Data Mining Tool To Help The Scientific Community Develop Answers To Covid-19 Queries

Rajath S*, Amit Kumar†, Mayank Agarwal‡,Sanjana Shekar § and VR Badri Prasad ¶
Department of Computer Science, PES University
Bangalore, India
*rajaths2510@gmail.com, †amitk526188@gmail.com, ‡mayankagarwal44442@gmail.com,
§sanjana.shekar99@gmail.com, ¶badriprasad@pes.edu

2021 Fifth International Conference On Intelligent Computing in Data Sciences (ICDS) | 978-1-6654-4238-1/21/$31.00 ©2021 IEEE | DOI: 10.1109/ICDS53782.2021.9626771

***Abstract*—Science has time and again proven to be one of the most powerful tools in finding solutions to the problems faced by the world. Let it be natural or man-made challenges, hard work put into finding efficient answers to tackle them has proven to safeguard the ecosystem. Sometimes the research community is put under pressure when humanity faces the challenge of survival like the Covid-19 pandemic. A great extent of published works needs to be studied to find an optimal solution to existing or new queries related to the virus. In this research work, we build an efficient data mining tool using the CORD-19 Dataset to help the community come up with answers to Covid-19 related questions. We use a combination of semantic and keyword search to reduce the solution space of our model. Our model makes use of parallelism, paraphrasing, and state-of-the-art natural language processing techniques which will serve as a time and energy-saving tool for the information need of all doctors and researchers who are trying to put an end to the pandemic and avoid future possible outbreaks.**

***Index Terms*—Bidirectional Encoder Representations from Transformers(BERT), Clustering, Paraphrasing, Natural Language Processing, Embeddings, CORD-19 Dataset, Deep Learning, Keyword Search, Semantic Search**

I. Introduction

With the growing world, humanity had to face multiple challenges and find efficient solutions for the same to win against them. Over the years, we have seen various unfortunate disasters occurring, one of which are diseases and illnesses. In 2020, we saw life coming to a standstill due to the Covid-19 pandemic. The outbreak started in November 2019 in Wuhan, China, and slowly found its way all over the world. The number of people infected and all the lives lost has put the whole world in grief.

A little about the virus, Coronaviruses are a group of RNA viruses that causes a range of diseases that have been proven to be fatal. Affecting mammals and birds by causing respiratory tract infections, these viruses have in the past proven to be lethal and difficult to cure. The first strain of coronavirus was seen in the 1920s in animals and the latest being Covid-19 in 2019.

With a wide range of common symptoms in humans like fever, dry cough, diarrhea, breathlessness, etc, the varieties of coronaviruses include:

- Severe Acute Respiratory Syndrome or SARS, infected more than 8000 people and 774 lost their lives in 2003.
- Middle East Respiratory Syndrome or MERS, occurred in 2013 and lasted till 2015 having a mortality rate of 34.
- Coronavirus Disease 2019 or SARS-CoV-2 or Covid-19, declared as a pandemic in March 2020, has infected more than 60 million people across the world and has claimed over 1 million lives in a course of one year

The scientific community is working hard in the battle against this virus giving their absolute best by working round the clock in stopping the transmission as well as finding a cure and developing vaccines to prevent them. Output on the virus has put a substantial strain on clinicians and researchers and others who must use this literature in the battle. Over 230,000 articles and information has been found provided by the World Health Organization(WHO) alone making conventional reading a challenge for everyone.

In such scenarios, automation using data mining addresses this wave of input. Data mining manages information overload with relevant document retrieval, information retrieval, text classification, and many more techniques. Traditional search techniques like TF-IDF [1] and BM25 [2] don't consider context, but are quick and serve the purpose of keyword search. Meanwhile, semantic search obtains the information that isn't explicitly written in the query. In this paper, we build a novel search engine using a combination of keyword as well as semantic search on a corpus (CORD-19 [3]) related to Covid-19 to help reduce the time and effort of a scientific individual in extracting relevant articles to find accurate answers for a Covid-19 related question or query.

