School of Computer Science and Engineering

DMA Course Project Report

On

COVID 19 OPEN RESEARCH DATASET (CORD 19)

Submitted by

1) Mr. Abhimalya Chowdhury	01FE18BCS007
2) Mr. Abhishek P.M	01FE18BCS010
3) Mr. Aditya. Girigoudar	01FE18BCS017
4) Mr. Amogh. Shetty	01FE18BCS035

Table of Contents

Introduction:

COVID 19. Literally a bane to Mankind. No one could have predicted the influence a virus could have across the Entire Globe. The Fear of living a life is on the Surge, the increasing of people falling ill, the rising death count, Catastrophe and havoc everywhere. Humanity is taking a toll.

CORD-19 aims to connect the machine learning community with biomedical domain experts and policy makers in the race to identify effective treatments and management policies for Covid-19. The goal is to harness these diverse and complementary pools of expertise to discover relevant information more quickly from the literature. Users of the dataset have leveraged a variety of AI-based techniques in information retrieval and natural language processing to extract useful information.

On the contrary, an extensive research on the virus has boosted interest among Academics, and now is the time to find answers to the most Intriguing Questions. With a large amount of research that could be done. It lies upon us to find the right answers to all the unanswered questions.

To make this happen, The Community of Kaggle is all in with the CORD 19 Dataset which contains resources of over 200,000 scholarly articles, including over 100,000 with full text, about COVID-19, SARS-CoV-2, and related corona viruses.

There is a growing urgency 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. Text and data mining tools that can help the medical community develop answers to high priority scientific questions are the Need of the Hour. Most of the tasks under this dataset have questions that could be answered using Text Mining tools from an enormous amount of resource papers available.

The Quest to find the Answers Begin.

Dataset Description:

The COVID-19 Open Research Dataset (CORD-19) is a growing resource of scientific papers on COVID-19 and related historical corona virus research. CORD-19 is designed to facilitate the development of text mining and information retrieval systems over its rich collection of metadata and structured full text papers. Since its release, CORD-19 has been downloaded over 200K times and has served as the basis of many COVID-19 text mining and discovery systems.

The literature survey of the Dataset includes People from all horizons contributing to the Open research challenge, for a foreseeable future towards the right answers.

Problem Statement:

Among the 17 Available tasks, the above two tasks were selected and answers to most pertaining questions in the chosen domain were to be realised. Task 1 Includes fetching documents of high similarity based on the query and Task 2 includes obtaining Summary Tables based on the query searched. The Quest to suppress intriguing questions is the problem that needs to be resolved and is the need of the hour.

Task 1: What do we know about diagnostics and surveillance?

Task 2: Create summary tables that address diagnostics for COVID-19

For Task 1, we retrieve the most similar documents from the available set of documents and check for their similarity based on the query in the task. This task includes searching for queries that are most related to diagnostics and Surveillance. For Task 2 Summary Tables with regard to addressing the diagnostics of Covid 19 and related queries have to be found. All the quesstionaries in the tasks have to be resolved to find the most similar documents possible.

Literature Survey:

1) Discovid.ai:

Journal: Studies in Natural Products Chemistry

DOI: 10.1101/2020.01.31.929042

Approach: A number of topics are discovered using LSA (Latent semantic analysis), followed by expert analysis of the output. This allows one to view each article as a mixture of these topics. This Particular Approach relies to show mapping a specific article into the topic space (a simplex with a topic in each corner), we can then find related articles.

Features

1) Similarity measure to show the confidence in the recommendations
2) Plots that show the topic-distribution of an article
3) Select a time range to limit the articles that are considered (you can decide if you want to find the latest publications or search for insights in past research)
4) Option to only suggest COVID-19-papers (those that contain COVID-19, SARS-CoV-2, 2019-nCov, SARS Corona virus 2 or 2019 Novel Corona virus in the text body)

2) Cord-19 ETL – Article SQLITE Database

The CORD-19 ETL notebook has a full overview of the raw CORD-19 dataset, parsing rules and other important information on how the data stored in SQLite is derived. It contains the full build process for both the SQLite database and embeddings index here but modularization was necessary as the dataset grew.

Approach:

1) An embedding's index is created with Fast Text + BM25
2) The embedding's index takes each COVID-19 tagged, not labeled a question/fragment, having a detected study type, tokenizes the text, and builds a sentence embedding.
3) A sentence embedding is a BM25 weighted combination of the Fast Text vectors for each token in the sentence.
4) The embedding's index takes the full corpus of these embeddings and builds a Faiss index to enable similarity searching.

Data Pre-Processing:

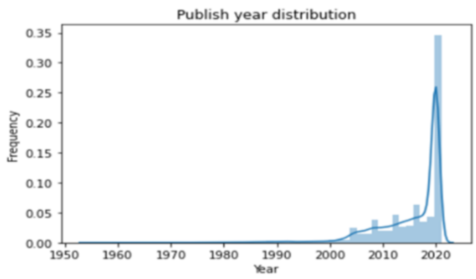
The dataset should be first cleaned so that further topic modelling can be done smoothly, to do this we first remove ambiguous data.

1. We import metadata file.
2. There is no Microsoft Academic papers published present in the dataset. So, this attribute ('mag_id') can be removed.
3. The rows which have the duplicate values for following attributes are remove
    - cord_Id : No Two Papers can have the same Id, As Cord_Id is Unique to Every Paper
    - title: No Two Papers can have the Same Title
    - Sha: Refers to the Web Link of the Paper and No two Papers can have the same 'SHA'.

We have taken a few steps to pre-process the data, so as to obtain the most factual data frame.

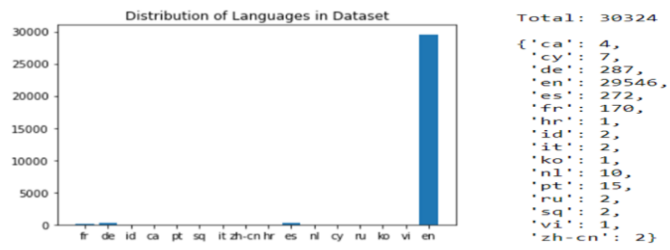
1. Extracting paper_id, abstract and full text from the json files and strings it in a data frame.
2. We drop the rows with duplicate paper_id and abstract.
3. We integrate metadata with json data frame using inner join on sha.
4. We replace missing string field with Nan values.
5. We replace rows with duplicate abstracts.
6. We only consider papers which are published in 2020.
7. We remove papers of languages other than English.
8. We merge title, abstract and full text into a new column.
9. We remove common and custom stop words.
10. Stemming and Lemmatization is done and the tokens are obtained from processed text
11. We have considered randomly selected 10000 rows.

ILLUSTRATIONS OF PREPROCESSING OF THE DATASET



Plot For the Number of Papers that are published during a particular Year

Fig. 1.1



```
Total: 30324

{'ca': 4,
 'cy': 7,
 'de': 287,
 'en': 29546,
 'es': 272,
 'fr': 170,
 'hr': 1,
 'id': 2,
 'it': 2,
 'ko': 1,
 'nl': 10,
 'pt': 15,
 'ru': 2,
 'sq': 2,
 'vi': 1,
 'zh-cn': 2}
```

Plot For the Number of Papers that are published in a particular language and the count of Each.

Fig. 1.2

An Unambiguous Data frame is now obtained for further Topic Modelling and Obtaining Results.

Task 1:

What do we know about diagnostics and surveillance?

Approaches:

1) Latent Dirichlet Allocation (LDA):

Each document is made up of various words, and each topic also has various words belonging to it. The aim of LDA is to find topics a document belongs to, based on the words in it.

