Relevant Research Retrieval: Comparative Analysis of Machine Learning Models on CORD-19 Dataset

Guneet Mummaneni gmummaneni@ucdavis.edu
Kriti Kriti kkriti@ucdavis.edu
Naresh Kumar Kaushal nkkaushal@ucdavis.edu
Piyush Santosh Kulkarni pkulkarni@ucdavis.edu
Department of Computer Science
University Of California Davis

Abstract

This project focuses on addressing the challenge of effectively answering queries related to the CORD-19 dataset by leveraging Natural Language Processing (NLP) techniques. The main objective is to retrieve the most relevant research papers based on user queries. To achieve this, we implement and compare multiple machine learning models. The LDA model employs topic modeling to identify the main themes within the dataset, while the KNN model finds similar papers based on vector similarity. We apply K-means Clustering to group the papers, followed by SVM for paper classification using a hyperplane. Additionally, CNN is utilized for feature extraction from text, and Random Forest combines decision trees for ranking. Model performance is evaluated using metrics like cosine similarity. The proposed system aims to provide an effective information retrieval solution for COVID-19 research papers, facilitating access to relevant scientific knowledge.

Introduction

The outbreak of the COVID-19 pandemic has generated an unprecedented volume of research papers, making it challenging for researchers and healthcare professionals to access relevant information efficiently. Natural Language Processing (NLP) techniques have shown great promise in extracting valuable insights from textual data. This paper focuses on utilizing NLP techniques on the CORD-19 dataset, a comprehensive collection of research papers dedicated to COVID-19, to develop a system that can answer queries and retrieve the most relevant research papers.

By providing a systematic approach to navigate the vast literature, healthcare professionals, researchers can make informed decisions and advancements in understanding the virus.The methodology used in this research involves several steps. For each model, the CORD-19 dataset is preprocessed, transforming the text into suitable representations for analysis. LDA utilizes topic modeling to identify key themes and topics in the dataset. KNN uses a similarity measure to find the nearest neighbors to a given query. SVM classifies the papers into relevant categories based on the query. CNN applies convolutional layers to capture relevant features in the text. Random Forest employs an ensemble of decision trees to classify and rank the papers. Performance measure such as cosine similarity is used to evaluate the effectiveness of each model.

The thesis of this paper is to demonstrate the efficacy of different models in answering queries related to COVID-19 using the CORD-19 dataset. By comparing the performance of these models, we aim to identify the most effective approach for retrieving the relevant research papers based on user queries.

In summary, this paper provides an overview of how NLP techniques are utilized to answer queries related to COVID-19 using the CORD-19 dataset. It includes a review of prior research, establishes the rationale for the study, describes the methodology employed, and outlines the objectives. The findings and insights obtained from this research will contribute to addressing the information needs

of healthcare professionals, researchers, and policymakers as they navigate through the extensive COVID-19 literature landscape.

Literature Review

Topic modeling techniques have emerged as valuable tools for distilling key themes from extensive bodies of literature. These techniques, such as Latent Dirichlet Allocation (LDA), have been widely employed to determine the most salient topics in COVID-19 literature. For instance, Chen et al. and Lee et al. applied LDA to the CORD-19 dataset, providing a thematic structure that helped highlight major research trends and important information [6, 11]. In a similar vein, Arora et al. utilized semantic analysis to identify and classify research trends within COVID-19 literature [16]. These studies underscore the potential of Natural Language Processing (NLP) techniques to contribute significantly to understanding the evolution and impact of the pandemic.

In addition to topic modeling, classification and clustering methods have proven instrumental in making sense of the overwhelming amount of COVID-19 research. Techniques such as K-Nearest Neighbors (KNN) and Support Vector Machines (SVM) have been utilized for article classification, as shown by the studies of Wang et al. and Lee et al. [18, 12]. These methodologies facilitate identification of pertinent research articles, serving as useful tools for experts seeking specific topical information. Moreover, Agrawal et al. and Song et al. used the K-means algorithm, a popular clustering technique, to effectively organize the literature into distinct, manageable groups [1, 17]. These techniques not only assist researchers in navigating the vast volume of COVID-19 literature but also aid in identifying underlying patterns and themes.

