Configuration Manual

MSc Research Project
Data Analytics

Utkarsh Mathur
Student ID: x19232977

School of Computing
National College of Ireland

Supervisor: Aaloka Anant

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	Utkarsh Mathur
Student ID:	x19232977
Programme:	Data Analytics
Year:	2021
Module:	MSc Research Project
Supervisor:	Aaloka Anant
Submission Due Date:	31/01/2022
Project Title:	Configuration Manual
Word Count:	1040
Page Count:	10

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	Utkarsh Mathur
Date:	31st January 2022

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Configuration Manual

Utkarsh Mathur
x19232977

1 Introduction

This is the configuration manual for the Research Project implemented on the topic "A Content Based Recommender System for Medicine using Machine Learning Algorithm". Major guidelines and references have been take from (Dai et al.; 2018), where author explains the use of machine learning techniques in healthcare industries and its benefits. The research project's software and hardware are described in detail in this configuration manual. In addition, it outlines the libraries that are used and provides a brief description of the dataset in Section 3. This explain how to reproduce the work on any machine that meets all of the requirements, which are explained in detail in the following sections.

Note: The entire Project is replicated on github and the repository is publicly available at `https://github.com/UtkarshMathur-git/Drug_Recommender_System`

2 Environment Specifications

A system on which the Recommender system model project runs must meet a set of specifications for both software and hardware, which are explained in detail in the following subsections.

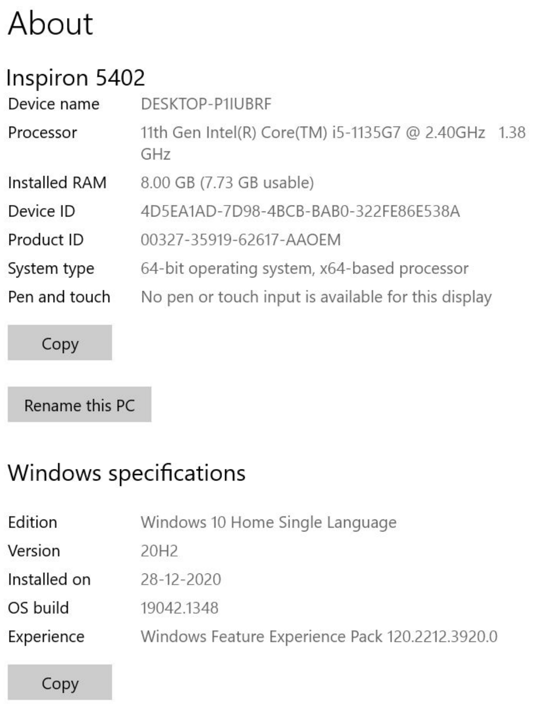
2.1 Hardware Specifications

Table 1 shows the hardware specifications of system which was used to run the recommender model smoothly.

Table 1: Hardware Details

HARDWARE	**CONFIGURATION**
System	Dell Inspiron5402
Operating System	Microsoft Windows 10 (64-bit OS)
Processor	Intel(R) Core i5
RAM	8GB, DD4, 3200MHz
Hard Disk	512GB M.2 PCIe NVMe Solid State Drive
Graphics Card	Intel Iris Xe Graphics

Figure 1 shows the details of computer's hardware detail used for making a Research project. These are the minimum hardware requirement to install different python package and run the machine learning algorithm.



About

Inspiron 5402

Device name	DESKTOP-P1IUBRF
Processor	11th Gen Intel(R) Core(TM) i5-1135G7 @ 2.40GHz 1.38 GHz
Installed RAM	8.00 GB (7.73 GB usable)
Device ID	4D5EA1AD-7D98-4BCB-BAB0-322FE86E538A
Product ID	00327-35919-62617-AAOEM
System type	64-bit operating system, x64-based processor
Pen and touch	No pen or touch input is available for this display

Windows specifications

Edition	Windows 10 Home Single Language
Version	20H2
Installed on	28-12-2020
OS build	19042.1348
Experience	Windows Feature Experience Pack 120.2212.3920.0

Figure 1: Screenshot of computer's hardware

2.2 Software Specifications

After hardware requirements are met now lets discuss about specific software requirement which is must for implementing the project. Table 2 shows the software requirement of the computer

Table 2: Software Details

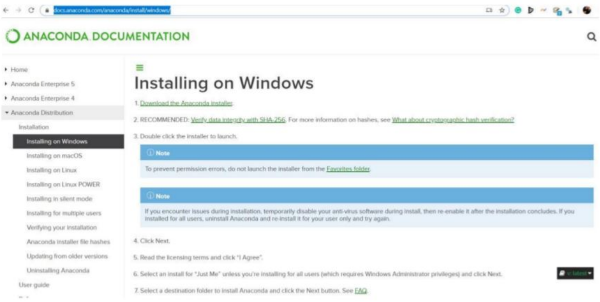
SOFTWARE	**CONFIGURATION**
Operating System	Microsoft Windows 10 (64-bit OS)
Python Notebook	Jupyter / Google Colab
IDE	Spyder
Coding Language	Python
Coding Language version	Python 3.9