We are using two variants of LDA: LDA BOW and LDA TF-IDF. We are generating multiple LDA models with different number of topics as a main hyper-parameter to then choose the best model based on the coherence and perplexity scores

1) From a Data frame containing randomly selected 10,000 Documents, the attribute '**Processed Text**' is the Key Feature. Tokens are made for all the sentences**,** Bigrams are obtained from the created Tokens
2) For the Bigram Model with Tokens, Dictionary is created by filtering extremes based on frequency, and Corpus is obtained.
3) To the Term Document Frequency Corpus Obtained, We Apply LDA to find Dominant Words Belonging to a Topic, Number of Topics is hyper parameter here.
4) Dominant Words in Each Topic Are Obtained, Best Fit LDA Model is applied to Each Documents to find the suitable Topic
5) From the Document Topic Frame, we take Tokens from query and finding similarity with word list of Topics To retrieve documents of that Topic
6) Final Model based on the Query Asked is Retrieved

APPROACH 1: IMPLEMENTING LATENT DIRICHLET ALLOCATION (LDA)

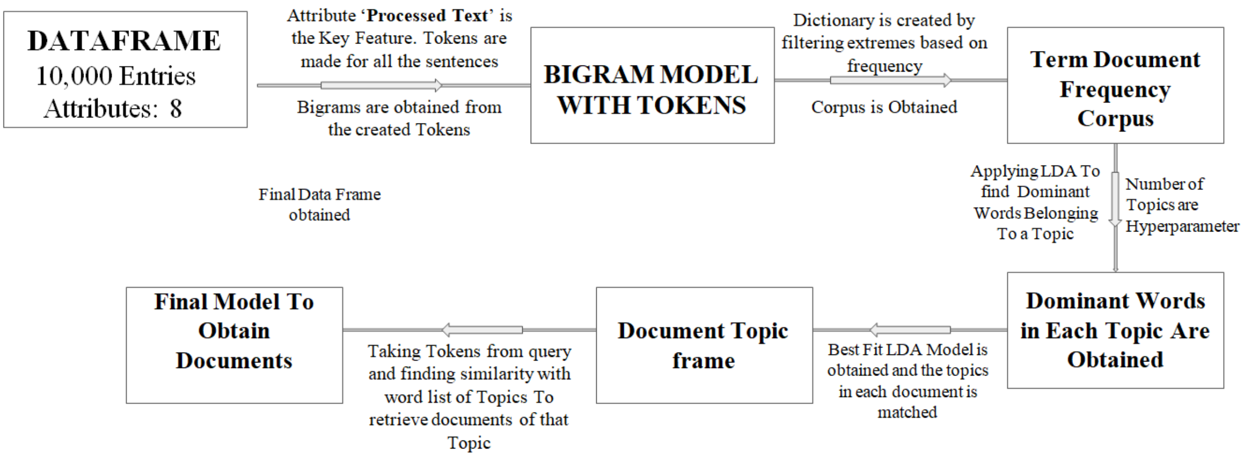
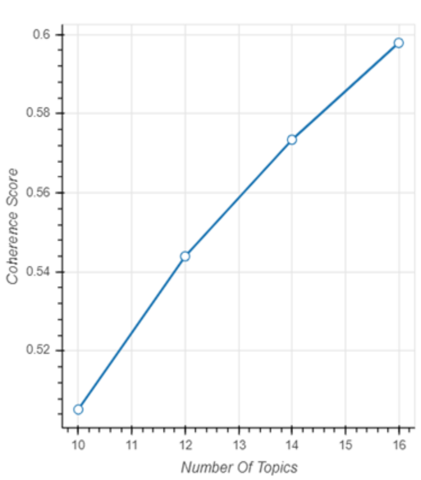

DATAFRAME
10,000 Entries
Attributes: 8
Attribute 'Processed Text' is the Key Feature. Tokens are made for all the sentences
Bigrams are obtained from the created Tokens
BIGRAM MODEL WITH TOKENS
Dictionary is created by filtering extremes based on frequency
Corpus is Obtained
Term Document Frequency Corpus
Applying LDA To find Dominant Words Belonging To a Topic
Number of Topics are Hyperparameter
Dominant Words in Each Topic Are Obtained
Best Fit LDA Model is obtained and the topics in each document is matched
Document Topic frame
Taking Tokens from query and finding similarity with word list of Topics To retrieve documents of that Topic
Final Model To Obtain Documents
Final Data Frame obtained

Fig. 2.1 Flowchart to implement Latent Dirichlet Allocation

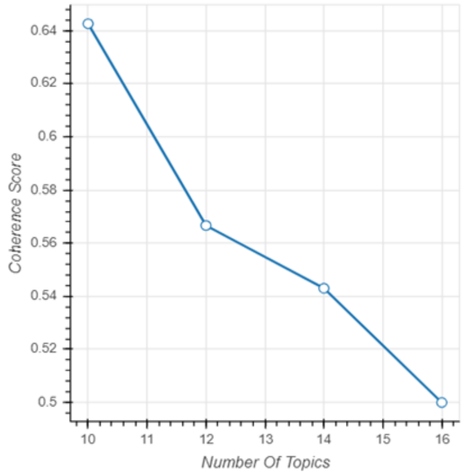
Results:



LDA BOW

Trained For 4 passes, With start topic as 10, and End Topic as 18, along with Step 2. As the coherence Value is Increasing. We Initiate it to Model 3 as it has a larger coherence value.

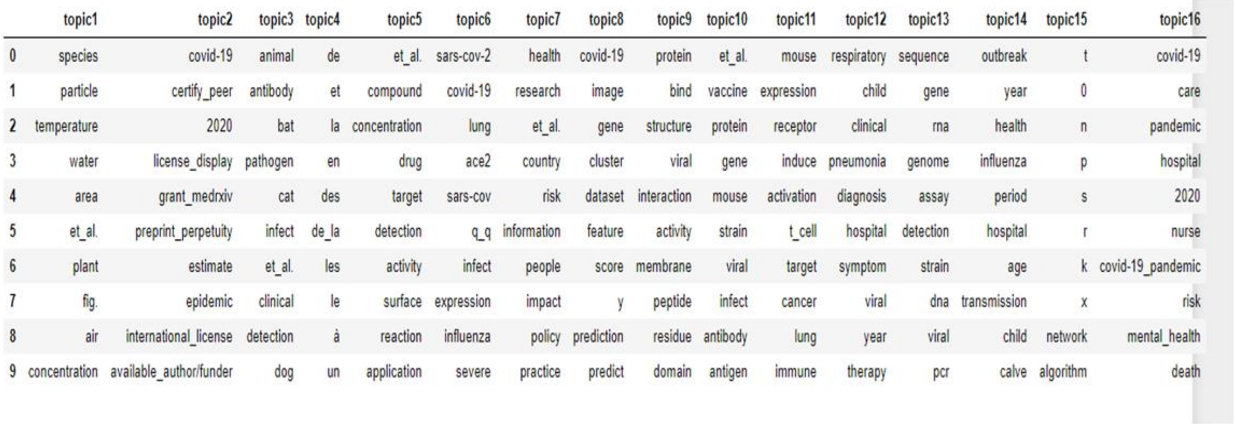
Fig 2.2



LDA TF-IDF

Trained For 4 passes, With start topic as 10, and End Topic as 18, along with Step 2. As the coherence Value is Decreasing. We Initiate it to Model 1 as it has a larger coherence value.

Fig 2.3

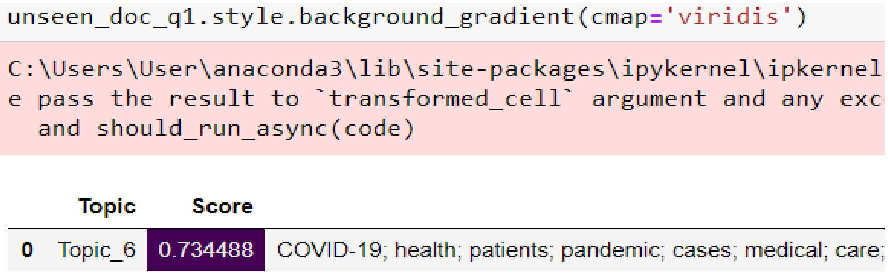


	topic1	topic2	topic3	topic4	topic5	topic6	topic7	topic8	topic9	topic10	topic11	topic12	topic13	topic14	topic15	topic16
0	species	covid-19	animal	de	et_al.	sars-cov-2	health	covid-19	protein	et_al.	mouse	respiratory	sequence	outbreak	t	covid-19
1	particle	certify_peer	antibody	et	compound	covid-19	research	image	bind	vaccine	expression	child	gene	year	0	care
2	temperature	2020	bat	la	concentration	lung	et_al.	gene	structure	protein	receptor	clinical	rna	health	n	pandemic
3	water	license_display	pathogen	en	drug	ace2	country	cluster	viral	gene	induce	pneumonia	genome	influenza	p	hospital
4	area	grant_medrxiv	cat	des	target	sars-cov	risk	dataset	interaction	mouse	activation	diagnosis	assay	period	s	2020
5	et_al.	preprint_perpetuity	infect	de_la	detection	q_q	information	feature	activity	strain	t_cell	hospital	detection	hospital	r	nurse
6	plant	estimate	et_al.	les	activity	infect	people	score	membrane	viral	target	symptom	strain	age	k	covid-19_pandemic
7	fig.	epidemic	clinical	le	surface	expression	impact	y	peptide	infect	cancer	viral	dna	transmission	x	risk
8	air	international_license	detection	à	reaction	influenza	policy	prediction	residue	antibody	lung	year	viral	child	network	mental_health
9	concentration	available_author/funder	dog	un	application	severe	practice	predict	domain	antigen	immune	therapy	pcr	calve	algorithm	death

