité et l'efficacité de notre approche. [21 Récemment, les graphiques des connaissances ont attiré

beaucoup d'attention à la fois dans le milieu universitaire et dans l'industrie. Le cadre de description des ressources (RDF) étant la norme de facto d'un graphe de connaissances, nous nous concentrons sur le référentiel RDF dans cet article. Un problème clé est de savoir comment accéder aux graphiques de connaissances et obtenir rapidement les informations souhaitées. Bien que SPARQL soit un langage de requête structuré sur les graphiques RDF, il n'est pas pratique pour les utilisateurs non professionnels d'interroger les graphiques RDF à l'aide de SPARQL, en raison de la complexité de la syntaxe

SPARQL et du schéma RDF. Nous illustrons un exemple de requête SPARQL sur DBpedia comme suit.

```
SELECT ?person
```

```
WHERE { ?person rdf:type Artist.
```

```
?person graduatedFrom Harvard University.}
```

2. Motivation

Pour masquer la complexité de la syntaxe SPARQL, les systèmes de questions / réponses (Q / A) RDF fournissent une interface facile à utiliser pour les utilisateurs, ce qui a attiré une grande attention dans les deux NLP (traitement du langage naturel) et les communautés DB (base de données). Une tâche difficile consiste à traduire les questions en langage naturel (NLQ) en requêtes structurées, telles que SPARQL, sur un grand graphe de connaissances G. L'un des outils efficaces est le «modèle», qui est utilisé dans de nombreux systèmes RDF Q / A existants. Un système typique est EVI (<http://www.evi.com/>, officiellement True Knowledge), qui a enregistré plus d'un million d'utilisateurs dans les quatre mois suivant son lancement en janvier 2012. EVI a été acquis par Amazon en octobre 2012 et fait maintenant partie du groupe d'entreprises Amazon. EVI vise à répondre directement aux questions posées en texte simple en anglais sur la base de connaissances. Il utilise une approche basée sur un modèle pour répondre aux questions en langage naturel, où un modèle décrit comment transformer une classe de questions en langage naturel en requêtes structurées correctes. Cependant, EVI doit définir manuellement les modèles de traduction. Il est clair que la qualité des modèles détermine la qualité de la réponse. Le défi est de savoir comment générer automatiquement un grand nombre de modèles de haute qualité. Peu d'ouvrages antérieurs

étudient cette question, mais tous doivent définir manuellement les modèles. De toute évidence, il est coûteux de définir manuellement ces modèles, en particulier pour les systèmes Q / A en domaine ouvert sur des graphiques de connaissances RDF à grande échelle. Dans ce document, nous étudions comment générer automatiquement des modèles. [1]

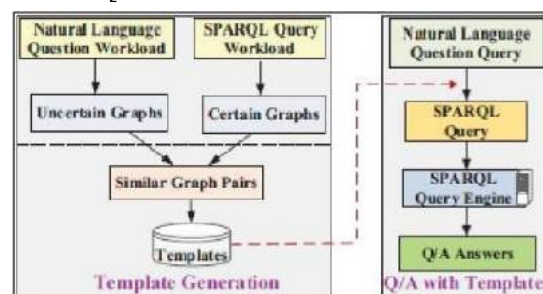


Figure 1 : Cadre pour Q / A RDF basé sur un modèle. III

La figure 1 présente le cadre de RDF Q / A à l'aide de modèles. Il se compose de deux tâches: comment générer des modèles et comment utiliser ces modèles dans RDF Q / A. Pour résoudre le premier problème (objet de cet article), nous proposons une approche basée sur la charge de travail des requêtes pour la génération automatique de modèles. Plus précisément, les entrées pour la tâche de génération de modèle sont deux charges de travail de requête, L'un est un ensemble D de requêtes SPARQL sur le référentiel RDF (tel que DBpedia workload). L'autre est un ensemble N de questions en langage naturel, qui peuvent être collectées à partir de certains sites de questions-réponses (Q&A) axés sur la communauté (tels que Yahoo Answers) ou de la charge de travail des requêtes des moteurs de recherche (telles que WebQuestions2). L'ensemble du processus se déroule en trois étapes comme suit.

Étape 1. Génération de graphique incertaine. Selon la méthode de, les questions en langage naturel peuvent être interprétées dans des graphes de requêtes sémantiques. En raison de l'ambiguïté de l'interprétation, nous modélisons le graphe de requête sémantique comme un graphe incertain, c'est-à-dire que chaque sommet / arête a plusieurs étiquettes possibles avec différentes probabilités. Ici, les étiquettes sont incertaines en raison de l'ambiguïté sémantique. Par exemple, considérons la question «quel acteur américain est marié à

Michael Jordan né dans une ville de New York» dans la figure 2.

