Advanced ML - Exercise 3 Report

Joel Liurner ID 346243579

Gil Zeevi ID 203909320

Introduction

Based on Covid19 research papers, during the project we investigated and developed a method to:

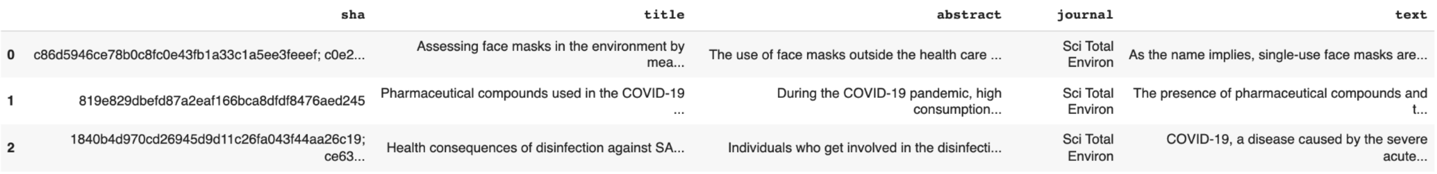
1. Compress and measure distance and similarity between the papers
2. Retrieve the K closest instances to a given paper.
3. Cluster them according to its mutual distance or similarity.

The full code can be found in the notebook repository.

Building the dataset

These are the steps taken during this section in order to build a dataset that we would work on.

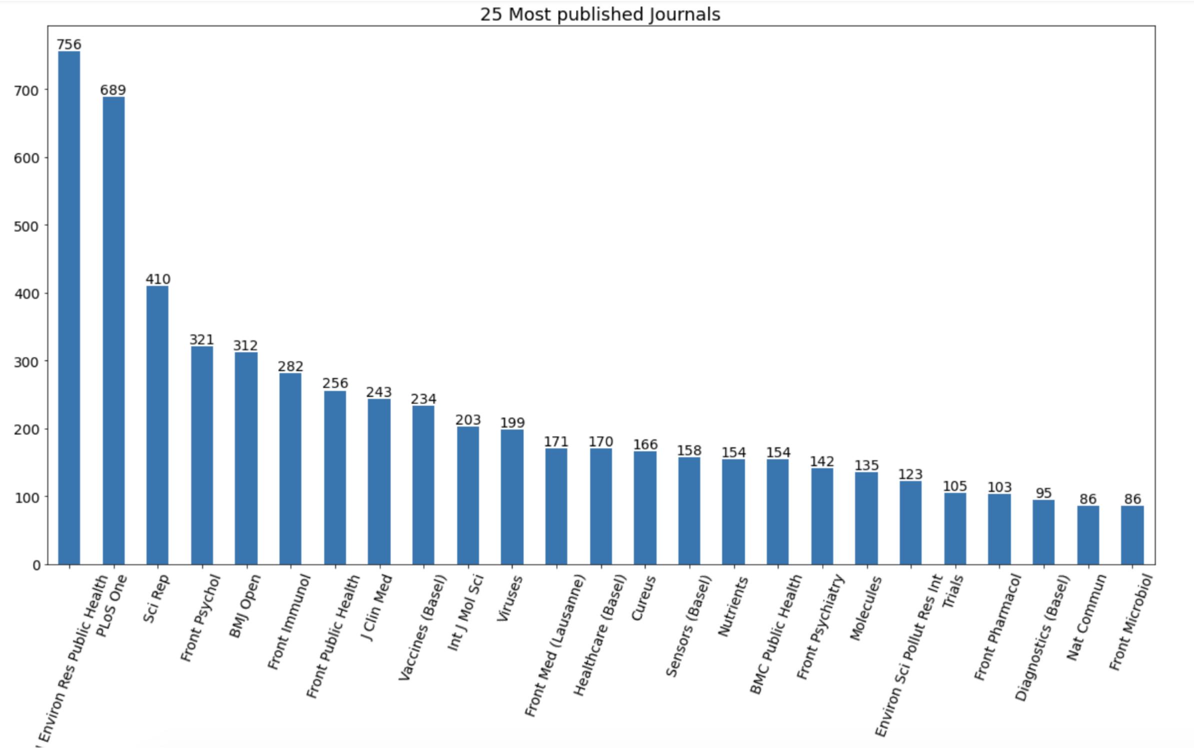
- **Download and extract** the full set of papers from Kaggle on CORD19 challenge
- **Data filtering**: based on the metadata csv that has information about all available papers:
    - Sort by publish date in order to get from the newest ones.
    - Remove inconsistent and corrupted datapoints such as
        - future publish dates
        - missing abstracts or actual file
        - missing sha identifier (unique index hash per paper on the dataset).
- **Paper selection**: chose the 20000 recently published papers and copied the pubMed central file and saved the text from them.
- **Dataset building:** dropped many columns from metadata that were not useful for the project and kept features such as sha identifier, title, text, abstract.



