DS5230 COIVD-19 Analysis Final Report

Jiachen Liu Kepan Gao Wenqing Xu Guangyi Chen Jianyue Chen Yian Ding

Abstract

The rapid acceleration of larger number of literature and cases in COVID-19's spread, to acquire the large updated information is inefficient for health researchers. Therefore, our goal is to build a literature search system bases on our exploratory data analysis of Louvain method, PCA and t-SNE dimensional reduction, and unsupervised clustering techniques of K-means, and Latent Dirichlet allocation method. The dataset is from CORD-19 [1], an open research dataset in response to the COVID-19 pandemic which includes over 51,000 scholarly articles and 40,000 related full texts. In this report, we have preliminary results include the data preprocessing and exploratory data analysis of co-author community detection, and the comparison between PCA with clustering and LDA.

1 Introduction

COVID-19, the world's common issue, has caused millions of deaths and still demands effective ways to solve it. Massive research results are published every day, and it has become essential and urgent to effectively organize these documents.

In this project, we will explore the CORD-19 dataset[1] and will implement some unsupervised machine learning algorithms to generate some new insights for this dataset. In response to the COVID-19 pandemic, the White House and a coalition of leading research groups have prepared the COVID-19 Open Research Dataset (CORD-19). This dataset is a resource of over 400,000 scholarly articles, including over 150,000 with full text, about COVID-19, SARS-CoV-2, and related coronaviruses. This dataset is provided to the global research community to apply recent advances in natural language processing and other AI techniques to generate new insights in support of the ongoing fight against this infectious disease. We will explore the dataset of metadata which contains some key attributes of papers' title, abstract, authors, journals and other related information.

There are three problems that we seek to answer in the project: Firstly, detect the author community of literature in the CORD-19 dataset. Secondly, perform literature clustering and visualization to obtain a clear, easy-to-read plot of similar academical results. Finally, build a literature search system based on our clustering outcomes.

By organizing these problems, we will simplify the similar search for authors and related articles, and will qualify the content of clusters. Therefore, people could quickly find related authors or publications and have meaningfully recommended insights.

2 Method

2.1 Co-authorship Network Analysis

Co-authorship networks are an important class of social networks and have been used extensively to determine the structure of scientific collaborations and the status of individual researchers. Co-authorship implies a temporal and collegial relationship that places it more squarely in the realm of social network analysis. [6]

In our project, co-authorship network analysis is adopted for mining the collaboration pattern for the scientists over all the world in the pandemic period. We constructed an undirected, weighted co-authorship network. Two authors are connected by a weighted link that equals the number of collaboration of these two authors.

2.1.1 Influential Scientists Mining

In order to measure prestige of an author, we introduced two common centrality metrics degree and eigenvector centrality:

Degree centrality of a node is defined as the total number of edges that are adjacent to this node. Degree centrality represents the simplest instantiation of the notion of centrality since it measures only how many connections tie authors to their immediate neighbors in the network.

Eigenvector centrality is used to measure the level of influence of a node within a network. The eigenvector centrality of a node i is defined as the i-th element of the eigenvector of the adjacency matrix which has the maximum eigenvalue. This score is relative to the number of connections a node will have to other nodes. Connections to high-scoring eigenvector centrality nodes contribute more to the score of the node than equal connections to low-scoring nodes. Google's PageRank[7] is a variant of the eigenvector centrality, which used to be the ranking mechanism at the heart of Google. In the project, we used this metric as well.

2.1.2 Community Detection

To find the association between authors, which might be helpful for mining underlying authorship community. We adopted the Louvain method for community detection to extract those author connections. The value to be optimized is modularity, defined as a value in the range $[0.5, 1]$ that measures the density of links inside communities compared to links between communities. For a weighted graph, modularity is defined as:

$$Q = \frac{1}{2m} \sum_{ij} [A_{ij} - \frac{k_i k_j}{2m}] \delta(c_i, c_j),$$

Where A_{ij} represents the edge weight between nodes i and j; k_i and k_j are the sum of the weights of the edges attached to nodes i and j; m is the sum of all of the edge weights in the graph; c_i and c_j are the communities of the nodes; δ is Kronecker delta function $\delta(x, y) = 1$ if $x = y$, 0 otherwise.

Louvain Method: Blondel et al.[2] introduced new algorithm that finds high modularity partitions of large networks in short time and that unfolds a complete hierarchical community structure for the network, thereby giving access to different resolutions of community detection. The algorithm is divided in two repeated phrases to maximize the value efficiently(Algorithm 1). First, each node in the network is assigned to its own community. Then for each node i, the change in modularity is calculated for removing i from its own community and moving it into the community of each neighbor j of i. The change in modularity is defined as:

$$\Delta Q = \left[\frac{\sum_{in} + k_{i,in}}{2m} - \left(\frac{\sum_{tot} + k_i}{2m} \right)^2 \right] - \left[\frac{\sum_{in}}{2m} - \left(\frac{\sum_{tot}}{2m} \right)^2 - \left(\frac{k_i}{2m} \right)^2 \right],$$

where $\sum_{in}$ in is the sum of the weights of the links inside C, $\sum_{tot}$ is the sum of the weights of the links incident to nodes in C, k_i is the sum of the weights of the links incident to node i, $k_{i,in}$ is the sum of the weights of the links from i to nodes in C and m is the sum of the weights of all the links in the network

Algorithm 1 Louvain Method

for each node **do**
  assign the node to its own community
end for
while the modularity $Q = \frac{1}{2m}\sum_{ij}\left[A_{ij} - \frac{k_i k_j}{2m}\right]\delta(c_i, c_j)$ doesn't converge **do**
  for each node i **do**
    for each neighbour j of node i **do**
      compute the increasing of the modularity if moving node i from C_i into C_j
      $\Delta Q = \left[\frac{\Sigma_{in}+2k_{i,in}}{2m} - \left(\frac{\Sigma_{tot}+k_i}{2m}\right)^2\right] - \left[\frac{\Sigma_{in}}{2m} - \left(\frac{\Sigma_{tot}}{2m}\right)^2 - \left(\frac{k_i}{2m}\right)^2\right]$
    end for
    if no increasing is possible **then**
      i remains in origin community
    else
      assign i to the community that resulted in the greatest modularity increase
    end if
  end for
end while

2.2 Topic Modeling by PCA + Clustering LDA

2.2.1 Preprocessing, PCA and t-SNE

Given the large number of COVID-19 literature, we will analyze the dataset from different aspects by performing exploratory data analysis. Aside from some important columns such as title and author, the text body will be the part that we put most effort on. We will parse the body of each document and turn document instances into feature vectors using TF-IDF so that we can implement unsupervised algorithms on them.