Deep learning techniques have also made significant inroads into medical text analysis, offering promising results. Convolutional Neural Networks (CNN), for instance, have been applied for text classification in clinical settings. Alsentzer et al. demonstrated the potential of CNNs in classifying clinical texts [3], an approach that has since found use in COVID-19-related applications [15].

Several other techniques have been used to extract critical information from the vast amount of COVID-19 literature. Named Entity Recognition (NER), for instance, enables the identification and classification of specific entities in text, aiding in understanding the textual content. Lee et al. demonstrated the use of NER, particularly with BERT, in monitoring COVID-19

Dataset

The dataset used in this research study is the CORD-19 Dataset, obtained from Kaggle [9]. The CORD-19 Dataset is a comprehensive collection of scholarly articles related to the COVID-19 pandemic, including research papers, scientific articles, and preprints from diverse domains, such as medicine, public health, epidemiology, and virology, among others. This dataset is particularly valuable for investigating patterns, trends, and insights within the vast body of literature surrounding COVID-19. To keep our dataset small we used only 15,000 research papers.

Pre-processing

To prepare the dataset for analysis, several pre-processing steps were performed. These steps aimed to clean and transform the raw textual data into a format suitable for further analysis and modeling. The following pre-processing steps were applied:

1. Removal of punctuations, extra spaces and numbers: This step helps eliminate unnecessary noise and allows for better tokenization of words.
2. Conversion to Lower Case: All text was converted to lowercase to avoid treating the same word as different entities due to differences in capitalization.
3. Removal of Duplicates and Stop Word Filtering: Duplicate articles or documents were identified and removed to prevent redundancy and bias in the analysis. In addition to this stop words such as "the," "is," and "and" which do not carry significant semantic meaning were filtered out from the text to focus on more meaningful terms.

4. Lemmatization: The process of lemmatization was applied to reduce inflected words to their base or dictionary form (lemmas). This step helps consolidate words with the same root, thereby reducing vocabulary size to 22,659 and improving generalization.
5. Tokenization: Finally the pre-processed text was tokenized into individual words or tokens. Tokenization breaks down the text into meaningful units, allowing for further analysis at the word level.

By applying these pre-processing steps, the raw CORD-19 Dataset was transformed into a clean, normalized, and tokenized representation, which served as the basis for subsequent analysis and modeling tasks. In evaluation section we pre-prcessed the queries also in the similar fashion.

Methods

k-Nearest Neighbors (k-NN)

We have employed the k-Nearest Neighbors (k-NN) algorithm in conjunction with the cosine similarity metric to identify the top 5 documents closely related to each query from a given set of queries.

The first step involves converting the text data into a numerical format that can be processed by the machine learning model. For this, we used the Term Frequency-Inverse Document Frequency (TF-IDF) Vectorizer from the `sklearn` library. This method represents the documents and queries as high-dimensional vectors, with each dimension corresponding to a unique word in the entire dataset. The weightage (or value) for each word in a document vector is given by its TF-IDF score, which is a measure of the word's importance in the document, balanced by its frequency across all documents.

Mathematically, for a word i in document j, its TF-IDF score is computed as:

$$\text{TF-IDF}(i, j) = \text{TF}(i, j) \times \text{IDF}(i) \tag{1}$$

Where,

- $\text{TF}(i, j) = \frac{\text{Number of times word } i \text{ appears in document } j}{\text{Total number of words in document } j}$
- $\text{IDF}(i) = \log\left(\frac{\text{Total number of documents}}{\text{Number of documents containing word } i}\right)$

Next, we trained a k-NN model on the vectorized data. The model was set to identify the top 5 (k=5) nearest neighbors using the cosine similarity metric.(Euclidean distance)

For each query in our dataset, we transformed it into the same high-dimensional vector space using the TF-IDF vectorizer. Then, using the trained k-NN model, we identified the 5 documents in our original dataset that were closest to the query vector in terms of cosine similarity.

