Department of Electrical and Computer Engineering

# North South University

CSE499

*Senior Design Project Report*

# SciRet: Scientific Information Made Easy for General Public

*Group Members*

| | |
|---|---|
| Kaysarul Anas Apurba | ID# 1811257042 |
| Md. Ahsanuzzaman | ID# 1813158042 |
| Rofiqul Alam Shehab | ID# 1831185042 |

*Supervisor*

DR. MOHAMMAD ASHRAFUZZAMAN KHAN

Assistant Professor, Department of Electrical and Computer Engineering

North South University, Dhaka, Bangladesh.

*Summer 2022*

# Declaration

It is hereby acknowledged that:

- No illegitimate procedure has been practiced during the preparation of this document.
- This document does not contain any previously published material without proper citation.
- This document represents our own accomplishments while being Undergraduate Students at the North South University.

Sincerely,

| | |
|---|---|
| **Kaysarul Anas Apurba**<br>ID: 1811257042 | **Rofiqul Alam Shehab**<br>1831185042 |
| **Md. Ahsanuzzaman**<br>1813158042 | |

# Approval

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation.

**Dr. Mohammad Ashrafuzzaman Khan**
**Assistant Professor**
Department of Electrical and Computer Engineering
North South University
Dhaka, Bangladesh

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation.

**Dr. Rezaul Bari**
Associate Professor & Chair
Department of Electrical and Computer Engineering
North South University
Dhaka, Bangladesh

# Abstract

The scarcity of authentic information has always been a major inconvenience. Very few of us have access to authentic information and guidelines. Scientific research papers have always been a great source of useful and authentic information. However, because they are complex and scarce, they are frequently useless to the general public, or we can say, the general public does not know how to properly utilize them. To solve this problem, we created SciRet, a passage retrieval system that retrieves data from datasets consisting of scientific research papers to provide easy-to-understand, authentic and informative information to the general public. To test and build the system, we took a document of 1,000,000 Corona Virus-related literature accessible through the "Coronavirus Open Research Dataset Challenge (CORD-19)." As the data was in JSON format, which could not be trained on our RAG model, we converted it to CSV. Before doing that, we applied NLP preprocessing techniques to clean the data to make it more usable for our model. To achieve our desired goal, we initially experimented with a question-answering model called "GPT-Neo." We trained "GPT-Neo" with a portion of data from our main dataset, and the result was quite good but not satisfactory. Based on our previous experiment, we developed another model that could retrieve all the passages related to a user's query. Based on the user's queries, our system will retrieve data from a dataset of scientific research papers using a fine-tuned RAG model and provide us with some passages related to the user's query. Our model opens a door to the world of research papers for the general public, which up until now was visible but not accessible or always usable. Our vision is to help the general public, science enthusiasts, and the front lines of every industry from every corner of this world with significant, authentic, and useful information.

## Table of Contents

**Declaration** ii

**Approval** iii

**Abstract** iv

**List of Tables** viii

**List of Illustrations** ix

**Acknowledgments** x

**List of Acronyms** Error! Bookmark not defined.

**Glossary** Error! Bookmark not defined.

**1 Introduction** **11**

**1.1 Background and Motivation** 11

**1.2 Question Answering System** 12

**1.3 Technical Concept** 13

**1.4 Project Aim & Objectives** 14

*1.4.1 Project Aim* *14*

*1.4.2 Project Objectives* *15*

**1.5 Cognitive Challenges** 15

**2 Related Works** **17**

**2.1 COBERT** 17

*Covid-19 question answering system using BERT:* *17*

**2.2 Smart Uniquitous Chatbot** 19

**2.3 Health Care Chatbot** 20

**2.4 DPR** 21

**3 Project Development Process** **24**

**4 Background & Design of The System** **26**

**4.1 Analysis of the design principles** 26

*4.1.1 Methodology* *27*

*4.1.2 Data Pipeline* *27*

*4.1.3 Models Evaluation* *31*

**4.2 Usability** 32

**4.3 Manufacturability** 33

**4.4 Sustainability** 33

**5 Project Implementation** **35**

**5.1 System Architecture** 35

*5.1.1 Implementation of GPT-Neo:* *35*

*5.1.2 Training GPT-Neo:* *36*

*5.1.3 Implementation of RAG:* *36*

*5.1.4 Encoder* *36*

*5.1.5 Document Store* *37*

*5.1.6 Retriever* *37*

*5.1.7 Training RAG:* *37*

**5.2 Models Implementation:** 38

**6 Project Impact** **39**

**6.1 Economic Impacts** 39

**6.2 Social Impacts** 40

**6.3 Political Impacts** 40

**6.4 Health Impacts** 40

**7 Environment considerations & Sustainability** **42**

**7.1 Environment Consideration** **Error! Bookmark not defined.**

**7.2 Environment Sustainability** **Error! Bookmark not defined.**

**8 Ethical & Professional Responsibility** **44**

**8.1 Ethical Responsibility** 44

**8.2 Professional Responsibility** 45

**9 Tools & Technology Used** **46**

**9.1 Software Requirements** 46

**9.2 Hardware Requirements** 46

**9.3 Project Management Tools** 47

**10 Result Analysis** **48**

**11 Conclusions ____ 51**

**11.1 Our Findings** ____ 51

**11.2 Future Works** ____ 52

**12 References** ____ Error! Bookmark not defined.

# List of Tables

Table 1 : Models and Arguments........................................................................38
Table 2 All Tech and Tools At A Glance.....................................................................47

# List of Illustrations

Figure 1: Google Question Answering....................13
Figure 2 Development Process....................24
Figure 3 ML Project Design Principles....................26
Figure 4  Counting All the Files Existing in Dataset....................28
Figure 5 Most Common Words in Title of papers in CORD-19 Dataset....................28
Figure 6 Most 3- Common Words In Title of papers in CORD-19 Dataset....................29
Figure 7 Most Common Date of Publication....................29
Figure 8  Word Cloud Function....................30
Figure 9  Sample Word Cloud....................30
Figure 10 Collecting The Papers Related to Covid Only....................31
Figure 11 System Diagram of GPT- Neo Implementation....................35
Figure 12 GPT Neo Training Args.....................36
Figure 13 System Architecture....................36
Figure 14 RAG Training....................37
Figure 15 Results Of GPT Neo....................48
Figure 16 Document Search Pipeline....................49
Figure 17 Retrieved Top 5 Passages Using Document Search Pipeline - Sample 1....................50
Figure 18  Retrieved Top 5 Passages Using Document Search Pipeline - Sample 2....................50

# Acknowledgments

Towards the accomplishment of this project, the constant support, encouragement, and guidance of Dr. Mohammad Ashrafuzzaman Khan, our honorable supervisor for this project were indispensable. We would like to express our profound gratitude to our honorable supervisor, Dr. Mohammad Ashrafuzzaman Khan, for his diligent guidance, valuable feedback, patience, and encouragement towards the completion of this project. At times we were in a state of great doubt about the completion of this project but Dr. Mohammad Ashrafuzzaman Khan always encouraged, directed, and supported us with conceivable solutions. Without his help, it would not have been possible for us to bring our project to its current state. Finally, we would like to thank everybody who supported us and provided us with counsel for the completion of this project.

# 1 Introduction

The matching of a user-generated query against a collection of text data is what is meant by the term "Passage retrieval." Any sort of record could be included in this collection, including a great amount of unstructured text, huge documentation, textual reports, newspaper stories, scientific journals, etc. User queries can be as long as several sentences or as short as a few words. Information retrieval methods include text retrieval as a subset. Most question answering system aims to locate answers from large collection of existing documents publicly available. Information Extraction (IE) and Information Retrieval (IR) components are typically combined to achieve this. Over the years, many researchers have been trying to use NLP for information retrieval, which has been argued to be inefficient and flimsy [1]. This section vividly describes the background motivation and challenges of our study.

## 1.1 Background and Motivation

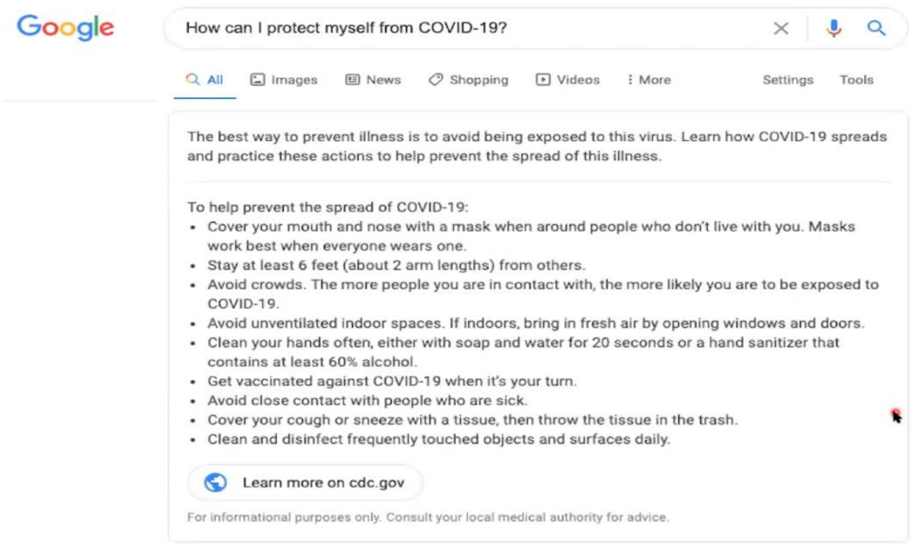
Scientific research plays a vital role in shaping the new world in better ways. Whenever we start doing research on a particular topic, information on that topic becomes the most important of all. Many people become demotivated while researching because they are not getting proper information. Sometimes researchers struggle to get proper information because they do not know where to find that information. Aside from that, if anyone wants to write an essay or paragraph or a blog on a topic, it becomes so messy with a lot of redundant information that they get confused about what to use and how to use that information.

Often, we lack the time to extract data in-depth for each attribute throughout the full corpus. As a result, information retrieval (IR) and standard question answering (QA) approaches are frequently combined to handle this type of task. We must start with passage retrieval, just like QA, where a passage might be anything from a sentence to a paragraph to a document. However, unlike QA, we have a set of fixed relations and a set of predefined expected answer types (e.g., a person's employer). This enables us to use IR's more advanced learning techniques to fine-tune the passage retrieval for each relationship of interest [2].

## 1.2 Question Answering System

The internet is a blessing for us, and because of the internet, we have come to a place where we can find a lot of data publicly available without any hassle. But the problem is searching for necessary data at a time. As public storages are huge, the quantity of data is also huge. So, exploring this huge amount of data to find any information is a very complex and expensive task. This problem has prompted the creation of new, more adaptable research instruments, such as Question Answering Systems. QA systems aim at satisfying users who are looking to answer a specific question in natural language [3].

The goal of a question-answering system is to build a system that automatically answers questions posed by Human In NLP. If we look into our daily life from using google to different social networks NLP based question answering systems or chatbots can be seen everywhere. If we see the diagram (Figure 1) below we will see that when a search engine is asked a question it answers in this manner.



The best way to prevent illness is to avoid being exposed to this virus. Learn how COVID-19 spreads and practice these actions to help prevent the spread of this illness.

To help prevent the spread of COVID-19:

- Cover your mouth and nose with a mask when around people who don't live with you. Masks work best when everyone wears one.
- Stay at least 6 feet (about 2 arm lengths) from others.
- Avoid crowds. The more people you are in contact with, the more likely you are to be exposed to COVID-19.
- Avoid unventilated indoor spaces. If indoors, bring in fresh air by opening windows and doors.
- Clean your hands often, either with soap and water for 20 seconds or a hand sanitizer that contains at least 60% alcohol.
- Get vaccinated against COVID-19 when it's your turn.
- Avoid close contact with people who are sick.
- Cover your cough or sneeze with a tissue, then throw the tissue in the trash.
- Clean and disinfect frequently touched objects and surfaces daily.

For informational purposes only. Consult your local medical authority for advice.

*Figure 1: Google Question Answering*

## 1.3 Technical Concept

Transfer learning is a machine learning technique in which a model created for one job is utilized as the basis for a model on a different task. Given the vast compute and time resources required to develop neural network models on these problems, as well as the huge jumps in skill that they provide on related problems, it is a popular approach in deep learning where pre-trained models are used as the starting point on computer vision and natural language processing tasks. Transfer learning can be done using two approaches [4].

1. Develop model approach

2. Pre-trained model approach

The Developing model approach requires four steps, which are selecting the source task, developing the source model, reusing the model, and tuning the model. The pre-trained model approach requires 3 steps, which are selecting the source model, reusing the mode, and tuning the model. Because these steps complicate and lengthen the model-building process, we took a different approach than other works currently available.

