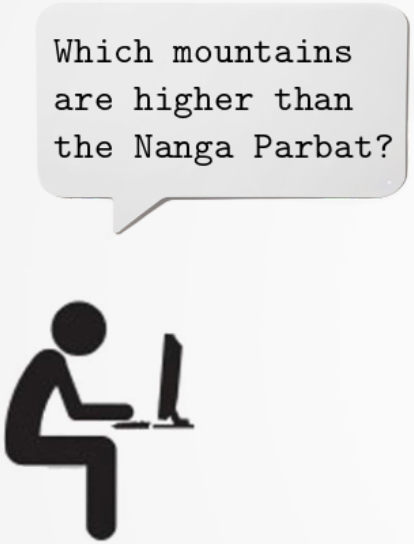
Universität Bielefeld

The long and winding road from natural language questions to SPARQL queries

Christina Unger

AG Semantic Computing
CITEC, Universität Bielefeld

The task




```
resource:Nanga_Parbat a dbpedia:Mountain .
resource:Nanga_Parbat dbpedia:elevation '8126' .
resource:K2 a dbpedia:Mountain .
resource:K2 dbpedia:elevation '8611' .
resource:Annapurna a dbpedia:Mountain .
resource:Annapurna dbpedia:elevation '8091' .
resource:Mount_Everest a dbpedia:Mountain .
resource:Mount_Everest dbpedia:elevation '8848' .
resource:Amsterdam a dbpedia:City .
resource:Amsterdam dbpedia:elevation '2' .
...
```

Given one or several RDF datasets and a natural language question, return the correct answers.

Which mountains are higher than the Nanga Parbat?

```
PREFIX dbo: <http://dbpedia.org/ontology/>
PREFIX res: <http://dbpedia.org/resource/>

SELECT DISTINCT ?uri WHERE {
  ?uri a dbo:Mountain .
  res:Nanga_Parbat dbo:elevation ?e1 .
  ?uri dbo:elevation ?e2 .
  FILTER (?e2 > ?e1) .
}
```

Eight mountains are higher than the Nanga Parbat: Mount Everest, Makalu, K2, ...

```
resource:Nanga_Parbat a dbpedia:Mountain .
resource:Nanga_Parbat dbpedia:elevation '8126' .
resource:K2 a dbpedia:Mountain .
resource:K2 dbpedia:elevation '8611' .
resource:Annapurna a dbpedia:Mountain .
resource:Annapurna dbpedia:elevation '8091' .
resource:Mount_Everest a dbpedia:Mountain .
resource:Mount_Everest dbpedia:elevation '8848' .
resource:Amsterdam a dbpedia:City .
resource:Amsterdam dbpedia:elevation '2' .
...
```

```
<http://dbpedia.org/resource/Mount_Everest>
<http://dbpedia.org/resource/Makalu>
<http://dbpedia.org/resource/K2>
...
```

Seems easy!

Seems easy! ...or not?

Outline

1. The challenges
   - QALD
   - OKBQA
   - BioASQ
2. From questions to answers (via templates)
3. OKBQA ideas

Outline

1. The challenges
   - QALD
   - OKBQA
   - BioASQ
2. From questions to answers (via templates)
3. OKBQA ideas

Outline

1. The challenges
   - QALD
   - OKBQA
   - BioASQ
2. From questions to answers (via templates)
3. OKBQA ideas

Benchmark

QALD

http://www.sc.cit-ec.uni-bielefeld.de/qald/

Dataset: DBpedia

- questions (en,es,de,nl,fr,it,ro)
- SPARQL queries
- answers (URIs or literals)

No challenge

Who developed Minecraft?

```
SELECT DISTINCT ?uri
WHERE {
    dbpedia:Minecraft dbpedia-owl:developer ?uri .
}
```

Lexical mismatches

How tall is Michael Jordan?

```
SELECT DISTINCT ?num
WHERE {
    dbpedia:Michael_Jordan dbpedia-owl:height ?num .
}
```

Lexical knowledge

Give me all taikonauts.

```
SELECT DISTINCT ?uri
WHERE {
     ?uri rdf:type dbpedia-owl:Astronaut .
     ?uri dbpedia-owl:nationality dbpedia:China .
}
```