TF-IDF: TF-IDF short for term frequency–inverse document frequency, is a numeric measure to score the importance of a word in a document based on how often did it appear in that document and a given collection of documents [3]. It is defined as:

$$tf - idf(t, d, D) = tf(t, d) \times idf(t, D)$$

Where t denotes the terms; d denotes each document; D denotes the collection of documents.

$$idf(t, D) = \log \frac{|D|}{1 + |\{d \in D : t \in d\}|}$$

Where D is inferring to our document space; $|\{d \in D : t \in d\}|$ implies the total number of times in which term t appeared in document d.

Clustering: Clustering is an approach that integrates massive literature and builds a search and recommendation system. We transformed the data from the high-dimensional space to a space of fewer dimensions, the clustered results can be represented by a scatter plot after dimensional reduction with PCA and t-SNE. PCA is a main linear technique for dimensionality reduction performs a linear mapping of the data to a lower-dimensional space in such a way that the variance of the data in the low-dimensional representation is maximized [5].

t-SNE: T-distributed Stochastic Neighbor Embedding (t-SNE) is a non-linear dimensionality reduction technique useful for visualization of high-dimensional datasets. In symmetric SNE, the pairwise similarities in the low-dimensional map q_{ij} are given by[4]:

$$p_{j|i} = \frac{\exp(-\|x_i - x_j\|^2/2\sigma^2)}{\sum_{k \neq i} \exp(-\|x_i - x_k\|^2/2\sigma^2)}$$

Where σ_i is the variance of the Gaussian that is centered on datapoint x_i. The method for determining the value of σ_i is presented later in this section.

In t-SNE, Van der Maaten and Hintonhai [4] used a Student t-distribution with one degree of freedom as the heavy-tailed distribution in the low-dimensional map to solve the crowding problems and optimize the problems of SNE. By using this distribution, the joint probabilities q_{ij} are defined as:

$$q_{ij} = \frac{(1 + \|y_i - y_j\|^2)^{-1}}{\sum_{k \neq l}(1 + \|y_i - y_l\|^2)^{-1}}$$

Van der Maaten and Hinton [4] provided the minimized gradient of the Kullback-Leibler divergence between P and the Student-t based joint probability distribution Q:

$$\frac{\delta C}{\delta y_i} = 4 \sum_j (p_{ij} - q_{ij})(y_i - y_j)(1 + \|y_i - y_j\|^2)^{-1}$$

2.2.2 K-means

The k-means algorithm is used to partition a given set of observations into a predefined amount of k clusters[9]. The algorithm is described to start with a random set of k center-points (μ). During each update step, all observations x are assigned to their nearest center-point. In the standard algorithm, only one assignment to one center is possible. If multiple centers have the same distance to the observation, a random one would be chosen.

$$S_i^{(t)} = \{x_p : \|x_p - \mu_i^{(t)}\|^2 \leq \|x_p - \mu_j^{(t)}\|^2 \ \forall j, 1 \leq j \leq k\}$$

Afterwards, the center-points are repositioned by calculating the mean of the assigned observations to the respective center-points.

$$\mu_i^{(t+1)} = \frac{1}{|S_i^{(t)}|} \sum_{x_j \in S_i^{(t)}} x_j$$

This means that the k-means algorithm tries to optimize the objective function. As there is only a finite number of possible assignments for the amount of centroids and observations available and each iteration has to result in better solution, the algorithm always ends in a local minimum.

$$J = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{nk} ||x_n - \mu_k||^2$$

The main problem of k-means is its dependency on the initially chosen centroids. The centroids could end up in splitting data points whilst other, separated points will get unexpectedly clustered together if the centroids are attracted by outliers.

1. Take uniformly a random data point from the data X and mark it as centroid c_1
2. Choose another centroid c_i with the probability $\frac{D(x)^2}{\sum_{x \in X} D(x)^2}$ where $D(x)$ denotes the shortest distance from the data point x to its closest, already chosen centroid.

3. Repeat 2. until all k initial centroids are chosen.

The most common approach is to perform multiple clusterings with different start positions. Afterwards the clustering, the one occurred most is considered as correct. Another, newer approach is the so called k-means++ by Arthur and Vassilvitskii [10]. This extension to the k-means algorithm tries to distribute the initial centroids over the given data to minimize the probability of bad outcomes.

