Exercise 3 – COVID19 papers (CORD-19)

The world is changing as we speak. The COVID-19 epidemic has been causing dramatic changes to the way we live our life. In this exercise we will focus on data related to the COVID-19 and will build unsupervised machine learning models.

The data that you will play with is a big corpus of academic papers, all dealing with COVID-19. The data is part of a worldwide research challenge (i.e., CORD-19), provided by Kaggle and is well described here.

Your mission is divided into two parts:

1. Design a method that will be able to retrieve the K most similar instances to a given instance (instance in our case is an academic paper). In addition, your method is required to retrieve the actual distance found between the K instances and the given instance.
   Your method must use **ONLY** the text data provided in the Kaggle competition. In addition, you **MUST** use, as the primal tool, ANY **compression** application or tool of your choice. Please note that you can plug (as a black-box) the compression application to a learning process, which in turn will act as a "meta" learning algorithm. The compression output will be used as the "distance" measure, based on which the K most similar instances will be returned.
2. You were hired by a research team that focuses on COVID-19 from a data perspective. Your manager asked you to design and build a machine learning solution based on the CORD-19 corpus. Your suggested solution must:
   a. Be unsupervised.
   b. Use the compression method you implemented in Part 1.

   Your manager guides you that your suggested solution might (but not must) be used for: analysis of COVID research papers along time, better understanding of the existing papers in the field, information retrieval of papers.
   Two examples of the valid solutions (both do not use the compression method from Part 1, but can be configured in a way they will):
   c. Papers clustering - A description and implementation is presented here.
   d. Papers similarity- A description and implementation is presented here.

   The exact way to use the compression method from Part 1 is part of the challenge, and left for you to decide. Your manager encourages you to be creative and open minded (after all, he is paying for your work☺).
   You are more than welcome to use get inspiration and ideas from the code posted in the Kaggle platform as part of this challenge.

Few important notes about the data maintained in the Kaggle challenge:

- Currently, there are ~200K full text academic papers in the website. Make sure you use only academic papers with full text (some of the academic papers that appear in the corpus contain only meta-information about the paper). Due to the huge amount of data, **you are expected to use only the latest 20K papers from the corpus**.
- Note that the information regarding the publication date can be found in the 'metadata.csv' file (a zipped file).
- Data is being updated over time. You can download the current data available in the website and treat it as frozen fixed dataset (without updating it).
- Data is organized in a very structured way, but the structure is a bit complex. Take some time to understand the structure of the data. Then, make sure you have preprocessing functions that are able to pull out the required data for your main task.
- Feel free to use Kaggle forums for technical questions as well as for data science related questions. Indeed, you can get inspiration by some of the solutions suggested in the website already.

Submission instructions

- The submission must contain your code (written in Python 3).
- Your code must be submitted in a jupyter notebook format. Indeed you can use an additional .py files for the code to run (including some of the utilities you built). We recommend you to use google colab platform and share with us the final product of your work.
  The code must be documented in a reasonable way and contain explanations throughout the notebook regarding the process you implemented. If you use special Python packages (on top of what Anaconda provides) – please document it well.
- The submission must also contain a short document (**up to 6 pages**, including all figures and tables) describing the work you did and an analysis of your solution.
- Your submission must include both a description of your work as well as results and an analysis of the results.
- The exercise has to be done in pairs. **Only one** of the students has to submit the HW. **Make sure to write your full names and IDs in the header of the PDF file**.
- Exercise deadline 14.4.2022, 23:30.

We will take different elements in consideration when checking your submission. Our focus will be on creativity and the usage of information theoretic considerations when designing your solution. Please submit even a partial solution in order to get a grade.