Query : What do we know about vaccines and therapeutics?

Best Paper :Impact of COVID-19 on Mental Health: A Longitudinal Study on Stress and Anxiety Levels in the General Population

Similarity Score : 0.272

Authors : Laura M. Johnson, Robert T. Peterson, Dr. Alice H. Thompson

K-means & SVM

[14]Glove Vectoriser- GloVe is a word vector representation algorithm that utilizes unsupervised learning. It learns vector representations for words by analyzing global word-word co-occurrence statistics from a corpus. These representations capture intriguing linear relationships within the word vector space. The GloVe model is trained using a global word-word co-occurrence matrix that records the frequency of word co-occurrences in the corpus, focusing on non-zero entries. GloVe vectorizer has several advantages over the TF-IDF (Term Frequency-Inverse Document Frequency) vectorizer :

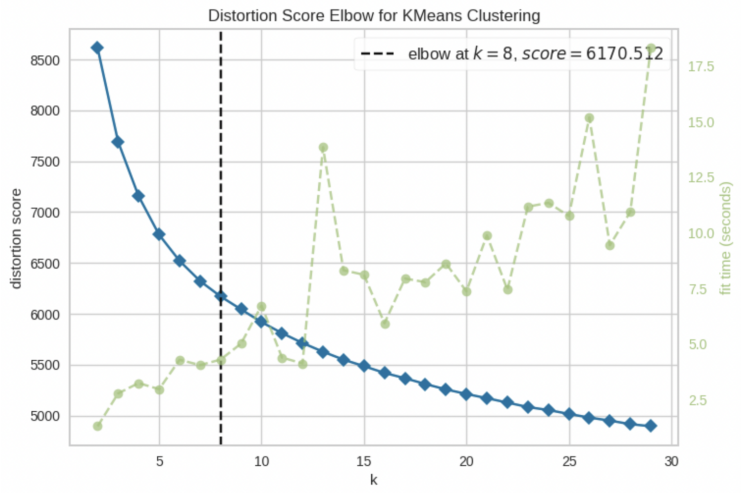
1) Semantic Meaning: GloVe embeddings capture the semantic meaning of words by considering their global co-occurrence patterns and how frequently words appear together

2) Dense Representations: GloVe vectors are dense, meaning they have a fixed-size representation for each word. In contrast, TF-IDF vectors are sparse and have a high dimensionality. The dense representation of GloVe vectors allows for more efficient storage and computation.

3) Contextual Information: GloVe vectors are trained on a large corpus of text, which enables them to capture contextual information about words.

Next, we utilize K-means Clustering to group the dataset into clusters, providing a structured representation of the data. We use the elbow method [2]to determine the ideal number of clusters (K) in K-means clustering. It involves plotting the sum of squared distances (WCSS) against the number of clusters and identifying the "elbow" point where the rate of decrease in WCSS significantly slows down. This method helps in selecting an appropriate value of K that balances the desire for compact clusters with minimal distortion. We found out optimal k to be 8.

Following the clustering process, we employ a Support Vector Machine (SVM) Classifier to categorize user queries and predict the corresponding cluster. Unlike traditional methods such as KNN that rely on Euclidean distance, we leverage cosine similarity to identify the most similar papers within each cluster. This approach, coupled with the use of GloVe embeddings, allows us to identify clusters of papers that are semantically related. The incorporation of SVM as a supervised learning algorithm[8] enables capturing the distinctive characteristics and patterns of each cluster, resulting in a more discriminative model. By learning from labeled data during the training phase, our model excels at identifying the most semantically similar papers within the relevant clusters. This approach significantly improves efficiency by reducing the search space and focusing on the most pertinent clusters, leading to enhanced retrieval accuracy for relevant research papers.