	sha	title	abstract	journal	text
0	c86d5946ce78b0c8fc0e43fb1a33c1a5ee3feeef; c0e2...	Assessing face masks in the environment by mea...	The use of face masks outside the health care ...	Sci Total Environ	As the name implies, single-use face masks are...
1	819e829dbefd87a2eaf166bca8dfdf8476aed245	Pharmaceutical compounds used in the COVID-19 ...	During the COVID-19 pandemic, high consumption...	Sci Total Environ	The presence of pharmaceutical compounds and t...
2	1840b4d970cd26945d9d11c26fa043f44aa26c19; ce63...	Health consequences of disinfection against SA...	Individuals who get involved in the disinfecti...	Sci Total Environ	COVID-19, a disease caused by the severe acute...

Exploratory Data Analysis

We performed some basic EDA in order to understand the data better and be able to work with it. Here are some findings, the distribution of the journals publishing those papers. This might be insightful about the topics each paper presents.



Compression and similarity

Compression

Initially we studied several compression algorithms that could be used, within them: Lempel Ziv Welch (LZW), RAR, LZMA and GZIP. According references and priori studies, all of them are useful for text compression and also vary on performance. The best performing is rar compression. We decided to implement GZIP which is simple and fast. It is good and efficient for text, uses DEFLATE algorithm (it similar of LZW with huffman coding). We used the available python package *gzip* and plugged it as a blackbox to the distance and similarity measures.

We decided to compress the full text of each paper, as it was the most informative data that was available.

Distance and Similarity measures

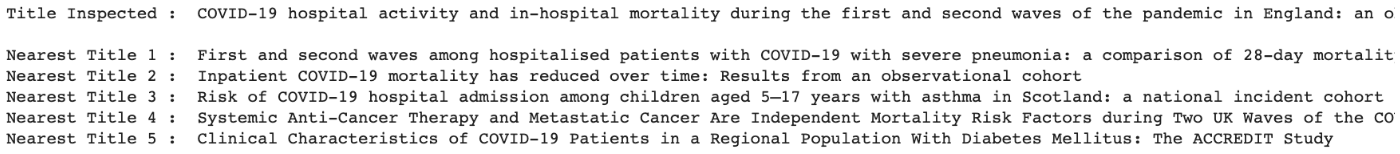
For distance calculation, we based the calculation on **BCN (Best Compression Neighbor** proposed by Dario Benedetto et. al. 2002) which compares a sample to a reference by compressing them together and checking the difference in length to the sample on its own. When engaged, by comparing the sample to all available papers in corpus and retrieving the closest one, what is going on behind the scenes is a process similar to KNN with K=1.

For similarity, we used **NCD (Normalized Compression Distance Li et. al. 2004)**. Given the way LZW, Huffman, hence gzip (and most of compression algorithms) work, It is based on the idea that if two texts compress better together than separately, then they must have many similarities - as with the information in one of them we can compress or describe the other one. It is calculated by normalizing the joint compression of two samples by the compression of them separately.

K most similar instances method

Given the previous setup, we created a method called `N_nearest_papers(paper,corpus,k,method='similarity')` that can retrieve the k nearest papers according to one of the proposed distance measures.