Multiliguality

한라산은 얼마나 높은가?

```
SELECT DISTINCT ?height
WHERE {
    dbpedia:Hallasan dbpedia-owl:elevation ?height .
}
```

Semantically light expressions

List all books by Kerouac.

```
SELECT DISTINCT ?uri
WHERE {
    ?uri dbpedia-owl:author dbpedia:Jack_Kerouac .
}
```

Semantically light expressions

List all movies by Kurosawa.

```
SELECT DISTINCT ?uri
WHERE {
    ?uri dbpedia-owl:director dbpedia:Akira_Kurosawa .
}
```

Semantically light expressions

List all songs by Blur.

```
SELECT DISTINCT ?uri
WHERE {
    ?uri dbpedia-owl:musicalArtist dbpedia:Blur_(band) .
}
```

Semantically light expressions

List all games by GMT.

```
SELECT DISTINCT ?uri
WHERE {
    ?uri dbpedia-owl:publisher dbpedia:GMT_Games .
}
```

Entity disambiguation

List all games by GMT.

```
SELECT DISTINCT ?uri
WHERE {
    ?uri dbpedia-owl:publisher dbpedia:GMT_Games .
}
```

- dbpedia:Greenwich_Mean_Time
- dbpedia:Geometric_measure_theory
- dbpedia:Giant_Magellan_Telescope
- …

Superlatives

What is the highest mountain?

```
SELECT DISTINCT ?uri
WHERE {
    ?uri rdf:type dbpedia-owl:Mountain  .
    ?uri dbpedia-owl:elevation ?e .
}
ORDER BY DESC(?e)
OFFSET 0 LIMIT 1
```

Superlatives

What is the second highest mountain?

```
SELECT DISTINCT ?uri
WHERE {
    ?uri rdf:type dbpedia-owl:Mountain .
    ?uri dbpedia-owl:elevation ?e .
}
ORDER BY DESC(?e)
OFFSET 1 LIMIT 1
```

Superlatives

Which mountain is the highest after the Annapurna?

```
SELECT DISTINCT ?uri
WHERE {
    ?uri rdf:type dbpedia-owl:Mountain .
    ?uri dbpedia-owl:elevation ?e1 .
    dbpedia:Annapurna dbpedia-owl:elevation ?e2 .
    FILTER (?e1 < ?e2) .
}
ORDER BY DESC(?e1)
OFFSET 0 LIMIT 1
```

Granularity

What is the largest city in South Korea?

```
SELECT DISTINCT ?uri
WHERE {
    onto:South_Korea onto:largestCity ?uri .
}
```

Granularity

What is the largest city in South Korea?

```
SELECT DISTINCT ?uri
WHERE {
    onto:South_Korea onto:largestCity ?uri .
}
```

```
SELECT DISTINCT ?uri
WHERE {
    ?uri rdf:type onto:City .
    ?uri onto:locatedIn onto:South_Korea .
    ?uri onto:populationTotal ?pop .
}
ORDER BY DESC(?pop)
OFFSET 0 LIMIT 1
```

Granularity (and lexical knowledge)

Give me the grandchildren of Bruce Lee.

```
SELECT DISTINCT ?uri
WHERE {
        dbpedia:Bruce_Lee dbpedia-owl:child ?child .
        ?child dbpedia-owl:child ?uri .
}
```

Granularity

Give me a list of all lakes in Denmark.

```
SELECT DISTINCT ?uri
WHERE {
        { ?uri rdf:type dbpedia-owl:Lake .
          ?uri dbpedia-owl:country dbpedia:Denmark . }
        UNION
        { ?uri rdf:type yago:LakesOfDenmark . }
}
```

To count or not to count?

How many children did Benjamin Franklin have?

```
SELECT COUNT(DISTINCT ?uri)
WHERE {
    dbpedia:Benjamin_Franklin dbpedia-owl:child ?uri .
}
```

To count or not to count?

