**Final Work Distribution**

|  |  |
| --- | --- |
| Name | Did |
| Jongyoon Choi | - Set up GitHub and get in touch with group mates  - Finish codes for question A, B-1, B-2, and B-3 (other entities) and interpret the results  - Finish Project Paper Introduction, Methods Intro, methods 2.1, 2.2, 2.3, Results 3.1, 3.2, 3.3, limitations, and bibliography |
| Chris McGowan |  |
| Sunny Patel |  |

Exploring PUBMED data

Yoon Choi, Chris McGowan, Sunny Patel

# **Introduction**

PubMed data is an important piece of information that we can utilize in order to analyze the health data. Our objective in this paper is to determine whether we can gain insights from the speculative dataset and covid-19 dataset in regard to correctly determine the data as speculative or not, and also to extract various topics involved in the dataset. In this paper, we extracted relevant information for the health data analysis, and finally created charts and figures to analyze it.

# **Methods**

We were given two datasets for Speculative data and COVID-19 data, in TOK and CSV format, respectively. The Speculative dataset contain four files: speculative data for training and testing, and non-speculative data for training and testing. It contains rows of free-text information that is delimited by a dot and a new line. The COVID-19 dataset contains various information in free-text format such as fields in titles, abstracts, and conclusions. With the data in our hands, our group decided the most efficient method for parsing and analyzing the dataset was to use Python, and especially Jupyter notebook, as our programming language and write scripts that iterate through all the relevant information. For analyzing the topics covered in the COVID-19 dataset, we utilized Jupyter Notebook in order to divide up the algorithms in places it takes a long time, i.e., building a LDA model.

# 2.1 Question A: Determining a Speculative Sentence with a Classifier

For analyzing the sentences in the testing speculative / non-speculative data, we utilized the sentences in the respective training datasets. Training the dataset in the training file allowed us to sufficiently build a model to test on the testing dataset and come up with the best F-1 score as possible. We imported the data with pandas framework so we can transform TOK file into CSV file. Then, because we first had to label the data, we added the ‘speculative’ column to distinguish between speculative and non-speculative sentences. We next merged training files with each other and testing files with each other while also shuffling their orders to build a better model. Once we had the data frame, we fed the data to the classifier. We utilized both LSTM and Word2Vec mode in order the analyze a sentence to be speculative or not.

# 2.2 Question B.1: Identifying Topics in COVID-19 Data

For identifying the topics covered (question B.1) we utilized LDA. We chose title and abstract because many researchers try to condense their big picture of the research paper into these two categories frequently, allowing us to sufficiently identify the topics covered with these two fields. After we loaded the COVID-19 Dataset, we pre-processed the data by lemmatizing using NLTK wordnet and removing a list of stop words. We also utilized a spacy model “en\_core\_web\_sm”, which is a NLP model for English languages and more efficient than “en\_core\_web\_trf”. Then we iterated the CSV data by extracting the title and the abstract from each observation and saving it in the variable ‘text\_data’ as a list of list of words. We encapsulated the mapping between normalized words and their integer ids and saved it in the variable ‘dictionary’. We then transformed ‘text\_data’ into the bag-of-words format (a list of tuples with token\_id and token\_count as the first and second element of each tuple pair). Bag of Words method allows us to extract features from text documents, in this case a combination of titles and abstracts. Using these data, we trained the model on 5 to 10 number of topics in order to determine the best number of topics to subdivide the data into. We determinined the number of topics as 10 because of high coherence score and low perplexity score in the graph below. Coherence scores measure the relative distance between words within a topic and analyzing the COVID-19 model with 10 topics demonstrated the most promising high score. Perplexity score measures how surprised a model is of new data it has not seen before and 10 topics also showed the lowest score out of all trials. We got respective scores by applying LDA model from the genism library to save a model, then outputting perplexity and coherence scores on each of them.

|  |  |
| --- | --- |
| number of topics | perplexity score |
| 5 | -8.4009 |
| 6 | -8.4365 |
| 7 | -8.5074 |
| 8 | -8.5939 |
| 9 | -8.7137 |
| 10 | -8.8793 |

|  |  |
| --- | --- |
| number of topics | coherence score |
| 5 | 0.5534 |
| 6 | 0.5466 |
| 7 | 0.5583 |
| 8 | 0.5767 |
| 9 | 0.5861 |
| 10 | 0.584 |