To validate its implementation here we have a random paper and inspect its closest 5 titles.



```
Title Inspected :  COVID-19 hospital activity and in-hospital mortality during the first and second waves of the pandemic in England: an o

Nearest Title 1 :  First and second waves among hospitalised patients with COVID-19 with severe pneumonia: a comparison of 28-day mortalit
Nearest Title 2 :  Inpatient COVID-19 mortality has reduced over time: Results from an observational cohort
Nearest Title 3 :  Risk of COVID-19 hospital admission among children aged 5–17 years with asthma in Scotland: a national incident cohort
Nearest Title 4 :  Systemic Anti-Cancer Therapy and Metastatic Cancer Are Independent Mortality Risk Factors during Two UK Waves of the CO
Nearest Title 5 :  Clinical Characteristics of COVID-19 Patients in a Regional Population With Diabetes Mellitus: The ACCREDIT Study
```

We can interpret they are somehow related in a matter that they talk about: covid waves, mortality and cohort studies.

Research on CORD19 corpus - Part 2

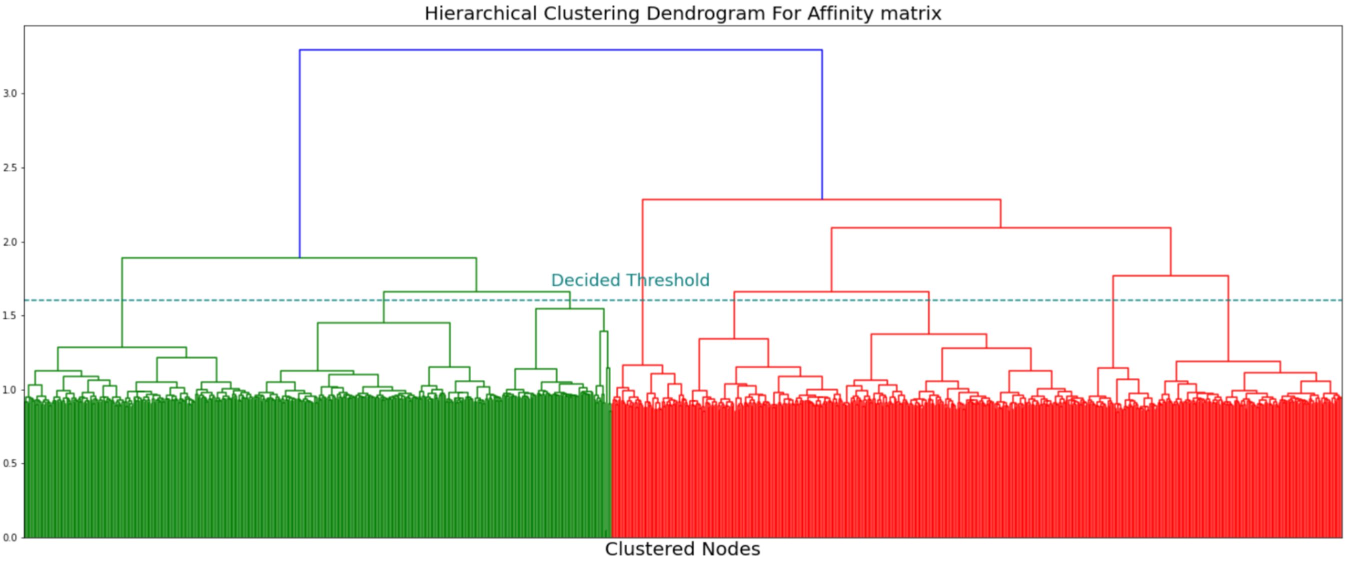
In this section we performed a study on the corpus in order to cluster them. As there was no label on each paper, the task was unsupervised. We based it on the ideas of compression and distance/similarity from previous part and due to limitation in resources of RAM, we reduced the dataset to 1000 random samples within the 20k papers.

Dataset calculated Matrices

After sampling 1000 papers, we calculated two matrices that would be used as base for the clustering: affinity matrix and distance matrix. **We assumed Symmetry** hence both were calculated as upper triangular matrices with the previous proposed methods, which save the NCD or BCN distance from each sample to each other. It must be noted that the diagonal is a 0 diagonal as it calculates the distance of each paper with itself.