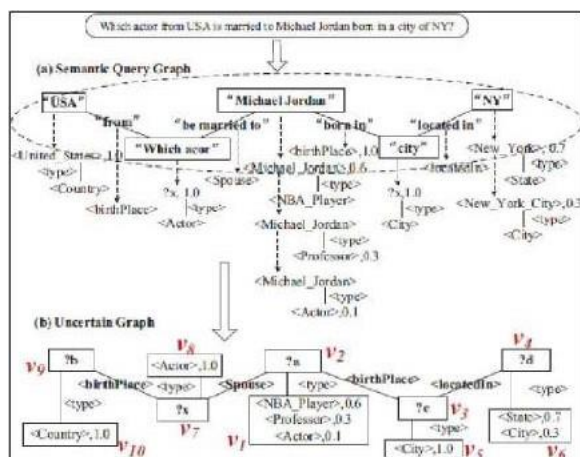


Figure 2 : Génération de graphique incertain. [1]

Il y a trois personnes nommées «Michael Jordan» dans le graphique DBpedia RDF, c'est-à-dire une Star de la NBA, un professeur et un acteur, chacun étant associé à une probabilité. «NY» a également deux correspondances possibles, à savoir l'état et la ville. Par conséquent, nous pouvons obtenir un graphe incertain contenant des nœuds avec plusieurs étiquettes, chacune associée à une probabilité.

Étape 2. Recherche de paires de graphiques similaires. Le but est de trouver certaines paires q, g , pour $q \in D_A$ et $g \in U$, où D correspond aux requêtes SPARQL et U sont les graphiques incertains dérivés des questions en langage naturel [3]. Ici, nous utilisons la distance d'édition du graphique, qui est largement utilisé pour mesurer la similitude des graphes, pour calculer la similitude entre q et g .

Étape 3. Génération de modèles avec des paires de graphiques similaires. Étant donné une paire retournée q, g , basée sur le mappage entre q et g (le mappage est trouvé lors du calcul de la distance d'édition du graphe à l'étape 2), nous pouvons construire les modèles. Afin de générer des modèles, le problème clé est de savoir comment trouver les paires de graphiques similaires efficacement (c'est-à-dire l'étape 2). C'est l'objet de cet article. Nous proposons une nouvelle approche, à savoir, Similarity Join sur des graphiques incertains, notée SimJ. Plus précisément, nous proposons une série de bornes effectives pour améliorer les performances. Compte tenu des modèles générés, une question en langage naturel est traduite en une requête SPARQL à l'aide des

modèles générés lors de la première tâche. Ensuite, la requête SPARQL est recherchée sur les graphiques de connaissances. Les systèmes existants, tels que Jena, RDF-3x, Virtuoso et gstore, peuvent être utilisés pour terminer la recherche. L'ensemble des traitements se déroule en deux étapes comme suit.

Traitement hors ligne: Pour activer le traitement en ligne d'extraction de relations sémantiques, un dictionnaire des relations RDF et de leurs paraphrases en langage naturel est automatiquement construit à l'avance. Par exemple, le directeur de relation peut être exprimé comme «dirigé par» ou «un film par». Nous suivons une approche d'analyse de dépendance simple basée sur trois points comme décrit dans [3]. Pour chaque triplet sujet-prédicat-objet (s, p, o) dans l'ensemble de données, ces phrases dans la codification de la confiance s et o sont récupérées. Ensuite, les phrases sont analysées par l'analyseur de dépendance, et les chemins les plus courts entre s et o sont sélectionnés comme paraphrases de la relation p . Un autre défi est l'ambiguïté du langage naturel. Une seule phrase dans la question peut souvent trouver plus d'un mappage dans le graphique RDF. Pour y faire face, nous introduisons les fragments RDF, qui sont des sous-graphes RDF semblables à des étoiles qui découvrent le voisinage de chaque relation, entité ou classe sémantique. Ensuite, nous pouvons à peu près savoir comment un certain élément sémantique est connecté dans le graphique à l'intérieur d'un seuil de longueur de pas. Les fragments RDF ont un bon pouvoir de filtrage et de désambiguïsation pour ces mappages inappropriés, tout en ayant des consommations d'espace acceptables dans de nombreuses entités de ce même type, qui partagent les mêmes fragments. Les fragments RDF peuvent être indexés par une liste inversée.