2.2.1 IDE Installation

The most recent version of Anaconda navigator has been installed in order to implement the models that are extremely demanding. Since, Spyder was used as IDE hence 'conda' needs to be installed.A sequence of various stages are required for its installation, which are outlined below:

- For Downloading and Installing Anaconda please visit to the given url in footnote.[1]. Figure 2 shows the website to download and install anaconda

[1] Anaconda link : `https://www.anaconda.com/products/individual`


docs.anaconda.com/anaconda/install/windows/
ANACONDA DOCUMENTATION
Home
Anaconda Enterprise 5
Anaconda Enterprise 4
Anaconda Distribution
Installation
Installing on Windows
Installing on macOS
Installing on Linux
Installing on Linux POWER
Installing in silent mode
Installing for multiple users
Verifying your installation
Anaconda installer file hashes
Updating from older versions
Uninstalling Anaconda
User guide
v: latest

Installing on Windows

1. Download the Anaconda installer.

2. RECOMMENDED: Verify data integrity with SHA-256. For more information on hashes, see What about cryptographic hash verification?

3. Double click the installer to launch.

Note

To prevent permission errors, do not launch the installer from the Favorites folder.

Note

If you encounter issues during installation, temporarily disable your anti-virus software during install, then re-enable it after the installation concludes. If you installed for all users, uninstall Anaconda and re-install it for your user only and try again.

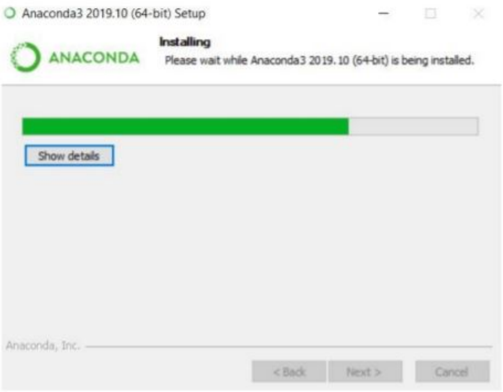
4. Click Next.

5. Read the licensing terms and click "I Agree".

6. Select an install for "Just Me" unless you're installing for all users (which requires Windows Administrator privileges) and click Next.

7. Select a destination folder to install Anaconda and click the Next button. See FAQ.

Figure 2: Homepage to download and install anaconda



Installing

Please wait while Anaconda3 2019.10 (64-bit) is being installed.

Figure 3: Final Installation of Anaconda

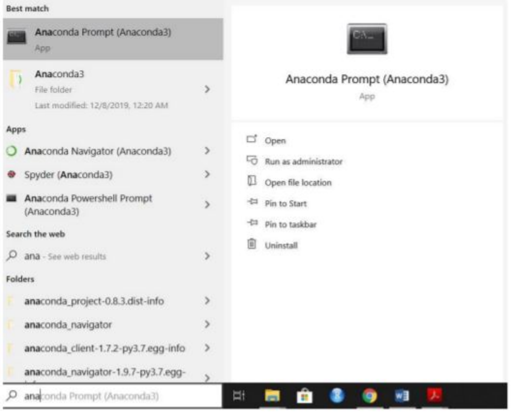
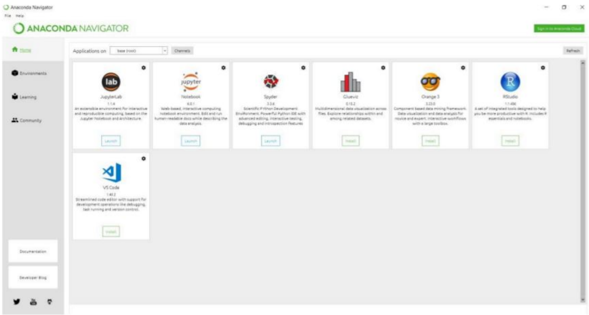

Best match
Anaconda Prompt (Anaconda3)
App
Anaconda3
File folder
Last modified: 12/8/2019, 12:20 AM
Apps
Anaconda Navigator (Anaconda3)
Spyder (Anaconda3)
Anaconda Powershell Prompt (Anaconda3)
Search the web
ana - See web results
Folders
anaconda_project-0.8.3.dist-info
anaconda_navigator
anaconda_client-1.7.2-py3.7.egg-info
anaconda_navigator-1.9.7-py3.7.egg-
anaconda Prompt (Anaconda3)
Anaconda Prompt (Anaconda3)
App
Open
Run as administrator
Open file location
Pin to Start
Pin to taskbar
Uninstall

Figure 4: Anaconda CLI

- Need to follow the steps further to install the anaconda navigator and cli will be installed as shown in Figure 3 and Figure 4



JupyterLab 1.1.4 An extensible environment for interactive and reproducible computing, based on the Jupyter Notebook and Architecture.

Notebook 6.0.1 Web-based, interactive computing notebook environment. Edit and run human-readable docs while describing the data analysis.

Spyder 3.3.6 Scientific PYthon Development EnviRonment. Powerful Python IDE with advanced editing, interactive testing, debugging and introspection features

Glueviz 0.15.2 Multidimensional data visualization across files. Explore relationships within and among related datasets.