2.2.3 Latent Dirichlet Allocation

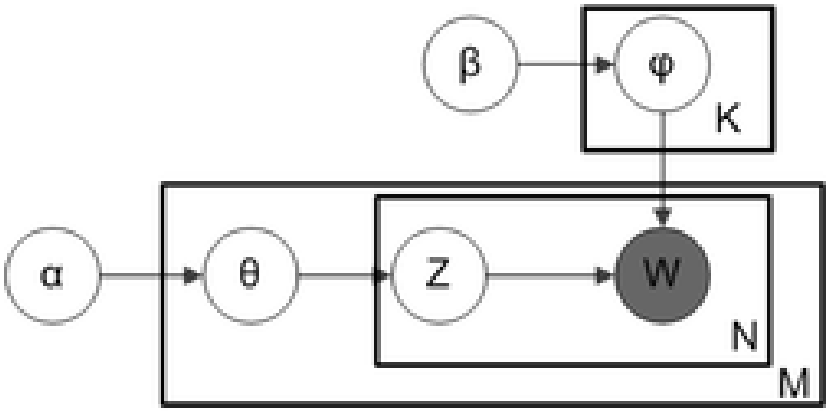


Figure 1: The number of publications per author

With plate notation, the dependencies among the many variables can be captured concisely. The boxes are "plates" representing replicates[8], which are repeated entities. The outer plate represents documents, while the inner plate represents the repeated word positions in a given document, each position is associated with a choice of topic and word. The variable names are defined as follows:

M denotes the number of documents,
N is number of words in a given document (document i has N_i words),
α is the parameter of the Dirichlet prior on the per-document topic distributions,
β is the parameter of the Dirichlet prior on the per-topic word distribution,
θ_i is the topic distribution for document i,
φ_k is the word distribution for topic k,
z_{ij} is the topic for the j -th word in document i,
w_{ij} is the specific word.

To actually infer the topics in a corpus, we imagine a generative process whereby the documents are created, so that we may infer, or reverse engineer, it. We imagine the generative process as follows. Documents are represented as random mixtures over latent topics, where each topic is characterized by a distribution over all the words. LDA assumes the following generative process for a corpus D consisting of M documents each of length N_i
1. Choose $\theta_i \sim Dir(\alpha)$, where $i \in \{1, \ldots, M\}$ and $Dir(\alpha)$ is a Dirichlet distribution with a symmetric parameter α which typically is sparse ($\alpha < 1$)
2. Choose $\varphi_k \sim Dir(\beta)$, where $k \in \{1, \ldots, K\}$ and β typically is sparse
3. For each of the word positions i, j, where $i \in \{1, \ldots, M\}$, and $j \in \{1, \ldots, N_i\}$
(a) Choose a topic $z_{i,j} \sim Multinomial\,(\theta_i)$.
(b) Choose a word $w_{i,j} \sim$ Multinomial $\left(\varphi_{i_{ij}}\right)$.
The lengths N_i are treated as independent of all the other data generating variables (w and z). The subscript is often dropped, as in the plate diagrams shown here.

2.2.4 Topic Coherence Score

The state-of-the-art in terms of topic coherence are the intrinsic measure UMass and the extrinsic measure UCI, both based on the same high level idea. Both measure compute the sum

$$Coherence = \sum_{ij} score(w_i, w_j)$$

of pairwise scores on the words $w_1, ..., w_n$ used to describe the topic, usually the top n words by frequency $p(w|k)$. This measure can be seen as the sum of all edges on complete graph. The UMass measure uses as pairwise score function

$$score_{UMass}(w_i, w_j) = \log \frac{D(w_i, w_j) + 1}{D(w_i)}$$

which is the empirical conditional log-probability $\log p(w_j|w_i) = \log \frac{p(w_i,w_j)}{p(w_j)}$ smoothed by adding one to $D(w_i, w_j)$.

The score function is not symmetric as it is an increasing function of the empirical probability $p(w_j|w_i)$, where w_i is more common than w_j, words being ordered by decreasing frequency $p(w|k)$. So this score measures how much, within the words used to describe a topic, a common word is in average a good predictor for a less common word.

As the pairwise score used by the UMass measure is not symmetric, the order of the arguments matters. UMass measure is computing $p(rareword|commonword)$, how much a common word triggers a rarer word. However, in human word association, high frequency words are more likely to be used as response words than low frequency words. It would be interesting to understand the effect of this choice by doing more experiments and comparing the two options.

3 Results

3.1 Co-authorship Network Analysis

3.1.1 Data Preprocessing

Since the data is from the multiple sources, name standardization is necessary before further analysis. Our solution is to only preserve the first 5 letter of the first name. Furthermore, we removed the name of organizations. After standardization, there are 592636 authors in the dataset, and 359709 authors have published papers on COVID-19.