This research work was mainly motivated by the abundance of COVID-19 related articles and the difficulty medical researchers face to find the right answers to their questions. Building a search tool that is efficient in terms of time and accuracy is the need of the hour and serves this purpose.

II. Related Work

Document classification has been a well-known area of research in terms of data mining. Many search engines make use of keyword search and topic modeling for finding related documents. This section outlines the research related to addressing data-mining problems, evolutionary algorithms and techniques, and then review literature related to building an efficient information retrieval system.

To mitigate, the impacts of COVID-19, several recent research initiatives have focused their attention on making an effective search engine for information retrieval to speed up research. Covidex [5] describes a neural ranking architecture based search engine which uses keyword search based ranking functions like BM25 to retrieve relevant papers followed by BioBERT [6] based semantic search to extract excerpts from such papers. This system was efficient enough to win the TREC-COVID challenge.

Esteva, Andre et al in their work CO-Search [7] build an encoder which is linearly composed of TF-IDF vectorizer and reciprocal-rank fused with a BM25 vectorizer to retrieve relevant papers. These papers and the query are passed to a question-answering model and an abstractive summarizer to obtain the excerpts which are then ranked based on answer match and retrieval scores. This system entails the need for heavy computational needs.

David Oniani et al. [8] proposed using transfer learning to train a GPT-2 model on the CORD-19 dataset which is then used as a question-answering system. Although, the result produced by the model is a lengthy piece of text. The answer produced by GPT-2 is further processed using semantic search on the answer which works best when BioBERT [6] or BERT [9] is used. Even though GPT models are state-of-the-art in the field, they are really heavy with millions and sometimes billions of parameters to be trained on a normal GPU or TPU.

Sonbhadra et al. [10] propose a method that involves clustering CORD-19 using k-means as it works best as shown by experiments followed by the use of OCSVM (One-Class SVMs) to map cluster to queries. Multiple OCSVMs are trained using individual cluster information. The query provided is classified to belong to one of those clusters and thereby the cluster contains relevant papers pertaining to the query.

Similarly, language model-based and NLP techniques have been proposed for data mining to cut down research time.

Clustering is an unsupervised learning approach used to create groups of relatively similar samples from the population. In this work, the following three clustering techniques have been considered:

- K-means [17] is an iterative algorithm that partitions the dataset into K pre-defined clusters
- Density-based spatial clustering of applications with noise (DBSCAN) [18] is a non-parametric algorithm that groups points that have nearby neighbors and mark the points in low-density regions as outliers.
- Hierarchical agglomerative clustering (HAC) [19] is a technique to build a hierarchy of clusters. It is a bottom-up approach wherein every element is considered as a cluster in the initial step and gradually during subsequent steps they are combined into larger clusters.

Embeddings are computed vector representations of spans of text that capture semantic and syntactic similarities between these texts. There are many types of embedding techniques like Word2Vec [12], Glove [13] and BERT [9]. Recently, a variety of different variants of BERT such as SBERT [11], BioBERT [6], DistilBERT [14], SqueezeBERT [15] have been introduced. While SBERT provides a way to get sentence embeddings using BERT, DistilBert and SqueezeBert aim to make these language models lighter and faster. BioBERT [6] is a domain-specific language representation model pre-trained on large-scale biomedical corpora In PEGASUS [16], Zhang, Jingqing, et al design a pre-training self-supervised objective for transformer encoder-decoder models to improve fine-tuning performance on abstractive summarization. These language models have been applied to capture the context of the text. It is vital in the field of medical science to preserve the meaning when answering a question. The loss of context can lead to an increase in false positives or false negatives seen in the results generated by the models resulting in incorrect practices which can turn out to be life-threatening. Therefore, semantic search is preferred over keyword search in such fields.