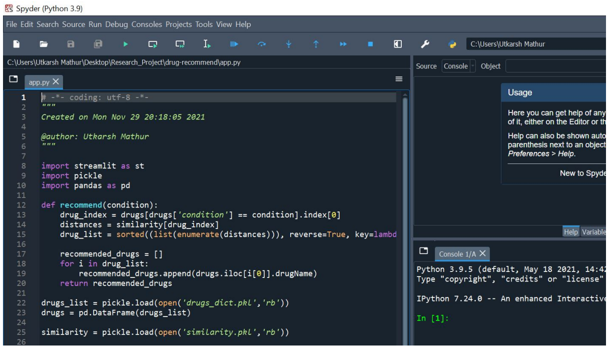
Orange 3 3.23.0 Component based data mining framework. Data visualization and data analysis for novice and expert. Interactive workflows with a large toolbox.

RStudio 1.1.456 A set of integrated tools designed to help you be more productive with R. Includes R essentials and notebooks.

VS Code 1.40.2 Streamlined code editor with support for development operations like debugging, task running and version control.

Figure 5: Anaconda Navigator

- Select Sypder from Anaconda Navigator and launch the application. It should always be noted that always a new environment needs to be created for preparing new project. This will help in keeping all the required libraries of python intact to the project and can be fetched whenever needs to be deployed.Figure 5 and Figure 6


Spyder (Python 3.9)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\Utkarsh Mathur
C:\Users\Utkarsh Mathur\Desktop\Research_Project\drug-recommend\app.py
app.py
Source Console Object
Usage
Help Variable
Console 1/A

```python
# -*- coding: utf-8 -*-
"""
Created on Mon Nov 29 20:18:05 2021

@author: Utkarsh Mathur
"""

import streamlit as st
import pickle
import pandas as pd

def recommend(condition):
    drug_index = drugs[drugs['condition'] == condition].index[0]
    distances = similarity[drug_index]
    drug_list = sorted((list(enumerate(distances))), reverse=True, key=lambd

    recommended_drugs = []
    for i in drug_list:
        recommended_drugs.append(drugs.iloc[i[0]].drugName)
    return recommended_drugs

drugs_list = pickle.load(open('drugs_dict.pkl','rb'))
drugs = pd.DataFrame(drugs_list)

similarity = pickle.load(open('similarity.pkl','rb'))
```

Here you can get help of any
of it, either on the Editor or th

Help can also be shown auto
parenthesis next to an object
Preferences > Help.

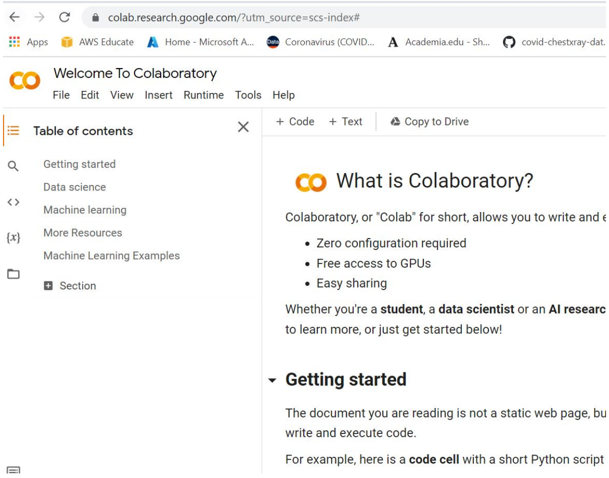
New to Spyde

```
Python 3.9.5 (default, May 18 2021, 14:42
Type "copyright", "credits" or "license"

IPython 7.24.0 -- An enhanced Interactive

In [1]:
```

Figure 6: Spyder IDE

2.2.2 Google Colab

Google colab can be accessed using `https://colab.research.google.com/` and further the 'Drug_Recommender_model.ipynb' file can be imported which is shared in Artefact zip folder. The Figure 7 below shows google colab notebook



What is Colaboratory?

Colaboratory, or "Colab" for short, allows you to write and e

- Zero configuration required
- Free access to GPUs
- Easy sharing

Whether you're a **student**, a **data scientist** or an **AI researc**
to learn more, or just get started below!

Getting started

The document you are reading is not a static web page, bu
write and execute code.

For example, here is a **code cell** with a short Python script

Figure 7: Google Colab Notebook

2.3 Python Libraries

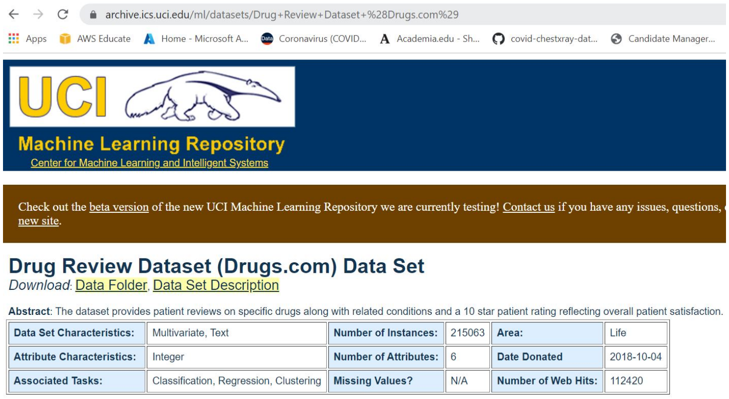
These are the dependencies which needs to be their on the system or virtual environment inorder to perform some operations. Hence, in order to install the requirements.txt file which is shared as artefacts and consists of all the libraries. Please use below command

pip install -r requirements.txt

3 Dataset Details

The Dataset is gathered from UCI Machine learning library and can be downloaded from Artefact zip file shared. Figure 8 shows the Dataset Source website