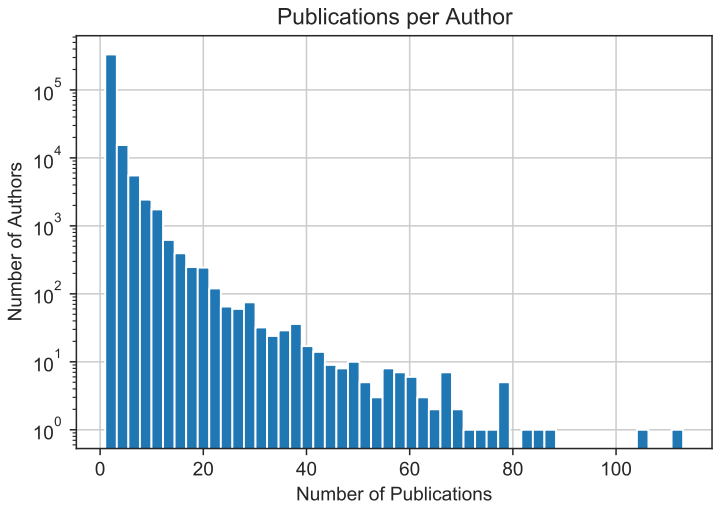


Figure 2: The number of publications per author

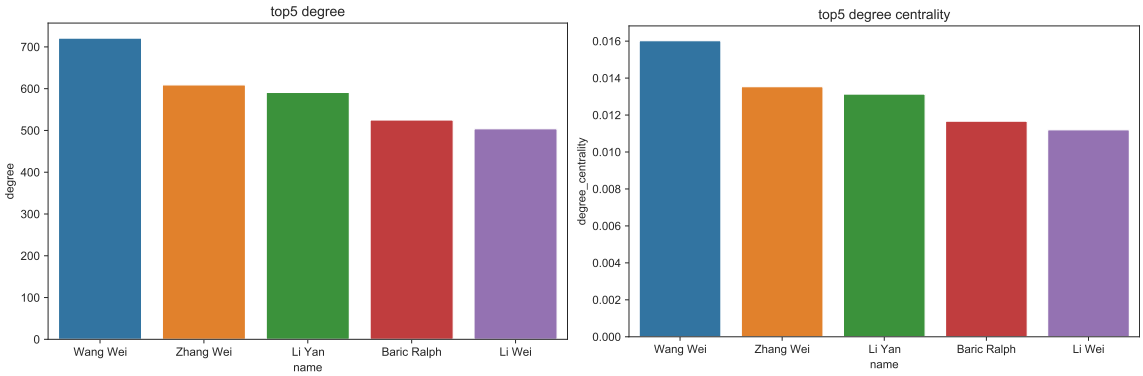
Figure 2 shows that most of authors just have published 1 or 2 papers on COVID-19, actually only 15% of the authors have published more than 3 papers and only 1% have published more than 10 papers. In order to save time and memory of the further analysis and preserve the connectivity, we just sampled the authors who have published more than 3 papers for network analysis. There are 47780 authors and 245673 related papers.

3.1.2 Network analysis

Overall statistics There are 47780 nodes and 643771 edges in the network composed of 1318 connected components. The largest connected component contains 94.18% nodes(44999) and 99.44% edges(640168). The following analysis is on the largest connected component.
The degree centrality for a node v is the fraction of nodes it is connected to, which is normalised by the maximum possible degree in a simple graph $n-1$ where n is the number of nodes in G.

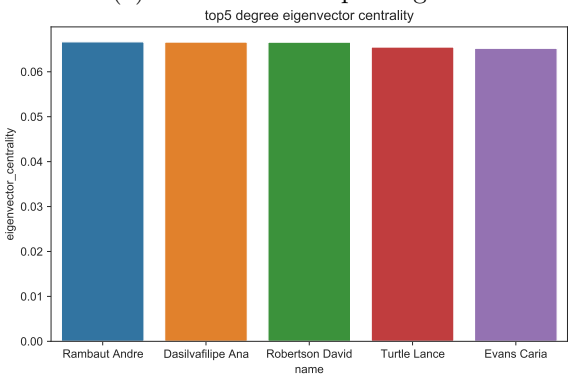
The eigenvector centrality or the power centrality of a node i is the i-th element of the eigenvector of the adjacency matrix which has the maximum eigenvalue.

Node statistics Figure 3 shows the top 5 authors in degree, degree centrality, eigenvector centrality and PageRanke value. Table 1 shows the generate descriptive statistics of nodes metrics.

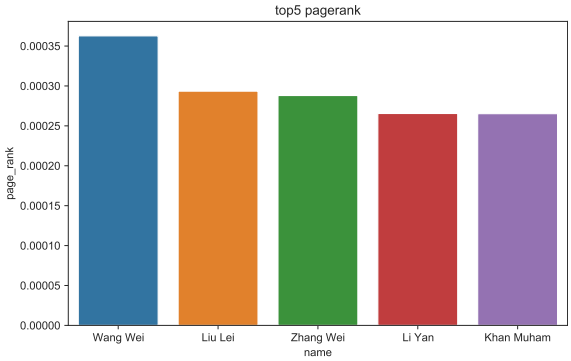


(a) Authors with top 5 degree

(b) Authors with top 5 degree centrality



(c) Authors with top 5 eigenvector centrality



(d) Authors with top 5 PageRank value

Figure 3: Top 5 authors

For the first two figures, we found that 4 out of 5 authors with top degree and centrality are Chinese. The other one is Ralph Baric, the so-called Dr. 'Coronavirus Hunter' by media, whose studies of Covid-19 are very famous. We needed to be aware that those 4 Chinese name are all very common names for Chinese so each of them might not represent only one researcher. However, even tough we knew about this, the data set did not have other support information for us to recognise researchers with same name written in English. As for eigenvector centrality, the top 5 authors differs from those of degree and degree centrality. This suggests that those authors publishes more important articles because the eigenvector centrality thesis reads: A node is important if it is linked to by other important nodes. That is to say, a node receiving many links does not necessarily have a high eigenvector centrality. When we looked up these authors online, we found that their articles had a high number of references, including by some equally important articles.

	degree	degree centrality	eigenvector centrality	PageRank value
mean	28.452543	0.000632	0.000418	0.000022
std	39.029366	0.000867	0.004696	0.000016
min	1.000000	0.000022	0.000000	0.000003
25%	8.000000	0.000178	0.000000	0.000013
50%	16.000000	0.000356	0.000001	0.000019
75%	33.000000	0.000733	0.000009	0.000026
max	721.000000	0.016023	0.066655	0.000363

Table 1: Node Statistics

3.1.3 Co-author Community Detection

We founds 109 partitions of this co-authorship network by Louvain method. According Figure 4, actually most of the author are in the partition indexed 1.