# 2.3 Question B.2: Extracting Drugs and Diseases from the COVID-19 Data

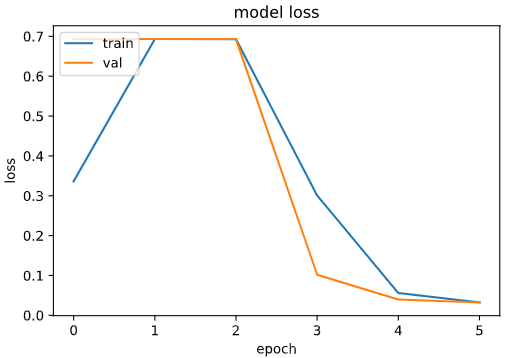
After pre-processing the COVID-19 data through lemmatization and removing stop words, we utilized open-source NLP software SciSpacy in order to extract drugs and diseases. We iterated through each row in the CSV file and saved titles and abstracts. After the iteration, we were left with a list of strings where each string was adjoined string of titles and abstracts. We then iterated through this list to get the entity and the entity label. In order to properly recognize the entity and the entity label, we made use of “en\_ner\_bc5cdr\_md” model for getting drugs and diseases. While diseases had its own entity label, drugs were under the label ‘chemical’, thus after getting a list of all chemicals, we manually sorted through the data to get the drugs because some were not drugs, such as ‘COVID-19’ and ‘oxygen’. In the ‘chemical’ labels, there were more often entities that were related to the chemicals of our body but not drugs.

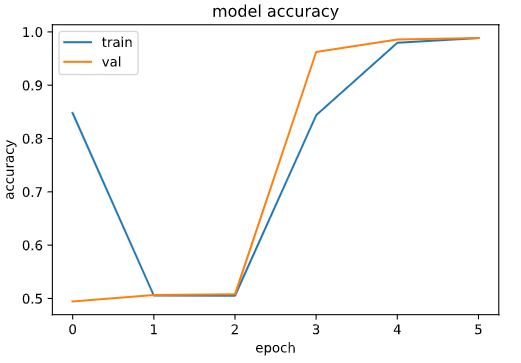
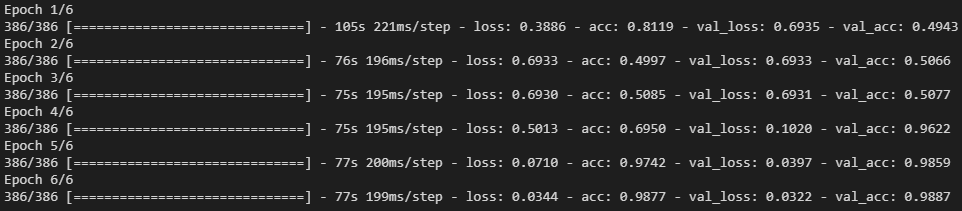
# 2.4 Question B.3: Another Topic

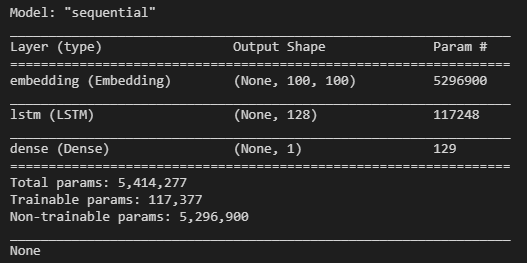
# 2.5 Question B.4: Another Topic

# **Results**

# 3.1 Question A Result:

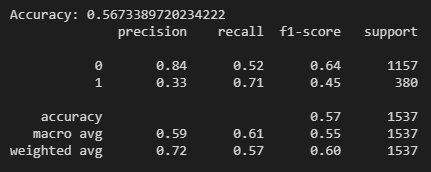
 Using the LSTM model, our model is summarized as below. Below we can see layer types along with output shapes and parameter numbers. There are over 100,000 trainable parameters and over 5,000,000 non-trainable parameters. We also iterated the data over six epochs. The result can be seen below with the graph of the model.