Furthermore, the Dataset is stored at Azure Cloud and the link for the same is passed in the python code.The link to download Dataset is testdata : `https://researchproject.`


archive.ics.uci.edu/ml/datasets/Drug+Review+Dataset+%28Drugs.com%29
Apps
AWS Educate
Home - Microsoft A...
Coronavirus (COVID...
Academia.edu - Sh...
covid-chestxray-dat...
Candidate Manager...
UCI
Machine Learning Repository
Center for Machine Learning and Intelligent Systems

Check out the beta version of the new UCI Machine Learning Repository we are currently testing! Contact us if you have any issues, questions,
new site.

Drug Review Dataset (Drugs.com) Data Set

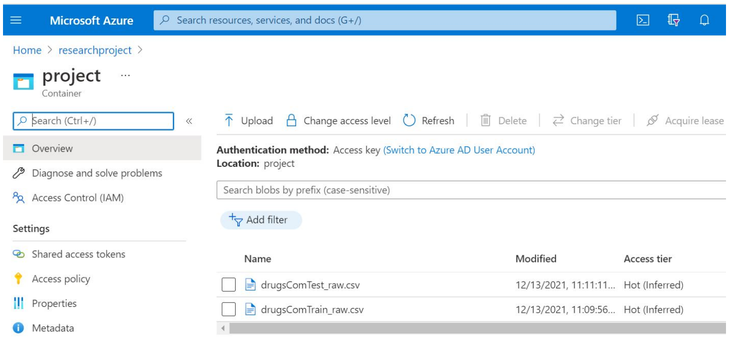
Download: Data Folder, Data Set Description

Abstract: The dataset provides patient reviews on specific drugs along with related conditions and a 10 star patient rating reflecting overall patient satisfaction.

Data Set Characteristics:	Multivariate, Text	**Number of Instances:**	215063	**Area:**	Life
Attribute Characteristics:	Integer	**Number of Attributes:**	6	**Date Donated**	2018-10-04
Associated Tasks:	Classification, Regression, Clustering	**Missing Values?**	N/A	**Number of Web Hits:**	112420

Figure 8: UCI ML Repository website

blob.core.windows.net/project/drugsComTest_raw.csv and traindata : https://researchproject.blob.core.windows.net/project/drugsComTrain_raw.csv Figure 9 shows the Dataset Source website



Authentication method: Access key (Switch to Azure AD User Account)
Location: project

	Name	Modified	Access tier
□	drugsComTest_raw.csv	12/13/2021, 11:11:11...	Hot (Inferred)
□	drugsComTrain_raw.csv	12/13/2021, 11:09:56...	Hot (Inferred)

Figure 9: Dataset stored in Blob Storage of Azure Cloud

4 Recommendation Engine Building

The Data preprocessing and cleaning done. Top 10 medicine were sorted out based on most reviewed by patients as shown in Figure 10

The Tags column was then created by concatenating the various other column to build metadata of drug in the form of a string as shown in Figure 11

The words were further stemmed out to in order to achieve a respective root word for all words in the tags, this done in order to make word into vectors as shown in Figure 12

Now using Count Vectorizer from sklearn library, words of tags are transformed into vectors and a numpy array hot created as shown in Figure 13

Cosine Similarity was calculated using Bag of Words Technique and comparing each vector with other vector in the matrix. The similarity score for one of the Drug is shown as below Figure 14