At the very beginning, we used a Generative Pre-trained Transformer (GPT) model, which does not need transfer learning, where the model is fine-tuned on task-specific data sets for specific tasks. This helped us to stay away from the complications of transfer learning and bought our model some valuable time. We used GPT-Neo, an open-source transformer model with 2.7 billion parameters. This model leverages the power of GPT-3 but without the hassle of going through the application process and so on.

GPT-Neo can be trained from scratch using a mesh-TensorFlow library. Parallelism is very vital, as these models have tons of data to train on and lots of parameters. So, we are able to run different segments of our training simultaneously rather than doing it one after another. This provides complete independence between different batches [5].

For recovering the passage from which the query was generated, we also used another pre-trained model called "Retrieval-augmented generation ("RAG")". This model is the combination of sequence-to-sequence models and pretrained dense retrieval (DPR). A seq2seq model is used by RAG models to obtain documents, which are then marginalized to produce outputs. Retrieval and generation can both adapt to subsequent tasks thanks to the retriever and seq2seq modules' cooperative fine-tuning of their initialization from pre-trained models [6].

## 1.4 Project Aim & Objectives

### 1.4.1 Project Aim

During the Corona Virus pandemic, the world has seen many lives getting lost without getting proper medical treatment. Even after much research and publication were available, people couldn't know all of the answers at their fingertips. So, the primary purpose of our project was to serve people with proper solutions related to Covid-19 so that they can keep themselves safe.

As coronavirus evolves day by day, researchers are releasing new information about this virus every day. But it is pretty challenging for medical professionals and the general public to know all this information quickly because reading and searching data from whole documents are time-consuming. So, our system will provide them with answers with the latest information. It will save a lot of time, and people can be aware easily.

The primary purpose of providing this passage retriever system design was to address COVID-19 issues and make it easier for people from the medical research field and medical workers to obtain accurate scientific information. Though we started our project with the vision of helping people during the COVID-19 pandemic, now our goal is to make a system where people from all sectors will get easier access to scientific information.

### 1.4.2 Project Objectives

Our study is targeted toward the following aspirations: -

- Make the dataset usable with proper information.
- Data Analysis.
- Creating embedding
- Suitable model selection for passage retrieving
- Suitable model selection for answer generation

## 1.5 Cognitive Challenges

First of all, the dataset collection was a very difficult task. As we were planning to do novel work that is related to future research, we were looking for a resource with a huge amount of authentic scientific data. When we get the COVID-19 dataset publicly available on Kaggle, we have to use many preprocessing steps before making that dataset usable for our model. The dataset contains a lot of information, from which we chose to use the important ones. Moreover, there was a lot of garbage information, which would have confused our model while generating answers. So, we had to look after that also. Another important task was choosing the right model, which could do our work fine. As there are a lot of existing models, we had to choose our model based on performance. Before finalizing our model, we tried many different models and finally settled on one. At that time, we have to work on our data once again. The model chosen by us needed a dataset with different parameters like vector embeddings. So, we had to create embedding for both dataset passages and queries asked by users. The dataset size was very large (64 GB), and to run the model we needed a high-configuration computational setup, which we did not have. But, through code optimization (details given in), we were able to finally do our experiments. We also used "Google Colab," a cloud platform for machine learning development, to conduct our experiments.

# 2 Related Works

This project required pretty heavy background studies as the resources needed to get started with the project were not abundant. We went through several research papers, online documentation, and tutorials and sought instructions from the professionals to get to know our project better. From a vast selection of research papers, some came in handy and provided us with many valuable insights, which were helpful while doing our project. We could compare the outputs of different models and algorithms and decide which ones to use in our project and which ones to avoid. In this part, we'll be sharing our learning and insights from three out of the many research papers we went through, as these three have many similarities with our project, and they were able to show us clear paths to get started with our work.

## 2.1 COBERT

**Covid-19 question answering system using BERT:**

With COBERT, the authors tried to present a retriever-reader dual algorithmic system that could answer complex queries by searching a document of 59 thousand Corona Virus-related literature, which they collected from the Coronavirus Open Research Dataset Challenge (CORD-19) [7]. They composed the retriever with a TF-IDF vectorizer which could capture the top 500 documents with optimal scores out of the abundance of documentation. In this project, the reader was a pre-trained Bidirectional Encoder Representations from Transformers (BERT) of the SQuAD 1.1 developer dataset, which was built on top of the Huggingface BERT transformers. This could refine the sentences from filtered documents, which after that got passed into ranker to compare the logits score to receive a short answer. The approach taken to execute this model was to take queries from the users as input and generate the most accurate responses consisting of a single line answer to the query, the title of the literature, and a whole paragraph from the scientific literature. [8]

We decided to follow this research for the information it offered and the promise it showed. The DistilBERT version used in this project outperformed previous pre-trained models by obtaining an Exact Match (EM) score of 80.6 and F1 score of 87.3 [9] . Even though we went through several other QA models, COBERT always stood apart from the others. Most other QA models were extractors and re-rankers and worked on small-sized datasets of texts as articles and sentences. But the framework of COBERT worked on a huge amount of data, or as they have mentioned in their paper, a huge corpse of literature. It coordinated best practices from information retrieval with BERT to create a framework focusing on an end-to-end closed domain question answering system and analysis on a standard benchmark showing improvement over the previous work. This model fine-tuned pre-trained versions of BERT with SQuAD and accomplished high scores in recognizing answers. [9]

To execute properly, firstly, they ran a retrieval throughout the whole corpus which was divided into paragraphs thus creating features based on TF-IDF focusing on bigrams and unigrams. Then, the embedding is used to calculate the cosine similarity with the query. By this the sequential comparison score are obtained which are then used to retrieve the top 500 documents with optimal scores as mentioned earlier. The third step is to perform comparisons batch wise to get the document that is most portable to contain the answer. The extraction of the documents is then done by the reader by splitting them into sentences and these sentences are then fine-tuned with BERT so that the automatically generated answers get refined based on the similarity between the query. In this step DistilBERT is also used for fine-tuning. To prepare for the final step, the ranker ranks the most accurate answers using weighted scores, threshold scores obtained from the retrieval and the reader. The scores from each candidate, answer from each paragraph is compared by the ranker. It also compares the logits of the best answer for each paragraph. And then finally, the best answer from that document is produced and given as output to the user. [9]

The main challenge that the authors kept in mind throughout this procedure is that QA models often fail to perform when asked to produce an answer for the question from a large input text. To address this, COBERT model was broken into two steps. Firstly, the input text got narrowed down to the top articles where the answer might be present using search (e.g., TF-IDF, BM25). Secondly, out of the narrowed down input text, the best potential answer was found using a QA model. [9]

## 2.2 Smart Uniquitous Chatbot

**A chatbot for Covid-19 assistance with deep learning sentiment analysis model during and after quarantine:**

This paper presented a smart ubiquitous chatbot called COVID-Chatbot, which would provide assistance during and after quarantine that communicates with a user to increase consciousness towards the real danger of this outbreak [10]. This would also be used to recognize and manage stress, during and after lock-down and quarantine period using Natural Language Processing (NLP).

This smart chatbot was made with some fixed visions. It was built to help people understand and accept the coronavirus quarantine in order to limit the rapid spread of the viral disease. To act on this vision, this model included four interdependent modules. Information Understanding Module (IUM), Data Collection Module (DCM), Action

Generator Module (AGM) and Depression Detector Module (DDM). Here, the NLP based tasks are done in the IUM. The DCM collects user's non-confidential information to be used later by the AGM. Then, the AGM generates the cat bots' answers which are managed through its three sub-modules [11]. Finally, the DDM detects anxiety in the text input through a deep learning sentiment analysis model. If anything is detected, the AGM deliver a reassurance message. The used NLP tasks are Tokenization, Part of Speech Tagging (POS tagging), Lemmatization and Stemming. In the Action Generator Module (AGM) consists of 3 parts. Response Classifier: which is the main sub-module of AGM; Daily Medical Follower: which stores the information on 14 quarantine days; and finally, Off-topic Input Manager: for returning a warning message if any non-serious aspect is shown by the user to avoid unnecessary discussions. At the end of the road, this project showed very promising results with an F1 score of 80.78% in LSTM layer, and accuracy of 92% in both LSTM and GRU models. The main model was the LSTM to get higher precision. [12]

## 2.3 Health Care Chatbot

**Healthcare chatbot using natural language processing:**

The proposed idea of this project was to create a healthcare chatbot using Natural Language Processing that can diagnose a disease and provide basic steps to take. The vision was to reduce healthcare costs and improve accessibility to medical knowledge. The system would provide text or voice assistance in a user's convenient language. Based on the user's symptoms it would provide medical assistance and food suggestions. Thus, people will have an idea about their health and have the right protection.

To execute the model, three algorithms were used. 1. N-gram Algorithm, 2. TF-IDF and 3. Cosine Similarity Algorithm. With the N-gram, the machine was helped to understand a word in the content to get a better understanding of the word. In a contiguous sequence of some items, it helped in predicting the next words in a sentence. The TF-IDF is used to score the relative importance of words. This divides the number of times a word is used during the document by the entire number of words within the document. This returns the Term Frequency (TF). This also determines the weight of unique words across all document in the corpus using Inverse Data Frequency (IDF). The final algorithm is the Cosine Similarity Algorithm which finds similarity between two non-zero vectors of an inner product space that measures the cosine of the angle between them. It also measures the cohesion among clusters within the field of information mining. [13]

## 2.4 DPR

**Dense Passage Retrieval for Open-Domain Question Answering:**

TF-IDF and BM25 are the most common ways of passage retrieval up until now [14]. But the authors of this paper brought something new to the table. They practically implemented retrieval using dense representations alone, where embeddings are learned from a small number of questions and passages by a simple dual encoder framework. Upon evaluation on a wide range of open-domain QA datasets, their dense retriever outperformed a strong LuceneBM25 system by an astonishing 9%-19% absolute in terms of top-20 passage retrieval accuracy.

As a result of reading comprehension models, a much simplified two-stage framework came up. A context retriever first selected a small subset of passages where some of them contained the answer to the question. A machine reader could thoroughly examine the retrieved contexts and identify the correct answer [15]. So, if the question "Who is the bad guy in Lord of The Rings?" is considered, which could be answered from the context "Sala Baker is best known for portraying the villain Sauron in Lord of The Rings trilogy", a term-based system would have difficulty retrieving such a context. But a dense retrieval system was able to better match “bad guy” with “villain” and fetch the correct context.

Before this research took place, ORQA was successful to show that dense passage can outperform BM25 by setting state-of-the-art results on multiple domains, but it suffered from two weaknesses. ICT pre-training is computationally intensive and it is not completely clear that regular sentences are good surrogates of questions in the objective function. As the context encoder is not fine-tuned using pairs of questions and answers, the corresponding representations could be suboptimal. The authors of this paper tried to outperform ORQA and tried to train a better dense embedding model using only pairs of questions and passages, without additional pre-training.

This research was executed in 2 major stages. First, the proper training setup was demonstrated, the question and passage encoders were simply fine-tuned on existing question-passage pairs which was sufficient to greatly outperform BM25. At this point, empirical results also did not show any necessity of additional pre-training. Second, it ensured that a higher retrieval precision actually translated to a higher end-to-end QA accuracy. They achieved better results on multiple QA datasets in an open-retrieval setting by applying a modern reader model to the top retrieved passages.

As data, Wikipedia dump from Dec. 20, 2018 was chosen as source document for answering questions. After cleaning an article of Wikipedia by pre-processing [15], the article was split into multiple, disjoint text blocks of 100 words as passages, which later served as retrieval unit. Going with this [16], 21,015,324 passages were received and each passage was prepended with the title of the Wikipedia article where the passage is from, along with an [SEP] token.

The question and passage encoders were trained for up to 40 epochs for large datasets and 100 epochs for small datasets, with a learning rate of 10-5 using Adam, linear scheduling with warm-up and dropout rate 0.1. At the end of their work, with the exception of SQuAD, DPR performed consistently better than BM25 on all datasets [17].

# 3 Project Development Process

The enormous amount of textual data that is readily available and being produced on a daily basis is one reason why NLP initiatives are becoming more and more popular [18]. In this part of the chapter, we will discuss how we carried out our whole NLP project. Starting doing research and then finally ending up with a promising project was not an easy task to do.

The process of planning and allocating resources to fully develop a project or product from concept to go live is known as project development. There are numerous things that come up during the project development process.  Here are the steps below that we undertook to accomplish our project.