Fig 2.4 Cluster of Top Dominant words in a Particular Topic

Question 1:

How widespread current exposure is to be able to make immediate policy recommendations on mitigation measures. Denominators for testing and a mechanism for rapidly sharing that information, including demographics, to the extent possible. Sampling methods to determine asymptomatic disease (e.g., use of serosurveys (such as convalescent samples) and early detection of disease (e.g., use of screening of neutralizing antibodies such as ELISAs).



```
unseen_doc_q1.style.background_gradient(cmap='viridis')
```

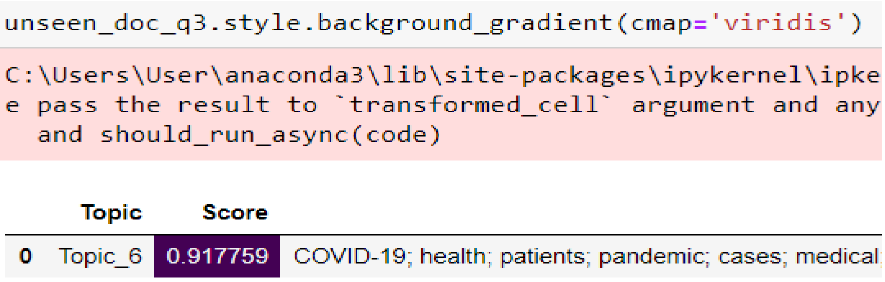
```
C:\Users\User\anaconda3\lib\site-packages\ipykernel\ipkernel
e pass the result to `transformed_cell` argument and any exc
  and should_run_async(code)
```

	Topic	Score	
0	Topic_6	0.734488	COVID-19; health; patients; pandemic; cases; medical; care;

Fig 2.5 Related Topic to Query 1

Question 2:

Efforts to increase capacity on existing diagnostic platforms and tap into existing surveillance platforms.



```
unseen_doc_q3.style.background_gradient(cmap='viridis')
```

```
C:\Users\User\anaconda3\lib\site-packages\ipykernel\ipke
e pass the result to `transformed_cell` argument and any
  and should_run_async(code)
```

	Topic	Score	
0	Topic_6	0.917759	COVID-19; health; patients; pandemic; cases; medical

Fig 2.6 Related Topic to Query 2

2) Semantic Search using Word Embedding's:

Description:

In this method we will use gensim Word2Vec in order to generate word embeddings using the abstract texts as our corpus. For each document, we calculate the centroid of its abstract and for each query word; we map it to a vector then calculate the word centroid similarity for the query and each document's abstract. The top ranked papers are shown as results.

Steps:

1) Loading and Pre-Processing the data: Basic data cleaning and pre-processing is done like removing duplicates, removing non-English articles and also handling null values.
2) Sentence tokenization using Spacy Tokenizer: Gensim's Word2Vec corpus should be in the form of separate sentences, so we use spacy tokenizer to tokenize the sentences.
3) Training the Word2Vec Model: We take the corpus and train it using gensims Word2Vec.
4) Centroid Calculation: We calculate the centroid for each abstract using the vectors of all words in the abstract.
5) Measuring the Cosine Similarity: We now measure the cosine similarities between the centroid of each document and the query.
6) Ranking the documents: We then rank the documents based on cosine similarity values and give out the results.

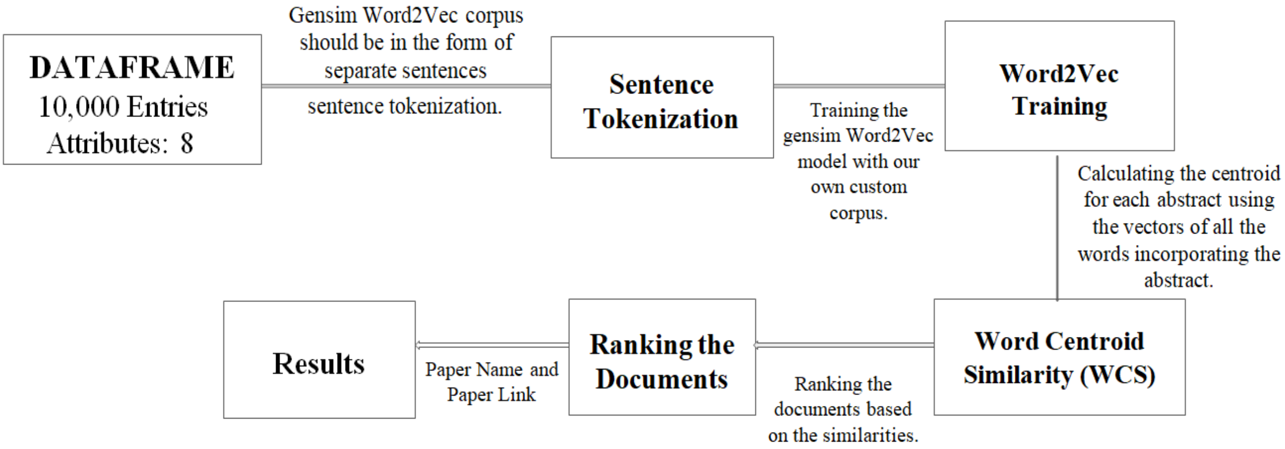
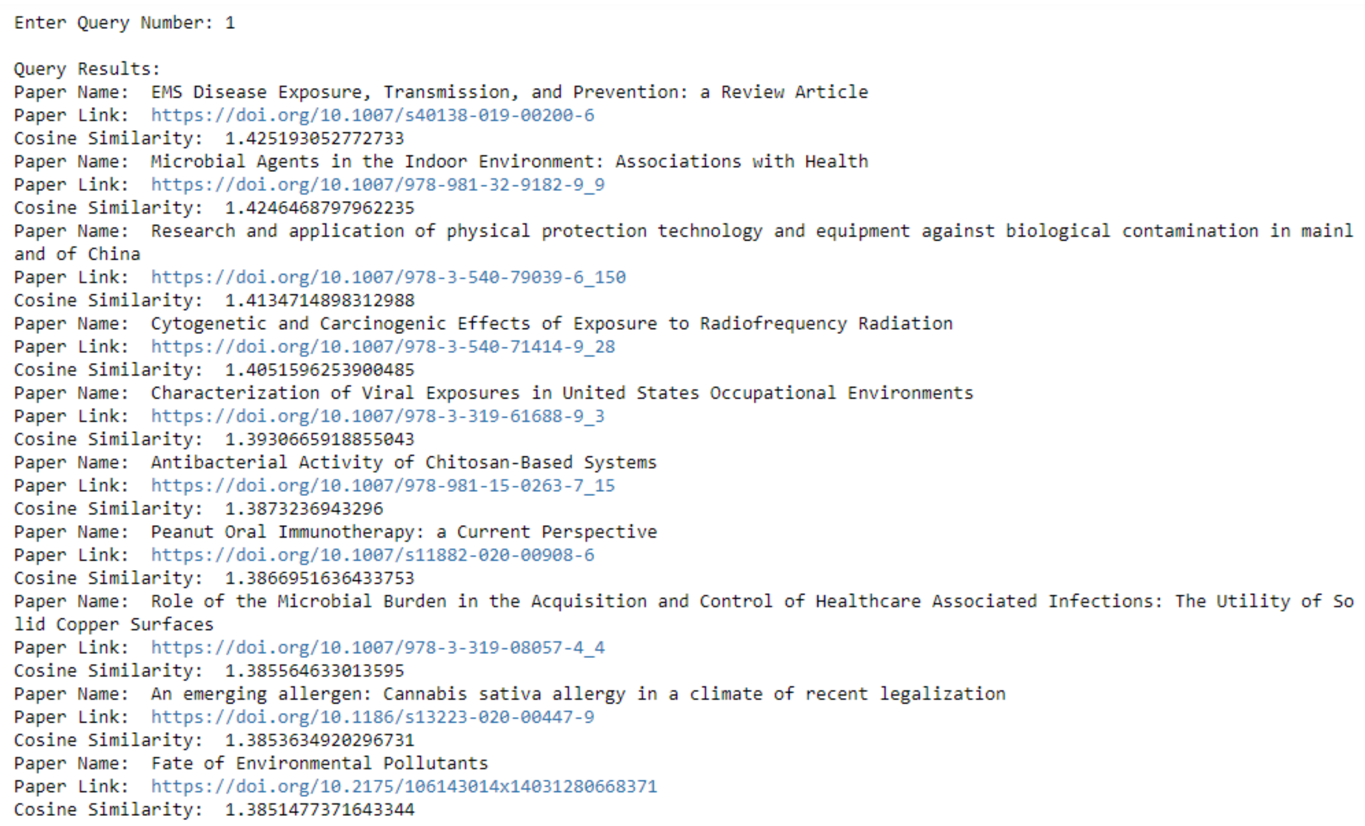

