CORD-19: NLP

Aditya Sharma (112676654)
Mukul Javadekar (112961738)

IDEA AND SIGNIFICANCE

- IDEA:
  - CORD-19 is a COVID-19 Open Research Dataset, it is a collection of over 47,000 scholarly articles, including over 36,000 with full text, about COVID-19, SARS-CoV-2, and related coronaviruses.
  - Our idea is to do data cleaning, exploratory data analysis, apply NLP techniques on data and share our insights from the results.
  - https://www.kaggle.com/allen-institute-for-ai/CORD-19-research-challenge

- SIGNIFICANCE:
  - With the rapid increase in the coronavirus literature there is a need to apply data analysis techniques which can help researchers get proper insights from the data.

TEAM EXPERTISE AND LEARNING

- Expertise Before class:
  - Aditya Sharma: Experience with C language
  - Mukul Javadekar: Beginner in programming language

- What we learnt during project:
  - Basics of Natural language processing
  - Tools used in NLP, Exploratory data analysis
  - Visualizing the data and topic modeling

TECHNIQUES AND TOOLS USED

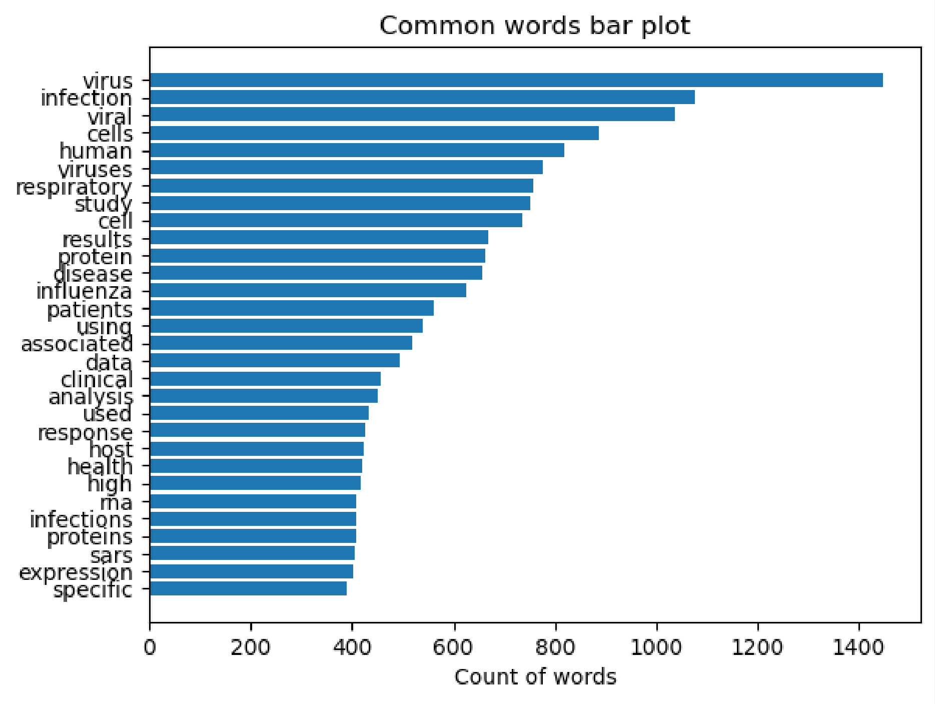
- Data Gathering: Obtained from Kaggle
  - Data has 19 columns, of which we are interested in cord_id and abstract

- Data Cleaning: Get data in clean standard format for analysis
  - Corpus: A collection of text using pandas
  - Document-term Matrix: Clean text by removing all numbers, letters not belonging to a-z, non-ASCII characters and next line using regular expressions. Tokenize it by word and put in matrix form using scikit-learn and count vectorizer (remove stop-words)
- Exploratory Data Analysis (EDA): Summarize the data
  - Word count – Aggregate word count in document-term-matrix by this we can get the top words
  - Bar plot – To visualize the top words using matplotlib
  - Word clouds – To visualize the top word in each paper after removing the common words such as virus, infection etc. and using wordcloud.