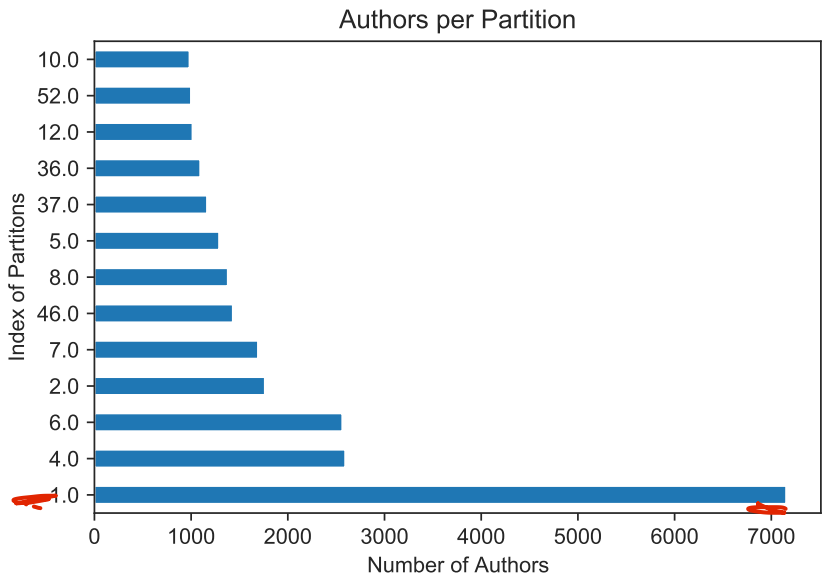


Figure 4: Top 10 populated community

We sampled and plotted (Linked Here) top 10 author (according to their degree) from the top 10 populated community.

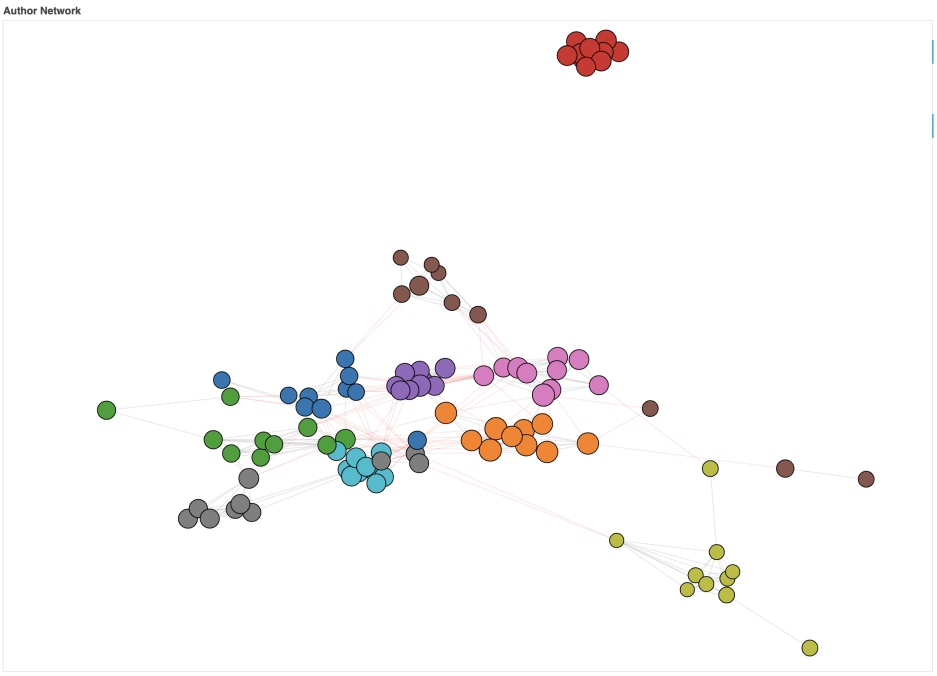


Figure 5: Network Visualisation

From the visualisation 5, there are some interesting findings: 1. All the authors from the red and orange community are from China. However, we can see that the authors from red

community just preserve the first letter of their first name while the authors from orange one meets the general format, which implies that these publications are from different sources. 2. The authors from the purple and pink community are from America. In addition, most of the authors from the purple one are virologists or biologists, for example, Ralph Baric, PhD are one of the leading scientists in coronaviruses, while the authors from pink one are Physicians or working for Centers for Disease Control and Prevention.

3.2 Clustering

3.2.1 PCA, K-Means and t-SNE on Abstract

As the first step towards creating an insightful model, we performed document clustering to group similar research articles together for simplifying related publications search. To achieve that, we first applied Principle Component Analysis to preprocessed data to reduce the dimension of document while maximally keeping the variance. We then used K-Means to label the documents with a loss function of sum of square errors (SSE). To visualize the results, we applied t-SNE to map the features into two-dimensional.

3.2.2 Abstract Preprocessing

After initial screening, we noticed several problems with the dataset, including multiple missing values, duplicate entries and multiple languages. We removed all entries with empty abstract and kept one entry in each group of duplicates. To detect languages, we used langdetect in Python and removed all entries in languages other than English. As the Figure 6 shown below, around 98.3 percent of abstracts were written in English.

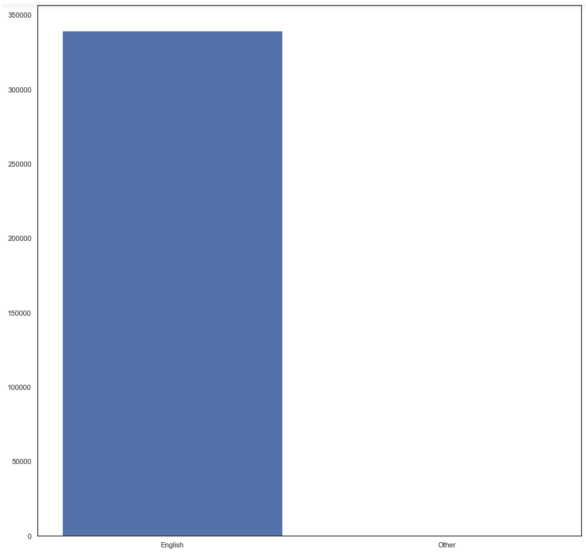


Figure 6: Language Distribution

To prepare for dimension reduction, we used word_tokenize and stopwords from nltk to turn abstracts into a feature vector and to remove the stopwords. We then used TfidfVectorizer from sklearn to transform our feature vector.

3.2.3 Dimension Reduction

We utilized two types of dimension reduction methods. Before applying K-Means, we decided to reduce the dimensionality of our data, which may temper the effects of curse of dimensionality on K-Means. We applied PCA on feature vectors while keeping 95% of variance.

