TP QA-part 1: Conception et implémentation d'un système de questions réponses en langue naturelle sur des données structurées

Les **systèmes question/réponse** sont des systèmes qui répondent à une question posée en langage naturel, par l'extraction d'une réponse précise à partir d'un corpus de documents. La discipline liée appartient aux domaines du **traitement automatique de la langue** et de la **recherche d'information**.

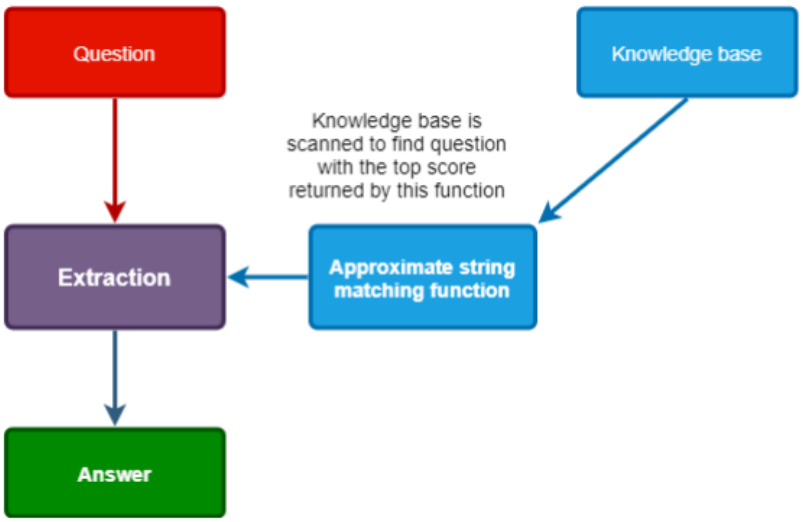
Les systèmes de réponse aux questions ont 3 buts principaux :

1. **Comprendre les questions en langue naturelle** (Analyser la question, quel est son type ? Quelle est son domaine ? On parlera de question fermée pour une question se portant sur un domaine spécifique (médecine, informatique...), et de question ouverte pouvant porter sur n'importe quoi et pour lesquelles on pourra vouloir faire appel à des systèmes d'ontologies généraux et des connaissances sur le monde.)

2. **Trouver les informations** (Soit au sein de base de données structurée (base de données spécialisées), soit au sein de textes hétérogènes (recherche sur internet)

3. **Répondre à la question** (Que ce soit par une réponse exacte, ou par la proposition de passages susceptibles de contenir la réponse.)

Lorsque un **système question/réponse** utilise une base de données pour répondre aux questions, il faut tout d'abord transformer une requête constituée de mots clés ou parfois d'une phrase en langage naturel en une requête formelle compréhensible par le système de gestion de base de données utilisé. Ensuite, pour répondre le plus précisément possible à un maximum de réponses, il faut d'une part disposer d'un large ensemble de connaissances et d'autre part être capable de reconnaître le plus possible de formes de requêtes correspondant aux réponses que l'on peut produire.



Le but de ce TP est de concevoir et implémenter un système de questions réponses (simple) en langue naturelle au dessus de la base DBpedia (http://fr.dbpedia.org/).
Cette base est constituées de triplets (sujet, prédicat, objet), décrites en RDF (Resource Description Framework) présentés comme des graphes, où sujets et objets sont les noeuds et les prédicats sont les arcs étiquetés par des labels. On parle aussi de relation entre deux entités. Ces graphes peuvent être interrogés par un langage de requête, tel que SPARQL.

La génération de ces requêtes en SPARQL requiert 1) d'identifier les entités et relations de la base mentionnés dans la question, 2) de les associer en triplets et 3) de construire la requête elle-même.

Tache 1. Analyse des questions

Même si on peut différencier de nombreux types de requêtes, on se focalisera uniquement sur les **questions factuelles** (exemples : *Où a été brûlée Jeanne d'Arc ?, Quelle est la capitale de la France?).*

Téléchargez le fichier *questions.xml* (sur Moodle), qui contient une liste de questions en langage naturel en plusieurs langues. Il s'agit d'un échantillon des questions du défi QALD (Question Answering over Linked Data). Voici le format de chaque question :

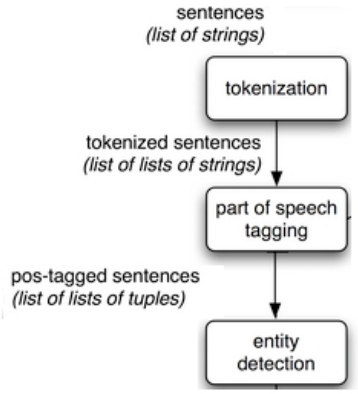
```
<question aggregation="false" answertype="resource" id="4" onlydbo="true">          IDENTIFIANT QUESTION
<string lang="en">Which river does the Brooklyn Bridge cross?</string>             QUESTION EN ANGLAIS
<string lang="de">Welchen Fluss überspannt die Brooklyn Bridge?</string>
<string lang="es">¿Por qué río cruza la Brooklyn Bridge?</string>
<string lang="it">Quale fiume attraversa il ponte di Brooklyn?</string>
<string lang="fr"> Quelle cours d'eau est traversé par le pont de Brooklyn? </string>   QUESTION EN FRANCAIS
<string lang="nl">Welke rivier overspant de Brooklyn Bridge?</string>
<keywords lang="en">river, cross, Brooklyn Bridge</keywords>
<keywords lang="de">Fluss, überspannen, Brooklyn Bridge</keywords>
<keywords lang="es">río, cruza, Brooklyn Bridge</keywords>
<keywords lang="it">fiume, attraversare, ponte di Brooklyn</keywords>
<keywords lang="fr">cours d'eau, pont de Brooklyn</keywords>
<keywords lang="nl">rivier, Brooklyn Bridge, overspant</keywords>
<query>
PREFIX dbo: <http://dbpedia.org/ontology/>
PREFIX res: <http://dbpedia.org/resource/>                                        SPARQL QUERY A LA BASE DBPEDIA
SELECT DISTINCT ?uri WHERE {
        res:Brooklyn_Bridge dbo:crosses ?uri .
        }
</query>
<answers>
<answer>
<uri>http://dbpedia.org/resource/East_River</uri>                                 REPONSE ATTENDUE (EXTRAITE DE DBPEDIA)
</answer>
```