Traitement en ligne : Au cours de l'étape en ligne, la question d'entrée qNL est introduite dans le pipeline en quatre étapes suivant: Premièrement, qNL est analysé dans un arbre de dépendance tNL. Deuxièmement, les phrases en qNL qui mentionnent toute relation sémantique sont reconnues en tNL avec l'aide du dictionnaire de paraphrase. Les phrases pour les entités et les classes sont ces arguments syntaxiques en tNL associés aux phrases de relation. Nous unifions également les phrases qui se réfèrent au même élément par une analyse de coréférence. Troisièmement, ces phrases sont mappées aux fragments RDF pour trouver leurs correspondances d'éléments

sémantiques dans le graphique RDF G. Les variables sont introduites par le focus de la question, les phrases d'en tête sans correspondance et les fragments RDF des classes sémantiques qui sont des variables à contrainte de type. Enfin, les fragments RDF sont joints à l'ordinateur sous-graphique raisonnable du graphique G, en vérifiant leur compatibilité en fonction de l'arbre de dépendance tNL. Les résultats sont classés selon le score de similitude et de cohérence sémantique, conduisant au SPARQL ciblé.

3. Linked data (web de données) :

Le Web des données (linked data, en anglais) est une initiative du W3C (Consortium World Wide Web) visant à favoriser la publication de données structurées sur le Web, non pas sous la forme de silos de données isolés les uns des autres, mais en les reliant entre elles pour constituer un réseau global d'informations.

Il s'appuie sur les standards du Web, tels que HTTP et URI- mais plutôt qu'utiliser ces standards uniquement pour faciliter la navigation par les êtres humains, le Web des données les étend pour partager l'information également entre machines. Cela permet d'interroger automatiquement les données, quels que soient leurs lieux de stockage, et sans avoir à les dupliquer.

Tim Berners-Lee, directeur du W3C, a inventé et défini le terme Linked Data ou Données liées et son synonyme Web of data au sein d'un ouvrage portant sur l'avenir du Web sémantique. En France, le terme Web des données est de plus en plus utilisé par la communauté des professionnels du domaine.

Pour les bibliothèques, il est nécessaire de participer au web des données afin de gagner en visibilité et afin de rendre ces données exploitables dans d'autres contextes. Le web des données permet ainsi de décloisonner les données des catalogues, afin qu'elles soient plus accessibles aux usagers et ceux en devenir. Il permet entre autres de relier des données d'archives, de musées ou des biographies à des textes présents dans le catalogue, offrant ainsi au chercheur une information plus complète.

Tim Berners-Lee a défini quatre piliers pour soutenir l'initiative « Web des données » :

1. utiliser des adresses URI uniques pour identifier les choses.
2. utiliser des adresses URI HTTP qui existent sur le Web (des URL, donc).

Une Erreur HTTP 404 indique simplement que l'adresse URI utilisée n'est pas documentée explicitement.

3. fournir à travers l'adresse URI des renseignements exploitables, lisibles par les humains et par les machines, en s'appuyant sur des formats ouverts comme RDF ou SPARQL. Par exemple en utilisant le mécanisme de redirection (code 302) et la variable User-Agent contenue dans les en-têtes des requêtes HTTP, un serveur peut afficher une page en XML ou RDF pour une machine ou une page HTML pour le navigateur d'une personne.
4. ailler l'adresse URI initiale en lui associant des adresses URI externes et ce pour améliorer la découverte d'autres informations sur le Web.

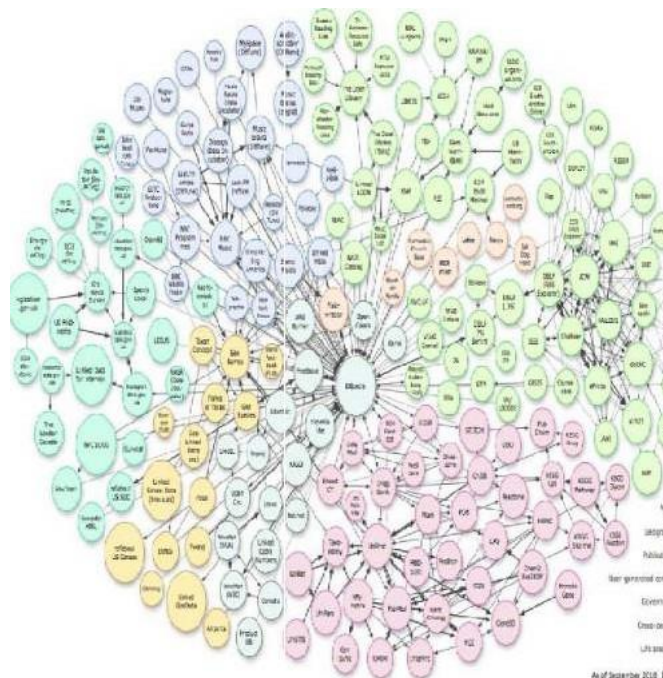


Figure 3 : Représentation de DBpedia en 2010.