Semantic similarity is a metric defined over a set of documents or terms, where the idea of the distance between items is based on the likeness of their meaning or semantic content as opposed to lexicographical similarity. There are various ways to measure semantic similarity. Cosine similarity measures the similarity between two vectors of an inner product space by the cosine of the angle between two vectors. Euclidean distance between two points in Euclidean space is the length of a line segment between the two points.

III. Dataset

The CORD-19 [3] dataset, provided by the White House and a group of leading research groups is used in this work. This dataset has been prepared in response to the Coronavirus disease 2019 pandemic.

This dataset contains more than 200,000 articles (including more than 100,000 with complete text), about the coronaviruses (COVID-19 and SARS-CoV-2 included). The dataset is freely available to the global research community so that it can be used with recent advances in NLP techniques to gain insights into supporting the fight against the pandemic. The rapid acceleration in corresponding literature makes this tool the need of the hour. The growing literature is hard to keep up with. A subset of the CORD-19 dataset is used in our research work due to computational constraints.

In this work, the articles present in CORD-19 are used. The dataset classifies the sources of these into two classes. The first class consists of articles sourced from PubMed [4] and the second from various other sources of publications. This classification is done as the articles hosted on PubMed provide the paper in XML format that can be accurately converted to JSON format. However, that is not the case for the second class. The second class uses a tool to convert PDF to XML called GROBID which is not perfect. Some of the drawbacks of this tool are:

- GROBID does currently not support line numbers
- It's a machine learning tool and it's currently not very good for preprints in general. The XML produced is now converted to JSON format.

The initial collection of papers in the CORD-19 dataset suffers from duplication and conflicting data. Preprocessing of the data is performed to get a better format out of the raw data. Data preparation involved loading the CORD-19 dataset into the system, performing necessary cleaning, and merging these articles/papers into a common dataset. Duplicate articles were removed based on the concatenation of titles and abstracts of every document in the dataset. The abstract was necessary in this case as it helped distinguish articles with the same titles but different abstracts.

IV. Proposed Methodology

This section will briefly discuss the work conducted in building the data mining tool to extract relevant articles for the cord-19 dataset for a user query. The architecture of the system designed is shown in Figure 1.

A. Inputs to the model

Inputs include:

- User query: The query inputted to the model by the user. It can be a complex query consisting of multiple subqueries or a simple, single query as well.
- Cord-19 Dataset: The dataset contains more than two lakh articles related to covid-19. This has to be passed to the model

B. Modules

This section introduces and briefs in detail all the modules and their working.

1) Paraphrasing and Subqueries Generation: This component was built with the idea that some queries would retrieve better outputs if paraphrased and would hence help to obtain enhance the overall accuracy of the system.

When a user inputs a query, the query is first checked if it has subqueries and is then split into subqueries post which each sub-query is paraphrased and sent to the multithreading module. Paraphrasing is done to rephrase the queries but maintain its semantic. These rephrased subqueries are generated from the original query and all the paraphrased output queries along with the initial query are passed to the multithreading module The user's query is rephrased using a pre-trained Google Pegasus model which is utilized to paraphrase each subquery to generate queries of the same semantic meaning. Thereby we generate multiple queries for the initial query.

Thus, this method helps us to retrieve all the relevant papers efficiently, as each query would go through the whole pipeline.

2) Multi-threading: To better the performance of our model, we parallelize the queries generated by the Paraphrasing module so that it reduces the time of retrieval of documents and increases efficiency. Each sub-query passes through the Document Retriever module in parallel.