We also applied t-SNE with two components on feature vectors and used these components as x and y axes, respectively.

3.2.4 Clustering Result and Interactive Plot

For the reason that there is not a effective way for us to find a suitable size of K. We used multiple starts with different K and judged by analysing the articles and key words inside each cluster. We found that when K is small such as K=20, some of the articles that are very likely to study on different fields are clustered together, while when K is big such as K=50, some of the clusters' keywords would be too limited to one or two words to actually makes sense. As a result, our final decision of K is 30. The clustering and T-SNE result are as follows.

Figure 7: Clustering Result

We also got the keywords of each topic by selecting those high-scoring words. There are some of the examples of keywords inside one cluster.The table of key words of all the clusters are in the appendix.

Cluster Index	Keywords
16	economic, policy, government ...
22	e-learning, universities, graduate ...
23	depression, anxiety, mental ...

Table 2: Examples of Cluster Keywords

From the keywords we actually got some idea about what articles inside these clusters are about. Cluster 16 is related to political and society effect of COVID-19 and 23 is very

likely to focus on psychological studies.

Given those features, what we created is an interactive search plot where user can browse the articles belong to each cluster and their keywords and filter them by typing one word that he/she likes.

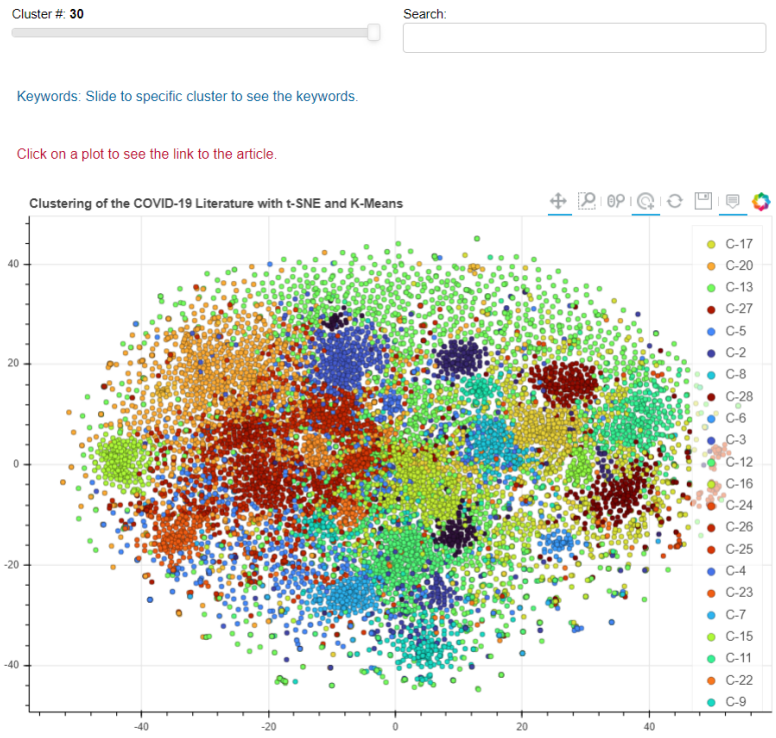
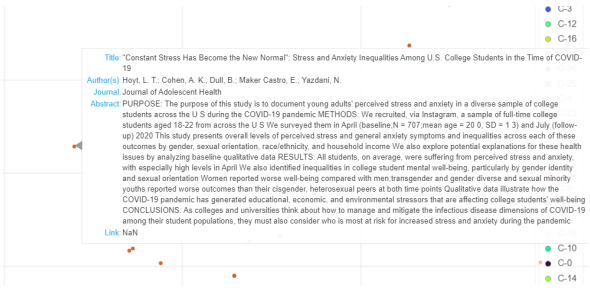


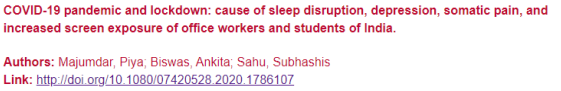
Figure 8: Interactive Plot

3.2.5 How to use this Plot?

The slider on the top-left is designed to help you view articles cluster by cluster, dragging it shows you keywords and articles of current cluster. After browsing the keywords, if you find something that you take interests in and want to narrow the search range a bit more, you can type any word you like in the top-right search bar such as "student". It will filter the articles and zoom in the plot for you. When you put your mouse onto the points, it shows the title, author and abstract. Clicking the one you would like to read and its title, authors and link will appear.



(a) Using Mouse to view Abstracts



(b) Clicking To Get The Link

Figure 10: How To Use The Interactive Plot

(a) Sliding To Browse Each Cluster

(b) Typing In Search Bar

3.3 Latent Dirichlet Allocation

As an algorithm that is frequently used in topic-modeling, Latent Dirichlet Allocation was performed in our analysis as well. After leaving out stop words and counting the word vector, we set the total number of components equals to 100 and ran LDA. Here is the top 10 topics LDA generated and their first 20 high-scoring keywords.