4. Algorithme de passage de la langue naturelle à une requête SPARQL :

Algorithm 1. [integrating Adj2ER into existing QA systems]

Input: X natural language question Q

Output: A SPARQL query for the input question Q

```
1: GENERATESPARQL(Q) S EnibingQASysfIII.GENEMTESPARQL(Q);
   for all (adj. C) pair in Q do Restrictions Adj2ER(adj, C); if S a
   Restrictions then continue;
```

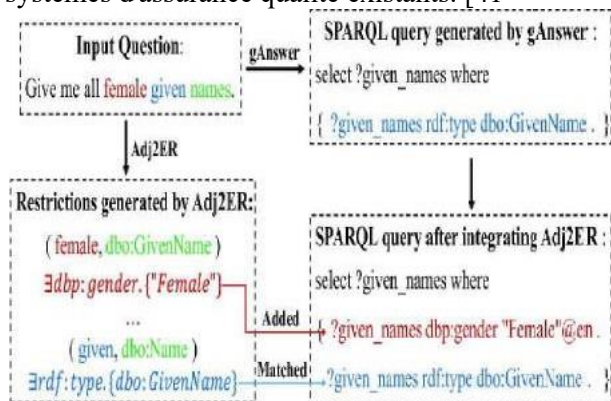
```
   Remove all for all restriction
```

```
   Restrictions do
```

```
       S' Resulting SPARQL for adding restriction to S; if
       isa rwn-empty querythen S Y ; break;
```

```
11: return S;
```

Stratégie d'intégration. Nous avons intégré notre approche de cartographie adjectivale dans deux systèmes d'assurance qualité de pointe, à savoir gAnswer et WDAqua. Qui se sont classés respectivement premier et deuxième du défi QALD-9. La procédure pour intégrer Adj2ER en eux est présentée dans l'algorithme 1. Un exemple d'intégration d'Adj2ER dans les systèmes d'assurance qualité existants. [4]



	70 QALD questions			30 YA questions		
	p	R	FI	p	R	FI
gAnswer	30.49%	35.48%	29.75%	16.34%	36.26%	13.9No
gAnswer+Adj2ER	44.03%	62.25%	43.02%	37.32%	56.50%	33.7/% 41.187,
	21.10%	26.64%	17.79%	23.53%	28.10%	
WDAW0+Adj2ER	33.28%	43.86%	32.03%	42.70%	44.88%	40.9% 35.77%

5. Conclusion :

Dans cet article, nous proposons un cadre basé sur des données graphiques pour répondre aux questions en langage naturel sur les graphiques RDF. Différent à partir des travaux existants, nous permettons l'ambiguïté phrases et structure dans la phase de compréhension des questions. Nous poussons la désambiguïsation dans l'évaluation des requêtes étape. Sur la base des résultats des requêtes sur les graphiques RDF, nous peut résoudre efficacement le problème d'ambiguïté. En d'autres termes, nous combinons la désambiguïsation et l'évaluation des requêtes dans un processus uniforme. Par conséquent, le graphique axé sur les données le cadre améliore non seulement la précision mais aussi les vitesses toutes les performances du système RDF Q / A.

References :

- [1] Weiguo Zheng, Lei Zou, Xiang Lian and al, Howto Build Templates for RDF Question/Answering—An Uncertain Graph Similarity Join Approach

- [2] Camille Pradel, Ollivier Haemmerlé, Nathalie Hernandez, Passage de la langue naturelle à une requête SPARQL dans le système SWIP

- [31] N. Nakashole, G. Weikum, and F. M. Suchanek. Patty: A taxonomy of relational patterns With semantic types. In EMNLP-CoNLL, pages 1135-1145, 2012.

- [4] J Ding et al : The Semantic Web - lswc 2019: 18th International Semantic Web Partie 1, Page 176. 2019.