DATAFRAME
10,000 Entries
Attributes: 8
Gensim Word2Vec corpus should be in the form of separate sentences
sentence tokenization.
Sentence Tokenization
Training the gensim Word2Vec model with our own custom corpus.
Word2Vec Training
Calculating the centroid for each abstract using the vectors of all the words incorporating the abstract.
Word Centroid Similarity (WCS)
Ranking the documents based on the similarities.
Ranking the Documents
Paper Name and Paper Link
Results

Fig. 3.1 Flow Diagram to Implement Semantic Search Using Word Embedding's

Results:



```
Enter Query Number: 1

Query Results:
Paper Name:  EMS Disease Exposure, Transmission, and Prevention: a Review Article
Paper Link:  https://doi.org/10.1007/s40138-019-00200-6
Cosine Similarity:  1.425193052772733
Paper Name:  Microbial Agents in the Indoor Environment: Associations with Health
Paper Link:  https://doi.org/10.1007/978-981-32-9182-9_9
Cosine Similarity:  1.4246468797962235
Paper Name:  Research and application of physical protection technology and equipment against biological contamination in mainl
and of China
Paper Link:  https://doi.org/10.1007/978-3-540-79039-6_150
Cosine Similarity:  1.4134714898312988
Paper Name:  Cytogenetic and Carcinogenic Effects of Exposure to Radiofrequency Radiation
Paper Link:  https://doi.org/10.1007/978-3-540-71414-9_28
Cosine Similarity:  1.4051596253900485
Paper Name:  Characterization of Viral Exposures in United States Occupational Environments
Paper Link:  https://doi.org/10.1007/978-3-319-61688-9_3
Cosine Similarity:  1.3930665918855043
Paper Name:  Antibacterial Activity of Chitosan-Based Systems
Paper Link:  https://doi.org/10.1007/978-981-15-0263-7_15
Cosine Similarity:  1.3873236943296
Paper Name:  Peanut Oral Immunotherapy: a Current Perspective
Paper Link:  https://doi.org/10.1007/s11882-020-00908-6
Cosine Similarity:  1.3866951636433753
Paper Name:  Role of the Microbial Burden in the Acquisition and Control of Healthcare Associated Infections: The Utility of So
lid Copper Surfaces
Paper Link:  https://doi.org/10.1007/978-3-319-08057-4_4
Cosine Similarity:  1.385564633013595
Paper Name:  An emerging allergen: Cannabis sativa allergy in a climate of recent legalization
Paper Link:  https://doi.org/10.1186/s13223-020-00447-9
Cosine Similarity:  1.3853634920296731
Paper Name:  Fate of Environmental Pollutants
Paper Link:  https://doi.org/10.2175/106143014x14031280668371
Cosine Similarity:  1.3851477371643344
```

Fig 3.2 Results for Each Questionnaire

3) Improvising LDA using K-Means, PCA and T-SNE:

Description

In this method, we use PCA to keep at least 95% variance and K means will be used to label each cluster with a different colour and T-SNE will be able to reduce the dimensions and plot it to 2 dimension. The combination of T-SNE and PCA with K-Means is used to obtain the graph with clusters. Now LDA is used to obtain most dominant words from each cluster.

Steps:

1) From the randomly Selected data frame of 10,000 documents, TF-IDF is applied to obtain Document Feature matrix.
2) To the Document Feature Matrix, We Apply T-SNE to find the Documents with similarity and reduce the high dimension data, Clustering of similar Documents and Dimension is reduced to 2D.

3) To the Document Feature Matrix, We Apply PCA to the data trying to preserve 95% Variance, this removes Outliers and reduced to only important dimensions.
4) A Document Feature Matrix with reduced dimensions is obtained.
5) K-Means is applied to find the clusters and obtain different labels (color) to Each Cluster
6) Labels to each Cluster is obtained, now this is in turn fed to the plot obtained by T-SNE
7) Scatter Plot in 2-D is Obtained with labels.
8) LDA is applied individually to each of the cluster and dominant words in each Cluster is obtained.

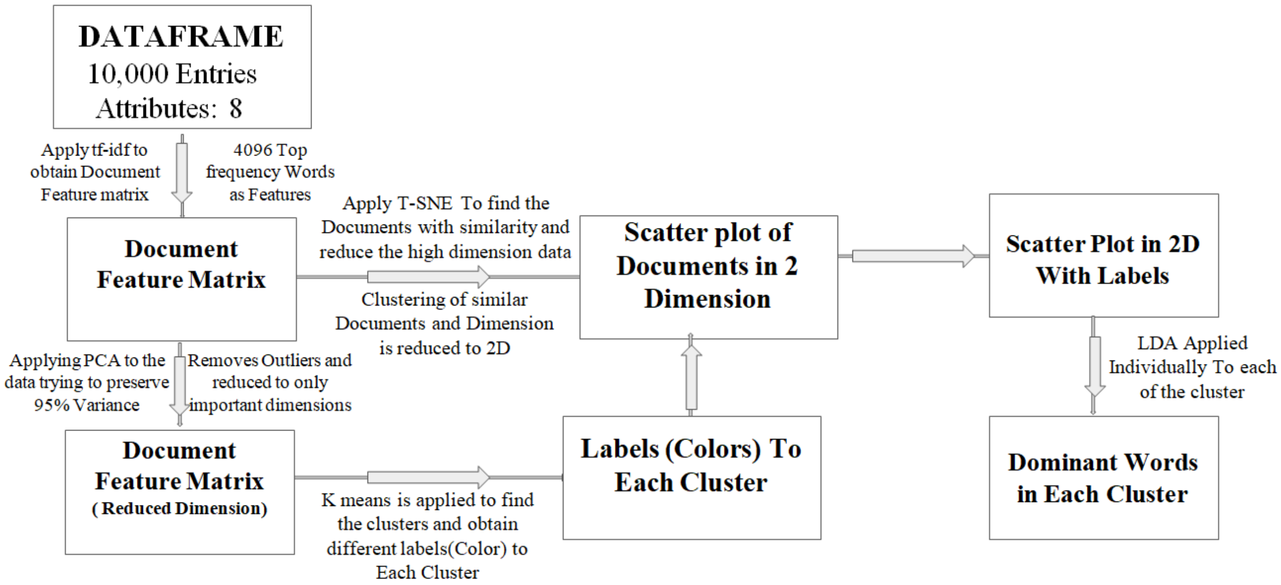
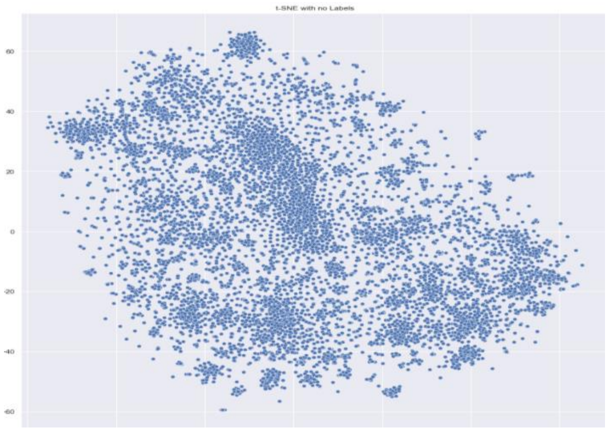