How many children did Benjamin Franklin have?

```
SELECT COUNT(DISTINCT ?uri)
WHERE {
    dbpedia:Benjamin_Franklin dbpedia-owl:child ?uri .
}
```

How many inhabitants does Maribor have?

```
SELECT DISTINCT ?num
WHERE {
    dbpedia:Maribor dbpedia-owl:populationTotal ?num .
}
```

Adjectives

Give me all Danish movies.

```
SELECT DISTINCT ?uri
WHERE {
     ?uri rdf:type dbpedia-owl:Film .
     ?uri dbpedia-owl:country dbpedia:Denmark .
}
```

Adjectives

Give me all communist countries.

```
SELECT DISTINCT ?uri
WHERE {
     ?uri rdf:type dbpedia-owl:Country .
     ?uri dbpedia-owl:governmentType dbpedia:Communism .
}
```

Adjectives

Give me all animals that are extinct.

```
SELECT DISTINCT ?uri
WHERE {
     ?uri rdf:type dbpedia-owl:Animal .
     ?uri dbpedia-owl:conservationStatus 'EX' .
}
```

Comparisons

Was the Cuban Missile Crisis earlier than the Bay of Pigs Invasion?

```
ASK
WHERE {
    dbpedia:Cuban_missile_crisis dbpedia-owl:date ?x .
    dbpedia:Bay_of_Pigs_Invasion dbpedia-owl:date ?y .
    FILTER (?x < ?y)
}
```

More of the same

Which U.S. states are in the same time zone as Utah?

```
SELECT DISTINCT ?uri
WHERE {
    ?uri rdf:type yago:StatesOfTheUnitedStates .
    ?uri dbp:timezone ?x .
    dbpedia:Utah dbp:timezone ?x .
    FILTER (?uri != dbpedia:Utah)
}
```

Temporal anaphora

Give me all world heritage sites designated within the past two years.

```
SELECT DISTINCT ?uri
WHERE {
        ?uri rdf:type dbpedia-owl:WorldHeritageSite .
      { ?uri dbp:year '2013'^^xsd:integer . }
      UNION
      { ?uri dbp:year '2014'^^xsd:integer . }
}
```

Missing or incomplete data

- Who invented the zipper?
- What is the average temperature on Hawaii?
- How many gold medals did Michael Phelps win at the 2008 Olympics?

Outline

1. The challenges
   - QALD
   - OKBQA
   - BioASQ
2. From questions to answers (via templates)
3. OKBQA ideas

Benchmark

NLQ400

http://www.okbqa.org

Dataset: Wikipedia and DBpedia

- questions (en,ko)
- answers (strings)

Reasoning

Who was the last princess of Joseon?

Jeopardy!-like questions

This person was born a princess between Emperor Gojong and the noblewoman Yang, and was known as the last princess of Joseon. She was forcibly taken to Japan in 1925, and entered into a marriage of convenience at the age of 19 with the head of the family of the feudal lord of Tsushima. She led a tragic life until she returned to Korea in 1962. Who was this person?

Anaphora

Former South Korean president Kim Dae-jung was the first Korean to win the Nobel Peace Prize for his efforts to establish democracy and human rights in Korea and East Asia. In what year did he win the prize?

Superfluous information

Former South Korean president Kim Dae-jung was the first Korean to win the Nobel Peace Prize for his efforts to establish democracy and human rights in Korea and East Asia. In what year did he win the prize?

Outline

1. The challenges
   - QALD
   - OKBQA
   - BioASQ
2. From questions to answers (via templates)
3. OKBQA ideas

Benchmark

BioASQ

`http://bioasq.org`

Dataset: PubMed Central articles

- questions (en)
- answers (strings and text snippets)

Non-factoid questions

- What is the treatment of infectious mononucleosis?
- What is the role of TRH in hypertension?
- How do histone methyltransferases cause histone modification?