3) Document Retriever: Document retriever helps in retrieving the relevant documents for an input query. We make use of BM25 for keyword search and SRoBERTa [11] for semantic search. Here, each document's embedding is obtained with the document's title and abstract. It is followed by dimension reduction and clustering. We set a hyperparameter k, which represents the number of clusters. This parameter k is set to 20 in our case. In the offline stage, documents are clustered into k clusters using the abstract-title embedding. These k clusters represent papers of a similar kind. Further, in this stage, a BM25 index for each cluster is produced considering the entire content of papers in that particular cluster. In the online or the inference stage, each query could fall into any one of these k clusters. Once a relevant cluster is obtained for a query, keyword search is performed using BM25 only on that relevant index to obtain relevant papers. Hence this technique helps reduce the solution search space by a factor of k. This approach is one of the fastest ways to search in a huge corpus of text as it initially considers the semantics of the text and later treats the piece of text as a bag-of-words i.e semantics is not considered but is mainly dependent on the co-occurrence of the terms of the query terms across the dataset. Thus this robust semantic keyword search is performed in 2 stages. In the first stage, we find the subset of articles that semantically belong to the query's domain of context, then a pure keyword search is used to pinpoint the relevant articles.

4) Relevant Snippet Selector: This module is responsible for returning the excerpts or the relevant sentences within each of the relevant articles returned by the Document Retriever for a given query.

The working of this stage is very similar to the way the target cluster was identified in the Retriever stage. Each one of the relevant articles returned by the Document Retriever is segmented sentence-wise and is vectorized using the clinicalBERT [20] model. This language model is a specialized BioBERT transformer-based model as it has been trained on the CORD-19 dataset as well as additional clinical data.

The query at hand is also vectorized using the same model. For every article, Euclidean distance is used to compute the similarity between the sentence embeddings and the query, and the top N sentences are selected for the respective article and returned to be highlighted at a later stage. Euclidean distance is selected here for similarity computation as cosine similarity works as a similarity measure when the vectors are normalized, thus to reduce additional normalization at this stage, Euclidean distance was used. The stage returns the relevant sentences ranked in order of relevance for each relevant paper which are also ranked in order of relevance.

C. Output

The user or the researcher is presented with the most relevant articles concerning his/her query with the excerpts highlighted in the respective articles for focused attention.

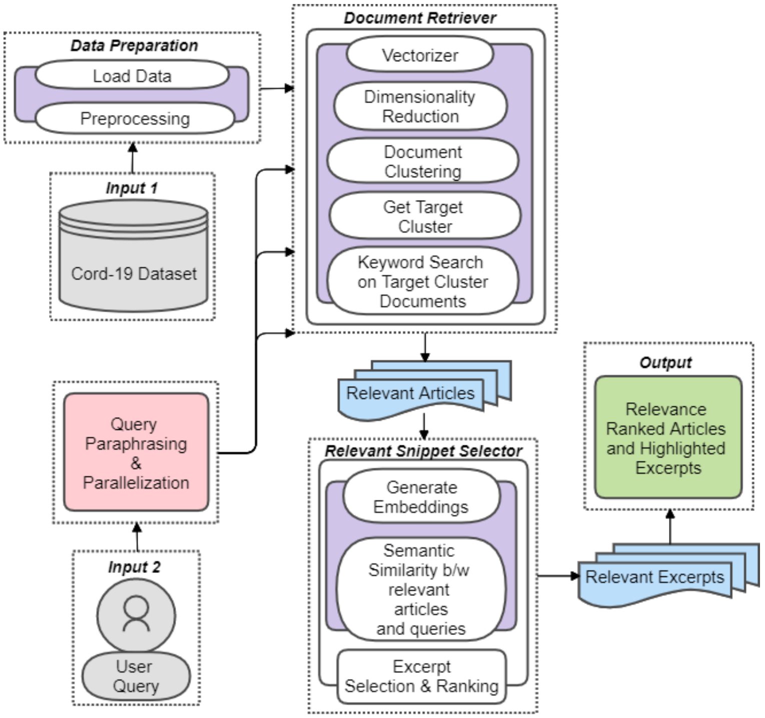
Data Preparation
Load Data
Preprocessing
Input 1
Cord-19 Dataset
Query Paraphrasing & Parallelization
Input 2
User Query
Document Retriever
Vectorizer
Dimensionality Reduction
Document Clustering
Get Target Cluster
Keyword Search on Target Cluster Documents
Relevant Articles
Relevant Snippet Selector
Generate Embeddings
Semantic Similarity b/w relevant articles and queries
Excerpt Selection & Ranking
Output
Relevance Ranked Articles and Highlighted Excerpts
Relevant Excerpts