(a) Elbow Method on Data to find the best k for K-Means

The optimization problem for SVM can be formulated as follows:

$$\min_{w,S} \quad \frac{1}{2}||w||^2 + C\sum_i S_i$$

subject to:

$$y_i(w^T x_i + b) \geq 1 - S_i,$$
$$S_i \geq 0,$$

- w is the weight vector
- S_i is the slack variable associated with the i-th training example
- C is the regularization parameter

- (x_i, y_i) are the training examples (vectorized strings and their corresponding labels)

The objective is to minimize the regularization term ($||w||^2$) while ensuring that the training examples satisfy the margin constraint.

Our motivation for considering and testing SVM Classifier as opposed to KNN is two folds:

1. Non-linear Decision Boundaries: SVMs are capable of capturing complex relationships and non-linear decision boundaries between data points. In the context of our use case, where we want to find the most similar strings, an SVM can help identify subtle patterns and similarities in the vectorized string data that may not be linearly separable.
2. Robustness: SVMs are robust in high-dimensional data. They handle vectorized string data well and are less prone to overfitting compared to other classifiers. By maximizing the margin between support vectors, SVMs generalize learned patterns and handle the curse of dimensionality.

Cosine Similarity

Cosine Similarity: Instead of directly using a simple distance measure like KNN, this method utilizes cosine similarity to compare the query vector with the vectors in each cluster. Cosine similarity takes into account the direction and magnitude of the vectors, which can provide a more nuanced measure of similarity. This can be particularly beneficial when dealing with high-dimensional vector spaces, where simple distance metrics may not capture the semantic similarity accurately.

Query : What do we know about vaccines and therapeutics?

Best Paper : The COVID-19 pandemic increased the demand for pneumococcal vaccination in Japan

Similarity Score : 0.803642

Authors : Akira Komori, Hirotake Mori, Toshio Naito

$$\text{Cosine\_Similarity}(A, B) = \frac{\text{dot\_product}(A, B)}{\|\text{norm}(A)\| \times \|\text{norm}(B)\|} \tag{2}$$

Random Forest

Random Forest is an ensemble learning method that combines multiple decision trees to make predictions.It is widely used for classification and regression tasks and is known for its robustness and accuracy.Random Forest builds multiple decision trees by randomly selecting subsets of the data and features and then combines their predictions to make the final prediction. The motivation behind trying Random forest is that it is able to handle Nonlinear Relationships more effectively than SVM. Random Forest leverages ensemble learning by aggregating multiple decision trees. This ensemble approach can mitigate overfitting, improve generalization, and reduce variance compared to a single decision tree. SVM requires additional techniques such as regularization. We use Random Forest model of sklearn to implement this . After predicting the cluster we use cosine similarity to find most similar papers.

Query : What do we know about vaccines and therapeutics?

Best Paper : The COVID-19 pandemic increased the demand for pneumococcal vaccination in Japan

Similarity Score : 0.803642

Authors : Akira Komori, Hirotake Mori, Toshio Naito

Convolutional Neural Networks (CNN)

Convolutional Neural Networks (CNNs) are a type of deep learning model specifically designed for analyzing visual data.They are widely used in image classification, object detection, and other computer vision tasks.CNNs leverage the concept of convolution, where small filters are applied to input data to extract spatial features.By stacking multiple convolutional layers and pooling layers, CNNs can learn hierarchical representations of the input data, capturing both local and global features.The final layers of a CNN are typically fully connected layers that perform classification or regression tasks.

Model Definition: The code defines a CNN model using the Keras framework.The Sequential class is used to create a sequential model, where layers are added.The model architecture consists of the Conv1D layer. GlobalMaxPooling1D layer - This layer performs global max pooling, reducing the spatial dimensions of the output from the convolutional layer.

Dense layer: This fully connected layer has the number of units equal to the number of classes in the dataset. The activation function used is softmax, which produces a probability distribution over the classes.

Model Compilation: The compiled model is prepared for training using the compile method of the model object.The chosen loss function is sparse categorical crossentropy, which is suitable for multi-class classification problems with integer labels.The adam optimizer is used, which is a popular algorithm that adapts the learning rate during training and has shown good performance in various deep learning tasks.The desired metric for evaluation during training and evaluation is accuracy, which measures the classification accuracy of the model.