Outline

1. The challenges
   - QALD
   - OKBQA
   - BioASQ
2. From questions to answers (via templates)
3. OKBQA ideas

Which mountains are higher than the Nanga Parbat?

```
PREFIX dbo: <http://dbpedia.org/ontology/>
PREFIX res: <http://dbpedia.org/resource/>

SELECT DISTINCT ?uri WHERE {
  ?uri a dbo:Mountain .
  res:Nanga_Parbat dbo:elevation ?e1 .
  ?uri dbo:elevation ?e2 .
  FILTER (?e2 > ?e1) .
}
```

Eight mountains are higher than the Nanga Parbat: Mount Everest, Makalu, K2, ...

```
resource:Nanga_Parbat a dbpedia:Mountain .
resource:Nanga_Parbat dbpedia:elevation '8126' .
resource:K2 a dbpedia:Mountain .
resource:K2 dbpedia:elevation '8611' .
resource:Annapurna a dbpedia:Mountain .
resource:Annapurna dbpedia:elevation '8091' .
resource:Mount_Everest a dbpedia:Mountain .
resource:Mount_Everest dbpedia:elevation '8848' .
resource:Amsterdam a dbpedia:City .
resource:Amsterdam dbpedia:elevation '2' .
...
```

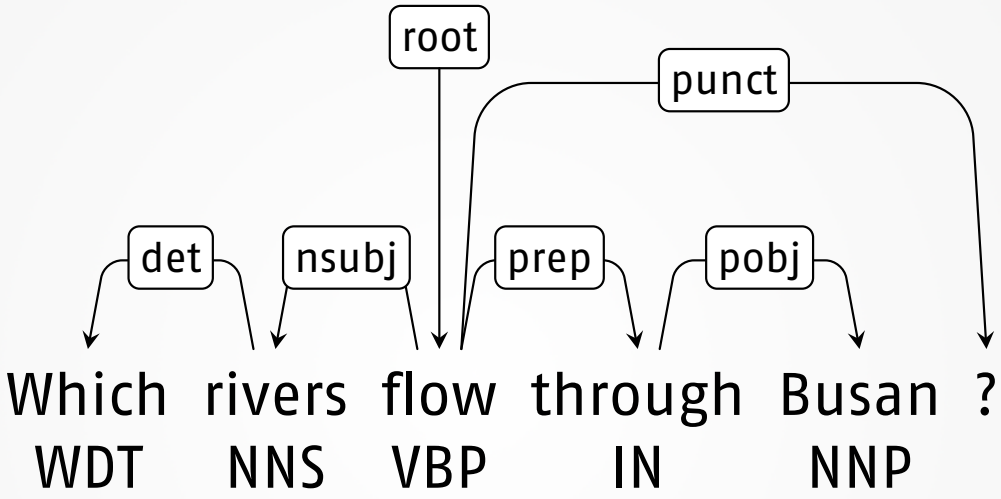
```
<http://dbpedia.org/resource/Mount_Everest>
<http://dbpedia.org/resource/Makalu>
<http://dbpedia.org/resource/K2>
...
```

Two-stage approaches

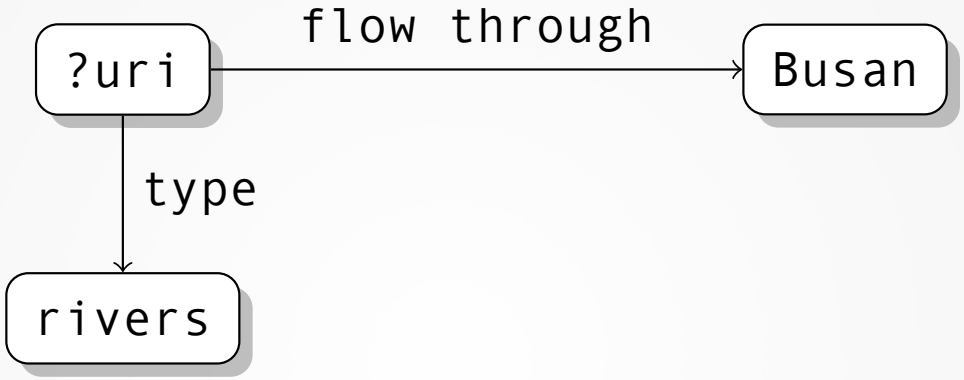
Example

Which rivers flow through Busan?