Fig. 1. Pipeline and Architecture

D. *Novelty in our work*

- Query paraphrasing followed by parallelizing the sub-queries.
- Building a Retriever module using Clustering followed by keyword search on target cluster to reduce the solution space.
- Using custom embeddings from the language model trained on CORD-19 dataset, for the input texts
- The pipeline/implementation along with reducing solution space using a combination of semantic clustering followed by keyword search.
- Breaking down complex queries to optimize on time for generating embedding.

V. Results

The method of evaluation devised for this QA system was to compute mAP (mean average precision) of the results generated for a query across the dataset considered. This mAP score has been computed at the article/paper level. Upon completion of the implementation of our solution, we obtained the following results.

In terms of the precision of the relevant articles generated, upon comprehensive evaluation, the model gave a mAP score of 0.63 or 63%. Due to the constraint of not having ground truth values, human evaluation was done to classify papers that are relevant in the results obtained. Few evaluated queries are provided in Table 1 and Table II where precision and performance is measured,

Table 1 shows the precision scores for a few of the evaluated queries with human annotated ground truth as reference.

S.No	Query	Precision
1	Are masks helpful in the fighting covid19?	0.53
2	What is the impact of covid19 on mental health?	0.77
3	What kind of patients are more prone to covid19?	0.8
4	What is the transmission rate of covid-19?	0.5
5	What is the incubation period of covid19?	0.6
6	What is the duration of treatment for covid19 patients?	0.6
7.	What are the symptoms of covid19?	0.61

TABLE I
Precision Scores for each query

To estimate the performance of our multithreading model, the speed up was calculated with the serial version as the reference. Speed up with multithreading is shown in Table II for individual queries.

S.No	Query	Speedup
1	Are masks helpful in the fighting covid19?	2.01
2	What is the impact of covid19 on mental health?	1.87
3	What kind of patients are more prone to covid19?	1.85
4	What is the transmission rate of covid-19?	1.2
5	What is the incubation period of covid19?	1.69
6	What is the duration of treatment for covid19 patients?	1.39
7	What are the symptoms of covid19?	1.2

TABLE II
Speedup on multi-threading for each query

The average speedup achieved due to parallelizing sub-queries through multi-threading is 1.6.

VI. Conclusion

Methods for text mining have matured significantly over the past few decades. With Covid-19, we can test these methods in the sort of time and resource constrained setting where automation or computational assistance may be most helpful. Results obtained in this work are promising considering that a subset of the original dataset is used. Since early March of 2020, several dozen production systems tailored to various aspects of search and retrieval have been released. Two shared tasks have been completed with more in progress and biomedical experts have been engaged to assess and evaluate many of the systems and tools that have been publicly deployed. Aiming to help researchers manage information overload, some systems use text mining techniques to assist with conducting rapid reviews on the Covid-19 literature. As we move forward, we encourage the community to make further developments in this area. We also remind the community to keep track of practical user needs as we develop text mining systems; though much progress has been made, significant improvements are needed to provide meaningful and actionable results in the fight against Covid-19.

VII. Future Work and Limitations

The following tasks could be done to embellish the project:

- Create a website and host the same to make it accessible to the general public.
- Along with the above, we want to compare models based on all hyper-parameters encountered (different similarity measures, different embedding techniques, pmc vs pdf).
- Run our model on other medical datasets to estimate the performance and make it available to the scientific community.