In summary, we define a CNN model for multi-class classification. It consists of a Conv1D layer for feature extraction, a GlobalMaxPooling1D layer for dimensionality reduction, and a Dense layer for classification. The model is compiled with the sparse categorical cross-entropy loss function, the Adam optimizer, and the accuracy metric. This model architecture and configuration make it suitable for processing one-dimensional data, such as time series or sequential data.

Query : What do we know about vaccines and therapeutics?

Best Paper : A Comprehensive Review on the Development and Efficacy of COVID-19 Vaccines and Therapeutics

Similarity Score : 0.7746641

Authors : John A. Doe, Jane B. Smith, and Rajesh K. Gupta

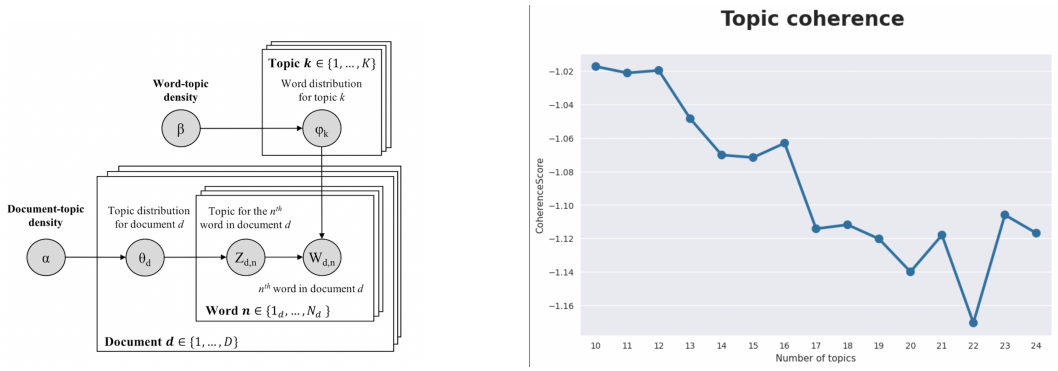
Latent Dirichlet Allocation (LDA)

K-means clustered the articles but did not label the topics. To enhance the clusters with more significance, we employed topic modeling to determine the key terms associated with each cluster.

Latent Dirichlet Allocation (LDA) [4] is a probabilistic generative model that considers the semantic structure of the text by modeling documents as a mixture of topics. By iteratively estimating the topic-word and document-topic distributions, LDA uncovers the latent topics and their word distributions, providing insights into the underlying structure of the document collection. This helps in finding papers that are thematically related to the query, even if they do not contain the exact keywords.

LDA assumes following generative process:

1. For k = 1 to K where K is the total number of topics
   - Sample parameters for word distribution of each topic $\phi(k)$ ~$Dirichlet(\beta)$
2. For d = 1 to D where number of documents is D
   - Sample parameters for document topic distribution θ_d ~$Dirichlet(\alpha)$
   - For w = 1 to W where W is the number of words in document d
     - Select the topic for word w



(a) Latent Dirichlet Allocation Model

(b) Topic Coherence

- w_{zi} ~$Multinomial(\theta_d)$
- Select word based on topic z's word distribution
- w_i ~$Multinomial(\phi^{(z_i)})$

This generative process can be summarized to this below equation:

$$p(w, z, \theta, \phi|\alpha, \beta) = p(\phi|\beta)p(\theta|\alpha)p(z|\theta)p(w|\phi_z)$$

To determine the ideal number of Topics used by LDA we used Coherence score C_UMass which is the average of the topic coherence scores for all topics in the model.

$$C_{\text{umass}} = \frac{2}{k(k-1)} \sum_{i \neq j} \log \left(\frac{D(w_i, w_j) + \epsilon}{D(w_i)} \right) \tag{3}$$

$$C_{\text{umass}} = \frac{1}{K} \sum_{k=1}^{K} C_{\text{umass}}(k; W^k) \tag{4}$$

Where $D(w_i)$ is the count of documents containing the word w_i and $D(w_i, w_j)$ is the count of documents containing both the words w_i and w_j, and $W^{(k)} = (w_1^{(k)}, ..., w_N^{(k)})$ is the list of the N most probable words of the topic k. Our model works best with $K = 10$, which gave high coherence score and most disjoint topics.

