Lab 4 - Topic Modeling and Named Entity Recognition

Deadline: March 13, 2023.

The last lab of the course! Here, we will cover the book's final chapters (don't worry, not *all* the pages). As we will soon enter the course's project phase, there will not be an implementation part related to the smart keyboard. Instead, you are encouraged to think about how you can apply the techniques you have learned so far in a larger project.

Setup

For the topic modeling task, you will need Gensim installed. You may also want to visualize using pyLDAvis. You can install these with:

```
pip install gensim
pip install pyldavis
```

Please refer to the documentation for more information:

- https://radimrehurek.com/gensim/auto_examples/index.html#documentation
- https://pyldavis.readthedocs.io/en/latest/readme.html#installation

Topic Analysis

(pages 304-308, 325-342)

1. What is the difference between supervised and unsupervised learning? Discuss some benefits and issues for each approach in the context of topic analysis.
2. You are given a corpus of 1 million documents and a vocabulary of 100,000 words. List some problems that you may encounter when using TF-IDF vectorization for clustering this corpus, and explain how you would deal with them.
3. Metrics are essential when dealing with machine learning. However, regarding unsupervised clustering (e.g., of topics), we cannot use the typical precision, recall, and f-measure metrics. What are the alternatives for this task?

Topic Modeling

(pages 349-356, 371-377)

Given the five sentences:

> "Macrosoft announces a new Something Pro laptop with a detachable keyboard."
>
> "Melon Tusk unveils plans for a new spacecraft that could take humans to Mars."
>
> "The top-grossing movie of the year Ramvel Retaliators."
>
> "Geeglo releases a new version of its Cyborg operating system."
>
> "Fletnix announces a new series from the creators of Thinger Strangs."

1. How would *you* (without programming) assign the listed sentences to separate topics? Explain your chain of thought.
2. Two algorithms for topic discovery are Latent Semantic Indexing (LSI) and Latent Dirichlet Allocation (LDA)

- What preprocessing steps should we consider before implementing these algorithms?

- Both require the user to specify the number of topic clusters. How can we *automatically* detect a reasonable number of topics? This is related to metrics for unsupervised clustering.

3. Using the corpus provided in Lab 3, `amazon_appliances_reviews`, implement an LSI and LDA model using Gensim. Specify 5 topics and print out the top 10 words for each topic. The package `pyLDAvis` will help you to visualize the topics!

- Which algorithm do you think is more accurate? Why?
- How do the topics compare to your interpretations? In your opinion, are there more or less than 5 *actual* topics?

Named Entity Recognition

(pages 384-392, 403-415)

1. In Lab 3, you learned about noun phrases. Noun phrases are typically named entities, such as "The quick brown fox" or "Mount Everest". Give examples of named entity categories that are typically *not* noun phrases.

2. Disambiguating (or entity linking) named entities is a crucial task to applications of NER and considers the problem of assigning an identifier to each entity, i.e., *linking* relevant entities together. The disambiguation process often incorporates external knowledge (*knowledge bases*).

   Consider the sentences:

   > "I ate an apple in New York"
   >
   > "New York Times wrote an article about Apple"
   >
   > "New York is also known as the Big Apple"

   How would you tackle the task of distinguishing the entities found here? Give a rough step-by-step explanation, either in text or by pseudo-code — no implementation required.

Finishing the labs

As a final part, please answer the following: (short answers are fine!)

- Was the setup of the lab environment too difficult?
- Was the keyboard implementation of the labs useful or any fun at all? Scrap it?
- Do you prefer the labs to be theoretical or practical?
- Any other comments?

Deliveries:

- A pdf or jupyter notebook with your answers and results.
- Supplementary code used to solve the exercises (if not included in the notebook)