The work's primary limitation is that it is designed to work specifically for the covid-19 pandemic. It will require models pre-trained on literature about the disease it is trying to answer queries for. Secondarily, the work is also limited by lack of compute resources.

References

[1] Robertson S. Understanding inverse document frequency: on theoretical arguments for IDF. Journal of documentation. 2004 Oct 1.

[2] Amati G. (2009) BM25. In: LIU L., ÖZSU M.T. (eds) Encyclopedia of Database Systems. Springer, Boston, MA.

[3] Wang, Lucy Lu, et al. ”CORD-19: The COVID-19 Open Research Dataset.” ArXiv (2020).

[4] Canese, Kathi, and Sarah Weis. ”PubMed: the bibliographic database.” The NCBI Handbook [Internet]. 2nd edition. National Center for Biotechnology Information (US), 2013.

[5] Zhang, Edwin, et al. ”Covidex: Neural ranking models and keyword search infrastructure for the covid-19 open research dataset.” arXiv preprint arXiv:2007.07846 (2020).

[6] Lee, Jinhyuk, et al. ”BioBERT: a pre-trained biomedical language representation model for biomedical text mining.” Bioinformatics 36.4 (2020): 1234-1240.

[7] Esteva, Andre et al. “CO-Search: COVID-19 Information Retrieval with Semantic Search, Question Answering, and Abstractive Summarization.” ArXiv abs/2006.09595 (2020): n. pag.

[8] Oniani, David, and Yanshan Wang. ”A qualitative evaluation of language models on automatic question-answering for covid-19.” Proceedings of the 11th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics. 2020.

[9] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805, 2018.

[10] Sonbhadra, Sanjay Kumar, Sonali Agarwal, and P. Nagabhushan. ”Target specific mining of COVID-19 scholarly articles using one-class approach.” Chaos, Solitons & Fractals 140 (2020): 110155.

[11] Reimers, Nils, and Iryna Gurevych. ”Sentence-bert: Sentence embeddings using siamese bert-networks.” arXiv preprint arXiv:1908.10084 (2019)

[12] Mikolov, Tomas, et al. ”Distributed representations of words and phrases and their compositionality.” Advances in neural information processing systems. 2013.

[13] Pennington, Jeffrey, Richard Socher, and Christopher D. Manning. ”Glove: Global vectors for word representation.” Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP). 2014.

[14] Sanh, Victor, et al. ”DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter.” arXiv preprint arXiv:1910.01108 (2019).

[15] Iandola, Forrest N., et al. ”SqueezeBERT: What can computer vision teach NLP about efficient neural networks?.” arXiv preprint arXiv:2006.11316 (2020).

[16] Zhang, Jingqing, et al. ”Pegasus: Pre-training with extracted gap-sentences for abstractive summarization.” International Conference on Machine Learning. PMLR, 2020.

[17] Jin X., Han J. (2011) K-Means Clustering. In: Sammut C., Webb G.I. (eds) Encyclopedia of Machine Learning. Springer, Boston, MA. https://doi.org/10.1007/978-0-387-30164-8_425

[18] Ester, Martin, et al. ”A density-based algorithm for discovering clusters in large spatial databases with noise.” Kdd. Vol. 96. No. 34. 1996.

[19] Zepeda-Mendoza M.L., Resendis-Antonio O. (2013) Hierarchical Agglomerative Clustering. In: Dubitzky W., Wolkenhauer O., Cho KH., Yokota H. (eds) Encyclopedia of Systems Biology. Springer, New York, NY. https://doi.org/10.1007/978-1-4419-9863-7_1371

[20] Emily Alsentzer, John Murphy, William Boag, Wei-Hung Weng, Di Jin, Tristan Naumann, and Matthew McDermott. 2019. Publicly available clinical BERT embeddings. In Proceedings of the 2nd Clinical Natural Language Processing Workshop, pages 72-78, Minneapolis, Minnesota, USA. Association for Computational Linguistics.