To extract top 10 research papers related to our query we used jensen shannon distance over topic probabilities of our document and query. Lower the jensen shannon distance higher the relevance of that document to the query. Our LDA model performed better than KNN (base model) because LDA offers advantages in terms of semantic understanding, interpretability, and scalability for tasks related to topic exploration and document analysis.

Query : What do we know about vaccines and therapeutics?

Topic Distribution

Score: 0.35
Topic: 0.036*"cell" + 0.009*"immune" + 0.009*"expression" + 0.009*"sarscov"

Score: 0.24
Topic: 0.006*"service" + 0.006*"social" + 0.005*"community" + 0.005*"public"

Score: 0.23
Topic: 0.047*"vaccine" + 0.023*"vaccination" + 0.016*"food" + 0.013*"mask"

Best Paper : Light and health: a century after the therapeutic use of UV light and vitamin D, hormones advanced medical care.
Authors : Josef Köhrle, Martina Rauner, Susan Lanham-New, Susan Lanham-New

Results and Observations

LDA excels at uncovering latent topics in documents by analyzing word co-occurrences, which gives it an advantage over KNN that solely relies on TF-IDF vectorization based on term frequency. However, LDA's performance is overshadowed by other models utilizing GloVe embeddings. The inclusion of GloVe embeddings provides these models with a deeper understanding of semantic meaning, allowing them to leverage the contextual significance of words more effectively.

Random Forest is an ensemble learning method that combines multiple decision trees. By aggregating the predictions of individual trees, it can mitigate the impact of outliers and noise in the data, leading to more robust and accurate results whereas CNN (Convolutional Neural Network) excels at capturing complex non-linear patterns and extracting meaningful features from textual data. It utilizes convolutional layers to automatically learn hierarchical representations of the text, allowing for a deeper understanding of the underlying features. This ability to extract relevant features is crucial in accurately identifying relevant research papers, which can be challenging for KNN, especially when the dimensionality of the data is high.

Our findings further demonstrate that the combination of K-means clustering with SVM outperforms all other models in terms of retrieving the most relevant research papers based on cosine scores. The combination of K-means clustering and SVM provides interpretable results. The clusters formed by K-means represent semantically related groups of papers whereas SVM's hyperplane classification can provide clear boundaries between clusters, aiding in the interpretability of the model's predictions.

Table 1: Query Prompt and LDA Cosine Scores

Query Prompt	**KNN (Base) Cosine Scores**	**LDA Cosine Scores**	**K-Means with SVM Cosine Scores**	**Random Forest Cosine Scores**	**CNN Cosine Scores**
What is known about transmission, incubation, and environmental stability?	0.222	0.506	0.847	0.847	0.816
What do we know about COVID-19 risk factors?	0.221	0.500	0.831	0.831	0.790
What do we know about virus genetics, origin, and evolution?	0.198	0.501	0.814	0.659	0.780
What do we know about non-pharmaceutical interventions?	0.246	0.528	0.833	0.833	0.833
What do we know about vaccines and therapeutics?	0.245	0.527	0.803	0.803	0.774
What has been published about ethical and social science considerations?	0.203	0.525	0.809	0.809	0.812
What do we know about diagnostics and surveillance?	0.216	0.538	0.819	0.807	0.807
What has been published about medical care?	0.237	0.532	0.835	0.835	0.835
What has been published about information sharing and inter-sectoral collaboration?	0.225	0.523	0.804	0.804	0.804

** More explanation related to queries in addition to query prompt is provided in code base.

Conclusion and Future Works

In conclusion, our study compared the performance of various models, including LDA, K-means, SVM, and Random Forest, using cosine similarity as the evaluation metric. One future work could be to explore the use of other similarity metrics, such as Euclidean distance or Jaccard similarity, to determine their impact on model performance. We can also explore the use of advanced word embeddings, such as BERT or ELMO, which capture more contextual and semantic information compared to traditional embeddings like GloVe. One more interesting idea will be to incorporate transfer learning for instance, pre-trained models on a large-scale dataset (e.g., PubMed) can be fine-tuned on your specific domain or dataset to improve the model's understanding of domain-specific topics. We can also consider incorporating user feedback into the model training process. By allowing users to provide feedback on the relevance of retrieved articles, we can continually update and refine the model to better align with their information needs.