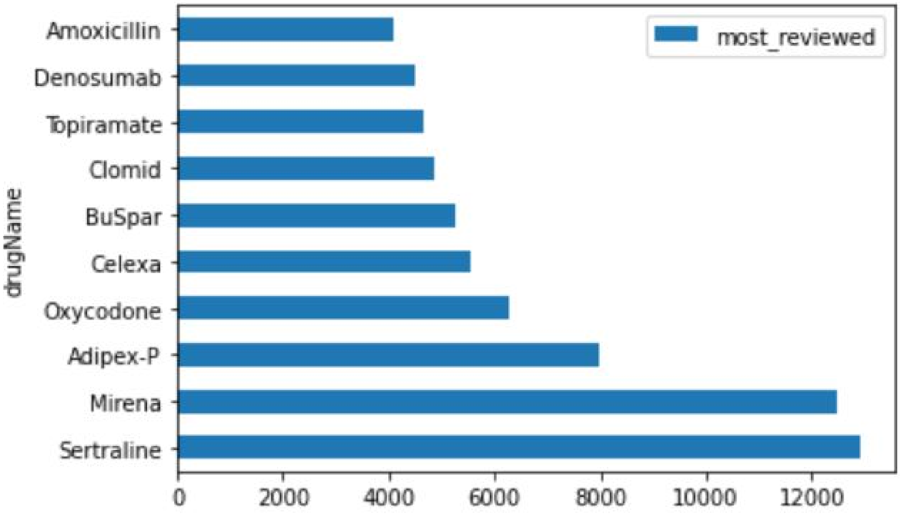
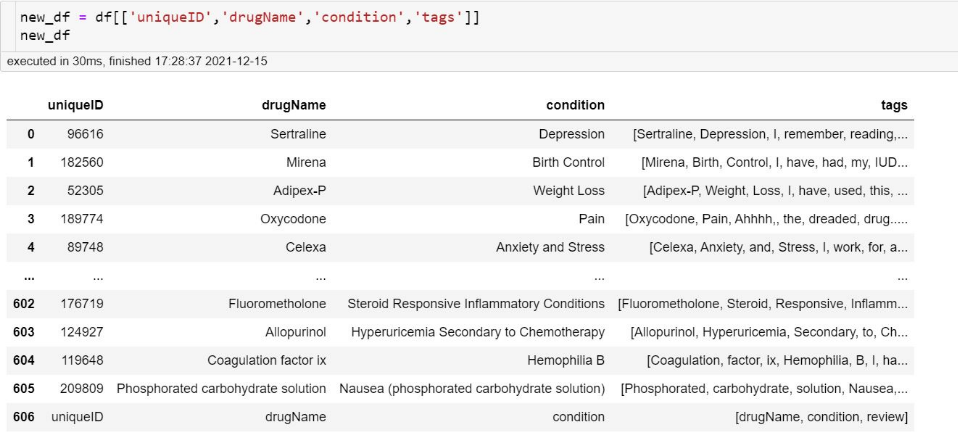


Figure 10: Top 10 most Reviewed Drugs/Medicine

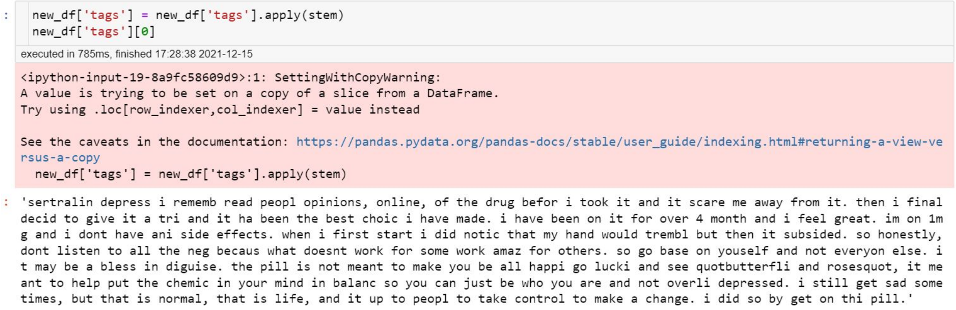


```
new_df = df[['uniqueID','drugName','condition','tags']]
new_df
```

executed in 30ms, finished 17:28:37 2021-12-15

	uniqueID	drugName	condition	tags
0	96616	Sertraline	Depression	[Sertraline, Depression, I, remember, reading,...
1	182560	Mirena	Birth Control	[Mirena, Birth, Control, I, have, had, my, IUD...
2	52305	Adipex-P	Weight Loss	[Adipex-P, Weight, Loss, I, have, used, this, ...
3	189774	Oxycodone	Pain	[Oxycodone, Pain, Ahhhh,, the, dreaded, drug.....
4	89748	Celexa	Anxiety and Stress	[Celexa, Anxiety, and, Stress, I, work, for, a...
...	...	...	...	...
602	176719	Fluorometholone	Steroid Responsive Inflammatory Conditions	[Fluorometholone, Steroid, Responsive, Inflamm...
603	124927	Allopurinol	Hyperuricemia Secondary to Chemotherapy	[Allopurinol, Hyperuricemia, Secondary, to, Ch...
604	119648	Coagulation factor ix	Hemophilia B	[Coagulation, factor, ix, Hemophilia, B, I, ha...
605	209809	Phosphorated carbohydrate solution	Nausea (phosphorated carbohydrate solution)	[Phosphorated, carbohydrate, solution, Nausea,...
606	uniqueID	drugName	condition	[drugName, condition, review]