DATAFRAME
10,000 Entries
Attributes: 8
Apply tf-idf to obtain Document Feature matrix
4096 Top frequency Words as Features
Document Feature Matrix
Apply T-SNE To find the Documents with similarity and reduce the high dimension data
Clustering of similar Documents and Dimension is reduced to 2D
Scatter plot of Documents in 2 Dimension
Scatter Plot in 2D With Labels
LDA Applied Individually To each of the cluster
Applying PCA to the data trying to preserve 95% Variance
Removes Outliers and reduced to only important dimensions
Document Feature Matrix (Reduced Dimension)
K means is applied to find the clusters and obtain different labels(Color) to Each Cluster
Labels (Colors) To Each Cluster
Dominant Words in Each Cluster

Fig. 4.1 Finding documents using K-Means, PCA and LDA

Results:



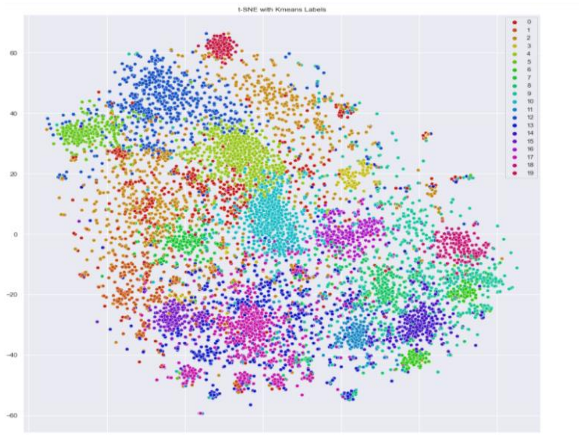
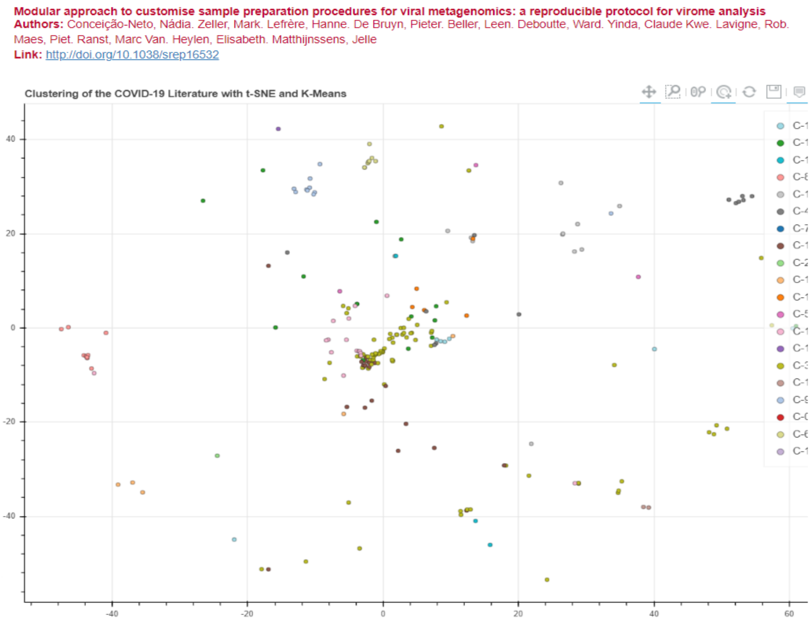


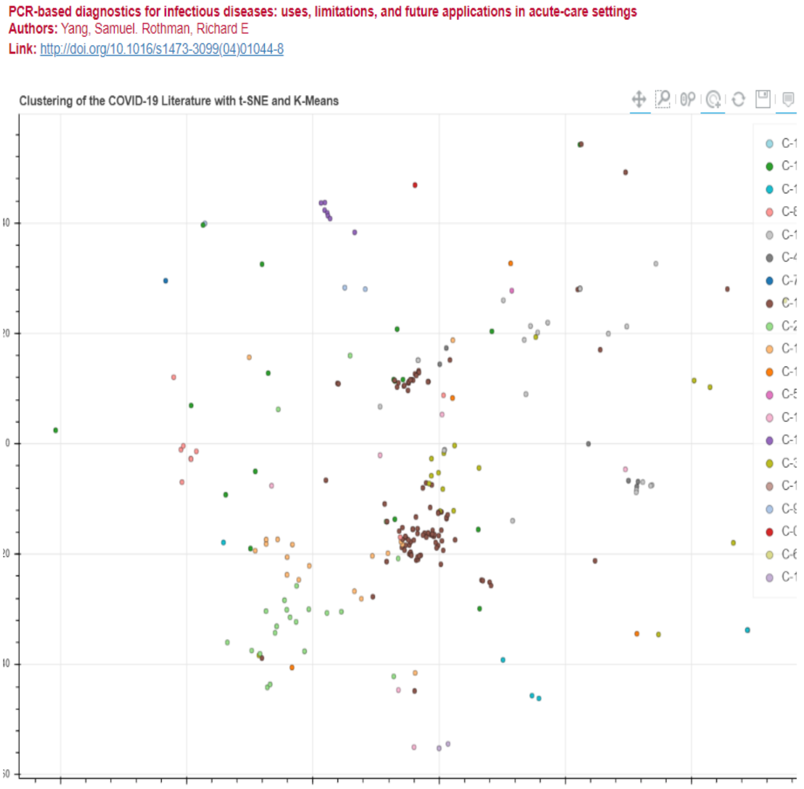
Fig 4.2 T-SNE Without Labels

Fig 4.3 T-SNE With Labels



Modular approach to customise sample preparation procedures for viral metagenomics: a reproducible protocol for virome analysis
Authors: Conceição-Neto, Nádia. Zeller, Mark. Lefrère, Hanne. De Bruyn, Pieter. Beller, Leen. Deboutte, Ward. Yinda, Claude Kwe. Lavigne, Rob. Maes, Piet. Ranst, Marc Van. Heylen, Elisabeth. Matthijnssens, Jelle
Link: http://doi.org/10.1038/srep16532

Fig 4.4 RESULTANT CLUSTER FOR SEARCH TERM ‘GENOMICS’



PCR-based diagnostics for infectious diseases: uses, limitations, and future applications in acute-care settings
Authors: Yang, Samuel. Rothman, Richard E
Link: http://doi.org/10.1016/s1473-3099(04)01044-8

Fig 4.5 RESULTANT CLUSTER FOR SEARCH TERM ‘DIAGNOSTICS’

Approach for Task 2:

Create summary tables that address diagnostics for COVID-19.

Specifically, the task requires to answer what the literature reports about:

1. What do we know about diagnostics and corona-virus?
2. New advances in diagnosing SARS-COV-2
3. Development of a point-of-care test and rapid bed-side tests
4. Diagnosing SARS-COV-2 with Nucleic-acid based tech
5. Diagnosing SARS-COV-2 with antibodies
6. How does viral load relate to disease presentations and likelihood of a positive diagnostic test?

Methodology:

Step 1: Ranking of documents using BM25 Okapi Algorithm:

In information retrieval, Okapi BM25 (BM is an abbreviation of best matching) is a ranking function used by search engines to estimate the relevance of documents to a given search query. It is based on the probabilistic retrieval framework developed in the 1970s and 1980s by Stephen E. Robertson, Karen Spark Jones, and others.

