**NAME : SAPTARSHI MUKHERJEE**

**REPORT**

**1: Decoding Covid-19 with Genome Analysis**

**PROBLEM STATEMENT**

You are one of the researchers responding to the White House Office of Science and

Technology Policy centre’s call to conduct advanced research on Covid-19. A dataset that

represents the most extensive machine-readable coronavirus literature collection available for

data and text mining to date, with over 29,000 articles, more than 13,000 of which have full

text.

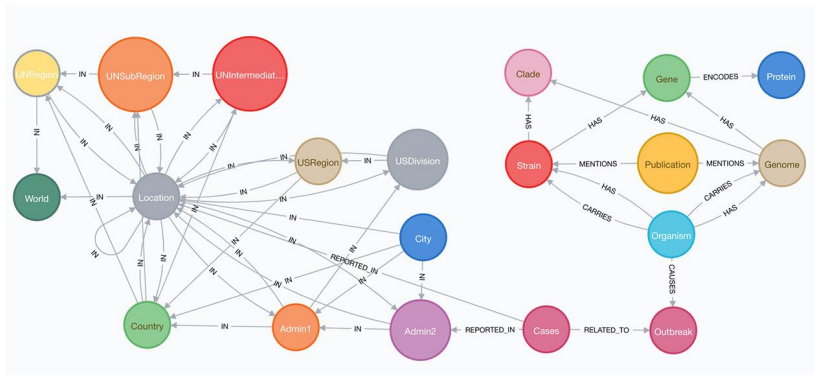
Using the CORD-NER dataset and Knowledge Graph, determine and map out the details of

the SARS-CoV-2 genome to assist understanding of the emergence, evolution and diagnosis

of this deadly virus.

Dataset used: <https://www.kaggle.com/code/xuanwangstat/cord-ne>

KNOWLEDGE GRAPH



**WORKFLOW OF MY NOTEBOOK 3: (Qna-using-bert llm model)**

**Initialization:**

* Set up logging to monitor the process.
* Define the root path and load metadata from the provided CSV file.
* Read all JSON files containing the articles' content.

**FileReader Class:**

* Create a class to parse JSON files containing article content.
* Extract paper ID, abstract, and body text from each file.

**Data Preparation:**

* Define a function (get\_date\_dt) to extract relevant information (paper ID, abstract, body text, authors, title, journal) from both JSON files and metadata CSV.
* Clean the data by handling exceptions and missing values.

**Word2Vec Model Initialization:**

* Initialize a Word2Vec model with specific parameters such as min\_count, window size, vector size, etc.

**Sentence Creation:**

* Tokenize and create sentences from the provided DataFrame using Phrases and Phraser.

**Building Vocabulary:**

* Build the vocabulary for the Word2Vec model based on the created sentences.

**Training Word2Vec Model:**

* Train the Word2Vec model using the created sentences.

**Text Preprocessing:**

* Perform text preprocessing including removing URLs, brief cleaning, lemmatization, and removing stopwords.

**Preprocessing for Word2Vec:**

* Prepare the text data for Word2Vec model training by preprocessing it and ensuring it meets certain criteria.

**Get Relevant Articles:**

* Define a function (get\_relevant\_articles) to retrieve relevant articles based on a query and their abstracts.
* Preprocess the query and abstracts.
* Embed the query into a vector representation.
* Calculate the similarity score between the query and each abstract using the Word2Vec model.
* Sort the articles based on their scores in descending order.

**Word2Vec Model Loading:**

* If to\_train is False, i load a pre-trained Word2Vec model along with pre-processed data and a Phraser.

**Query Processing:**

* preprocess the query and articles' abstracts to prepare them for similarity calculations.

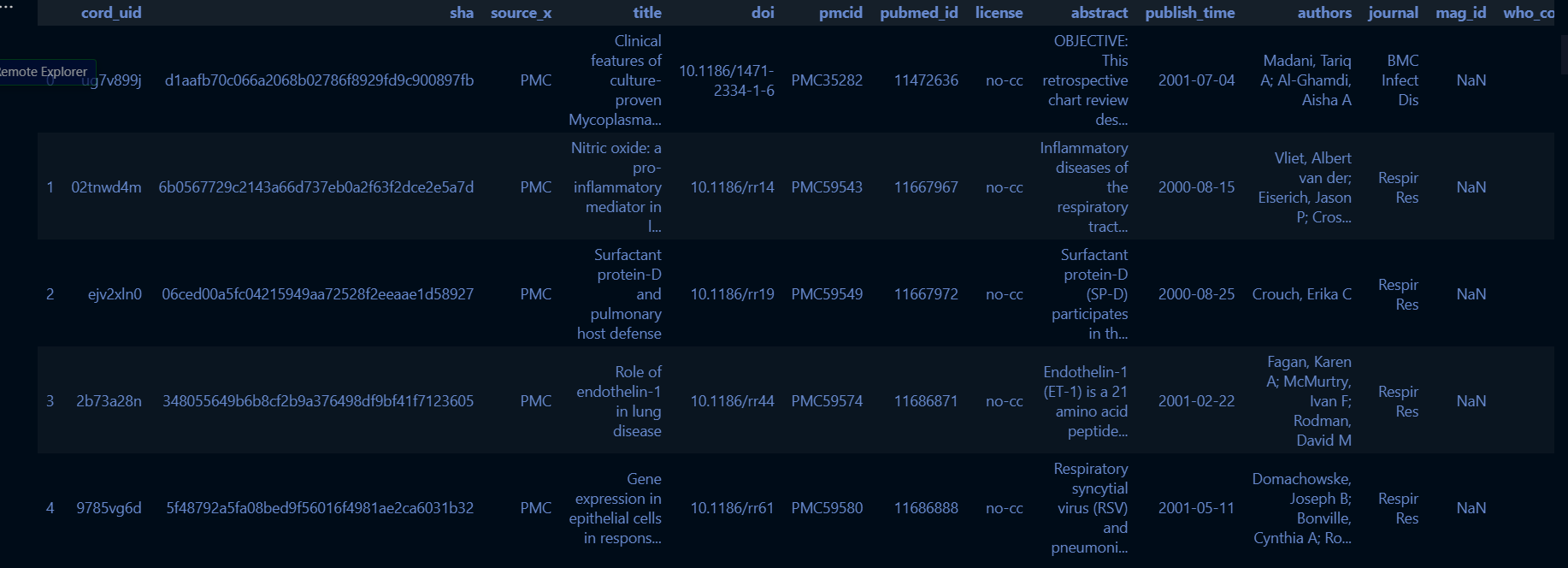
**Similarity Calculation:**

* I calculate the cosine similarity between the query and each article's abstract using the Word2Vec model.

**BERT-based Question Answering:**

* I use a BERT-based question-answering model to find answers to the query within the relevant articles' paragraphs.

**RESULTS**

****







**CONCLUSION**

The developed system effectively utilizes Word2Vec models and BERT-based question-answering techniques to efficiently retrieve relevant articles and extract pertinent information in response to user queries, facilitating comprehensive exploration and understanding of COVID-19 research literature.