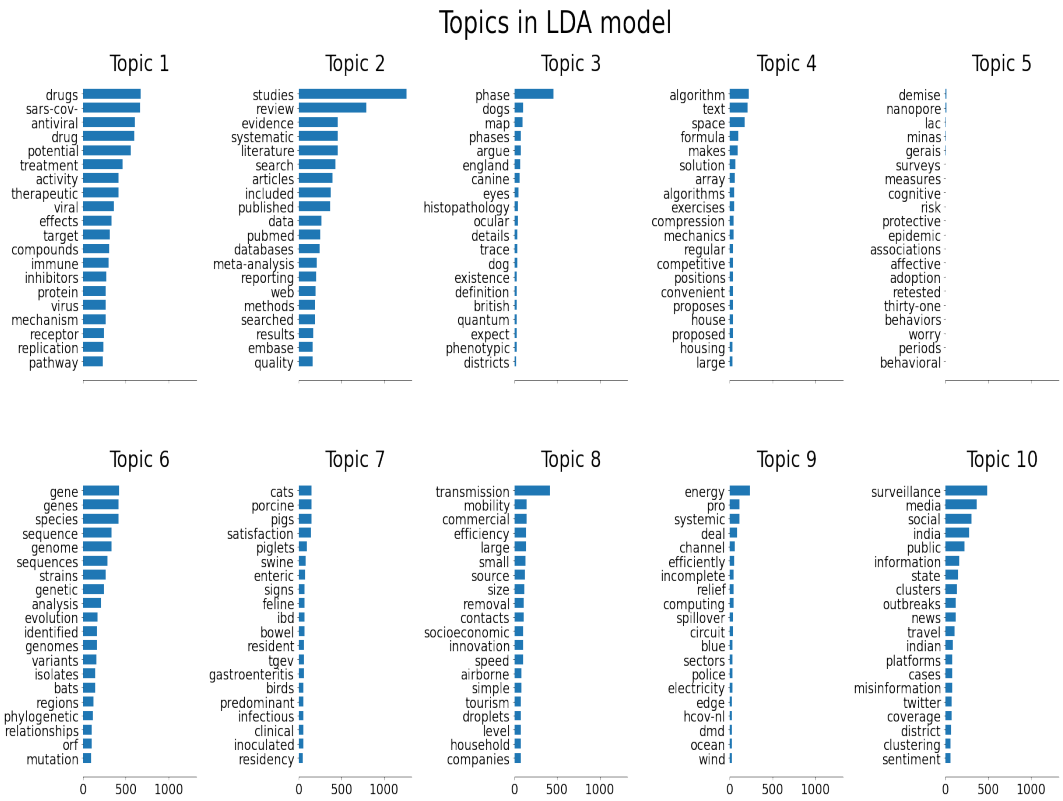


Figure 11: Top 10 topics for LDA of 100 components

We found out that there are some new topics that PCA+Clustering model did not find. In order to compare them more precisely, we took the 10 most frequently occurring topics in both algorithms and checked the keywords that they shared in common and unique words in each of them. To show the outcomes more clearly, we plotted this chart.

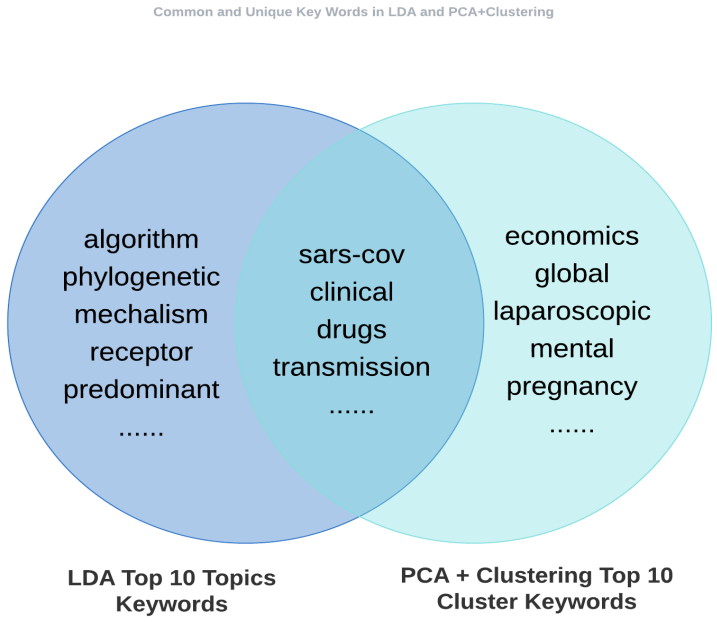


Figure 12: Common and Unique Words in both algorithms

From what we have seen, it is hard to say which one has better performance just by human judgment. We need a non-subjective approach to compare their results. As a result, we introduced topic coherence score. After comparing their U-Mass score, we came to a conclusion that their performances are very close because their scores are nearly equal.

height	PCA+Clustering	LDA
U-Mass Score	-18.768062824707048	-18.57168277874653

4 Discussion

4.1 Co-authorship Network Analysis

According to the result of network analysis and community detection, we found that scientists all over the world collaborate during this pandemic period. Over 25% of the scientists in the dataset has collaborated with more than 30 other researchers. Therefore, it could explain that specific researchers have such the frequent number of publications.

As the result of centrality analysis shows, we found some influential scientists play a core role in the co-authors' network, for example, Dr. Ralph Baric, Dr. Andrew Rambaut etc. Based on the result of community detection which has total 109 communities, we have visualised some centroid locating authors in the populated community and confirmed that this result is reasonable.

However, there are still some limitations on the co-author network analysis. First, we use authors' name as the unique keys, while some authors may share the same name, especially Chinese authors, for example, in the leading authors, Wei Wang, Wei Zhang, Wei Li are common name in China. It needs a better method to eliminate this influence to have more accurate results.

4.2 Topic-mdoeling with PCA+Clustering and LDA

The prepossessing procedure can be improve if we further pre-truncate the words. In other words, plurals and different tenses of verbs should be removed and be seen as the same word or same root. In our result so far, we have seen 'study' and 'studies' both show up in the same cluster, which is supposed to be settle for a better result in the future.

Due to computational limitation, we cannot analyse the full-length articles. It might losses some important components to only analyse the abstracts. Furthermore, we haven't found an effective approach to determine the number of clusters for K-means. 30 is a number that we accepted for now based on human judgment.

Having generated some useful topics with both PCA+Clustering and LDA, we compared their performance and found out that their coherence scores are very close. Two algorithms have their own pros and cons. PCA can tell us which article belongs to which cluster but it needs more prepossessing and depends more on chance (depends on the initial centroids). As for LDA, it is more reliable for topic modeling and needs less prepossessing. However, LDA is not able to give one single topic to each article in the data set, which makes it unsuitable for building an interactive search tool like the plot that we built.

