biotec
Biotechnology Center TU Dresden

Intelligente Systeme

WS 2021/2022

TECHNISCHE
UNIVERSITÄT
DRESDEN

Prof. Michael Schroeder
Contact: is_teachers@lists.biotec.tu-dresden.de

IS Group Project Guidelines

Submission date: 26.01.2022

Group Assignment

- This project is entirely optional and a bonus.
- The project must be coded in Python.
- Students must form groups of maximum 5 people.
- In Opal, each group will have a 'Folder' to share files and a 'Forum' to communicate among themselves. The Forum is for communicating with only group members and not teachers.

Project Ideas

The groups should implement text mining tasks on scientific articles retrieved from PubMed. The project ideas are listed below. Each group picks one of them to implement in their project. Please notice that each one can be picked by a maximum of 3 groups.

1. **Journal Prediction**
   Predict from title and abstract, whether an article is published in a certain journal such as Bioinformatics, Nature, Cell, Science, Oncotarget, Plos One, etc.

2. **Impact Factor Prediction**
   Use machine learning to predict from title and abstract whether an article is published in a journal with a high-impact factor.

3. **Sentence Boundaries**
   Use machine learning to classify sentence boundaries. Compare against NLTK sent_tokenize. Compute precision and recall of your classifier, NLTK, and a simple dot-space-uppercase classifier.

4. **Q&A**
   Extract questions and answers from abstracts.

5. **Letter Variations in First Names**
   Retrieve all first names and cluster them. Derive a substitution matrix for common letter variations. A substitution matrix is a matrix that describes the rate at which a letter is replaced with another letter (e.g. d-t,p-b, etc.).

6. **Definition Extraction**
   Extract definitions from abstracts. Search for patterns such as "is a" and "such as". Can you build a hierarchy such as A is a B, B is a C, D is a C? Cluster textual definitions and find a representative one, compare them against definitions in Wikipedia.

7. **Profile of a Scientist**
   Generate a profile of a scientist. Extract all affiliations and cluster them, extract representative affiliations and together with the year of publication build a “CV” of the scientist.

8. **Collaboration Network**
   Build a network of co-occurring cities in publication affiliations. Visualise the network and find interesting relationships. Which are the strongest co-occurrences across countries? Does geography play a role? Use relative optionally log-odds ratios in your analysis. Draw the network on a map.

9. **Automatic Sentence Completion**
   Automatic sentence completion (such as Gmail providing suggestions to complete your sentence while writing emails). Learn how to extend sentences from abstracts.

10. **Drug-Disease Network**
    Extract a drug-disease network (e.g. co-occurrences of drug and disease names in positive sentences).

Submission Guidelines

- Students must write a project report which describes their project (e.g. their data set, data cleaning/preparation stages, methods or machine learning models that they implemented, results etc.). The report can be in PDF format or Google Colab. Using Google Colab is preferable. However, if you do not use Google Colab, please include your Python codes in the appendix of the report in PDF format.

- Students must prepare a short presentation video (approx. 5-6 min.). The video must demonstrate the proposed solution and show important pieces of their code (or the complete code if relevant). The video should address the following questions:

  - What was the problem to be solved?
  - What is the proposed solution?
  - Did you use any concepts or algorithms from the lecture?
  - Did you use any external algorithm or data to solve the problem?
  - Which python libraries did you use and which functionality do they provide?
  - Has someone done something similar to your work? If yes, which literature you based your work on?
  - What are the advantages/disadvantages of your proposed solution?

- Please upload your video and your report in ZIH own cloud and provide us with a PDF containing your information (group number, names, and student numbers of the member of groups) as well as the link to your video and Google Colab / PDF report). Please send us this PDF to is_teachers@lists.biotec.tu-dresden.de by the submission date.

- For general questions regarding the project, please use the OPAL forum, and for specific questions regarding your project, contact is_teachers@lists.biotec.tu-dresden.de.

PubMed Query Script

The script below retrieves articles from PubMed. Define your query in PubMed syntax. You can use [ti] for title, [tiab] for title or abstract. For more options, please check PubMed.

- Here are some examples for **myQuery** in the code below:
  - clustering[ti] algorithm
  - Nature[tiab]
- For the code to run you need to replace **myEmail** with your personal mail address. In case of excessive usage of the E-utilities, this gives NCBI the chance to contact you before blocking access for you.
- With **maxPapers** in the code below, you can limit the number of papers retrieved.

```python
#install and import Entrez and Medline first
try:
  from Bio import Entrez,Medline
except:
  !pip install Bio
  from Bio import Entrez,Medline


def getPapers(myQuery, maxPapers, myEmail ="xxxxx@xxxxxxxx.xx"):
  # Get articles from PubMed
  Entrez.email =myEmail
  record =Entrez.read(Entrez.esearch(db="pubmed", term=myQuery, retmax=maxPapers))
  idlist =record["IdList"]
  print("\nThere are %d records for %s."%(len(idlist), myQuery.strip()))
  records =Medline.parse(Entrez.efetch(db="pubmed", id=idlist, rettype="medline",
                                        retmode="text"))
  # records is iterable, which means that it can be consumed only once.
  # Converting it to a list, makes it permanently accessible.
  return list(records)

myQuery ="your query string"+"[tiab]" #query in title and abstract
maxPapers =1000 #limit the number of papers retrieved
records =getPapers(myQuery, maxPapers)
```