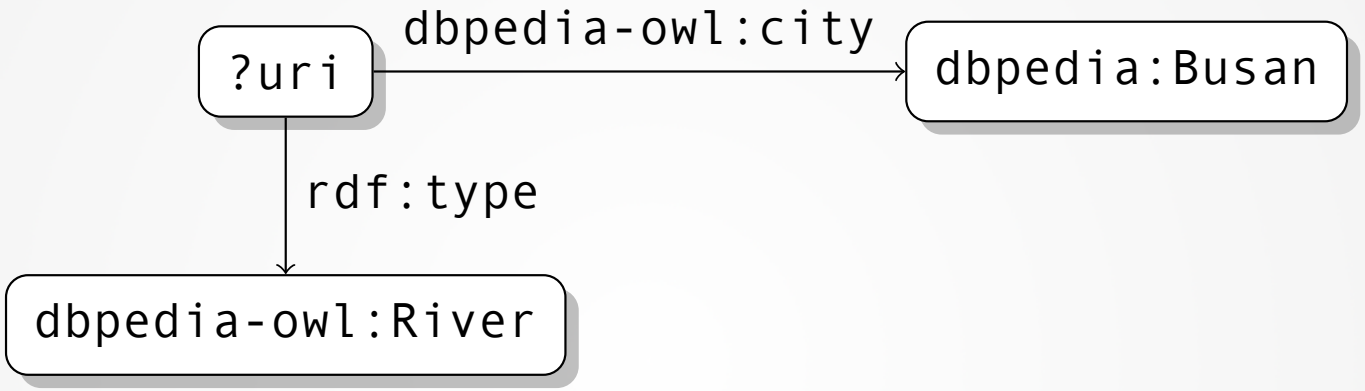
Example


root
punct
det
nsubj
prep
pobj
Which rivers flow through Busan ?
WDT NNS VBP IN NNP

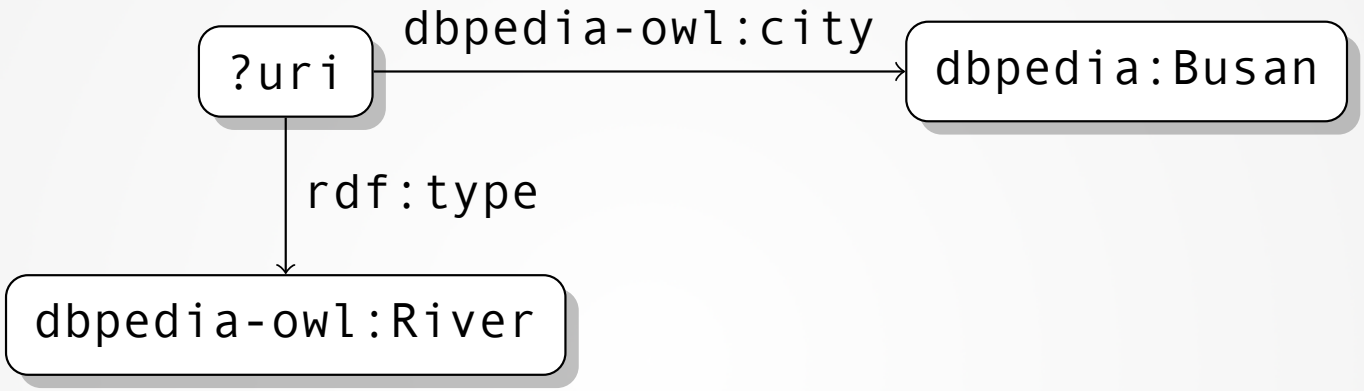
Example



Example


?uri
dbpedia-owl:city
dbpedia:Busan
rdf:type
dbpedia-owl:River

Example




```
SELECT ?uri WHERE {
       ?uri rdf:type dbpedia-owl:River .
       ?uri dbpedia-owl:city dbpedia:Busan .
}
```

Template-based question answering

Goal: NL question $\rightarrow$ SPARQL query

1. Template generation: NL question $\rightarrow$ Template
   (schema-agnostic, i.e. dataset-indepedent)
2. Instantiation: Template $\rightarrow$ SPARQL query
   (schema-specific, i.e. dataset-dependent)