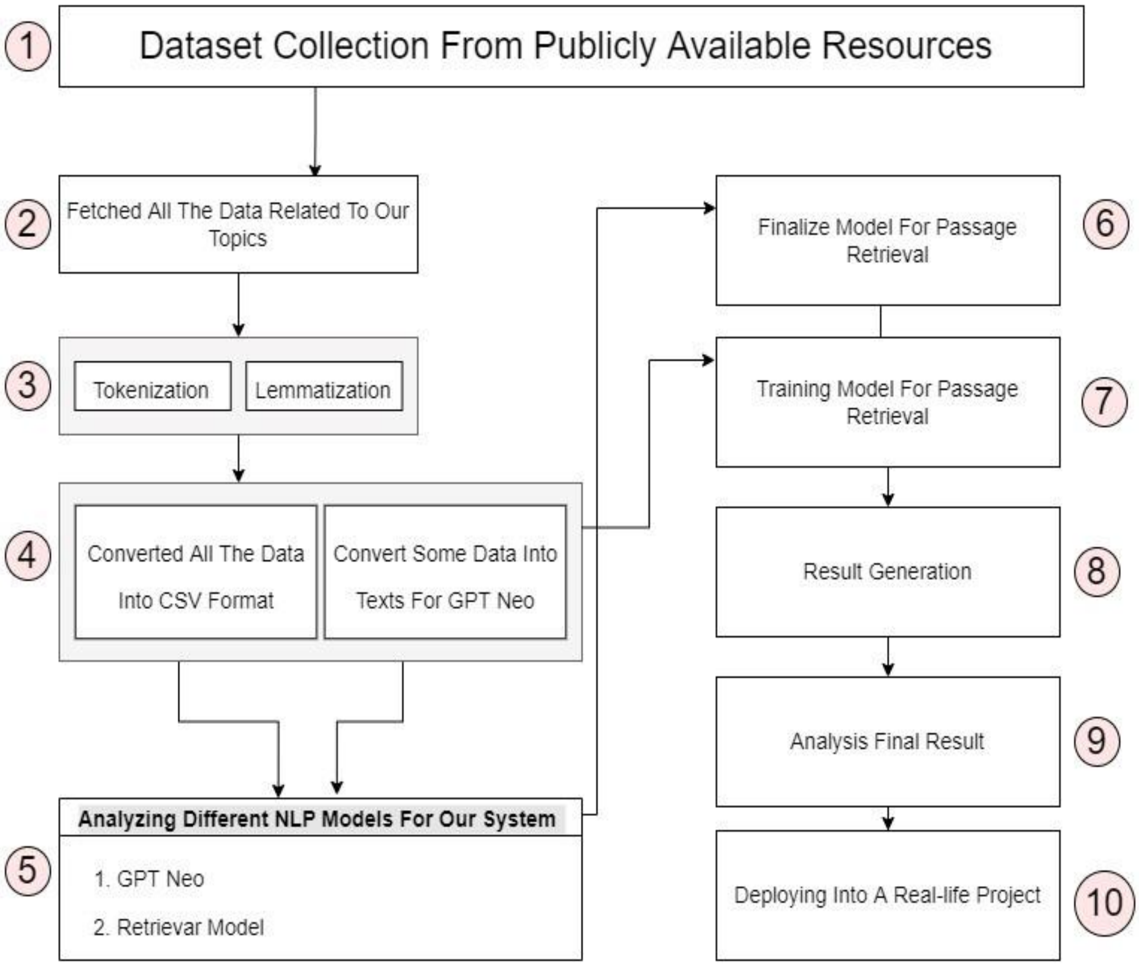

1
Dataset Collection From Publicly Available Resources
2
Fetched All The Data Related To Our Topics
3
Tokenization
Lemmatization
4
Converted All The Data Into CSV Format
Convert Some Data Into Texts For GPT Neo
5
Analyzing Different NLP Models For Our System
1. GPT Neo
2. Retrievar Model
6
Finalize Model For Passage Retrieval
7
Training Model For Passage Retrieval
8
Result Generation
9
Analysis Final Result
10
Deploying Into A Real-life Project

*Figure 2 Development Process*

The figure shows the sequence of the tasks approached by us during this study.

First, we selected our dataset from Kaggle [19]. Then we have selected the category of data we are going to choose for our project. We have generated a CSV file with 115000 rows of text data bearing four important columns, such as paper id, title, abstract, and body_text. By applying basic NLP techniques, we selected all COVID-related papers only [details given in chapter 4.1.2 (Data Pipeline)]. Once we have collected all our data, we might discover that the data is unstructured. Processing the language to make it clear, succinct, and meaningful is the next crucial step. Which we have done by applying tokenization and lemmatization.

After that, we have taken different approaches at different times to select the proper model. We trained our model with preprocessed data and generated excellent results. Finally, we have deployed our project to give it a more real-life visualization. A detailed discussion of these steps is shown in the following sections.

# 4 Background & Design of The System

## 4.1 Analysis of the design principles

We all know that NLP is a part of machine learning. When a machine learning system is created, it is created in a way that it can generate the most business value. Most of the time, developers are solely concerned with attaining state-of-the-art (SOTA) performance and creating creative model designs. In practice, things are a little different, and data scientists have a lot more work to do [20].

For the machine-learning designer, choosing and refining features is frequently a significant activity. The task must be sufficiently stated before we can fully decide what characteristics we need, and of course, both the objective and the features are limited by the kinds of workable models we can create [20].

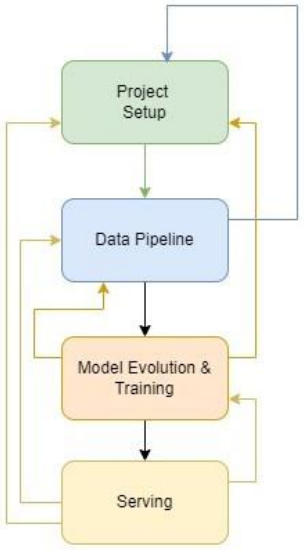


*Figure 3 ML Project Design Principles*

A machine learning system's design approach is iterative. Even though there isn't a one-size-fits-all solution, it's typically wise to keep your system updated. We can see in the figure above that the output of each phase feeds back to the stages before it, depending on the way we seek to increase the product's overall quality with optimizations, error fixes, and general quality upgrades.

### 4.1.1 Methodology

Before we start to build models, we should be ready to gather enough information about the project we are going to work on. One of them is to set goals. What are we hoping to gain with this project? should be the first question that comes to our minds. Aside from that, we should consider user experience and performance constraints, which will aid in determining the acceptable tolerance for errors. Another important thing to think about is model evaluation, which includes a solid understanding of our model in production. And finally, the project constraints, like how much computational power is available for training and deploying, available tools that could be used, etc.

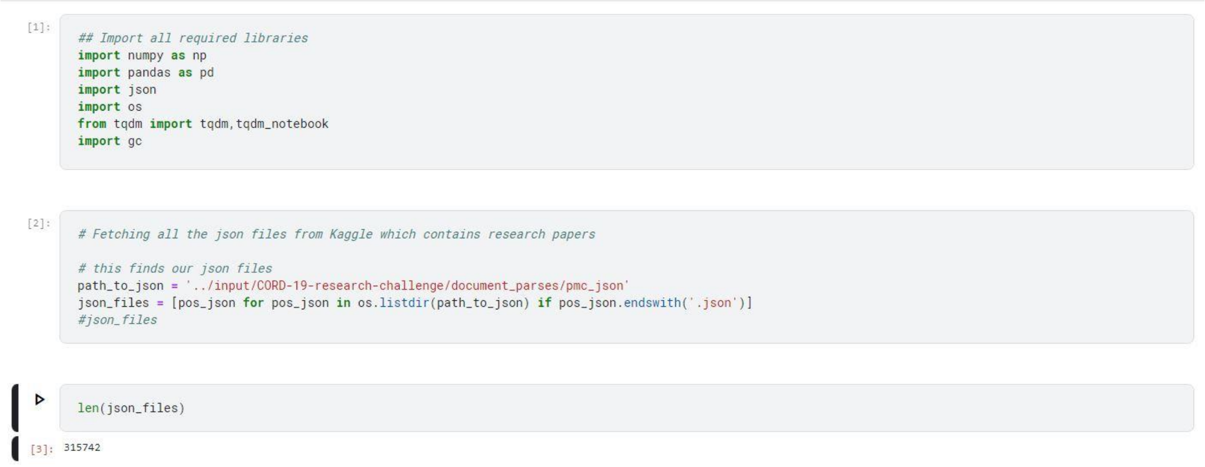
### 4.1.2 Data Pipeline

**Dataset Collection:**

The dataset was obtained from the open-source Kaggle. The dataset name is "COVID-19 Open Research Dataset Challenge (CORD-19)" [19]. The dataset contained various types of files, including the document parsing file containing the PMC Jason file. The size of the document parsing file was around 3 lakh 15 thousand.

**Data Analysis:**

CORD-19 is a resource of over 400,000 scholarly articles, including over 150,000 with full text, about COVID-19 and the coronavirus group. The dataset contained various types of files, including the document parsing file containing the PMC Jason file. The size of the document parsing file was around 3 lakh 15 thousand.



```
[1]:
## Import all required libraries
import numpy as np
import pandas as pd
import json
import os
from tqdm import tqdm,tqdm_notebook
import gc
```

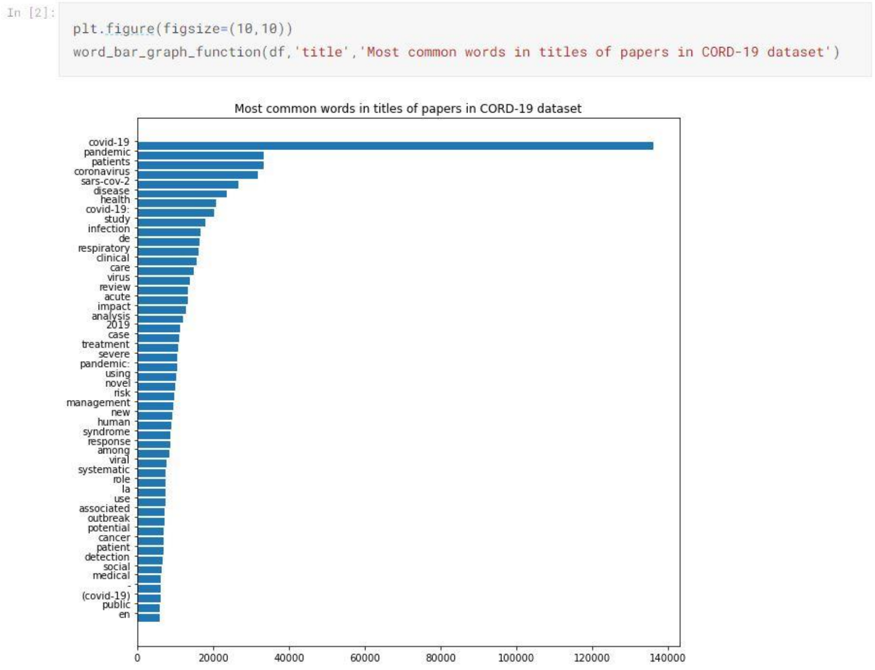
```
[2]:
# Fetching all the json files from Kaggle which contains research papers

# this finds our json files
path_to_json = '../input/CORD-19-research-challenge/document_parses/pmc_json'
json_files = [pos_json for pos_json in os.listdir(path_to_json) if pos_json.endswith('.json')]
#json_files
```

```
len(json_files)
```

```
[3]: 315742
```

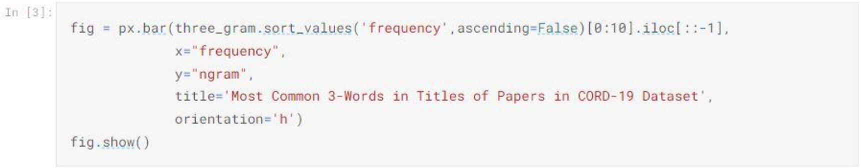
*Figure 4  Counting All the Files Existing in Dataset*

Before selecting the dataset for future use, we tried to analyze the dataset in depth. We tried to see the most common words in the title, then we looked for most common three words in title which will help us future to select the papers which will be useful to us.



```
In [2]:
plt.figure(figsize=(10,10))
word_bar_graph_function(df,'title','Most common words in titles of papers in CORD-19 dataset')
```

*Figure 5 Most Common Words in Title of papers in CORD-19 Dataset*



```
fig = px.bar(three_gram.sort_values('frequency',ascending=False)[0:10].iloc[::-1],
             x="frequency",
             y="ngram",
             title='Most Common 3-Words in Titles of Papers in CORD-19 Dataset',
             orientation='h')
fig.show()
```

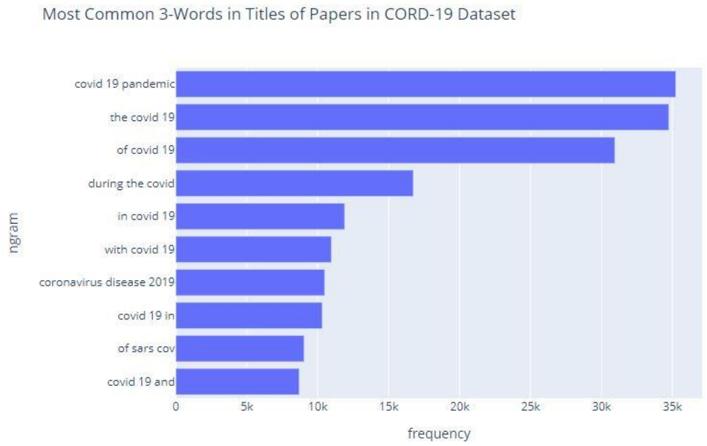


*Figure 6 Most 3- Common Words In Title of papers in CORD-19 Dataset*

We also checked the most common year of all the journals' publication. The Covid-19 pandemic was heated at the beginning of the year 2020. We wanted to make sure that all the papers were up-to-date. We felt it was very important for us to ensure that the information we provide to our users is updated.