Figure 11: Dataframe with Tags/Metadata column



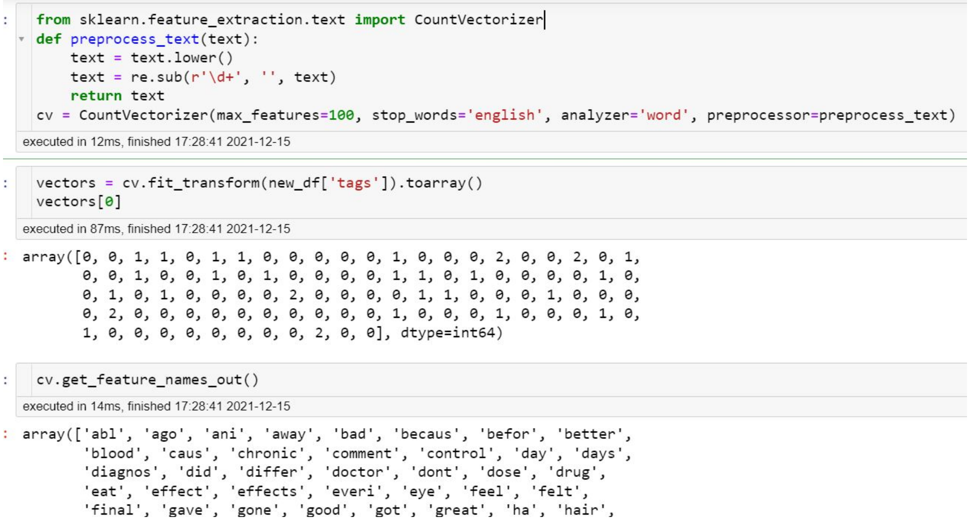
```
:   new_df['tags'] = new_df['tags'].apply(stem)
    new_df['tags'][0]
```

executed in 785ms, finished 17:28:38 2021-12-15

```
<ipython-input-19-8a9fc58609d9>:1: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-ve
rsus-a-copy
  new_df['tags'] = new_df['tags'].apply(stem)
```

```
: 'sertralin depress i rememb read peopl opinions, online, of the drug befor i took it and it scare me away from it. then i final
  decid to give it a tri and it ha been the best choic i have made. i have been on it for over 4 month and i feel great. im on 1m
  g and i dont have ani side effects. when i first start i did notic that my hand would trembl but then it subsided. so honestly,
  dont listen to all the neg becaus what doesnt work for some work amaz for others. so go base on youself and not everyon else. i
  t may be a bless in diguise. the pill is not meant to make you be all happi go lucki and see quotbutterfli and rosesquot, it me
  ant to help put the chemic in your mind in balanc so you can just be who you are and not overli depressed. i still get sad some
  times, but that is normal, that is life, and it up to peopl to take control to make a change. i did so by get on thi pill.'
```

Figure 12: Metadata after getting stemmed



```
from sklearn.feature_extraction.text import CountVectorizer
def preprocess_text(text):
    text = text.lower()
    text = re.sub(r'\d+', '', text)
    return text
cv = CountVectorizer(max_features=100, stop_words='english', analyzer='word', preprocessor=preprocess_text)
```

executed in 12ms, finished 17:28:41 2021-12-15

```
vectors = cv.fit_transform(new_df['tags']).toarray()
vectors[0]
```

executed in 87ms, finished 17:28:41 2021-12-15

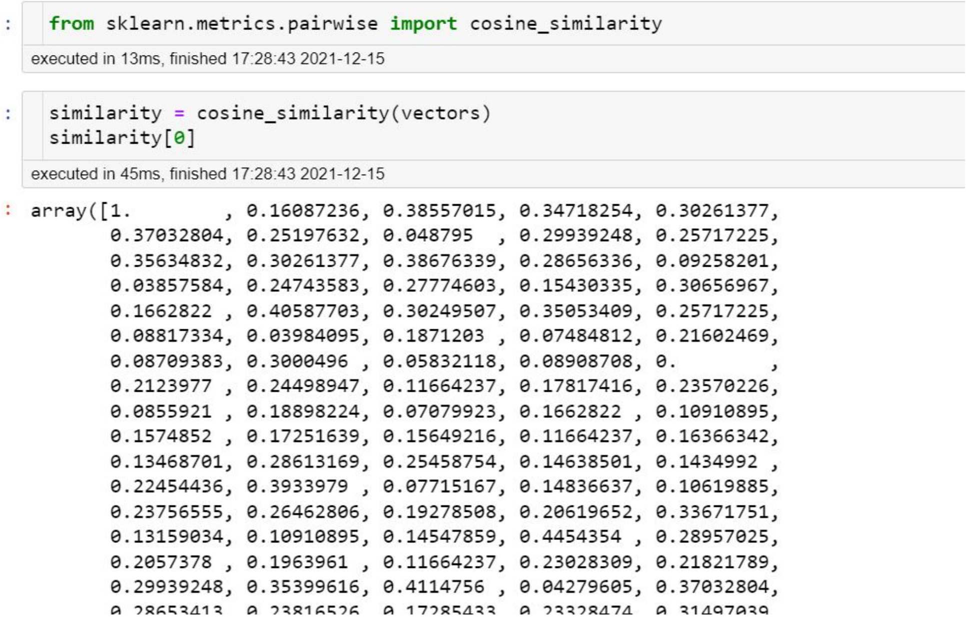
```
array([0, 0, 1, 1, 0, 1, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 2, 0, 0, 2, 0, 1,
       0, 0, 1, 0, 0, 1, 0, 1, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, 0, 1, 0,
       0, 1, 0, 1, 0, 0, 0, 0, 2, 0, 0, 0, 0, 1, 1, 0, 0, 0, 1, 0, 0, 0,
       0, 2, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 1, 0,
       1, 0, 0, 0, 0, 0, 0, 0, 0, 2, 0, 0], dtype=int64)
```

```
cv.get_feature_names_out()
```

executed in 14ms, finished 17:28:41 2021-12-15

```
array(['abl', 'ago', 'ani', 'away', 'bad', 'becaus', 'befor', 'better',
       'blood', 'caus', 'chronic', 'comment', 'control', 'day', 'days',
       'diagnos', 'did', 'differ', 'doctor', 'dont', 'dose', 'drug',
       'eat', 'effect', 'effects', 'everi', 'eye', 'feel', 'felt',
       'final', 'gave', 'gone', 'good', 'got', 'great', 'ha', 'hair',
```