References

[1] Wang, L.L., Lo, K., Chandrasekhar, Y., Reas, R., Yang, J., Eide, D., Funk, K., Kinney, R.M., Liu, Z., Merrill, W., Mooney, P., Murdick, D., Rishi, D., Sheehan, J., Shen, Z., Stilson, B.B., Wade, A.D., Wang, K., Wilhelm, C., Xie, B., Raymond, D.M., Weld, D.S., Etzioni, O., Kohlmeier, S. (2020). CORD-19: The Covid-19 Open Research Dataset. ArXiv.

[2] Blondel, Vincent D, Guillaume, Jean-Loup, Lambiotte, Renaud and Lefebvre, Etienne."Fast unfolding of communities in large networks." Journal of Statistical Mechanics:Theory and Experiment 2008

[3] Claude Sammut, Geoffrey I. Webb. "Encyclopedia of Machine Learning." Springer, Boston, MA. https://doi.org/10.1007/978-0-387-30164-8832

[4] Laurens van der Maaten and Geoffrey Hinton. "Visualizing Data using t-SNE ." Journal of Machine Learning Research 9 (2008): 2579–2605.

[5] Jolliffe I. "Principal Component Analysis." In: Lovric M. (eds) International Encyclopedia of Statistical Science. Springer, Berlin, Heidelberg.(2011): https://doi.org/10.1007/978-3-642-04898-2455

[6] Xiaoming Liu, Johan Bollen, Michael L. Nelson, Herbert. "Co-authorship networks in the digital library research community" Information Processing Management. 2005

[7] Page, L., & Brin, S. "The anatomy of a large-scale hypertextual Web search engine". In Proceedings of the seventh international World-Wide Web conference(1998).

[8] Blei, David M.; Ng, Andrew Y.; Jordan, Michael I (January 2003). Lafferty, John (ed.). "Latent Dirichlet Allocation". Journal of Machine Learning Research. 3 (4–5): pp. 993–1022. doi:10.1162/jmlr.2003.3.4-5.993. Archived from the original on 2012-05-01. Retrieved 2006-12-19.

[9] MacQueen, J. B. (1967). Some methods for classification and analysis of multivariate observations. In L. M. Le Cam J. Neyman (Eds.), Proceedings of the fifth Berkeley symposium on mathematical statistics and probability (Vol. 1, pp. 281–297). California: University of California Press.

[10] Arthur, D., and Vassilvitskii, S. 2007. k-means++: the advantages of careful seeding. In SODA.

5 Appendix

#	Keywords
0	occlusion, aneurysms, thrombectomy, stroke, days, ischemic, embolization ...
1	laparoscopic, fluid, complications, surgical, pressure, hernia, renal, pain, ablation, mesh, weight ...
2	insomnia, patients, quality, depression, group, depressive, disorder, disturbances, body ...
3	respiratory, activity, infection, study, pedv, gene, virus, different, group, results, plasma, calves ...
4	mers-cov, cases, infection, outbreak, camels, antibodies, protein, covid-, cov, countries ...
5	pandemic, adults, healthcare, sentiment, factors, misinformation, covid-, young, public ...
6	participants, covid, study, research, data, studies, articles, women, health, nurses, surgical ...
7	drug, repositioning, targets, cov, network, sars, pathway, antibodies, therapeutics, risk ...
8	risk, covid-, sars, women, pregnancy, symptoms, score, mortality, liver, persons, rates, peak ...
9	covid-, foram, com, foi, care, sars-cov-, los, health, del, abstract, sobre, pandemia, spread ...
10	children, viruses, virus, outbreaks, study, vaccination, zoonotic, covid-, adults, cases, new ...
11	infection, cell, gene, lung, diseases, mechanisms, covid-, disease, increased, expression, ifn- ...
12	patient, case, right, images, patients, x-ray, disease, infections, pneumoniae, kawasaki, source ...
13	clinical, pediatric, syndrome, neurological, headache, healthcare, airway, metabolic, immunoglobulin ...
14	disease, interventions, information, different, results, supply, management, production, distribution ...
15	protein, rna, translation, proteins, orf, virus, sequences, leader, cell, frame, acid, spike ...
16	economic, crisis, global, food, policy, china, gender, chinese, urban, government, safety, ethical ...
17	covid-, sars-cov-, patients, pcr, respiratory, positive, rt-lamp, cats, antibody, sample, assays ...
18	virus, infection, patients, cell, covid-, human, inflammation, lung, lines, cancer, tumour ...
19	bat, evolution, expression, species, genetic, genome, pathogens, animal, birds, parasites, host ...
20	study, patients, health, children, clinical, treatment, different, analysis, review, countries ...
21	article, protected, reserved, rights, doi, corrects, copyright ...
22	learning, faculty, clinical, e-learning, virtual, universities, graduate, students, sessions, campus ...
23	depression, covid-, distress, study, anxiety, suicide, mental, health, care, methods, mood ...
24	protein, receptor, rbd, binding, human, model, disease, species, angiotensin, plasma, structure, bind ...
25	mortality, symptoms, pandemic, cancer, liver, injury, olfactory, ill, neurological, auc, treatment ...
26	visits, covid-, telehealth, healthcare, people, hospice, testing, dermatology, attitudes, rehabilitation ...
27	covid-, vaccines, sars-cov-, ibv, health, mice, epitopes, chickens, peptide, oral, vaccination ...
28	covid-, cases, deaths, vitamin, wuhan, air, china, temperature, medical, admissions ...
29	sars, covid-, infection, cov, sars-cov, protein, viral,delivery, sars-cov-, mothers, index ...

Table 3: Key Words List of All 30 Clusters