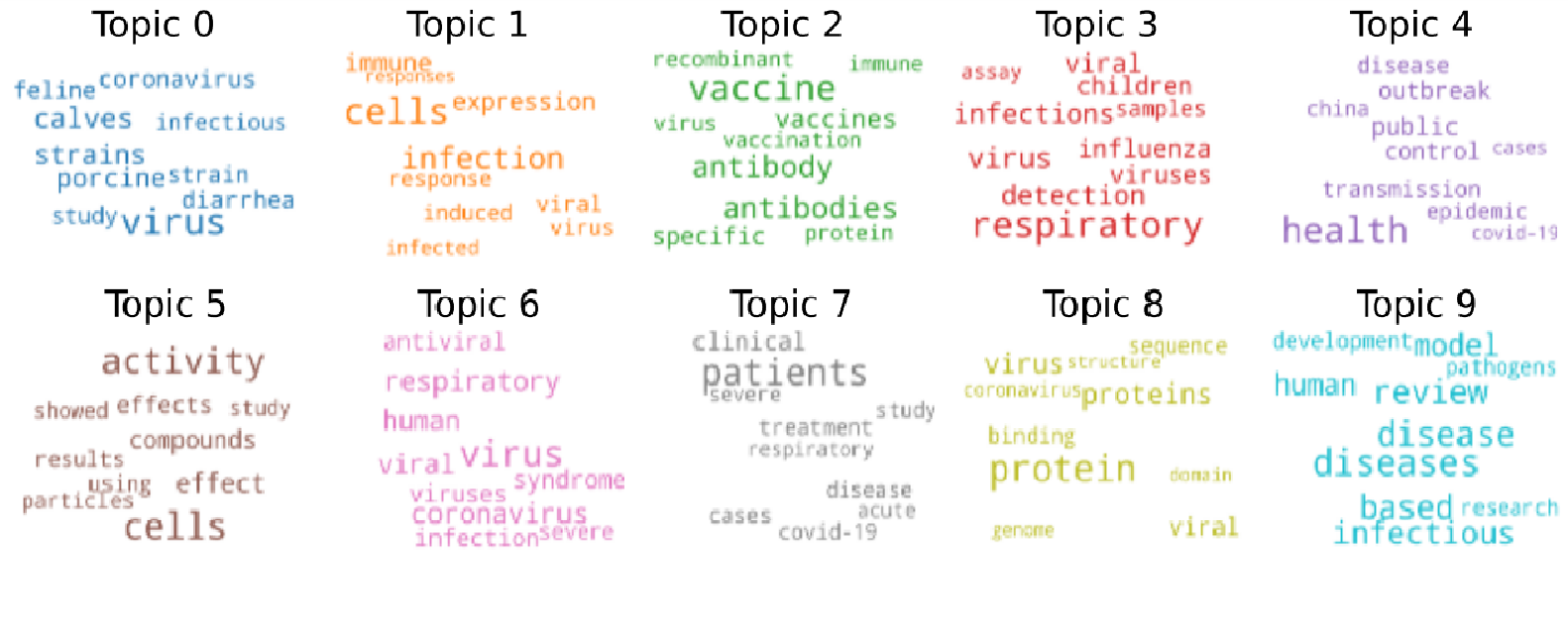


We can see that we have a model loss actually increasing when we are at first and second epoch, meaning that we are already overfitting with the data, and we should stop. Thus, we can deduce that we should look at epoch of 0. At epoch 0, we can see that our model accuracy is at around 0.5. This is not the best outcome, but we believe it is a respectable value from the trained data.

We also created a model pipeline to assess the model and get a f1-score. We utilized genism word2vec model for the model creation because word2vec is a commonly used machine learning algorithm with a respectable result. We ran the model with it to come up with a f1-score and we outputted on a terminal like the figure below. We can see that the f1-score accuracy is 0.57 and we were able to predict non-speculative data better: non-speculative data had a f1-score of 0.64 and speculative data had a f1-score of 0.45.