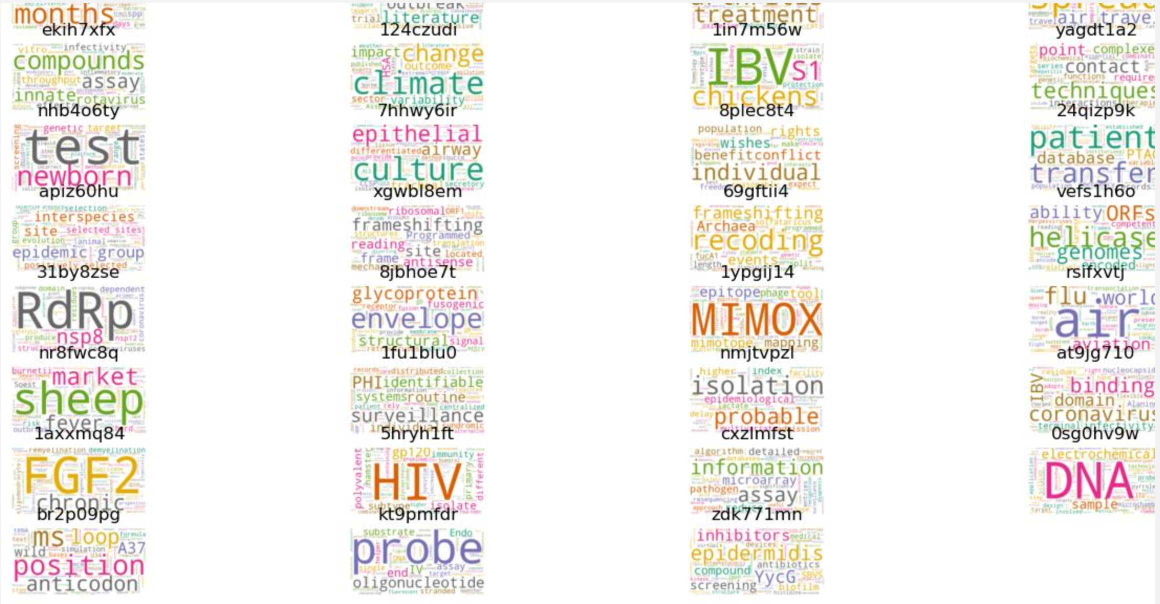
TECHNIQUES AND TOOLS USED

- Topic Modelling: Input is document term matrix as each topic has a set of words where order doesn't matter. We use gensim here which use Latent Dirichlet Allocation (LDA) for topic modelling. We used nltk for tagging.
- LDA learns topic mix in every document and word mix in every document. We start with a random number of topics, it goes through every word and randomly assigns it to a topic. It further updates the topic for every word based on how frequently that topic occurs in that document and how often the word occurs in the topic overall. Does in multiple iteration.
- nltk used to filter out just the nouns or filtering nouns and adjectives for topic modelling

PICS AND GRAPHS



Top 30 words of 5000 items



Sample for word cloud for several papers

OUTCOMES AND RESULT

- Sample topic generated: We generated 20 topics

```
[(0,
  '0.048*"rna" + 0.036*"viral" + 0.033*"replication" + 0.013*"virus" +
0.013*"host" + 0.012*"synthesis" + 0.011*"cellular" + 0.010*"transcription" +
0.009*"protein" + 0.009*"translation"')
, (1,
  '0.022*"viruses" + 0.020*"virus" + 0.016*"human" + 0.013*"viral" +
0.013*"species" + 0.012*"new" + 0.011*"genome" + 0.011*"disease" +
0.011*"host" + 0.010*"pathogens“’),
```

- Example for paper and topic, each cord_id assocated with some topics:

```
([0, 6], 'zjufx4fo')
([0, 7, 8], 'ymceytj3')
```

CHALLENGES AND FUTURE WORK

- While cleaning data, adding stop words, finding topics, domain related vocabulary and knowledge will be useful as large amount of data
- Few topics can be selected and analysis on papers belonging to that topic can be performed

THANK
YOU