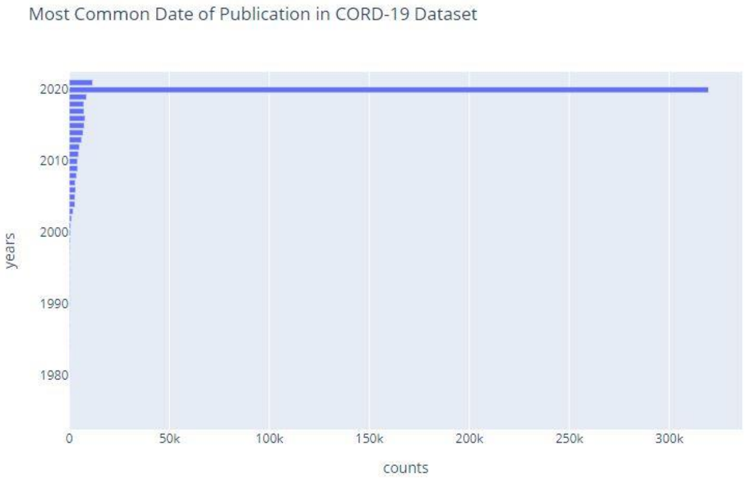
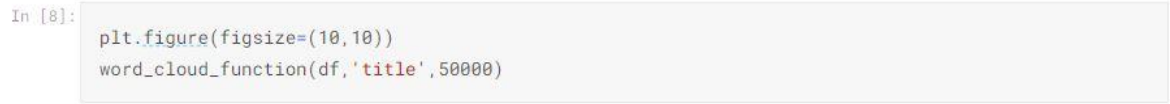


*Figure 7 Most Common Date of Publication*

Finally, we generated a word cloud using word cloud function to have a look on the overall scenario.



```
In [8]:
plt.figure(figsize=(10,10))
word_cloud_function(df,'title',50000)
```

*Figure 8  Word Cloud Function*

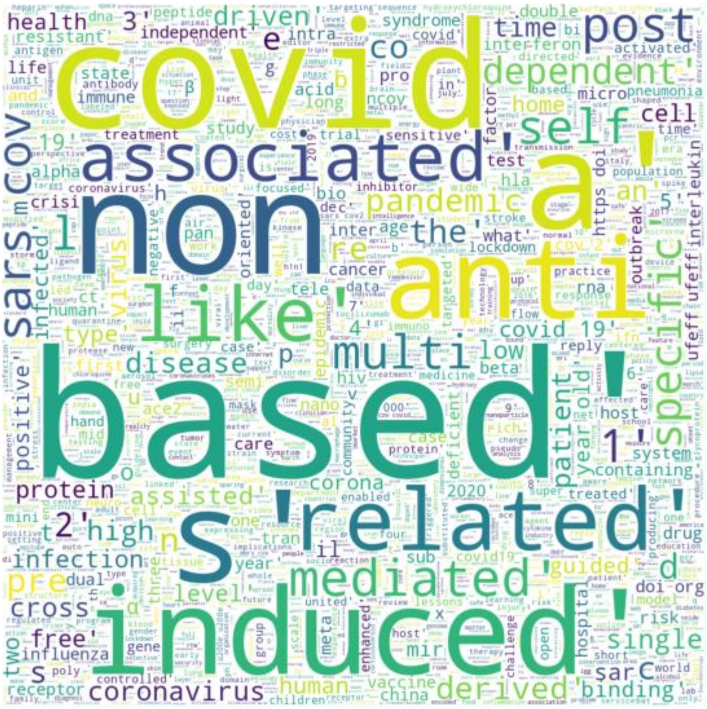
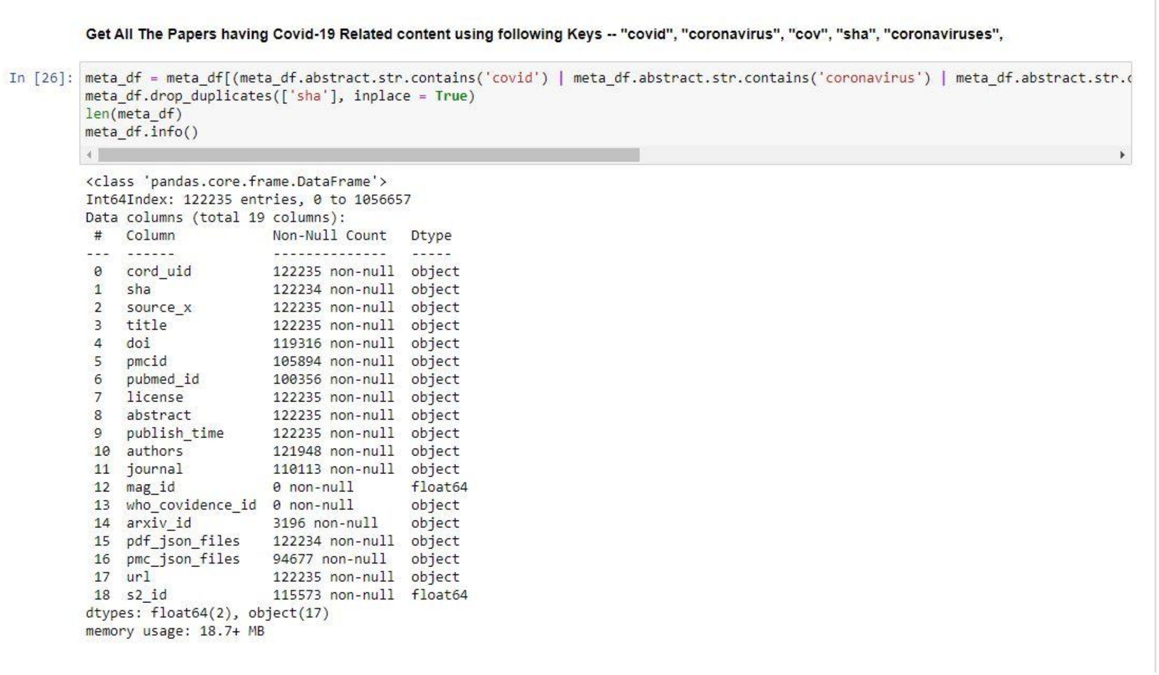


*Figure 9  Sample Word Cloud*

### Dataset Pre-Processing:

At first, we took 1000 files from the PMC_Json folder, where we took only the body text and title columns of those files and converted them into a CSV file. After that, we converted the CSV file into a (.txt) file to feed the data to our first model. As the model itself had built-in functions like tokenization and lemmatization for cleaning the data, we did not have to implement any regular NLP approach for cleaning the data.

For the second model, we generated a CSV file having 1,15,000 rows bearing only COVID-related paper information. First, we created indexes for the dataset and stored it in a database using a library called "FaissIndex Factory" [21]. Then we load the dataset to feed our model. Here are details of CSV file we generated using covid-19 related keywords.



**Get All The Papers having Covid-19 Related content using following Keys -- "covid", "coronavirus", "cov", "sha", "coronaviruses",**

```
In [26]: meta_df = meta_df[(meta_df.abstract.str.contains('covid') | meta_df.abstract.str.contains('coronavirus') | meta_df.abstract.str.c
         meta_df.drop_duplicates(['sha'], inplace = True)
         len(meta_df)
         meta_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 122235 entries, 0 to 1056657
Data columns (total 19 columns):
 #   Column             Non-Null Count   Dtype  
---  ------             --------------   -----  
 0   cord_uid           122235 non-null  object 
 1   sha                122234 non-null  object 
 2   source_x           122235 non-null  object 
 3   title              122235 non-null  object 
 4   doi                119316 non-null  object 
 5   pmcid              105894 non-null  object 
 6   pubmed_id          100356 non-null  object 
 7   license            122235 non-null  object 
 8   abstract           122235 non-null  object 
 9   publish_time       122235 non-null  object 
 10  authors            121948 non-null  object 
 11  journal            110113 non-null  object 
 12  mag_id             0 non-null       float64
 13  who_covidence_id   0 non-null       object 
 14  arxiv_id           3196 non-null    object 
 15  pdf_json_files     122234 non-null  object 
 16  pmc_json_files     94677 non-null   object 
 17  url                122235 non-null  object 
 18  s2_id              115573 non-null  float64
dtypes: float64(2), object(17)
memory usage: 18.7+ MB
```

*Figure 10 Collecting The Papers Related to Covid Only*

### 4.1.3 Models Evaluation

**GPT:**

We introduce the 20 billion parameter autoregressive language model GPT-NeoX-20B, which was trained on the Pile. GPT-Neo 125M is a transformer model designed using EleutherAI's replication of the GPT-3 architecture. GPT-Neo refers to the class of models, while 125M represents the number of parameters of this particular pre-trained model. The Pile, a sizable curated dataset produced by EleutherAI [22] specifically for the purpose of training this model, served as the training data for GPT-Neo 125M.

We used GPT neo for our initial experiment where we used GPT-Neo for generating answers based on the user's questions. Here are some results we yield with GPT-neo.

More on GPT-Neo implementation will be discussed later in project implementation chapter.

**RAG:**

We used RAG for passage retrieval. In this case, we trained our model using a dataset having 115000 rows. We used the haystack API to train and generate answers. The RAG could easily identify the passage using similarity functions. So, we finalize the RAG model for our system. More on GPT-Neo implementation will be discussed later in project implementation chapter [6].

## 4.2 Usability

For any project that is considered to be beneficial for the general public, usability becomes a great concern. SciRet is made to be a useful tool for every user, rather than just being just a retrieval model. The starting from the dataset that is going into the system to the answers which are retrieved and shown to the user, is well thought out and fine-tuned. Firstly, to make the system retrieve more useful data overlooking the less informative ones, we fine-tuned our dataset and made a custom data frame. Among the 1 million literatures available in CORD-19 dataset, we only kept the ones related to covid-19 so that the accuracy and the usability of the output improves. On top of that, as our system will fetch data from multiple scientific researches for a singular question, the users will be relieved from facing the complexity of the process of reading and understanding research journals on their own. This not only saves time, but also gives the user opportunity to work on other things, given that the data fetched with our system is authentic and easy to get. We also added the title of each scientific research retrieved, so that the user can cross check the authenticity and usability of the output for additional assurance. This will also help the students and professionals who are newly entering the research field to explore their retrieved data from more depth. In addition to all these, if anyone wants to build a better model using SciRet as their system, all that needs to be done will be providing a fine-tuned dataset to SciRet, as the rest is already prepared.

## 4.3 Manufacturability

SciRet is very easily manufacturable and scalable. We used Python as our programming language and our text data is stored in a CSV data frame, which makes SciRet compatible with every hardware and software available. The system is already deployed in a user-friendly User Interface made of Flux. In addition to that, as our system is already providing good output, it's ready to use in Chatbots, QA systems based on NLP, Information generating websites and applications, Virtual Governmental information desks and lots. As quick and bona fide information is a necessity in every field, we made SciRet in an easy and quickly manufacturable way, so that it can be of service to the mass population. Furthermore, as Natural Language Processing is a relatively new field and almost every system with Artificial Intelligence needs a retriever to be able to fetch and execute, SciRet can be a convenient and useful model for them to use.