Clustering

A simple approach was to use an algorithm such as k-means. However, it threw initial results that were not promising and we decided to go to an alternative. We finally decided and implemented Ward Method for Hierarchical Clustering with affinity calculation based on NCD. *Ward's minimum variance method* can be even interpreted as the ‘Kmeans of hierarchial clustering’ as its minimum variance method, defined to be squared Euclidean distance between points which resembles in a way the popular k-means algorithm. Hierarchical Clustering demands a distance matrix as an input but we plugged in the affinity matrix instead. It could be done as the NCD similarity resembles distance in a manner which the smallest value holds for ‘closer’/similar object. We finally proposed a threshold that suggested 8 clusters.



Cluster Analysis

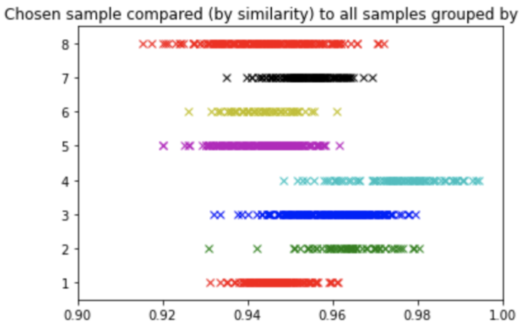
To analyze results, we took one sample and performed three different checks:

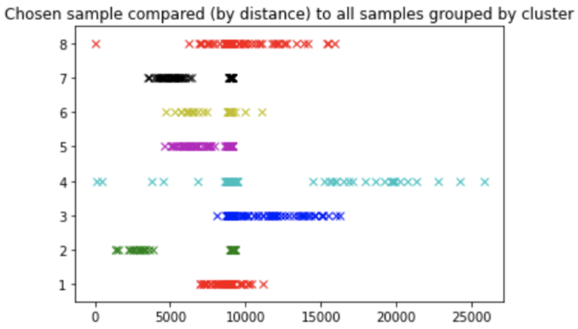
1. Compare its distance to all other samples, grouped per cluster
2. Comparte some titles within a cluster
3. Check most relevant words in the text of papers per cluster

Distance to each cluster

We can observe interesting results. As per the left image, there is some kind of overlap when considering affinity calculation (similarity). The means of the distance of the sample to each cluster are different, whereas the variance does not add relevant information. Anyways, all of them are larger than 0.90, even within the clustering itself, which does not show that much of compression or similarity.

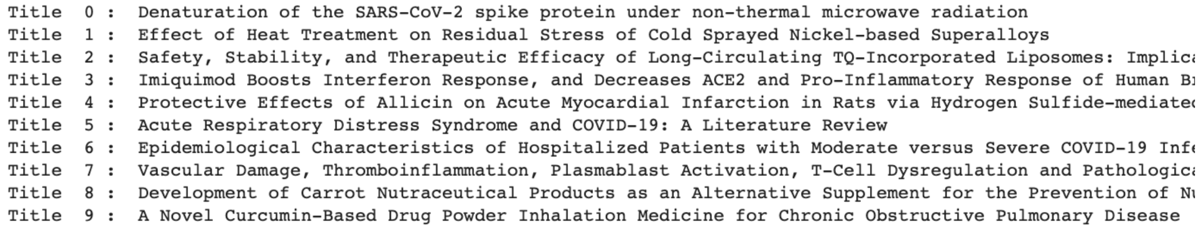
If we consider the second figure, we can see that the performed clustering does not show relevance when comparing the sample to each cluster by BCN distance.