# 3.2 Question B.1 Result: Identifying Topics in COVID-19 Data

The figure below shows the results we got from having 10 topics on the COVID-19 Dataset.

There are general words mixed into the results such as ‘study’, ‘activity’, ‘effects’, ‘showed’, and etc. Not only that, there are clear overlaps between the words across the topics such as ‘disease’ and ‘corona virus’. To the best of our ability, we tried to describe what each topic may be representing in the table below by highlighting the words that seemed most significant. We believe while there are clear similarities between the topics due to inseparableness of similarity of vocabularies used in the data.

|  |  |
| --- | --- |
| Topics Name | Words and Scores |
| Study on Animals (Topic 0) | 0.020\*"virus" + 0.012\*"calves" + 0.011\*"strains" + 0.010\*"**porcine**" + 0.008\*"coronavirus" + 0.008\*"diarrhea" + 0.008\*"**feline**" + 0.008\*"strain" + 0.007\*"study" + 0.007\*"infectious" |
| How infection induces Certain Responses (Topic 1) | 0.033\*"cells" + 0.023\*"**infection**" + 0.015\*"immune" + 0.014\*"expression" + 0.012\*"viral" + 0.011\*"**response**" + 0.011\*"**induced**" + 0.011\*"virus" + 0.009\*"infected" + 0.007\*"**responses**" |
| Study on treatments or Cures through vaccination (Topic 2) | 0.036\*"**vaccine**" + 0.024\*"**antibody**" + 0.024\*"**antibodies**" + 0.018\*"vaccines" + 0.015\*"specific" + 0.012\*"recombinant" + 0.012\*"protein" + 0.012\*"**vaccination**" + 0.012\*"virus" + 0.011\*"immune" |
| Virus impact in on children’s body (Topic 3) | 0.033\*"**respiratory**" + 0.020\*"virus" + 0.018\*"infections" + 0.018\*"detection" + 0.017\*"viral" + 0.015\*"influenza" + 0.015\*"viruses" + 0.014\*"**children**" + 0.012\*"samples" + 0.010\*"**assay**" |
| The world’s response and interpretation of the virus (Topic 4) | 0.024\*"**health**" + 0.009\*"**public**" + 0.009\*"**outbreak**" + 0.008\*"control" + 0.008\*"transmission" + 0.007\*"disease" + 0.007\*"**epidemic**" + 0.007\*"**china**" + 0.005\*"cases" + 0.005\*"covid-19" |
| The Study on Virus (very general) (Topic 5) | 0.018\*"cells" + 0.017\*"activity" + 0.008\*"effect" + 0.007\*"compounds" + 0.007\*"effects" + 0.006\*"results" + 0.006\*"using" + 0.006\*"particles" + 0.006\*"showed" + 0.005\*"study" |
| Virus itself (very general) (Topic 6) | 0.054\*"virus" + 0.029\*"coronavirus" + 0.029\*"respiratory" + 0.026\*"viral" + 0.026\*"human" + 0.022\*"syndrome" + 0.021\*"viruses" + 0.021\*"infection" + 0.020\*"antiviral" + 0.017\*"severe" |
| Virus’s impact on public (Topic 7) | 0.036\*"**patients**" + 0.013\*"clinical" + 0.010\*"cases" + 0.010\*"**treatment**" + 0.010\*"covid-19" + 0.009\*"disease" + 0.009\*"severe" + 0.009\*"study" + 0.008\*"acute" + 0.008\*"respiratory" |
| Virus’s biochemical structure (Topic 8) | 0.030\*"protein" + 0.017\*"**proteins**" + 0.016\*"virus" + 0.012\*"viral" + 0.009\*"**binding**" + 0.009\*"**sequence**" + 0.007\*"coronavirus" + 0.006\*"**genome**" + 0.006\*"domain" + 0.006\*"structure" |
| How researchers are trying to understand the virus (Topic 9) | 0.011\*"diseases" + 0.010\*"disease" + 0.009\*"**review**" + 0.009\*"based" + 0.008\*"infectious" + 0.007\*"**model**" + 0.007\*"human" + 0.006\*"development" + 0.006\*"**research**" + 0.006\*"pathogens" |