The problem that BM25 (Best Match 25) tries to solve is similar to that of TFIDF (Term Frequency, Inverse Document Frequency), that is representing our text in a vector space (it can be applied to field outside of text, but text is where it has the biggest presence) so we can search/find similar documents for a given document or query.

BM25 improves upon TFIDF by casting relevance as a probability problem. A relevance score, according to probabilistic information retrieval, ought to reflect the probability a user will consider the result relevant.

Workflow:

In this method, we will first tokenize the entire body text for each of the documents. Also, each of the queries must be tokenized into words. The tokenized query as well as the text will be the input to our BM25 Okapi algorithm.

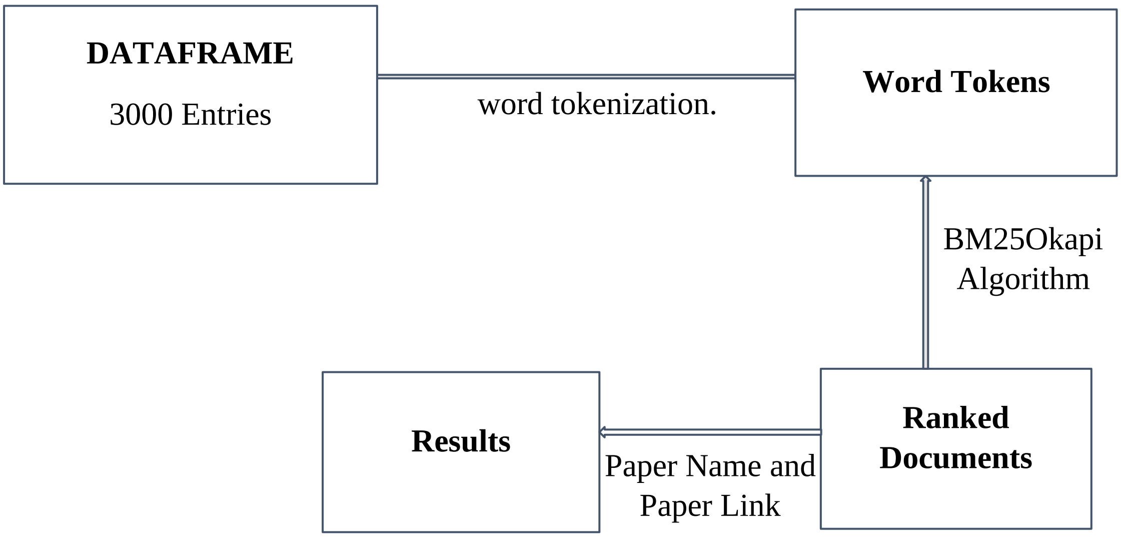


Fig 5.1 BM25 Algorithm

BM25 (Best Match 25) function scores each document in a corpus according to the document's relevance to a particular text query. For a query Q, with terms q1…, qn, the BM25 score for document D is:

$$\mathrm{BM25}(D,Q)=\sum_{i=1}^{n} IDF(q_i,D)\frac{f(q_i,D)\cdot(k_1+1)}{f(q_i)+k_1\cdot(1-b+b\cdot|D|/d_{avg}))}$$

where:

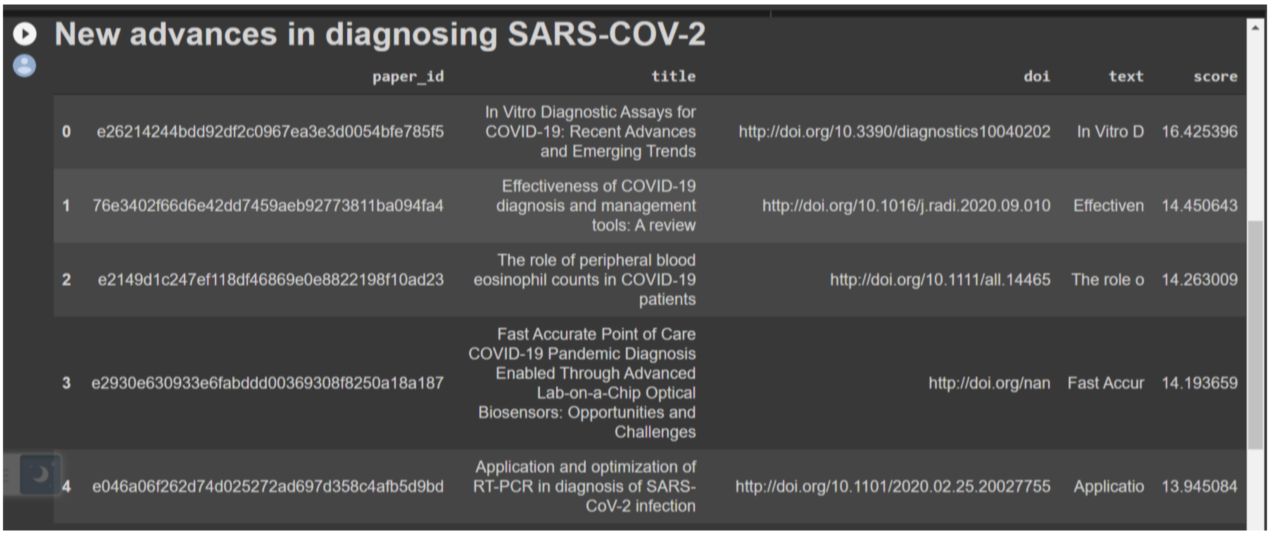
- $f(q_i,D)$ is the number of times term q_i occurs in document D.
- $|D|$ is the number of words in document D.
- d_{avg} is the average number of words per document.
- b and k_1 are hyperparameters for BM25.

k_1 **and b can be tuned by cross-validation, typical values are k_1=2 and b=0.75**

$$\mathrm{IDF}(q_i) = \ln\left(\frac{N - n(q_i) + 0.5}{n(q_i) + 0.5} + 1\right)$$

where N is the total number of documents in the collection, and $n(q_i)$ is the number of documents containing q_i.

Results:



New advances in diagnosing SARS-COV-2

	paper_id	title	doi	text	score
0	e26214244bdd92df2c0967ea3e3d0054bfe785f5	In Vitro Diagnostic Assays for COVID-19: Recent Advances and Emerging Trends	http://doi.org/10.3390/diagnostics10040202	In Vitro D	16.425396
1	76e3402f66d6e42dd7459aeb92773811ba094fa4	Effectiveness of COVID-19 diagnosis and management tools: A review	http://doi.org/10.1016/j.radi.2020.09.010	Effectiven	14.450643
2	e2149d1c247ef118df46869e0e8822198f10ad23	The role of peripheral blood eosinophil counts in COVID-19 patients	http://doi.org/10.1111/all.14465	The role o	14.263009
3	e2930e630933e6fabddd00369308f8250a18a187	Fast Accurate Point of Care COVID-19 Pandemic Diagnosis Enabled Through Advanced Lab-on-a-Chip Optical Biosensors: Opportunities and Challenges	http://doi.org/nan	Fast Accur	14.193659
4	e046a06f262d74d025272ad697d358c4afb5d9bd	Application and optimization of RT-PCR in diagnosis of SARS-CoV-2 infection	http://doi.org/10.1101/2020.02.25.20027755	Applicatio	13.945084

Fig 5.2 For each query, documents are displayed in descending order of the scores obtained

Discussion:

BM25 score is a linear weighted combination of scores for each of the words that make up the query. The total score is the sum of score of each of the words in the query. The query ‘New advances in diagnosing SARS-COV-2’ is tokenized, and for each of the query term, ‘new’, ‘advances’, ‘in’ …. ‘sars-cov-2’ BM25 score is calculated. Words, such as ‘in’, ‘the’, are referred to as stop-words and they appear in almost every English documents. Hence, such words contribute negligibly to the overall BM25 score, since, their IDF score comes out to be nearly zero. On the other hand, unique words such as ‘sars-cov-2’, ‘advances’ contribute highly to the BM25 score, their IDF score being high. The final score of a document is determined by summing up the scores obtained for each of the terms present in the query.