## 4.4 Sustainability

When it comes to technology and other scientific inventions, sustainability is always a matter to examine further and we believe SciRet is a sustainable invention. For a system built with Artificial Intelligence, one major concern is if it will be harmful to society by taking over jobs and even lives. To address this concern, United Nations - UN Development announced the UN's Sustainable Development Goals. Upon fulfilling these goals, a system with Artificial Intelligence is accepted as a sustainable development and can become a valuable tool worldwide. [23]

We believe that SciRet is eligible to fulfill each goal as its made for the sake of the general public's necessity. SciRet is able to retrieve data from a large chunk of dataset and provide valuable answers, which for a human would be very time consuming; for a machine would be very expensive and it would not be an environmentally friendly decision. We already saw the misery of people in the covid pandemic caused by lack of authenticity and scarcity of information. If sooner or later the world faces another pandemic, SciRet will be ready to provide help to the mass population. One major bottleneck in Artificial Intelligence field is a chronic shortage of talent able to improve AI capabilities. To address this problem, SciRet is made as a self-sufficient system which does not require additional monitoring once it has got a data frame to work with. And it will also be a risk-free system. Because there is no scope of manipulation which can harm the society, as SciRet only retrieves data from authentic scientific literature published by well-known publishers upon cross examination.

# 5 Project Implementation

In this section we will discuss our overall journey throughout the project implementation.

## 5.1 System Architecture

### 5.1.1 Implementation of GPT-Neo:

At the very beginning we trained GPT-neo for 100 epochs. The dataset we fed to GPT-Neo while training was made from only 1000 rows out of 315000 rows of data from the main dataset. Even with this small amount of data it took 27 hours to fine tune the pre-trained model. Here is the block diagram of how we trained GPT-Neo. [24]

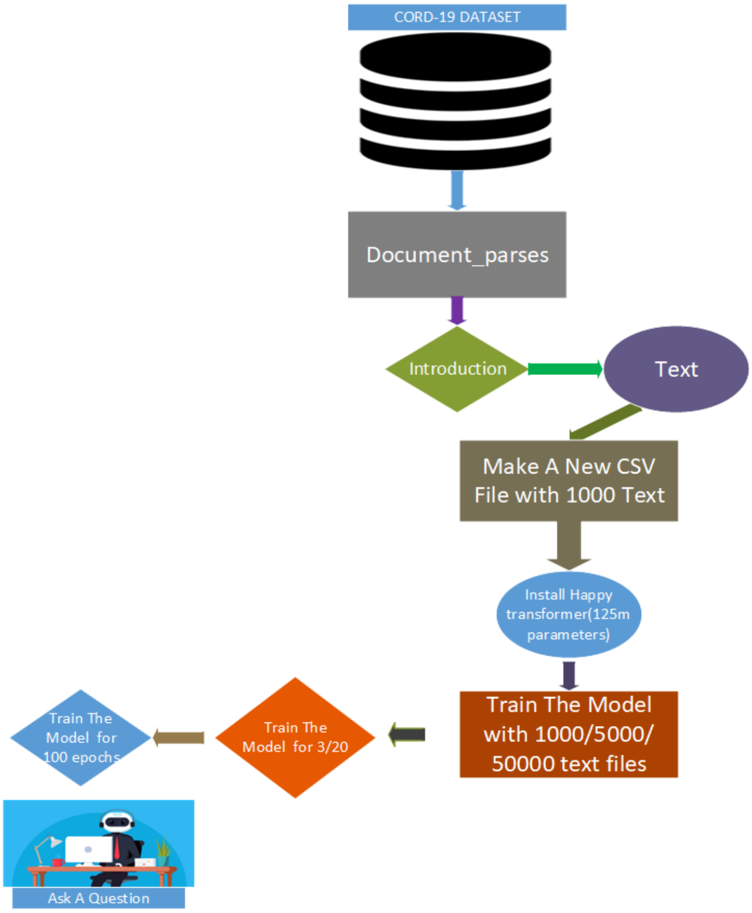
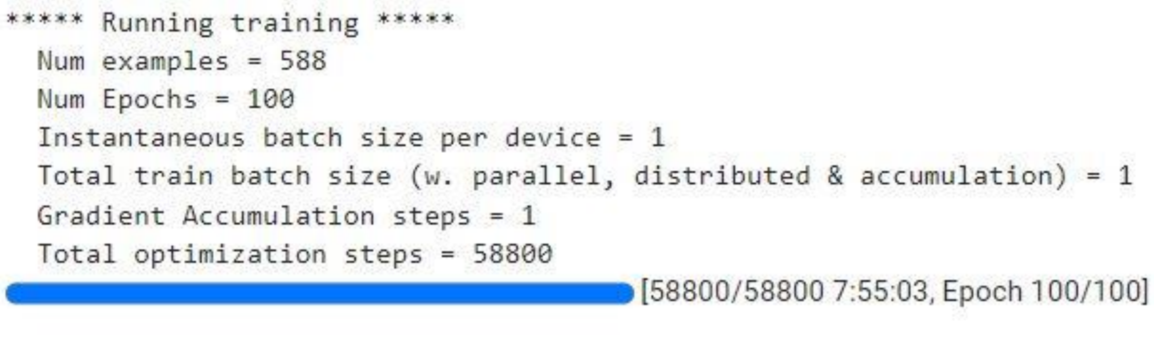

CORD-19 DATASET
Document_parses
Introduction
Text
Make A New CSV File with 1000 Text
Install Happy transformer(125m parameters)
Train The Model with 1000/5000/ 50000 text files
Train The Model for 3/20
Train The Model for 100 epochs
Ask A Question

*Figure 11 System Diagram of GPT- Neo Implementation*

### 5.1.2 Training GPT-Neo:

Here are the training parameters of GPT Neo



```
***** Running training *****
  Num examples = 588
  Num Epochs = 100
  Instantaneous batch size per device = 1
  Total train batch size (w. parallel, distributed & accumulation) = 1
  Gradient Accumulation steps = 1
  Total optimization steps = 58800
[58800/58800 7:55:03, Epoch 100/100]
```

*Figure 12 GPT Neo Training Args.*

### 5.1.3 Implementation of RAG:

In our RAG implementation, the whole system consists of a dataset, encoders, a document store, retriever model, rank generator, Pre-trained QA model.

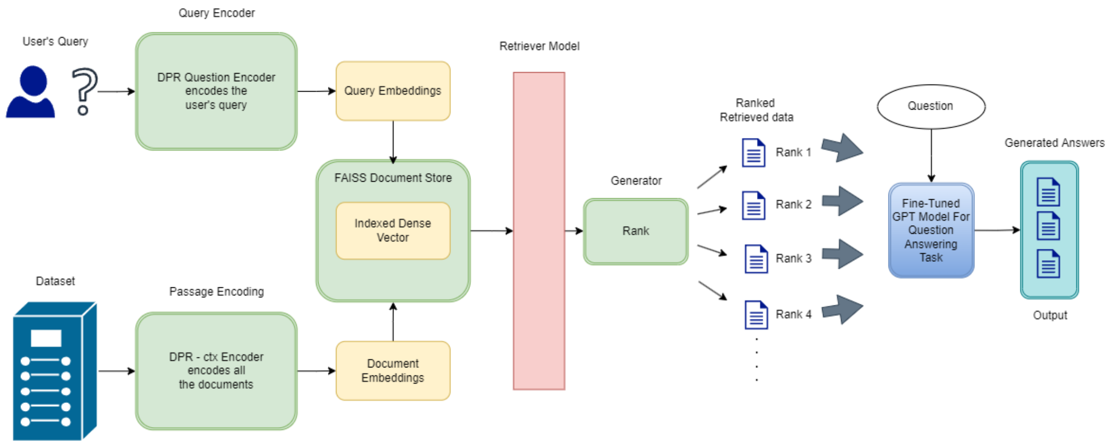


*Figure 13 System Architecture*

### 5.1.4 Encoder

We utilized DPRQuestionEncoder and DPRContextEncoder for the encoder portions [17]. The Tokenizer conducts end-to-end tokenization with punctuation splitting and the workpiece and is identical to the BertTokenizer. With the question encoder, we created embeddings for the user's query and stored them in our document store. With the Context encoder, we created embeddings for our dataset and stored them in our document store.

### 5.1.5 Document Store

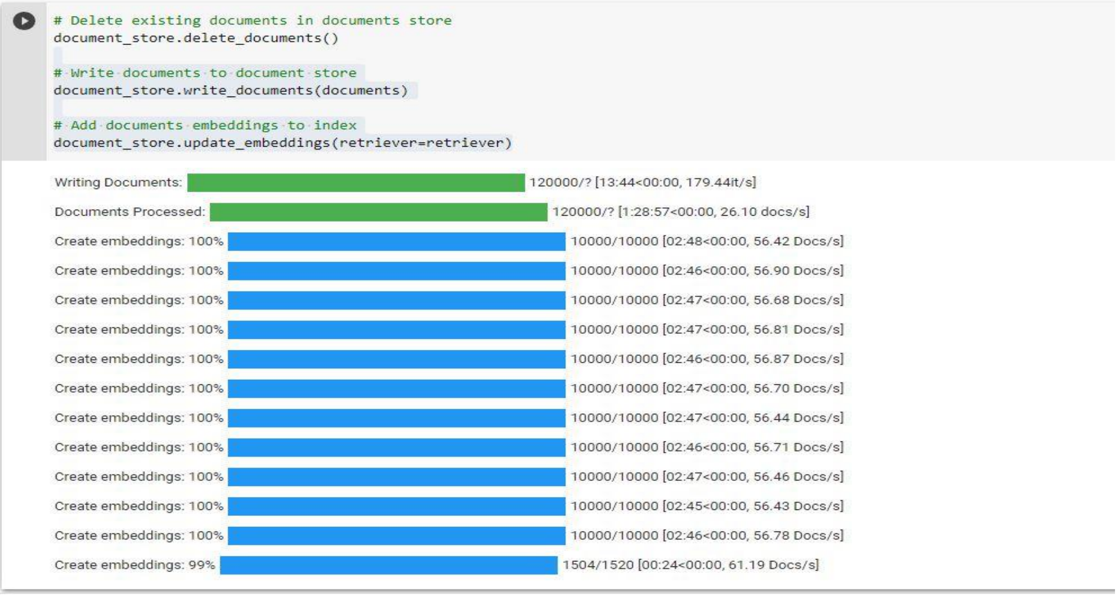
We required a quick and optimized document store which could contain our embeddings of the whole dataset along with an index. We used “FaissIndexFactory” [21] for our document storage purposes. More on the Faissindex store has been discussed in Chapter 9(Tools and technology used.

### 5.1.6 Retriever

We used the RAG retriever for retrieving our passage based on the user's query. The model uses a similarity function to rank the passages that match the users' query embeddings. Finally, using the haystack document search pipeline, we can generate and retrieve documents based on that rank.

### 5.1.7 Training RAG:

Here are the training steps of RAG



```
# Delete existing documents in documents store
document_store.delete_documents()

# Write documents to document store
document_store.write_documents(documents)

# Add documents embeddings to index
document_store.update_embeddings(retriever=retriever)
```

```
Writing Documents: 120000/? [13:44<00:00, 179.44it/s]
Documents Processed: 120000/? [1:28:57<00:00, 26.10 docs/s]
Create embeddings: 100% 10000/10000 [02:48<00:00, 56.42 Docs/s]
Create embeddings: 100% 10000/10000 [02:46<00:00, 56.90 Docs/s]
Create embeddings: 100% 10000/10000 [02:47<00:00, 56.68 Docs/s]
Create embeddings: 100% 10000/10000 [02:47<00:00, 56.81 Docs/s]
Create embeddings: 100% 10000/10000 [02:46<00:00, 56.87 Docs/s]
Create embeddings: 100% 10000/10000 [02:47<00:00, 56.70 Docs/s]
Create embeddings: 100% 10000/10000 [02:47<00:00, 56.44 Docs/s]
Create embeddings: 100% 10000/10000 [02:46<00:00, 56.71 Docs/s]
Create embeddings: 100% 10000/10000 [02:47<00:00, 56.46 Docs/s]
Create embeddings: 100% 10000/10000 [02:45<00:00, 56.43 Docs/s]
Create embeddings: 100% 10000/10000 [02:46<00:00, 56.78 Docs/s]
Create embeddings: 99% 1504/1520 [00:24<00:00, 61.19 Docs/s]
```

*Figure 14 RAG Training*

## 5.2 Models Implementation:

Here we tried to show all the approaches we tried to train and analyze our models through different approaches from the begging to end.

| Model Name | Number of Epoch | Number of Documents | Comments | Data set |
|---|---|---|---|---|
| GPT-Neo | 20 | 1000 Text | Generate the Answer | (CORD-19) |
| GPT-Neo | 100 | 1000 Text | Generate the Answer | (CORD-19) |
| RAG | Fine Tune | 1000 Text | Generate the Answer | (CORD-19) |
| RAG | Fine Tune | 5000 Text | Generate the Answer and try to catch the entire passage | (CORD-19) |
| RAG | Fine Tune | 50000 Text | Generate the Answer and able to catch the right passage | (CORD-19) |

*Table 1 : Models and Arguments*

# 6 Project Impact

This section illustrates the effects of our research across several industries and examines how these effects might reduce human effort, how it can be useful to the general people, how it is beneficial to health and so on.