Example

Which rivers flow through Busan?

```
SELECT DISTINCT ?uri
WHERE {
        ?uri rdf:type ?c .
        ?uri ?p ?r .
}
```

- ?c CLASS river
- ?p PROPERTY flow through
- ?r RESOURCE Busan

Example

Which rivers flow through Busan?

```
SELECT DISTINCT ?uri
WHERE {
        ?uri rdf:type ?c .
        ?uri ?p ?r .
}
```

- ?c CLASS river → dbpedia-owl:River
- ?p PROPERTY flow through → dbpedia-owl:city
- ?r RESOURCE Busan → dbpedia:Busan

Example

Which rivers flow through Busan?

```
SELECT DISTINCT ?uri
WHERE {
        ?uri rdf:type dbpedia-owl:River .
        ?uri dbpedia-owl:city dbpedia:Busan .
}
```

Example

Which rivers flow through Busan?

```
SELECT DISTINCT ?uri
WHERE {
        ?uri rdf:type dbpedia-owl:River .
        ?uri dbpedia-owl:city dbpedia:Busan .
}
```

uri
http://dbpedia.org/resource/West_Nakdong_River

Template generation

https://github.com/okbqa/templator

NL question

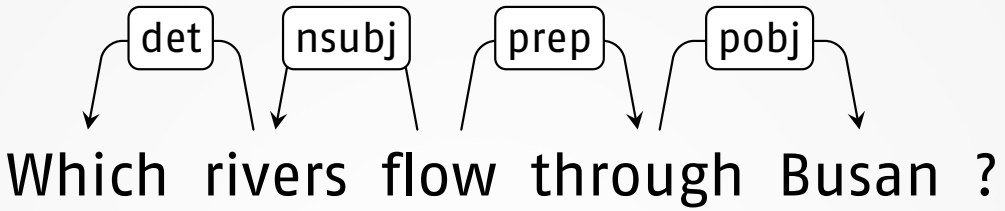
→ dependency graph + semantic role labels

→ query template

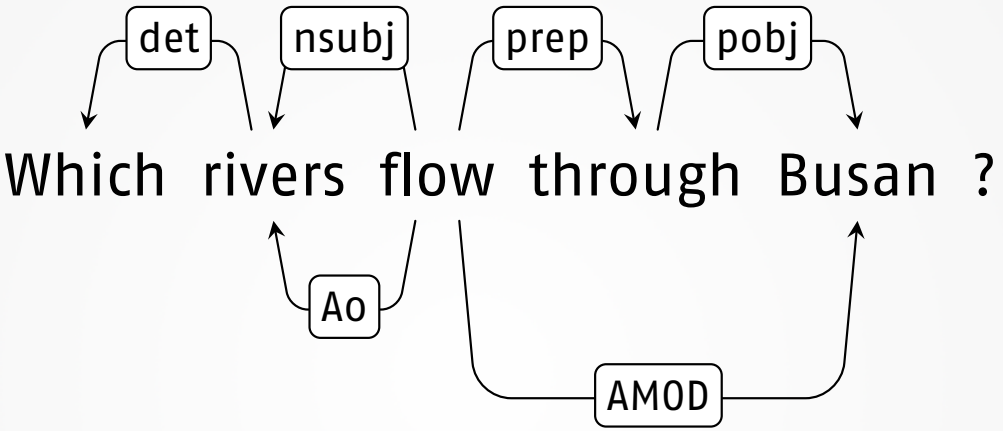
Example

Which rivers flow through Busan?