Figure 13: Code Snippet for count vectorizer and vectors to numpy array



```
from sklearn.metrics.pairwise import cosine_similarity
```

executed in 13ms, finished 17:28:43 2021-12-15

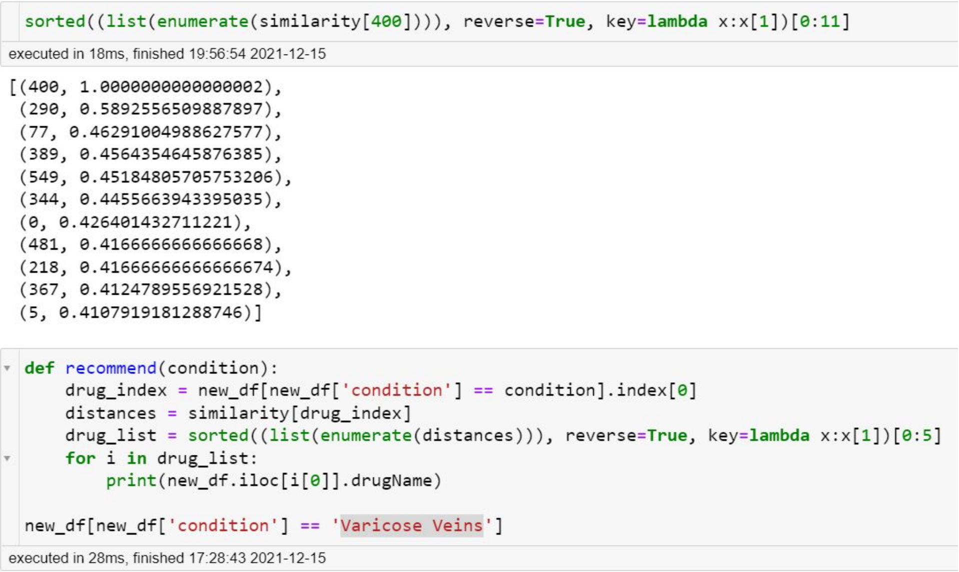
```
similarity = cosine_similarity(vectors)
similarity[0]
```

executed in 45ms, finished 17:28:43 2021-12-15

```
array([1.        , 0.16087236, 0.38557015, 0.34718254, 0.30261377,
       0.37032804, 0.25197632, 0.048795  , 0.29939248, 0.25717225,
       0.35634832, 0.30261377, 0.38676339, 0.28656336, 0.09258201,
       0.03857584, 0.24743583, 0.27774603, 0.15430335, 0.30656967,
       0.1662822 , 0.40587703, 0.30249507, 0.35053409, 0.25717225,
       0.08817334, 0.03984095, 0.1871203 , 0.07484812, 0.21602469,
       0.08709383, 0.3000496 , 0.05832118, 0.08908708, 0.        ,
       0.2123977 , 0.24498947, 0.11664237, 0.17817416, 0.23570226,
       0.0855921 , 0.18898224, 0.07079923, 0.1662822 , 0.10910895,
       0.1574852 , 0.17251639, 0.15649216, 0.11664237, 0.16366342,
       0.13468701, 0.28613169, 0.25458754, 0.14638501, 0.1434992 ,
       0.22454436, 0.3933979 , 0.07715167, 0.14836637, 0.10619885,
       0.23756555, 0.26462806, 0.19278508, 0.20619652, 0.33671751,
       0.13159034, 0.10910895, 0.14547859, 0.4454354 , 0.28957025,
       0.2057378 , 0.1963961 , 0.11664237, 0.23028309, 0.21821789,
       0.29939248, 0.35399616, 0.4114756 , 0.04279605, 0.37032804,
       0.28653413  0.23816526  0.17285433  0.23328474  0.31497039
```

Figure 14: Cosine Similarity of 1st Drug with others

Finally, the model was build using the reverse sorting similarity score, which means more the similarity between two drugs it will come first. The Recommend function is defined which will take input as condition (Symptoms or diseases) and recommend the drugs as per most 5 similar ones. This is shown in Figure 15



```
sorted((list(enumerate(similarity[400]))), reverse=True, key=lambda x:x[1])[0:11]
```

executed in 18ms, finished 19:56:54 2021-12-15

```
[(400, 1.0000000000000002),
 (290, 0.5892556509887897),
 (77, 0.46291004988627577),
 (389, 0.4564354645876385),
 (549, 0.45184805705753206),
 (344, 0.4455663943395035),
 (0, 0.426401432711221),
 (481, 0.4166666666666668),
 (218, 0.41666666666666674),
 (367, 0.4124789556921528),
 (5, 0.4107919181288746)]
```

```
def recommend(condition):
    drug_index = new_df[new_df['condition'] == condition].index[0]
    distances = similarity[drug_index]
    drug_list = sorted((list(enumerate(distances))), reverse=True, key=lambda x:x[1])[0:5]
    for i in drug_list:
        print(new_df.iloc[i[0]].drugName)

new_df[new_df['condition'] == 'Varicose Veins']
```