## 6.1 Economic Impacts

In addition to money, funding field research requires a number of other problems, such as finding staff, traveling, and planning the project over an extended period of time, depending on the subject. However, machine learning has recently brought about a huge shift in practically every industry. Although it doesn't take much human work, using ML to solve real-world problems requires data particular to that situation. Data is a challenge in classic machine learning problems because open-source data that is specific to a problem is limited.

Since our data is publicly available, we didn’t have to do anything else to collect the data. But, to preprocess the data, we had to use some strong resources, which is not much if we think about the whole situation. Apart from that, we used all the models that were already trained. So, we didn’t have to train any models from scratch. Training a model from scratch is very time-consuming, especially when the model is much bigger than regular ML models. It takes a lot of power, time, and manpower to maintain or look after it. We only fine-tuned the trained models that are publicly available for research purposes, so we saved a huge amount of money. This process requires minimal computational power, so we didn't have to use additional computers or resources. Another important thing is that we used the cloud for all the tasks, which also saves a lot of time.

## 6.2 Social Impacts

Our system has huge social impacts. When we started to build this project, we were thinking about how to make a system that would be beneficial to the general public. We made this system to give the general population proper information. People in society generally lack access to proper information due to a lack of availability. Another issue is that they do not know the importance of scientific information. So, they mostly collect information that is not scientifically correct. As a result, people are enslaved by superstition.

Our system will provide authentic scientific information, which will help people to create a superstition-free society. Our system will benefit many people in society because our goal is to make scientific information available to the general public.

## 6.3 Political Impacts

We are creating a system to make people's lives much easier than ever through providing proper information at their fingertips. We think it won't affect any political people or organizations. Because we do not intend to harm any political organization, our system has no political implications.

## 6.4 Health Impacts

Our project has a great health impact. Our project will help researchers in the medical field to extract the proper information from scientific papers. New researchers from medical science can use our system to get authentic information quickly, which will save a lot of time for new innovations. Another important thing is that, using this system, people will get necessary information so quickly that they won't have to sit for hours in front of a computer to study thousands of scientific papers to get their desired information.

When we started this project, the world had just recovered from the COVID-19 pandemic. People from all sectors still had a lack of knowledge regarding this situation. That’s the reason we used the COVID-19 dataset as a prime example to showcase our project usability to general people and make them understand the importance of knowing things scientifically. We think this will help them greatly in maintaining a healthy life by knowing the scientific information in an easier way. This will reduce their tension about not having proper knowledge. They don't even have to learn by heart all the information because it will be publicly available for free.

# 7 Environment considerations & Sustainability

Technological advancement always comes with a price, and often the price to pay is degradation of the environment. Advanced technology needs large computational resources which causes large energy consumption. So, these become costly to train and to develop, both financially and environmentally, due to the cost of hardware and carbon footprints [25]. As many of the NLP models are still trained and tested on heavy-duty GPUs and powerful CPUs, the environmental cost to pay is often higher than realized while building a model. To address this concern, we used a new route, which could potentially eliminate the environmental harm and will make our model more sustainable.

Firstly, we choose a pre-trained model over an untrained model to work with. This helped us to cut down the mammoth computational power needed to build, train and test an NLP model. It's a proven fact that re-training a model needs more computational power and hardware support, which makes the process of model building costly and unsustainable. We used a pre-trained RAG model from Haystack platform, which helped us to cut down the re-training cost and also gave us the opportunity to use less GPU and CPU power and helped us to remove potential carbon footprint from the environment [26] . This might not be a great deal for our project as SciRet is still in its initial stages. But our thought is to make SciRet a household name and to serve as many people as possible. If we went with using a model which needed re-training, it would not be sustainable in the long run.

Our second step taken for making SciRet more environment friendly and sustainable was going with a cloud-based environment to develop our project rather than using powerful hardwires available. Even though cloud computing has been proven to be harmful for the environment to some extent, researchers around the world have come up with Green

Cloud Computing, which is able to serve the developers with their computational needs while protecting the environment [27]. Recently Google has also converted all its cloud computing-based services into Green Cloud Computing, by which they are able to reduce the impact on environment and lessen the amount of carbon footprint. Hence, we decided to use Google Colab as our virtual environment and used the GPU and RAM provided on the platform. This step has made SciRet more cost effective and sustainable, while protecting the climate. And as developers we were able to use the necessary computational power without causing unnecessary carbon emission and energy consumption.

# 8 Ethical & Professional Responsibility

## 8.1 Ethical Responsibility

Our primary priority throughout the process has been maintaining our moral integrity. When creating SciRet, we resisted using any unethical tactics. We were truthful in our explanations of the dataset, methodologies, and procedures and did not fake or manipulate any data in order to ultimately benefit science and the public. We also kept ourselves accessible to criticism by being very transparent about our information, concepts, methods, resources, and findings.

We ensured that the public only receives true information by using SciRet. We used a dataset made up of research articles that were published by reputable publishers for model development and testing in order to guarantee the accuracy of the data that is made available to the public. We could also guarantee that the material presented to the audience is not something made up out of thin air because we gathered all the information from scientific literature, which has been thoroughly reviewed and verified.

By fulfilling our promises to the public and being sincere during the dataset modification and model development processes, we maintained our integrity. We concentrated on giving the populace accurate and practical information. We anticipate that it will lessen social harms and reduce disputes between general public members over the veracity of information.

We have to remain firm on our moral principles when working with the works of outstanding and creative authors because we are dealing with scientific research articles. To protect your anonymity, we only used literature that was freely available.

Obtaining consent from every author of the more than a million research papers in our dataset is not practicable. Therefore, we used CORD-19-organized publicly available data. We respect the intellectual property (copyrights and patents) of other scientists. As a result, we avoided using previously unpublished data, techniques, or results without authorization and always gave full acknowledgment.

We took great effort to avoid mistakes and carelessness since we knew that the information produced by our system would end up in the hands of people who couldn't tell the difference between incorrect and correct information.

## 8.2 Professional Responsibility

We carried out our work on SciRet while abiding by our moral and legal obligations. We made use of our experience to help the general public without disseminating false information onto them. We did not reveal any author or company's private information to the general public. We merely used the information that was openly accessible.

Despite the fact that many opportunities may be created with a system like this, we did not view SciRet as a business prospect. We did not prioritize our own requirements over those of the project and the general public. To avoid conflicts of interest, we created and carefully thought out SciRet. Every time we saw a potential breach in this plan, we devised a new method of execution. We gave the project's methods and findings to other researchers who were working to assist the broader population.

Every researcher whose assistance paved the way for a successful endeavor has received the due credit. All the possible outcomes of the project and the points where we failed to execute are penned in this paper.

# 9 Tools & Technology Used

## 9.1 Software Requirements

**Python**: Python is proven to be a great language for data analysis. Its extensive library [Pandas, Numpy, Pytorch] access made our pre-processing work sufficient and quick. Its simplicity and consistency made our work process less tiring and time consuming. Python is also platform independent. As a result, SciRet will get the versatility offered by Python. Python also has one of the largest communities worldwide. So whenever, we faced any obstacles, we could get quick and effective solutions. Even though R, Scala, Julia and Java are also used for machine learning projects, we choose Python for the added benefits. [28]

**Haystack:** We used Haystack API to train RAG mode. We deployed it as a REST API and it helped us by providing us the QA functionalities. Haystack can automate the extraction of pertinent data from a group of documents that are on the same subjects but concern several entities. It can apply a set of standard questions to each document in a store & return a NO_ANSWER if a given document does not contain the answer to a question.

**FaissIndex**: Faiss library is used in our project for the similarity search function. Faiss is a system developed by Facebook AI where we can search for an index's most similar vectors given a set of vectors. Faiss offers a number of similarity search techniques that cover a wide range of use trade-offs, the speed and memory utilization of Faiss have been optimized. And the best part is, for the most important indexing techniques, Faiss provides a cutting-edge GPU implementation.

## 9.2 Hardware Requirements

**GPU:** We had to use a Graphics Processing Unit as we were working with a dataset consisting of huge number of documents. A GPU is built from the ground up to almost exclusively render high-resolution graphics and images, a task that doesn't involve a lot of contexts switching. Instead, GPUs emphasize concurrency, which is the division of large jobs into smaller ones that can be executed concurrently. We used GPU [NVIDIA K80, P100, P4, T4, V100] provided by Google Colab at <colab.research.google.com>.

**RAM:** We used 32 GB ram provided by Google Colab Pro. When cloud computing is not an option for processing speed, 16GB of RAM or more is helpful because waiting for your algorithm to finish is a genuine nuisance. Meanwhile, in-memory data science job gets much easier if a developer has 32GB or RAM, especially if a big data platform is unavailable or inaccessible. Considering all the parameters and the long-term plan, we choose to work with 32 GB RAM.

**Google Colab:** After thinking of environmental considerations and sustainability, we wanted to use a Green Cloud Computing based virtual platform for building our project. Google recently made all of its cloud-based services green to lessen carbon footprint and the platforms are now able to provide more computational power to the end user. We decided to work on Google Colab. We initially failed to train our model with standard google colab, so we switched to Google Colab pro. This offered us more computational power and here we trained our model for 9 hours and 45 minutes. Google Colab offers pre-installed libraries, free GPU and TPU use, and easy collaboration of the developers

**Project Management Tools**

Github: We used Github as project management tool. Github offered us easy collaboration and made the management process of our project very easy and less time-consuming.

Link to our Github repository: https://github.com/Anaskaysar/SciRet-Scientific-Information-Made-Easy

| | |
|---|---|
| Language | Python, Html, CSS, Bootstrap |
| API | Haystack, Rest API |
| Framework | Flux |
| Library | Pandas, Numpy, Pytorch, Faiss Index |
| Hardware Requirement | GPU: NVIDIA P100, P4, T4, V100<br>RAM: 32GB |
| Virtual Platform | Google Colab Pro, Jupyter, Github |

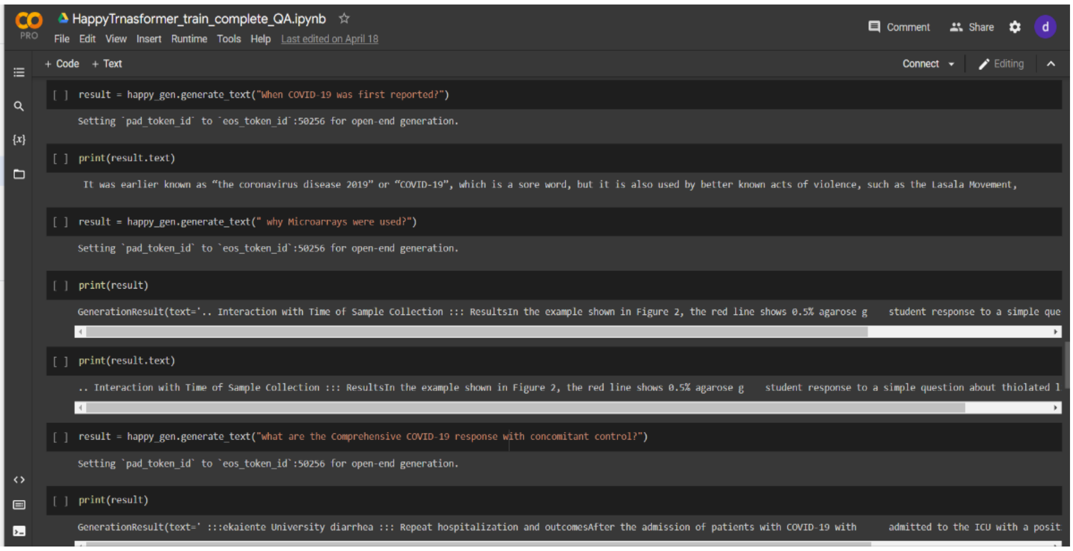
*Table 2 All Tech and Tools At A Glance*

# 10 Result Analysis

This chapter describes all the results we have gotten during our whole development process. As during the project development stage, we changed our approaches many times to get the desired results, we will be showing all the outcomes here.