# 3.3 Question B.2 Result: Extracting Drugs and Diseases from the COVID-19 Data

Below table demonstrates the results we obtained by extracting drugs and diseases from the COVID-19 dataset.

|  |  |  |
| --- | --- | --- |
| Drugs | frequency | proportions of diseases |
| amino acid | 1644 | 0.403139 |
| IFN | 928 | 0.227563 |
| ribavirin | 469 | 0.115007 |
| chloroquine | 264 | 0.064738 |
| oseltamivir | 212 | 0.051986 |
| steroid | 139 | 0.034085 |
| angiotensin II | 125 | 0.030652 |
| corticosteroids | 109 | 0.026729 |
| vitamin D | 95 | 0.023296 |
| niclosamide | 93 | 0.022805 |

|  |  |  |
| --- | --- | --- |
| DISEASE | frequency | perc |
| SARS | 6604 | 0.284043 |
| Pneumonia | 5282 | 0.227183 |
| diarrhea | 4110 | 0.176774 |
| cancer | 3434 | 0.147699 |
| RSV | 1136 | 0.04886 |
| ARDs | 724 | 0.03114 |
| FIP | 710 | 0.030538 |
| Fever | 606 | 0.026065 |
| gastroenteritis | 378 | 0.016258 |
| H1N1 | 266 | 0.011441 |

As we can see from the previous page, amino acid and IFN seems to be the most frequent drugs mentioned in the papers. We can also see chloroquine, steroid, and vitamin D, all of which are familiar to people. When looking at the diseases, we can see that SARS and pneumonia are the top two most frequently mentioned diseases. We can also see that diarrhea, cancer, fever, gastroenteritis and H1N1, all of which are familiar to people are present in the list. We can deduce that COVID may affect people by making them have gastrointestinal diseases due to ‘diarrhea’ and ‘gastroenteritis’. Also, we can see it may affect people’s lung capacity due to ‘pneumonia’ and ‘fever’.

# Limitations

As with all NLP parsing problems on free-texts, we were faced with limitations on current progress of natural language processing while using the open-source software library ‘SciSpacy’. However, as the software is the best in its field, we thought the limitations were nothing to be concerned about. The examples include similarities of words and ambiguities of the words. For example, if the diseases have several names, we had to manually check to make sure their data was merged.

Secondly, we can see from question B.1 that our answer derives from our best interpretation we have at the moment. This means that what we have as our results may seem incorrect for others. However, we believe our thorough interpretations justify such limitations and we are confident that one could see the points we argue for in the results section.

# Conclusions

There is various information available through PubMed data we can utilize in order to analyze health data. We used speculative / non-speculative dataset and COVID-19 dataset to build a speculative classifier, identify topics in COVID-19, extract drugs and diseases, and more.

# Bibliography

1. “scispacy”. [Online]. Available: <https://allenai.github.io/scispacy/>. [Accessed: 25-April-2021].
2. “Named Entity Recognition”. [Online]. Available: <https://en.wikipedia.org/wiki/Named-entity_recognition>. [Accessed: 27-April-2021].
3. “csv — CSV File Reading and Writing”. [Online]. Available: https://docs.python.org/3/library/csv .html. [Accessed: 12-March-2021]
4. “Perplexity”. [Online]. Available: <https://en.wikipedia.org/wiki/Perplexity>. [Accessed: 27-April-2021].
5. “Creation of a new longitudinal corpus of clinical narratives”. [Online]. Available: https://pubm ed.ncbi.nlm.nih.gov/26433122/. [Accessed: 12-March-2021]
6. “os.path — Common pathname manipulations”. [Online]. Available: https://docs.python.org/3/ library/os.path.html. [Accessed: 20-March-2021].
7. “7. Input and Output — Python 3.9.2 documentation”. [Online]. Available: https://docs.pytho n.org/3/tutorial/inputoutput.html. [Accessed: 20-March-2021].
8. “IMDB Dataset of 50K Movie Reviews”. [Online]. Available: [IMDB Dataset of 50K Movie Reviews | Kaggle](https://www.kaggle.com/lakshmi25npathi/imdb-dataset-of-50k-movie-reviews). [Accessed: 2-May-2021].