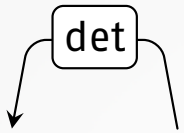
Example



Example



Example



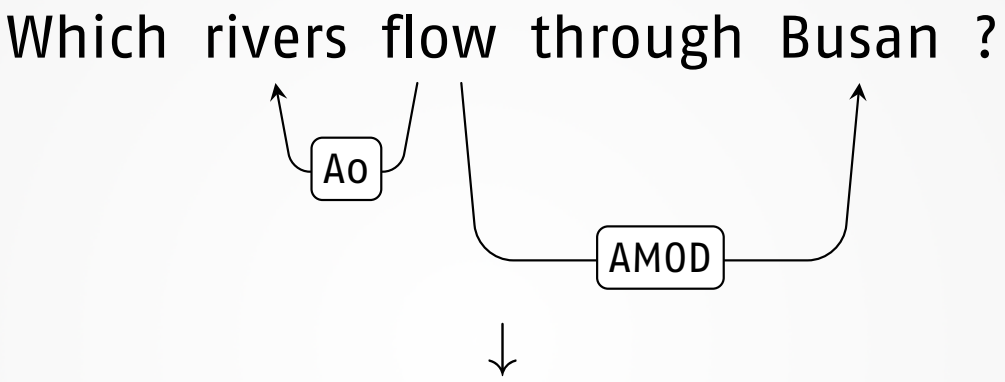
Which rivers flow through Busan ?

↓

```
?x rdf:type ?c .
```

- ?c=rivers

Example



?x ?p ?y .

- ?p=flow
- ?y=Busan

Example

```
SELECT ?x WHERE
{
        ?x rdf:type ?c .
        ?x ?p ?r .
}
```

- ?c CLASS river
- ?p PROPERTY flow
- ?r RESOURCE Busan

Dealing with structural varieties

Template rewriting

Example

```
SELECT ?x WHERE
{
        ?x rdf:type ?c .
        ?x ?p ?r .
}
```

- ?c CLASS river
- ?p PROPERTY flow
- ?r RESOURCE Busan

Example

```
SELECT ?x WHERE
{
        ?x rdf:type ?c .
        ?x ?p ?r .
}
```

- ?c CLASS river
- ?p PROPERTY flow
- ?r RESOURCE Busan

Example

```
SELECT ?x WHERE
{
        ?x rdf:type ?c .
        ?r ?p ?x .
}
```

- ?c CLASS river
- ?p PROPERTY flow
- ?r RESOURCE Busan

Example

```
SELECT ?x WHERE
{
        ?x rdf:type ?c .
        ?x ?p ?r .
}
```

- ?c CLASS river
- ?p PROPERTY flow
- ?r RESOURCE Busan

Example

```
SELECT ?x WHERE
{
        ?r ?p ?x .
}
```

- ?p PROPERTY river
- ?r RESOURCE Busan

From templates to queries

1. Discovery of potential slot fillers
   - DBpedia Spotlight, FOX
   - Lexical knowledge
     `https://github.com/ag-sc/lemon.dbpedia`
     `http://dblexipedia.org`
2. Disambiguation
   - Agdistis

Outline

1. The challenges
   - QALD
   - OKBQA
   - BioASQ
2. From questions to answers (via templates)
3. OKBQA ideas

The template generation module

- includes an English and Korean dependency parser
  - TODO REST service for the latter still running?

The template generation module

- includes an English and Korean dependency parser
  - TODO REST service for the latter still running?
- implements a basic rule set for mapping dependency/SRL graphs to templates
  - TODO check results for QALD (and OKBQA) benchmark
  - TODO extend English rule set
  - TODO translate QALD questions into Korean
  - TODO extend Korean rule set

The template generation module

- includes an English and Korean dependency parser
  - TODO REST service for the latter still running?
- implements a basic rule set for mapping dependency/SRL graphs to templates
  - TODO check results for QALD (and OKBQA) benchmark
  - TODO extend English rule set
  - TODO translate QALD questions into Korean
  - TODO extend Korean rule set
- works for single questions as well as more text-like questions
  - TODO remove irrelevant information (e.g. unconnected triples) from templates

The template generation module

- includes an English and Korean dependency parser
  - TODO REST service for the latter still running?
- implements a basic rule set for mapping dependency/SRL graphs to templates
  - TODO check results for QALD (and OKBQA) benchmark
  - TODO extend English rule set
  - TODO translate QALD questions into Korean
  - TODO extend Korean rule set
- works for single questions as well as more text-like questions
  - TODO remove irrelevant information (e.g. unconnected triples) from templates
- TODO template rewriting (with appropriate scoring)

Template instantiation

- TODO exploit ontology lexicon for URI lookup