ΥΣ19 Artificial Intelligence II (Deep Learning for Natural Language Processing)
Fall Semester 2022-2023
Homework 4
25% of the course mark
Announced: January 13, 2023 Due: March 12, 2023 at 23:59

1. Develop a sentiment classifier for the dataset `imdb-reviews.csv` of Homeworks 1, 2 and 3 by fine-tuning the pretrained BERT-base model available on Hugging Face.[1] You have to compute precision, recall and F1 for each class.

   (5/10 marks)

2. In this and the following question you will develop a question answering engine for simple questions over the knowledge graph Wikidata.[2]

   Start by reading the paper

   > Denis Lukovnikov, Asja Fischer and Jens Lehmann. *Pretrained Transformers for Simple Question Answering over Knowledge Graphs.* CoRR abs/2001.11985 (2020). Available at https://arxiv.org/pdf/2001.11985.pdf.

   This paper discusses techniques for creating such a question answering engine over the knowledge graph Freebase.[3] After you read the paper, implement the techniques of Section 2 (except 2.4) for entity span prediction and relation prediction.

   The dataset that this paper uses is available at https://github.com/askplatypus/wikidata-simplequestions/tree/master/SimpleQuestions_v2. This dataset has been transformed into a form appropriate for the Wikidata knowledge graph by the authors of the paper

   > Diefenbach, Dennis and Tanon, Thomas and Singh, Kamal and Maret, Pierre. *Question Answering Benchmarks for Wikidata.* ISWC 2017 posters and demos. Available at https://ceur-ws.org/Vol-1963/paper555.pdf.

[1] https://huggingface.co/models

[2] https://www.wikidata.org/wiki/Wikidata:Main_Page

[3] Freebase is knowledge graph built in 2007 by the American company Metaweb. Freebase was bought by Google in 2010 and was used as a basis for the Google knowledge graph. In 2014, Freebase was shutdown by Google and its data were moved to Wikidata. See https://en.wikipedia.org/wiki/Freebase_(database) for more details.

and is available at https://github.com/askplatypus/wikidata-simplequestions.

You will evaluate your implemented modules for entity span prediction and relation prediction, and present the results using a table like Table 1 of the first paper (page 9). Your table will have only one row since you will only be implementing the BERT-based modules.

(5/10 marks)

3. Now you are ready to implement the question answering engine. To do this, you will need to implement:

   - a simple interface that allows a user to pose a natural language question over the knowledge graph Wikidata e.g., “what type of music does david ruffin play?”.
   - a *query builder module* that produces a SPARQL query equivalent to the natural language question posed by the user. The module will utilize the modules you developed in Question 2 and the techniques of Section 2.4 of the paper.
   - a *query executor module* that executes the query built by the query builder module over the Wikidata SPARQL endpoint https://query.wikidata.org/.

   Since most of you will not be familiar with SPARQL, we will cover the details in the tutorial.

   (5 bonus marks)

Note: Your solutions should be implemented in PyTorch. You should hand in: (i) a pdf with a detailed report of your solution for the three questions, including an explanation of the methods you used, an evaluation and comparison of different approaches you tried and citations to relevant literature that you might have used in developing your solutions. (ii) Colab notebooks (ipynb files using https://colab.research.google.com/) containing your code for Questions 1 and 2. (iii) your code for Question 3. Please do not use code that you might find on the Internet since you will be penalized; implement your own solutions.