Title comparison

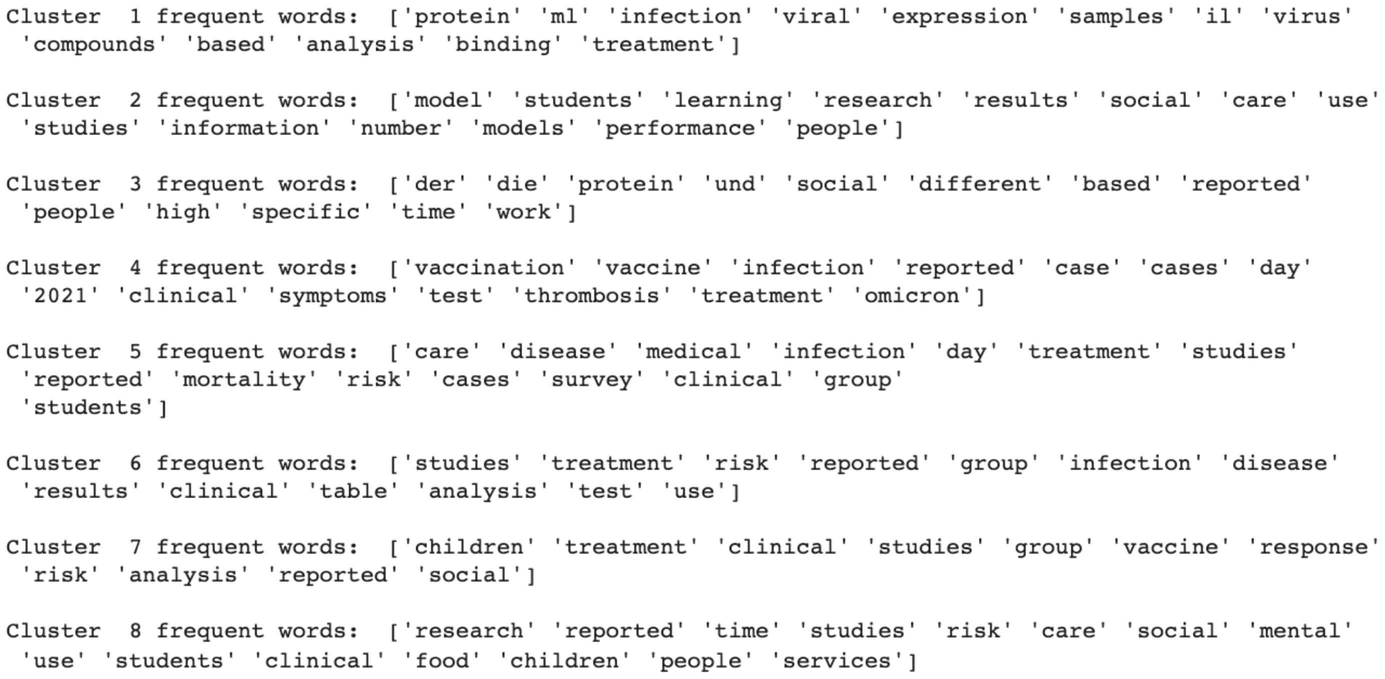
Within the cluster of the sample we are analyzing, we extract 10 titles of different papers to check their relation. At a glance, we could say that the topics they talk about are related to “heat” and “studies about treatments”.



```
Title  0 :  Denaturation of the SARS-CoV-2 spike protein under non-thermal microwave radiation
Title  1 :  Effect of Heat Treatment on Residual Stress of Cold Sprayed Nickel-based Superalloys
Title  2 :  Safety, Stability, and Therapeutic Efficacy of Long-Circulating TQ-Incorporated Liposomes: Implica
Title  3 :  Imiquimod Boosts Interferon Response, and Decreases ACE2 and Pro-Inflammatory Response of Human Br
Title  4 :  Protective Effects of Allicin on Acute Myocardial Infarction in Rats via Hydrogen Sulfide-mediate
Title  5 :  Acute Respiratory Distress Syndrome and COVID-19: A Literature Review
Title  6 :  Epidemiological Characteristics of Hospitalized Patients with Moderate versus Severe COVID-19 Infe
Title  7 :  Vascular Damage, Thromboinflammation, Plasmablast Activation, T-Cell Dysregulation and Pathologica
Title  8 :  Development of Carrot Nutraceutical Products as an Alternative Supplement for the Prevention of Nu
Title  9 :  A Novel Curcumin-Based Drug Powder Inhalation Medicine for Chronic Obstructive Pulmonary Disease
```