## 10.1 GPT Neo Results

During 499A, we started our project implementation by training "GPT-Neo: A Fine-Tuned Transformer Model" for our question answering system. Though it could generate some good results related to the questions we asked, it could not provide the exact answers we were expecting. Here are some examples of answers generated by GPT-Neo.



HappyTrnasformer_train_complete_QA.ipynb
PRO
File Edit View Insert Runtime Tools Help Last edited on April 18
Comment Share
+ Code + Text
Connect
Editing

```
[ ] result = happy_gen.generate_text("When COVID-19 was first reported?")

    Setting `pad_token_id` to `eos_token_id`:50256 for open-end generation.

[ ] print(result.text)

     It was earlier known as “the coronavirus disease 2019” or “COVID-19”, which is a sore word, but it is also used by better known acts of violence, such as the Lasala Movement,

[ ] result = happy_gen.generate_text(" why Microarrays were used?")

    Setting `pad_token_id` to `eos_token_id`:50256 for open-end generation.

[ ] print(result)

    GenerationResult(text='.. Interaction with Time of Sample Collection ::: ResultsIn the example shown in Figure 2, the red line shows 0.5% agarose g    student response to a simple que

[ ] print(result.text)

    .. Interaction with Time of Sample Collection ::: ResultsIn the example shown in Figure 2, the red line shows 0.5% agarose g    student response to a simple question about thiolated l

[ ] result = happy_gen.generate_text("what are the Comprehensive COVID-19 response with concomitant control?")

    Setting `pad_token_id` to `eos_token_id`:50256 for open-end generation.

[ ] print(result)

    GenerationResult(text=' :::ekaiente University diarrhea ::: Repeat hospitalization and outcomesAfter the admission of patients with COVID-19 with     admitted to the ICU with a posit
```

*Figure 15 Results Of GPT Neo*

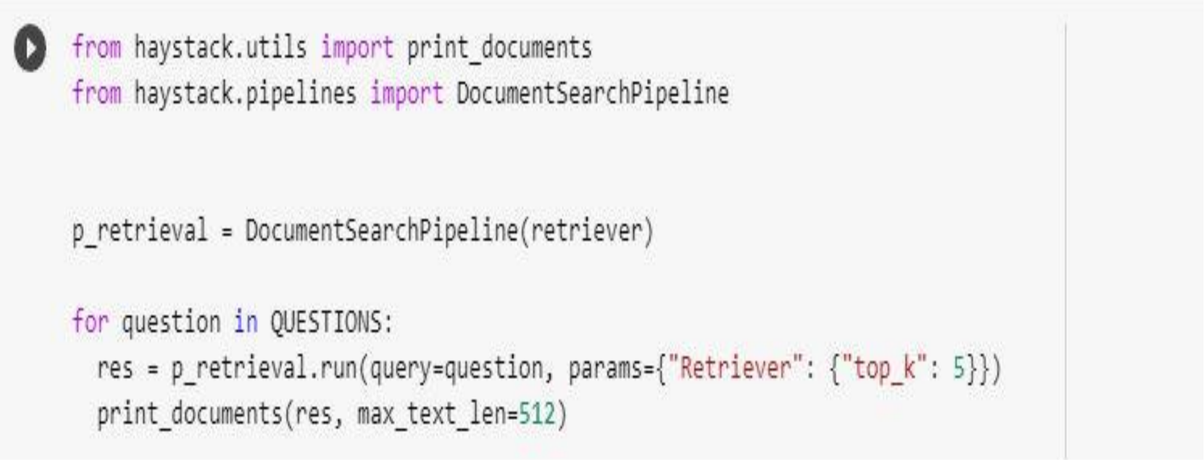
As our system wasn’t providing answers according to our desires, we tried different approaches. Rather than generating direct answers, we started looking for the passages from where our model was generating answers. To achieve this goal, we started working with RAG (Retriever Augmented Generated-Model) [6].

## 10.2 RAG Results

Our entire system was created to retrieve a passage for any question and also to provide a solid response. So that, we can find a passage to answer any query. In essence, the system gives us the top 5 or 10 passages that are relevant to the question, and the passage system also includes a confidence score. Whichever has the greatest confidence score will be placed first, and each passage will return to the system based on the query. Thus, how close is the query to the generated output?

We used haystack's document search pipeline to find out the passages based on user's query. It uses Similarity search [21] to match the document vector with query vectors stored in the document store. After finding similarity between query embeddings and passage embeddings our system provide ranks the passages based on confidence score.

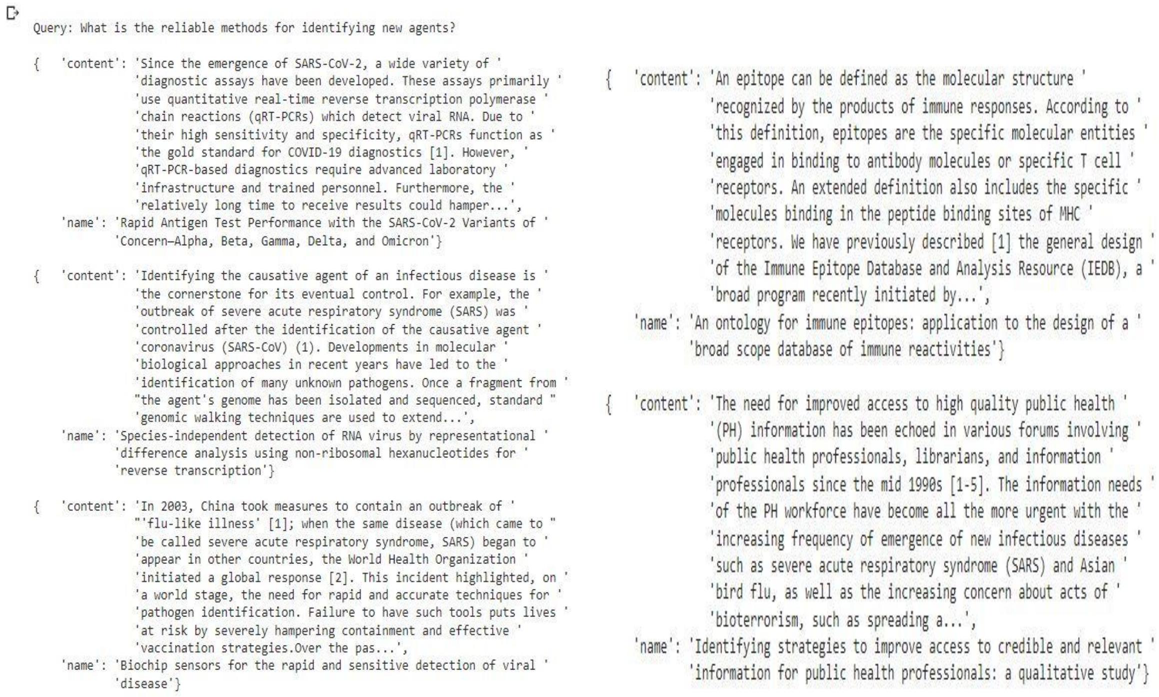
Here is the code for retrieving passages. We are passing topK=5 to our retriever model RAG. This function will provide us 5 best ranked passages which has similarity between user's query and existing passages.



```
from haystack.utils import print_documents
from haystack.pipelines import DocumentSearchPipeline


p_retrieval = DocumentSearchPipeline(retriever)

for question in QUESTIONS:
  res = p_retrieval.run(query=question, params={"Retriever": {"top_k": 5}})
  print_documents(res, max_text_len=512)
```

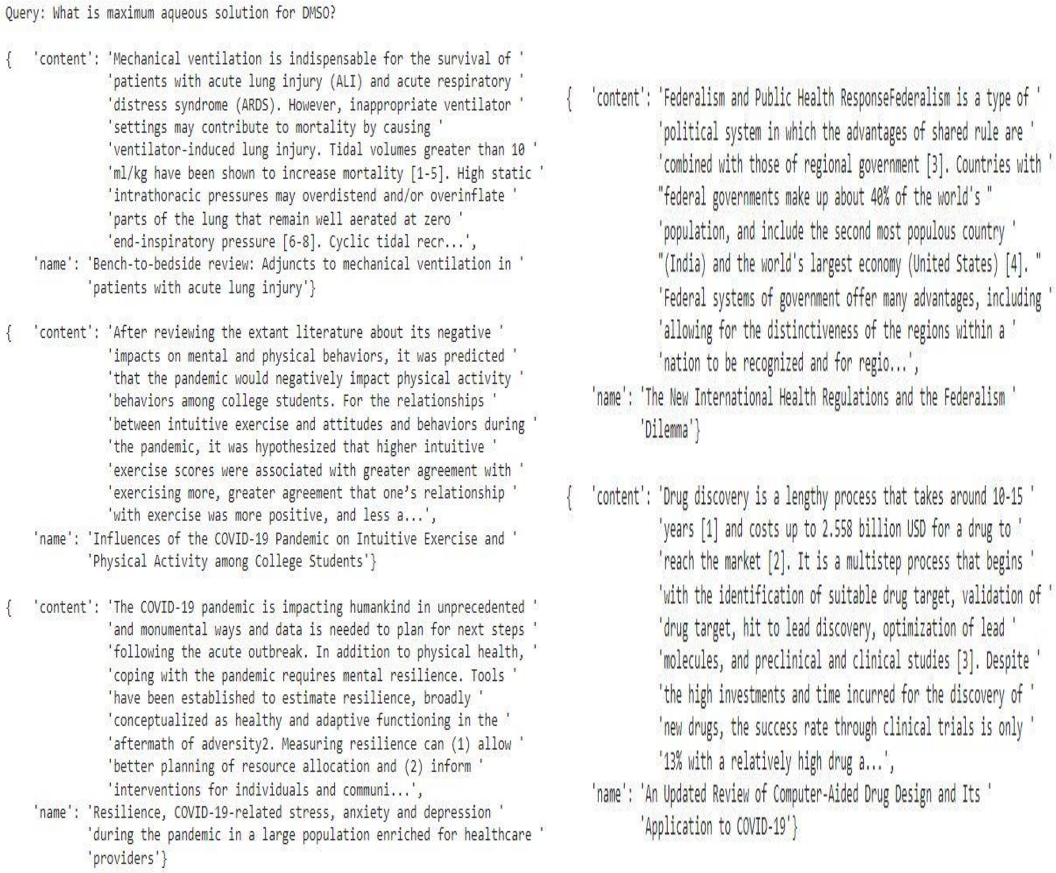
*Figure 16 Document Search Pipeline*

Here are some retrieved results…



```
Query: What is the reliable methods for identifying new agents?

{   'content': 'Since the emergence of SARS-CoV-2, a wide variety of '
               'diagnostic assays have been developed. These assays primarily '
               'use quantitative real-time reverse transcription polymerase '
               'chain reactions (qRT-PCRs) which detect viral RNA. Due to '
               'their high sensitivity and specificity, qRT-PCRs function as '
               'the gold standard for COVID-19 diagnostics [1]. However, '
               'qRT-PCR-based diagnostics require advanced laboratory '
               'infrastructure and trained personnel. Furthermore, the '
               'relatively long time to receive results could hamper...',
    'name': 'Rapid Antigen Test Performance with the SARS-CoV-2 Variants of '
            'Concern—Alpha, Beta, Gamma, Delta, and Omicron'}

{   'content': 'Identifying the causative agent of an infectious disease is '
               'the cornerstone for its eventual control. For example, the '
               'outbreak of severe acute respiratory syndrome (SARS) was '
               'controlled after the identification of the causative agent '
               'coronavirus (SARS-CoV) (1). Developments in molecular '
               'biological approaches in recent years have led to the '
               'identification of many unknown pathogens. Once a fragment from '
               "the agent's genome has been isolated and sequenced, standard "
               'genomic walking techniques are used to extend...',
    'name': 'Species-independent detection of RNA virus by representational '
            'difference analysis using non-ribosomal hexanucleotides for '
            'reverse transcription'}

{   'content': 'In 2003, China took measures to contain an outbreak of '
               "'flu-like illness' [1]; when the same disease (which came to "
               'be called severe acute respiratory syndrome, SARS) began to '
               'appear in other countries, the World Health Organization '
               'initiated a global response [2]. This incident highlighted, on '
               'a world stage, the need for rapid and accurate techniques for '
               'pathogen identification. Failure to have such tools puts lives '
               'at risk by severely hampering containment and effective '
               'vaccination strategies.Over the pas...',
    'name': 'Biochip sensors for the rapid and sensitive detection of viral '
            'disease'}

{   'content': 'An epitope can be defined as the molecular structure '
               'recognized by the products of immune responses. According to '
               'this definition, epitopes are the specific molecular entities '
               'engaged in binding to antibody molecules or specific T cell '
               'receptors. An extended definition also includes the specific '
               'molecules binding in the peptide binding sites of MHC '
               'receptors. We have previously described [1] the general design '
               'of the Immune Epitope Database and Analysis Resource (IEDB), a '
               'broad program recently initiated by...',
    'name': 'An ontology for immune epitopes: application to the design of a '
            'broad scope database of immune reactivities'}

{   'content': 'The need for improved access to high quality public health '
               '(PH) information has been echoed in various forums involving '
               'public health professionals, librarians, and information '
               'professionals since the mid 1990s [1-5]. The information needs '
               'of the PH workforce have become all the more urgent with the '
               'increasing frequency of emergence of new infectious diseases '
               'such as severe acute respiratory syndrome (SARS) and Asian '
               'bird flu, as well as the increasing concern about acts of '
               'bioterrorism, such as spreading a...',
    'name': 'Identifying strategies to improve access to credible and relevant '
            'information for public health professionals: a qualitative study'}
```

