Question Answering and Chatbots

2nd Practical exercise – Working with Natural Language

Aleksandr Perevalov

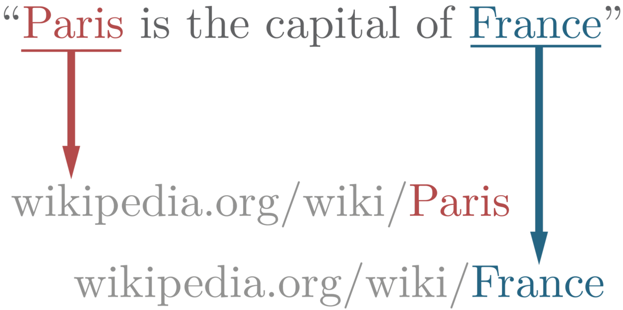
aleksandr.perevalov@hs-anhalt.de

October 4, 2021

Hochschule Anhalt
Anhalt University of Applied Sciences

Named Entity Linking (NEL)

NEL – is the task of determining unique identity to entities (people, locations, songs, etc.) mentioned in text.



"Paris is the capital of France"

wikipedia.org/wiki/Paris

wikipedia.org/wiki/France

Named Entity Linking (NEL)

NEL consists of several steps:

Named Entity Linking (NEL)

NEL consists of several steps:

1. Named Entity Recognition (NER) – spot text spans in that contain Named Entity label;

Named Entity Linking (NEL)

NEL consists of several steps:

1. Named Entity Recognition (NER) – spot text spans in that contain Named Entity label;
2. Named entity search – given the text form, find possible candidates in database;

Named Entity Linking (NEL)

NEL consists of several steps:

1. Named Entity Recognition (NER) – spot text spans in that contain Named Entity label;
2. Named entity search – given the text form, find possible candidates in database;
3. Candidates ranking – order candidates according to a parameter e.g. PageRank, Views per month.

Named Entity Linking (NEL)

NEL consists of several steps:

1. Named Entity Recognition (NER) – spot text spans in that contain Named Entity label;
2. Named entity search – given the text form, find possible candidates in database;
3. Candidates ranking – order candidates according to a parameter e.g. PageRank, Views per month.

Sometimes 1st step is not included.

Named Entity Linking (NEL)

NEL consists of several steps:

1. Named Entity Recognition (NER) – spot text spans in that contain Named Entity label;
2. Named entity search – given the text form, find possible candidates in database;
3. Candidates ranking – order candidates according to a parameter e.g. PageRank, Views per month.

Sometimes 1st step is not included.
Tools: DBpedia Spotlight, TagMe, AGDISTIS, etc.

Text preprocessing

- Tokenizaiton – dividing text into tokens (words);

Text preprocessing

- Tokenizaiton – dividing text into tokens (words);
- Stop words removing (but, how, or, etc.);

Text preprocessing

- Tokenizaiton – dividing text into tokens (words);
- Stop words removing (but, how, or, etc.);
- Special characters removing (!@#$%);

Text preprocessing

- Tokenizaiton – dividing text into tokens (words);
- Stop words removing (but, how, or, etc.);
- Special characters removing (!@#$%);
- Lemmatization – “changing” → “change”;

Text preprocessing

- Tokenizaiton – dividing text into tokens (words);
- Stop words removing (but, how, or, etc.);
- Special characters removing (!@#$%);
- Lemmatization – "changing" → "change";
- Stemming – "changing" → "chang";

Text preprocessing

- Tokenizaiton – dividing text into tokens (words);
- Stop words removing (but, how, or, etc.);
- Special characters removing (!@#$%);
- Lemmatization – "changing" → "change";
- Stemming – "changing" → "chang";

We don't need this if we use DNNs, for example BERT. Because text preprocessing doesn't affect the model quality.

Any questions?

Exercise 2

Task – depending on your exercise **variant** manually write a script, which takes for input a list of questions and outputs for each question:

1. preprocessed question (tokenization, stopwords and special characters removing, lemmatization);
2. a dictionary `'uri':'text'` of linked named entities from the question (use DBpedia Spotlight).

Randomly pick 5 questions from your variant and do manual analysis (see GitHub). **The variants** are available in my **GitHub repository**.
Also, in the repository, you can find the the format of the script output.
Link to the repo: https://github.com/Perevalov/qa_chatbots_exercises

Exercise 2

To submit your solution, please, use corresponding form in the **Moodle**. If you don't have an access to the Moodle, then use e-mail.

Let's do the exercise. Ask me if you have a question.

Plan for the Exercise 3: Question classification (Rule-Based/ML)

Task for TODAY: think/draw/code a text classification algorithm for **relation (predicate) prediction** you can use rule-based or machine learning approach.

Question: "In what music genre does Boris Brejcha making his songs?" → Relation: `dbo:genre`.

Example dataset is available in the GitHub repo.

The complete task will be published in 1-2 days.

1. SPARQL;
2. **Work with Natural Language (NER)**;
3. Questions classification (ML is possible);
4. Web-Service, Front-end;
5. Simple QA system;
6. Tests for QA system;
7. Docker;
8. Qanary Framework;
9. ...