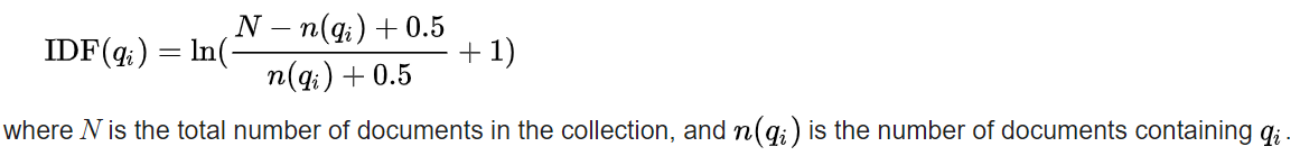
We will now discuss how BM25 improves upon TF-IDF, a common scoring function for determining search relevance.

The IDF formula for TF-IDF is given by

idf(t) = log [n / df(t)] + 1

where n is the total number of documents in the document set and df(t) is the document frequency of t

whereas the IDF formula for BM25 is given by



$$\text{IDF}(q_i) = \ln\left(\frac{N - n(q_i) + 0.5}{n(q_i) + 0.5} + 1\right)$$

where N is the total number of documents in the collection, and $n(q_i)$ is the number of documents containing q_i.

The IDF component of BM25 scoring function measures how often a term occurs in all of the documents and “penalizes” terms that are common.

The IDF component of BM25 scoring function measures how often a term occurs in all of the documents and “penalizes” terms that are common.

Let us now turn our focus to the two important hyperparameters the scoring function takes: **‘k1’** and **‘b’.**

k1: k1 is a variable which helps determine term frequency saturation characteristics. That is, it limits how much a single query term can affect the score of a given document. It does this through approaching an asymptote.

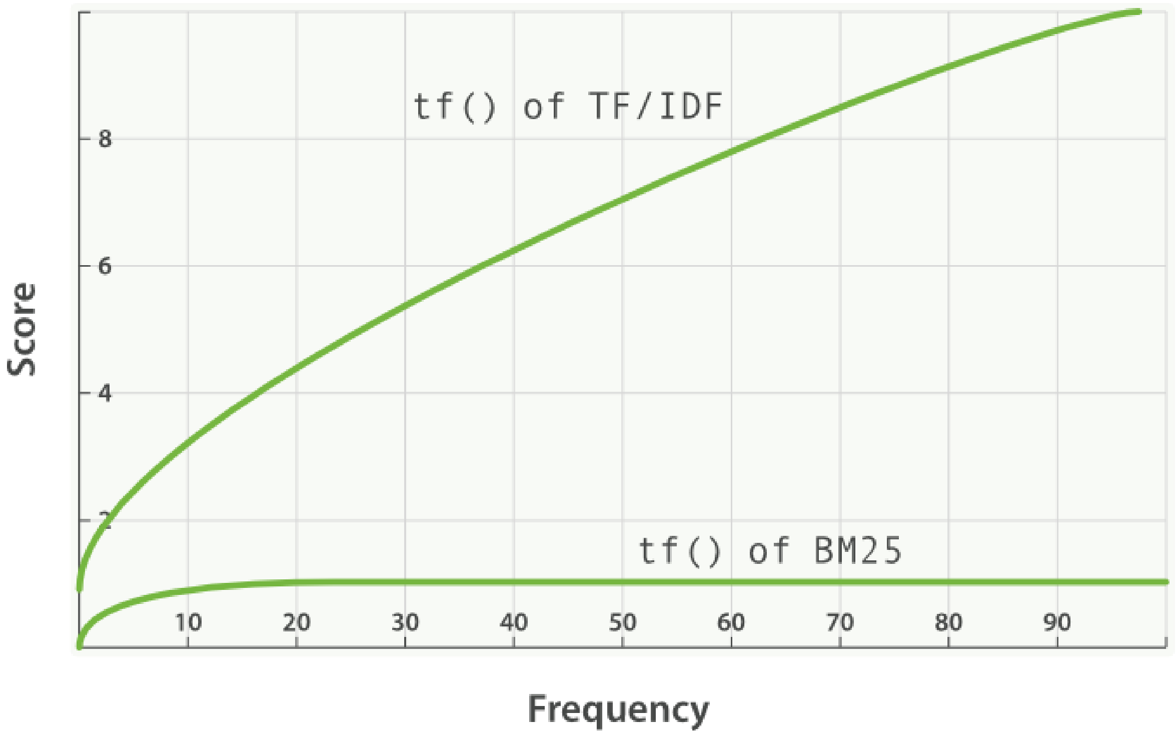


Fig 5.3 Plot to show tf() score of TF-IDF and BM-25 vs the frequency of words

Now, the question is, "when do we think a term is likely to be saturated?"

For very long documents, it's very likely to have a lot of different terms appear several times in a work, even when the term isn't the primary subject of the document. This is the case when our documents, particularly, the ‘**body-text**’ of the json files that we have taken in our dataset, span for more than 10 pages or so. However, this is rarely the case. We may not want terms to be saturated as quickly in this situation, so, the **k1** should generally trend toward larger numbers, when the text is a lot longer and more diverse. On the opposite side of things, **k1** should be set on the lower side. It's very unlikely that a documents having only 2-3 pages of body-text would have a term multiple times without being highly related to the subject of the document.

b: The parameter **b** (bound 0.0 ~ 1.0) in the denominator is multiplied by the ratio of the document length we just discussed. If **b** is bigger, the effects of the document length compared to the average length are more **amplified**. We can imagine if we set **b** to 0, the effect of the length ratio would be completely **nullified**.

Now, the question is, "when do we think a document is likely to be very long, and when should that hinder its relevance to a term?"

Documents which are highly specific for example, which speaks about how effective certain kinds of vaccines are against SARS-COV-2, are lengthy in order to be **more specific** about the subject. Their length is unlikely to be detrimental to the relevance and **lower b** may be more appropriate. On the other end of the spectrum, documents which touch on several different topics in a broad way such as recent advancements in technologies in diagnosis, trends of symptoms of the virus, etc. often benefit by choosing a **larger b** so that **irrelevant topics** to a user's search, including **news articles** and the like, are penalized.

Step 2: Creating summary from each of the ranked documents

- For creating summary, we have extracted the relevant answer-excerpts for each of the queries from the ranked documents.
- For this purpose, we have used the **BertForQuestionAnswering** class from the transformers library.
- We have used a number of models from this class, including "bert-large-uncased-whole-word-masking-finetuned-squad", "bert-base-uncased"," bert-base-cased"," bert-large-cased"," bert-base-cased-finetuned-mrpc"

Workflow

- We begin the task by computing the input_ids by encoding the question and the paragraph text.
- We find the tokens from the question and the paragraph text
- Then we calculate the segment_ids which separates the question from the paragraph text
- We feed the input_ids and segment_ids to the model to calculate the start and end scores
- Finally, we find the answer by indexing the tokens between the highest start and end scores

Input Format

For the Question Answering task, BERT takes the input question and passage as a single packed sequence. The input embeddings are the sum of the token embeddings and the segment embeddings. The input is processed in the following way before entering the model:

Token embeddings: A [**CLS**] token is added to the input word tokens at the beginning of the question and a [**SEP**] token is inserted at the end of both the question and the paragraph.

Segment embeddings: A marker indicating segment A or segment B is added to each token. This allows the model to distinguish between segments.