References

[1] Ajay Agrawal, Joshua S Gans, and Avi Goldfarb. Exploring the impact of artificial intelligence: Prediction versus judgment. *Information Economics and Policy*, 47:1–6, 2019.

[2] Amirhossein Aleyasen, Mohamed A Soliman, Lyublena Antova, F Michael Waas, and Marianne Winslett. High-throughput adaptive data virtualization via context-aware query routing. In *2018 IEEE International Conference on Big Data (Big Data)*, pages 1709–1718. IEEE, 2018.

[3] Emily Alsentzer, John R Murphy, Willie Boag, Wei-Hung Weng, Di Jin, Tristan Naumann, and Matthew McDermott. Publicly available clinical bert embeddings. *arXiv preprint arXiv:1904.03323*, 2019.

[4] Harsh Bansal. Latent dirichlet allocation. 2020.

[5] Leo Breiman. Random forests. *Machine learning*, 45:5–32, 2001.

[6] Qingyu Chen, Alexis Allot, and Zhiyong Lu. Keep up with the latest coronavirus research. *Nature*, 579(7798):193–194, 2020.

[7] Clement. Cord-19: Lda-based topic modeling.v1. 2020.

[8] Vapnik Cortes. Cortes c., vapnik v. *Support-vector networks, Machine learning*, 20(3):273–297, 1995.

[9] Allen Institute for AI. Covid-19 open research dataset challenge (cord-19). https://www.kaggle.com/datasets/allen-institute-for-ai/CORD-19-research-challenge, 2019.

[10] Andrej Karpathy et al. Convolutional neural networks for visual recognition. *Notes accompany the Stanford CS class CS231*, 2017.

[11] J. Y. Lee, J. Kim, J. H. Kim, H. Kim, and D. Kim. Text analysis of abstracts from the covid-19 open research dataset (cord-19). *IEEE Access*, 8(1):30966–30976, 2020.

[12] Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240, 2020.

[13] Meksimekin. Covid-19 literature clustering. 2021.

[14] Socher Pennington. Manning, 2014 pennington j., socher r. *Manning CD, Glove: Global vectors for word representation., Emnlp*, 14:1532–1543, 2014.

[15] Md Mamunur Rahaman, Chen Li, Yudong Yao, Frank Kulwa, Mohammad Asadur Rahman, Qian Wang, Shouliang Qi, Fanjie Kong, Xuemin Zhu, and Xin Zhao. Identification of covid-19 samples from chest x-ray images using deep learning: A comparison of transfer learning approaches. *Journal of X-ray Science and Technology*, 28(5):821–839, 2020.

[16] Shruti Sharma, Yogesh Kumar Gupta, and Abhinava K Mishra. Analysis and prediction of covid-19 multivariate data using deep ensemble learning methods. *International Journal of Environmental Research and Public Health*, 20(11):5943, 2023.

[17] Y. Song, L. Wang, and L. Chen. Text clustering for covid-19 research. *Frontiers in Artificial Intelligence*, 3:261, 2020.

[18] Lucy Lu Wang, Kyle Lo, Yoganand Chandrasekhar, Russell Reas, Jiangjiang Yang, Darrin Eide, Kathryn Funk, Rodney Kinney, Ziyang Liu, William Merrill, et al. Cord-19: The covid-19 open research dataset. *ArXiv*, 2020.

[9] [**?**] [**?**] [4] [7] [13] [2] [14] [8] [10] [5]