Vous pouvez choisir de travailler sur les questions en Anglais ou en Français.

Sous-tache 1a. Pre-traitement des questions.

Comme première étape, vous devez implémenter un système d'extraction d'informations simple (voir figure en bas). Le texte brut de la question est subdivisée en mots à l'aide d'un **tokenizer**. Ensuite, chaque phrase est étiquetée avec des balises de partie de discours (**PoS**

tagger), ce qui s'avérera très utile à l'étape suivante, la **détection d'entités nommées (NER)**. Dans cette étape, nous recherchons des mentions d'entités potentiellement intéressantes dans chaque phrase, qui souvent représentent l'objet sur lequel se porte la question.
Pensez a stocker toutes les informations extraites, elles vous seront utiles par la suite !



Vous pouvez choisir d'utiliser le tokenizer, PoS tagger, et NER de :

- **NLTK** (pour l'Anglais, pour le Français il n'y a pas de NER)

https://www.nltk.org/book/ch03.html#chap-words
https://www.nltk.org/book/ch05.html#chap-tag
https://www.nltk.org/book/ch07.html

- SPACy (pour l'Anglais et le Français)

https://spacy.io/api

Sous-tache 1b. Identifier le type de réponse attendu
Le type de la réponse attendue correspond à l'identification de l'objet de la question (souvent à partir de la reconnaissance du type d'entité nommée) ou du type de la phrase attendu.

Exemple de type d'objets

Personne	« Qui ... », « Quel ministre ... »
Organisation	« Qui ... », « Quelle compagnie ... »
Lieu	« Où ... », « Dans quelle région ... »
Date	« Quand ... », « En quelle année ... »

En vous appuyant sur les entités nommés identifiées dans la question (plus quelques heuristiques, par ex. ``si la question commence avec QUI, le type de réponse attendu est une personne''), écrivez des expressions réguliers pour identifier le type de réponse attendu pour chaque question.

Remarque : Essayez de ne pas écrire des regex trop ciblées sur ce jeu de questions (la dernière séance du TP je vous donnerai un autre jeu de questions pour tester si votre système generalise bien).

Préparation aux sous-taches 1c et 1d. Identifier les entités et les relations de la base mentionnés dans la question

En préparation de ces sous-taches qu'on abordera pendant la partie II du TP, commencez a vous familiariser avec la base DBpedia. Le langage standard d'interrogation de cette base de connaissances est **SPARQL, langage proche du SQL,** qui permet d'interroger les données RDF des ressources du Web sémantique. Ne vous inquiétez pas, on n'écrira que de requêtes simples (que de SELECT). Lisez ce court tutoriel : https://www.w3.org/2009/Talks/0615-qbe/

Pour vous entrainer a t**ester vos requêtes SPARQL sur DBpedia** (vous pouvez copier et tester les requêtes du le jeu de questions questions.xml) : https://dbpedia.org/sparql

Querying DBpedia via API: https://wiki.dbpedia.org/OnlineAccess#1%20Querying%20DBpedia

Pour faire le matching entre les entités nommées que vous avez identifié dans votre question et les entités dans la base DBpedia, servez-vous du DBpedia Lookup Service - Find DBpedia URIs for keywords (https://wiki.dbpedia.org/lookup). Ce service de recherche DBpedia peut être utilisé pour rechercher des URI DBpedia à l'aide de mots-clés associés. Associé signifie que soit le libellé d'une ressource correspond, soit un texte d'ancrage fréquemment utilisé dans Wikipédia pour faire référence à une ressource spécifique correspond (par exemple, la ressource `http://dbpedia.org/resource/United_States` peut être recherchée par la chaîne "USA").

Pour ce qui en est des relations de la base, ici (http://mappings.dbpedia.org/server/ontology/classes/) vous pouvez trouver l'ontologie de DBpedia, qui contiens toutes les relations de la base. Pendant le prochain TP il vous sera demandé de faire le match entre la relation détectée dans la question (qui vous permettra de trouver la réponse) et la bonne relation dans la base.
Cette tâche pose le problème des variations lexicales entre les labels associés aux entités et relations de la base et les termes employés par l'utilisateur, puisque celui-ci n'est pas guidé par la connaissance du schéma de la base. Se pose aussi le problème de résolution d'ambiguïtés sémantiques, car un même terme peut faire référence à différents objets ou prédicats. Par exemple, le verbe *married to* peut faire référence aux relations *dbo:spouse*, *dbo:partner*, *dbp:wife*, *dbp:husband*, *dbp:union*, *dbp:relationship*.
A suivre...