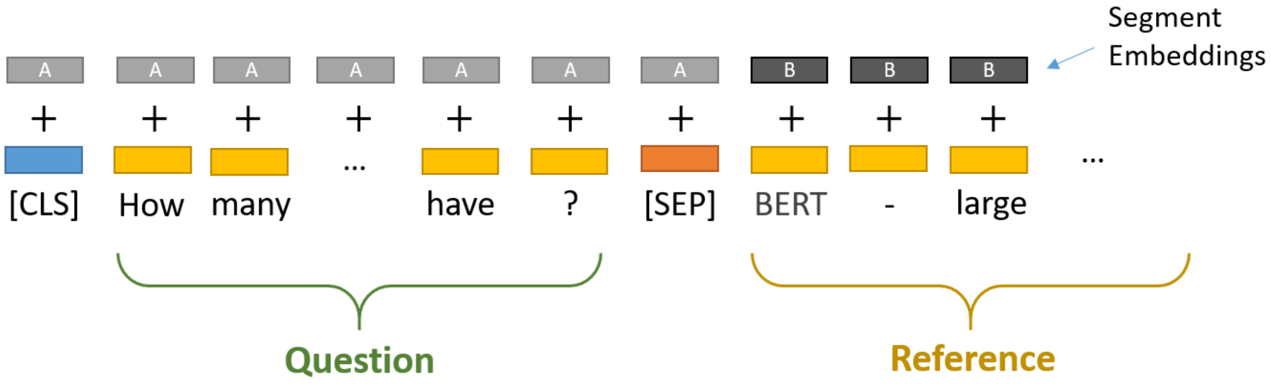
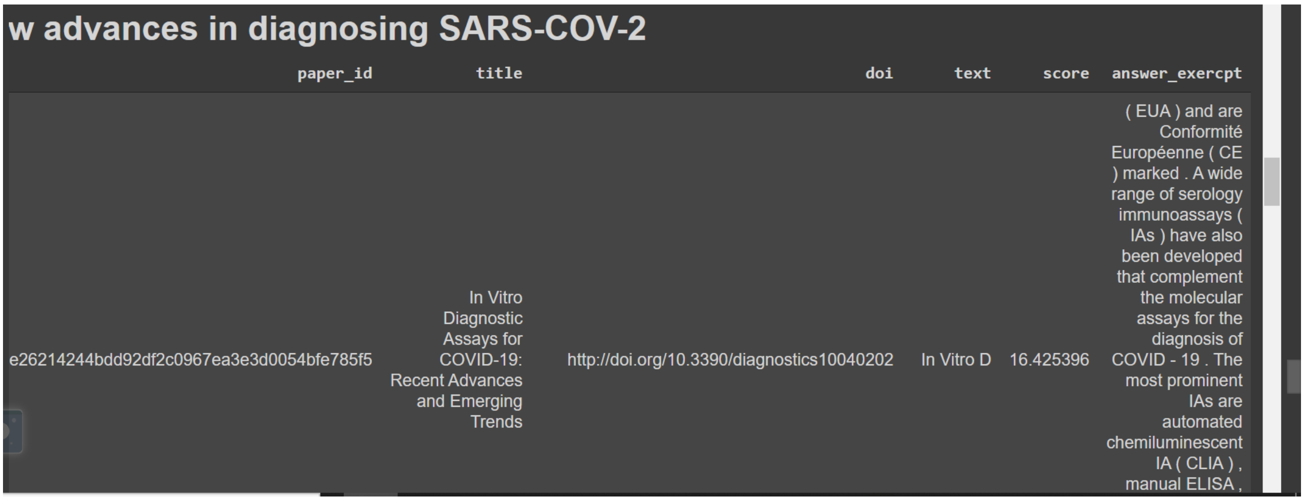


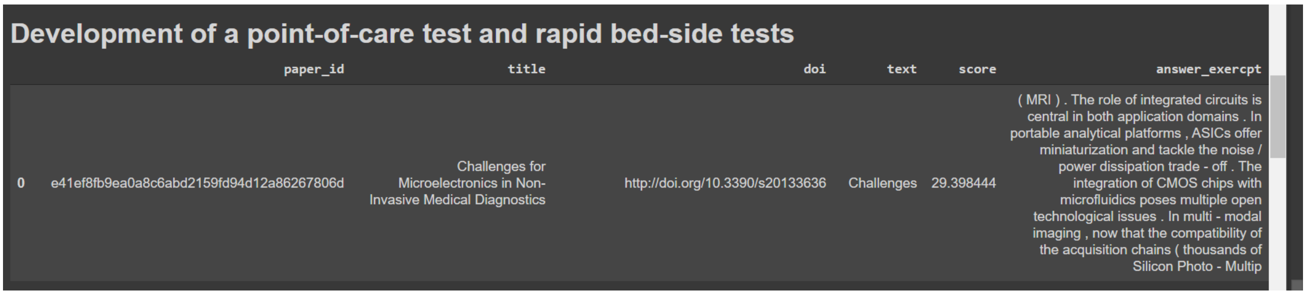
Fig 5.4 Diagram to show how the input is processed before feeding it to the model

Results



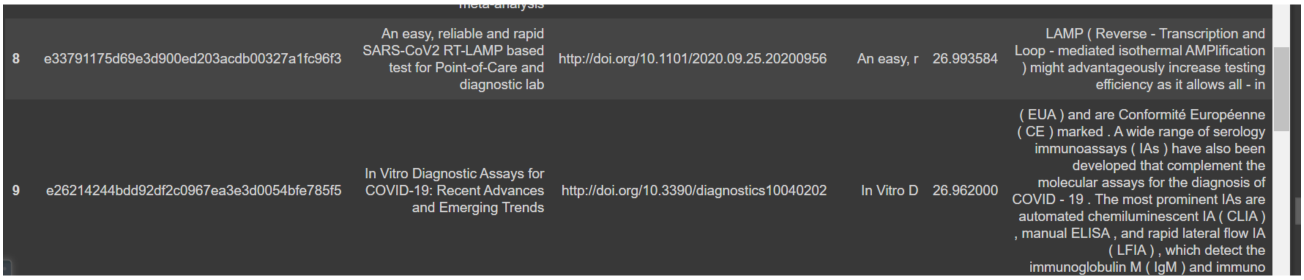
w advances in diagnosing SARS-COV-2

paper_id	title	doi	text	score	answer_exercpt
e26214244bdd92df2c0967ea3e3d0054bfe785f5	In Vitro Diagnostic Assays for COVID-19: Recent Advances and Emerging Trends	http://doi.org/10.3390/diagnostics10040202	In Vitro D	16.425396	(EUA) and are Conformité Européenne (CE) marked . A wide range of serology immunoassays (IAs) have also been developed that complement the molecular assays for the diagnosis of COVID - 19 . The most prominent IAs are automated chemiluminescent IA (CLIA) , manual ELISA ,



Development of a point-of-care test and rapid bed-side tests

	paper_id	title	doi	text	score	answer_exercpt
0	e41ef8fb9ea0a8c6abd2159fd94d12a86267806d	Challenges for Microelectronics in Non-Invasive Medical Diagnostics	http://doi.org/10.3390/s20133636	Challenges	29.398444	(MRI) . The role of integrated circuits is central in both application domains . In portable analytical platforms , ASICs offer miniaturization and tackle the noise / power dissipation trade - off . The integration of CMOS chips with microfluidics poses multiple open technological issues . In multi - modal imaging , now that the compatibility of the acquisition chains (thousands of Silicon Photo - Multip



		meta-analysis				
8	e33791175d69e3d900ed203acdb00327a1fc96f3	An easy, reliable and rapid SARS-CoV2 RT-LAMP based test for Point-of-Care and diagnostic lab	http://doi.org/10.1101/2020.09.25.20200956	An easy, r	26.993584	LAMP (Reverse - Transcription and Loop - mediated isothermal AMPlification) might advantageously increase testing efficiency as it allows all - in
9	e26214244bdd92df2c0967ea3e3d0054bfe785f5	In Vitro Diagnostic Assays for COVID-19: Recent Advances and Emerging Trends	http://doi.org/10.3390/diagnostics10040202	In Vitro D	26.962000	(EUA) and are Conformité Européenne (CE) marked . A wide range of serology immunoassays (IAs) have also been developed that complement the molecular assays for the diagnosis of COVID - 19 . The most prominent IAs are automated chemiluminescent IA (CLIA) , manual ELISA , and rapid lateral flow IA (LFIA) , which detect the immunoglobulin M (IgM) and immuno

Fig 5.5 Results obtained after applying “bert-base-cased-finetuned-mrpc” pre-trained model from Hugging face Transformers library to our dataset. The span of text obtained for each query is shown under the column heading ‘answer-excerpts’.

Conclusion

In Task 1, We were able to retrieve all the task specific documents and in task 2 summary tables for each of the questions in the task were answered and hence The Two tasks was completed successfully and hope could be of help for Medical Research.