Exercise 6 - Topic Modeling

Topics and Trends in NLP

Deadlines

The deadline for Exercise 6 is **15.12.2024, 23:59**
The deadline for the peer review is **23.12.2024, 23:59.**

Learning goals

This exercise is about topic modeling, more specifically about Latent Dirichlet Allocation (LDA) and topic modeling based on pre-trained language models (PLM). By completing this exercise, you should…

- understand how topic modeling is used as a text-mining tool.
- be able to apply LDA and PLM-based topic models.

Please keep in mind that you can always consult and use the exercise forum if you get stuck (note that we have a separate forum for the exercises).

Deliverables

We encourage you to use Colab to develop your notebooks since there you have access to GPU time. **After finishing the assignment, download your notebook as a .ipynb file.** That way your reviewers can see your already executed code.

Please hand in your code in a notebook and name it as follows. The notebook should contain a lab report at the end in a markdown cell.

- ex06_tm.ipynb

The lab report should contain a description of the approaches you have used to solve this exercise. Please also include the results. **For sections in green, we expect a more detailed statement (e.g., discussing issues you've encountered).**
Please note that your peers need to be able to run your code. Otherwise, you will not be able to obtain the maximum number of points. Also, please DO NOT submit the data files!

Data

For this exercise, you will work with the dblp: a large database containing metadata on computer science (and related) publications. You will perform topic modeling on the titles (**not on the full text itself!**) of computer science publications to detect important topics and trends in the field.

Given

For the exercise, you are given this notebook. It already contains code for downloading the dataset, and instructions on how to proceed.

Part 1: Topic Modeling using LDA

The given notebook limits the number of titles to a reasonable amount and divides publications into three time periods: before 1990, from 1990 to 2009, and from 2010 onwards. Perform topic modeling on the three time periods.

Please experiment with varying numbers of topics (starting with at least five) and with different ways of preprocessing.

For each period, assign a name to each generated topic based on the topic's top words. List all topic names in your report. If a topic is incoherent to the degree that no common theme is detectable, you can just mark it as incoherent (i.e., no need to name a topic that does not exist).

- **Do the topics make sense to you? Are they coherent? Do you observe trends across different time periods? Discuss in 4-6 sentences.**

Part 2: Topic Modeling using Combined Topic Models (CTMs)

Bianchi et al. 2021 propose a topic modeling approach that makes use of pre-trained language models such as BERT. The authors provide a simple colab tutorial demonstrating how to use the CTM library that implements their method.

Again, perform topic modeling for the 3 time periods. This time using the CTMs. Use the same number of topics as before. You can copy and adjust code from the author's tutorial. We suggest using "sentence-transformers/paraphrase-mpnet-base-v2" for CTM.

- **Again: Assign a name to each topic based on the topic's top words (for each period). List all topic names in your report.**
- **Bianchi et al. 2021 claim that their approach produces more coherent topics than previous methods. Let's test this claim by comparing the coherence of the topics produced by CTM with the topics produced by LDA. Describe your observations in 3-4 sentences.**
- **Do the two models generate similar topics? Can you discover the same temporal trends (if there are any)? Discuss in 5-6 sentences.**
- **Can you suggest an alternate model apart from paraphrase-mpnet-base-v2? What could be some of the possible advantages and disadvantages of using an alternate model? Hint: Look at some of the models here. Note: You do not need to execute the code for an alternate model.**

Submission & Peer Review Guidelines:

Peer Reviews will be carried out on OLAT.
As soon as the deadline for handing in the exercise expires, you will have time to review the submissions of your peers. You need to do **2 reviews** to get the maximum points for this exercise.
Here are some additional rules:

- **All file submissions are anonymous (for peer review purposes): Do not write your name into the Python scripts.**
- **ONLY ONE** member of each team submits on OLAT.
- Please submit a zip folder containing all the deliverables.

Groups & Peer Reviews:

- You should work in the same group (pair) as for exercises 1 to 5. Each member should contribute equally!
- Only **one team member submits the solutions**.
- Only the submitting team member will have access to the peer review, however, you should distribute the workload evenly!
- If you do not submit 2 reviews, the maximum number of points you can achieve is 0.75 (out of 1).
- Please use full sentences when giving feedback.

- Provide constructive and helpful comments to your peers! If you criticise something, you should also be able to provide actionable and realistic suggestions for how something can be improved.
- Please answer all the review questions of the peer review.