

Task 1: Question Answering Systems

Given are the following five queries:

- a: What movies has Leonardo DiCaprio acted in?
- b: Give me the movies whose music composers have won the BAFTA Award for Best Film Music.
- c: Who wears the number 5 of Germany?
- d: Has Germany won the FIFA World Cup 2014?
- e: Who is the US president?

Task 1.1: Describe what kind of different result types you expect for each query.

Task 1.2: Name potential challenges for each query (if there are any).

Task 1.3: Execute the queries on all of the following Question Answering systems and describe the results.

- WDAqua: wdaqua-frontend.univ-st-etienne.fr
- Wolfram Alpha: <http://www.wolframalpha.com>
- START: <http://start.csail.mit.edu/index.php>
- Google: www.google.com

Query	WDAqua	Wolfram Alpha	START	Google
a				
b				
c				
d				
e				

Task 2: Semantic Question Answering Example System

Task 2.1: As in Task 1, query for “What movies has Leonardo DiCaprio acted in?” on wdaqua-frontend.univ-st-etienne.fr and list three results.

Task 2.2: Use the “Did you mean” button to list potential interpretations of the entities included in the query.

Task 2.3: Use the “Q” button to find Wikidata SPARQL queries generated for this natural language query. Find a SPARQL query that gives correct results and one query that gives wrong results and confirm this using the Wikidata SPARQL endpoint (<https://query.wikidata.org/>).

Task 2.4: What is the role of the relation linking in this example?

Task 3: Question Answering Pipeline

Given is the natural language query “Which papers has Michelle written?” and the following triple database:

Charlie_Carpenter, write, Contributions_of_Michelle
Charlie_Carpenter, write, Keyword_Search_in_XML
Michael_Richardson, write, Keyword_Search_in_XML
Michelle, write, Keyword_Search_in_XML
Michelle, write, Pattern_Matching_in_XML
Michelle, write, Algorithms_for_TopK_Query
Keyword_Search_in_XML, cite, Contributions_of_Michelle
Pattern_Matching_in_XML, cite, Contributions_of_Michelle
Keyword_Search_in_XML, cite, Pattern_Matching_in_XML
Pattern_Matching_in_XML, cite, Algorithms_for_TopK_Query
Keyword_Search_in_XML, cite, Algorithms_for_TopK_Query
Contributions_of_Michelle, type, Paper
Keyword_Search_in_XML, type, Paper
Pattern_Matching_in_XML, type, Paper
Algorithms_for_TopK_Query, type, Paper
Charlie_Carpenter, type, Person
Michael_Richardson, type, Person
Michelle, type, Person
Charlie_Carpenter, type, Woman
Michelle, type, Woman
Michael_Richardson, type, Man

Task 3.1: What should be the output of the entity linking step?

Task 3.2: What should be the output of the relation linking step?

Task 3.3: What should be the top ranked SPARQL query?

Task 4: Template-based Benchmark Generation

For each of the following three templates¹:

- Create an example query based on the dataset in Task 3.
- Write down the corresponding natural language question for your query.
- Write down the expected results retrieved by your query.

- a: Simple query

```
SELECT DISTINCT ?uri WHERE {  
  <a> <b> ?x .  
  ?x <c> ?uri  
}
```

- b: Count query

```
SELECT DISTINCT COUNT(?uri) WHERE {  
  ?uri <a> <b> .  
  ?uri rdf:type <class> .  
}
```

- c: Ask query

```
- ASK WHERE {  
  ?a ?b ?c .  
}
```

¹ <https://github.com/AskNowQA/LC-QuAD/blob/develop/templates.py>