*Figure 17 Retrieved Top 5 Passages Using Document Search Pipeline -Sample 1*



```
Query: What is maximum aqueous solution for DMSO?

{   'content': 'Mechanical ventilation is indispensable for the survival of '
               'patients with acute lung injury (ALI) and acute respiratory '
               'distress syndrome (ARDS). However, inappropriate ventilator '
               'settings may contribute to mortality by causing '
               'ventilator-induced lung injury. Tidal volumes greater than 10 '
               'ml/kg have been shown to increase mortality [1-5]. High static '
               'intrathoracic pressures may overdistend and/or overinflate '
               'parts of the lung that remain well aerated at zero '
               'end-inspiratory pressure [6-8]. Cyclic tidal recr...',
    'name': 'Bench-to-bedside review: Adjuncts to mechanical ventilation in '
            'patients with acute lung injury'}

{   'content': 'After reviewing the extant literature about its negative '
               'impacts on mental and physical behaviors, it was predicted '
               'that the pandemic would negatively impact physical activity '
               'behaviors among college students. For the relationships '
               'between intuitive exercise and attitudes and behaviors during '
               'the pandemic, it was hypothesized that higher intuitive '
               'exercise scores were associated with greater agreement with '
               'exercising more, greater agreement that one’s relationship '
               'with exercise was more positive, and less a...',
    'name': 'Influences of the COVID-19 Pandemic on Intuitive Exercise and '
            'Physical Activity among College Students'}

{   'content': 'The COVID-19 pandemic is impacting humankind in unprecedented '
               'and monumental ways and data is needed to plan for next steps '
               'following the acute outbreak. In addition to physical health, '
               'coping with the pandemic requires mental resilience. Tools '
               'have been established to estimate resilience, broadly '
               'conceptualized as healthy and adaptive functioning in the '
               'aftermath of adversity2. Measuring resilience can (1) allow '
               'better planning of resource allocation and (2) inform '
               'interventions for individuals and communi...',
    'name': 'Resilience, COVID-19-related stress, anxiety and depression '
            'during the pandemic in a large population enriched for healthcare '
            'providers'}

{   'content': 'Federalism and Public Health ResponseFederalism is a type of '
               'political system in which the advantages of shared rule are '
               'combined with those of regional government [3]. Countries with '
               "federal governments make up about 40% of the world's "
               'population, and include the second most populous country '
               "(India) and the world's largest economy (United States) [4]. "
               'Federal systems of government offer many advantages, including '
               'allowing for the distinctiveness of the regions within a '
               'nation to be recognized and for regio...',
    'name': 'The New International Health Regulations and the Federalism '
            'Dilemma'}

{   'content': 'Drug discovery is a lengthy process that takes around 10-15 '
               'years [1] and costs up to 2.558 billion USD for a drug to '
               'reach the market [2]. It is a multistep process that begins '
               'with the identification of suitable drug target, validation of '
               'drug target, hit to lead discovery, optimization of lead '
               'molecules, and preclinical and clinical studies [3]. Despite '
               'the high investments and time incurred for the discovery of '
               'new drugs, the success rate through clinical trials is only '
               '13% with a relatively high drug a...',
    'name': 'An Updated Review of Computer-Aided Drug Design and Its '
            'Application to COVID-19'}
```

*Figure 18  Retrieved Top 5 Passages Using Document Search Pipeline -Sample 2*

# 11 Conclusions

SciRet: Scientific Information Made Easy for General Public can perform a crucial part for a modern world because it can retrieve a passage very successfully. However, it can be applied for any data set and you can retrieve a passage by given your desire question. In This way, our modern world can take benefited from this kind modern work. This study was conducted as a part of senior design project by the ECE department of North South University. During the experimentation and analysis, we came across some findings and outcomes which might be beneficial to future research on this topic.

## 11.1 Our Findings

Before getting into the actual implementation, the dataset was obtained from the open-source Kaggle. The Dataset name is COVID-19 Open Research Dataset Challenge (CORD-19). The dataset contained various types of files and the document parses file contained the PMC Jason file, the size of the document parses file was around 2 lakh 61 thousand but only 1000 Jason files was taken. Only the introduction and metadata feature were extracted from the document parses the file. After that metadata was dropped and created a new CSV file with introduction text .and converted the CSV file into a text file. The hugging page was used to install the happy transformer because GPT new was used to generate the Human-like question answering system. The model used to train the dataset was GPT Neo-125 M. Questions were created from the PMC Jason file and asked the model to answer those questions, before that a Covid 19 dataset was used to ask the question, where the question was already given. The model was trained multiple times like 3,10,50 and 100 epochs. We observed that the output result was superior to the previous training result after 100 iterations. Then, we switch to the RAG model. For the RAG model, we applied the same methodology and data set. Additionally, we discovered that our results are now superior to all previous results after training with varied sized data sets. We were able to forecast exact passages with greater accuracy because of the RAG model, but our question-and-answer system is still insufficient.

## 11.2 Future Works

A Natural Language Question Answering system doesn't give the user a finished document like conventional keyword searches do. Users instead pose a natural language enquiry and get a precise response in reply.[2] We are still having issues with our question-answering system. So, in the future, we will train our rag model output in GPT-NEW or GPT -3. We hope to be able to resolve the issue with the question-answer system. Because, it is a Generative QA model, it can be used in Chatbots, Question Answering Systems, Guideline systems, Information poles, Governmental Information sites, educational websites, and many other applications that generate revenue while also assisting the general public. This can be used as a QA system for any topic simply by changing the data frame and keeping the model intact. If another pandemic occurs in the near future, this model will be ready to use as soon as we have a new dataset of information. Our implementation has been designed to be generic enough to aid in other related research. Our implementation has been made public to encourage future research and can be found

# 12 References

[1] B. L. J. S. u. r. t. i. p. i. q. Katz, Proceedings of the {ACL} 2003 Workshop on Natural Language Processing in Biomedicine, Sapporo, Japan: Association for Computational Linguistics, 2003.

[2] W. X. R. G. L. Zhao*, "Passage Retrieval for Information Extraction".

[3] D. B. Abdelghani BOUZIANEa, " “Question Answering Systems: Survey and Trends,” .," *ScienceDirect, p. 375, ,* 2015.

[4] “. J. Brownlee, A Gentle Introduction to Transfer Learning for Deep Learning,, [Online]. Available: [Online]. Available: https://machinelearningmastery.com/transfer-learning-for-deep-learning/?fbclid=IwAR3zLbhN4rF3k2WSf7BGTZWc0YSQ-RHlCLJ9tyxd8MhWxlTNoQfQwT8R75E.

[5] M. RUSTAGI, "“A Beginner’s Guide to GPT Neo (With Python Codes,”," [Online]. Available: https://analyticsindiamag.com/a-beginners-guide-to-gpt-neo-with-python-codes/?fbclid=IwAR0DjLe0ROANi5KnoQ2rdPwaeHKMBVJB8VbwlgTqRjVBWCbgrluvNjO9Xqw.

[6] facebook, "RAG," [Online]. Available: https://huggingface.co/docs/transformers/model_doc/rag.

[7] V. N. I. M. R. C. X. S. G. P. N. K. R. B. P. B. G. K. Colby Wise, COVID-19 Knowledge Graph: Accelerating Information Retrieval and Discovery for Scientific Literature, 24 Jul 2020.

[8] K. D. a. S. K. Emre, Knowledge and attitudes hospital pharnacusts about COVID-19, 2020.

[9] K. Emre, K. Demirkan and Serhat, "Knowledge and attitudes hospital pharmacists about COVID-19.," 2020.

[10] A. M. H. B. G. S. C. Nourchene Ouerhani, "Smart Ubiquitous Chatbot for COVID-19 Assistance with Deep learning Sentiment Analysis Model during and after quarantine," *Research Gate.*

[11] J. R. Quinlan, "Programs for Machine," *Morgan Kaufmann Publishers Inc,* San Francisco,1993).

[12] A. M. H. B. G. S. C. N. Ouerhani, Smart Ubiquitous Chatbot for COVID-19 Assistance with Deep Learning Sentiment Analysis Model During and After Quarantine, Research Gate.

[13] R. 2. A. J. P. D. A. B. P. Mahajan1, Healthcare Chatbot using Natural Language Processing, International Research Journal of Engineering and Technology (IRJET), 2020.

[14] S. R. a. H. Zaragoza, "The probabilistic relevance framework: BM25 and beyond. Foundations and Trends in Information Retrieval,," p. 333–389, 2009..

[15] A. F. J. W. a. A. B. [Danqi Chen, "Reading Wikipedia to answer opendomain questions. In Association for Computational Linguistics (ACL),," p. 1870–1879, 2017.

[16] P. N. X. M. R. N. a. B. X. higuo Wang, "Multi-passage BERT: A globally normalized bert model for open-domain question answering.," 2019.

[17] B. O. S. M. P. L. L. W. S. E. D. C. W.-t. Y. Vladimir Karpukhin, "Dense Passage Retrieval for Open-Domain Question Answering," 2020.

[18] F. Malik, "End To End Guide For NLP Project," 6 Jul 2019. [Online]. Available: https://medium.com/fintechexplained/end-to-end-guide-for-nlp-project-55e6765f63b5.

[19] C. M. G. N. &. T. W. H. An AI challenge with AI2, "COVID-19 Open Research Dataset Challenge (CORD-19)," [Online]. Available: https://www.kaggle.com/datasets/allen-institute-for-ai/CORD-19-research-challenge.

[20] S. Saha, "Principles of Good ML System Design," 19 Jul 2021. [Online]. Available: https://medium.com/@_sumitsaha_/principles-of-good-ml-system-design-cdf98c4b2035.

[21] E. S. A. G. A. McCallum, "Energy and Policy Considerations for Deep Learning in NLP," *University of Massachusetts Amherst,* 2019.

[22] C. F. G. H. EvaGarcía-Martín, "Estimation of energy consumption in machine learning," *Journal of Parallel and Distributed Computing,* August, 2019.

[23] M. C. V. j. Dr. V. Bindhu, "GREEN CLOUD COMPUTING SOLUTION FOR OPERATIONAL COST EFFICIENCY AND ENVIRONMENTAL IMPACT REDUCTION," *Journal of ISMAC,* 2019.

[24] "Stardome Observatory Planetarium," [Online]. Available: https://www.stardome.org.nz/wp-content/uploads/2016/10/WEB-PDFs_solar-system-chart_2016.pdf.

[25] M. RUSTAGI, "A Beginner’s Guide to GPT Neo (With Python Codes," [Online]. Available: https://analyticsindiamag.com/a-beginners-guide-to-gpt-neo-with-python-codes/?fbclid=IwAR0DjLe0ROANi5KnoQ2rdPwaeHKMBVJB8VbwlgTqRjVBWCbgrluvNjO9Xqw.

[26] [Online]. Available: https://arxiv.org/abs/2007.12731.

[27] R. J. 2. A. S. 2. P. P. 3. M. G. Jafar A Alzubi 1, "COBERT: COVID-19 Question Answering System Using BERT," pp. 1-11, 2021.

[28] R. W. A. J. P. D. A. B. Papiya Mahajan1, "Healthcare Chatbot using Natural Language Processing," *International Research Journal of Engineering and Technology (IRJET),* p. 6, 2020.