Relevant words

Within the samples of each cluster, we performed TFIDF (Salton and Buckley, 1988), in which coordinates of the vector space correspond to individual words. Each word is scored by the algorithm and we choose the most relevant ones, prior removing useless or meaningless words such as (“et al”, “sars”, “covid”).



```
Cluster  1 frequent words:  ['protein' 'ml' 'infection' 'viral' 'expression' 'samples' 'il' 'virus'
 'compounds' 'based' 'analysis' 'binding' 'treatment']

Cluster  2 frequent words:  ['model' 'students' 'learning' 'research' 'results' 'social' 'care' 'use'
 'studies' 'information' 'number' 'models' 'performance' 'people']

Cluster  3 frequent words:  ['der' 'die' 'protein' 'und' 'social' 'different' 'based' 'reported'
 'people' 'high' 'specific' 'time' 'work']

Cluster  4 frequent words:  ['vaccination' 'vaccine' 'infection' 'reported' 'case' 'cases' 'day'
 '2021' 'clinical' 'symptoms' 'test' 'thrombosis' 'treatment' 'omicron']

Cluster  5 frequent words:  ['care' 'disease' 'medical' 'infection' 'day' 'treatment' 'studies'
 'reported' 'mortality' 'risk' 'cases' 'survey' 'clinical' 'group'
 'students']

Cluster  6 frequent words:  ['studies' 'treatment' 'risk' 'reported' 'group' 'infection' 'disease'
 'results' 'clinical' 'table' 'analysis' 'test' 'use']

Cluster  7 frequent words:  ['children' 'treatment' 'clinical' 'studies' 'group' 'vaccine' 'response'
 'risk' 'analysis' 'reported' 'social']

Cluster  8 frequent words:  ['research' 'reported' 'time' 'studies' 'risk' 'care' 'social' 'mental'
 'use' 'students' 'clinical' 'food' 'children' 'people' 'services']
```

Here the results per cluster. On high level, we can identify topics of the papers per cluster. As we can see, these are not extremely conclusive and sometimes overlap but can represent the group.

- Cluster 1 topics: *virus and its structure*
- Cluster 2 topics: *Research and studies*
- Cluster 3 topics: *Statistics and measured results*
- Cluster 4 topics: *Vaccine and treatments*
- Cluster 5 and 6 topics: *Risks results and mortality*
- Cluster 7 and 8 topics: *Treatments and studies on children*

Clustering the rest of the data

Now, as we've clustered a sample of 1000 random papers, we can use the clustered samples as a concept of 'trained' data so that we could use it for an 'inference' stage as we cluster the complete dataset based on the clustered samples. We suggest the following rule in order to cluster a sample from dataset:

$$argmin_{j} \frac{\sum_{i=1}^{size(j)} NCD(x, y_{i})}{size(j)} , j \in [1, 2,.., n]$$

$x - inspected\ sample$

$y_{i} - an\ instance\ i\ in\ cluster\ j$

where j stands for an instanced cluster in the total number of found clusters, meaning we're clustering a sample by applying similarity to all members of cluster j, and then we average the similarity score (calculated by NCD). we do that for all the clusters and then we cluster the sample based on the minimal average similarity score from the set of n clusters.
For example we take the following sampled paper:

```
Expanded Pharmacy Practice Implementation: Lessons from Remote Practice
```

we plug it into our method of clustering and we get:

```
By Similarity, the Closest Cluster is 8
labels
1    0.949934
2    0.948419
3    0.972144
4    0.954155
5    0.932908
6    0.928895
7    0.931558
8    0.925285
```

But in general, as already shown, we can say that by average, the scores of similarity for each cluster are pretty 'close' and there is no **clear cut** between clusters, as a decision based on a score of 0.925 rather than 0.928 seems a bit obscure.

Discussion and future work

- We inspected 2 proximity by compression methods: NCD, BCN. by creating an affinity matrix and distance matrix we did hierarchical clustering and evaluated the nature of clusters qualitatively. in general we conclude that those metrics proximity arent robust enough for the task of clustering as the scores , similarity for instance, are far too close to 1 (unsmiliar) rather then similar (tend to 0).

- It is interesting to inspect NCD and BCN on papers from different subject areas rather than all from the same main topic. only them ,their efficiency can be inspected more thoroughly. in general, it is interesting to compress and cluster papers from completely different subject area and see if there is a clear cut in clustering and measure the errors in a more clear way.

- It would definitely be interesting to use word embedding (using SciBert or SPECTER) and then compare the similarities and clusters between compression and embedding. It is kind of an unfair comparison as all pretrained transformer models use attention and extract semantics from papers but still interest us.

- Clustering the full 20,000 dataset estimated time was ~45 hours so we’ve decided only to sample through it and not perform the complete task. in future work, we would definitely try clustering the full data and see if the results differ.