executed in 28ms, finished 17:28:43 2021-12-15

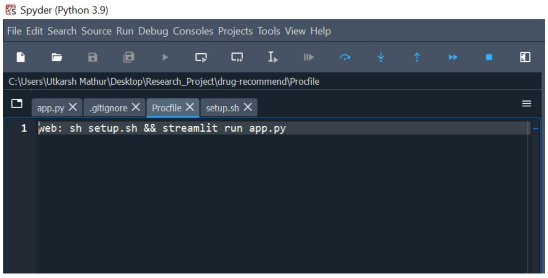
Figure 15: Defining a Function to recommend drugs

5 Application Deployment on Heroku Cloud

After the Recommender model was built, the dataframe file and similarity matrix file was exported in the form of .pkl file. There were various other files which were also created using Spyder IDE as a dependency file for Heroku CloudApp deployment.All these files are shared in Artefacts and also uploaded on github link provided in Section 1.

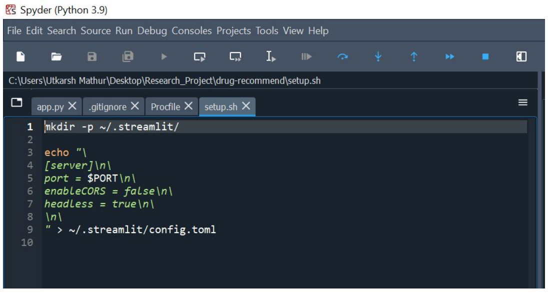
Figure 18 shows the app.py file which contains the code for running the application and mechanism behind it. In this technique, Streamlit [2] was used to design a website.

[2] Link : `https://streamlit.io/`


Spyder (Python 3.9)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\Utkarsh Mathur\Desktop\Research_Project\drug-recommend\Procfile
app.py
.gitignore
Procfile
setup.sh

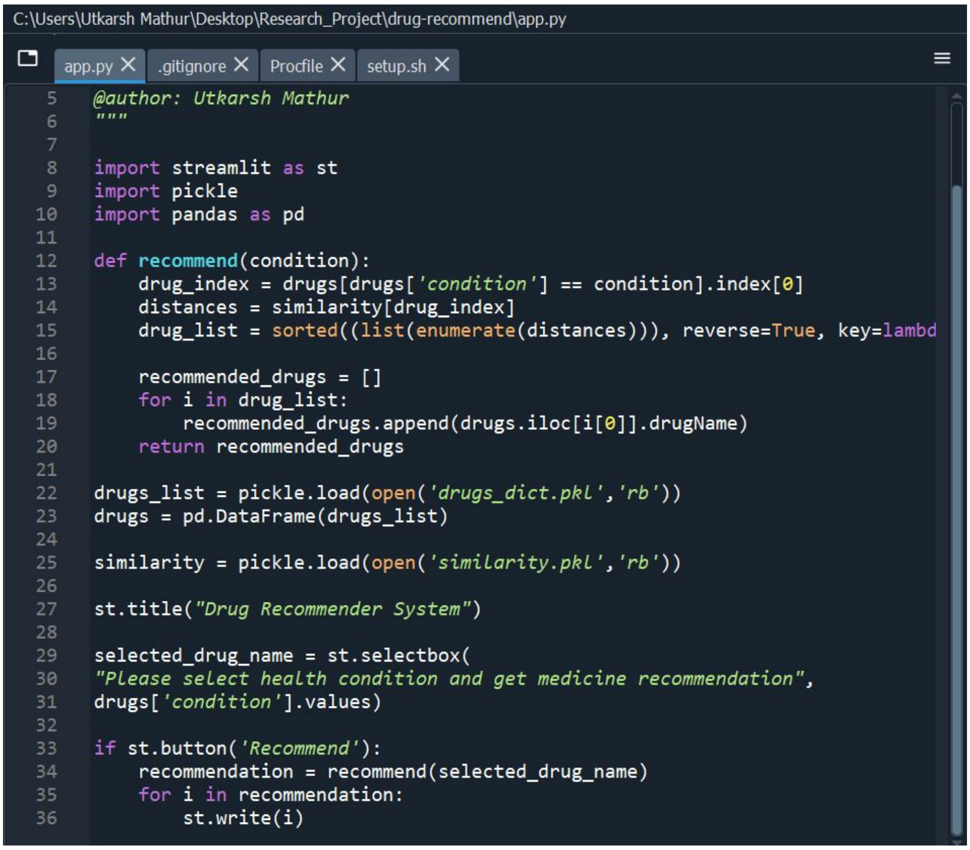
```
web: sh setup.sh && streamlit run app.py
```

Figure 16: Procfile


Spyder (Python 3.9)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\Utkarsh Mathur\Desktop\Research_Project\drug-recommend\setup.sh
app.py
.gitignore
Procfile
setup.sh

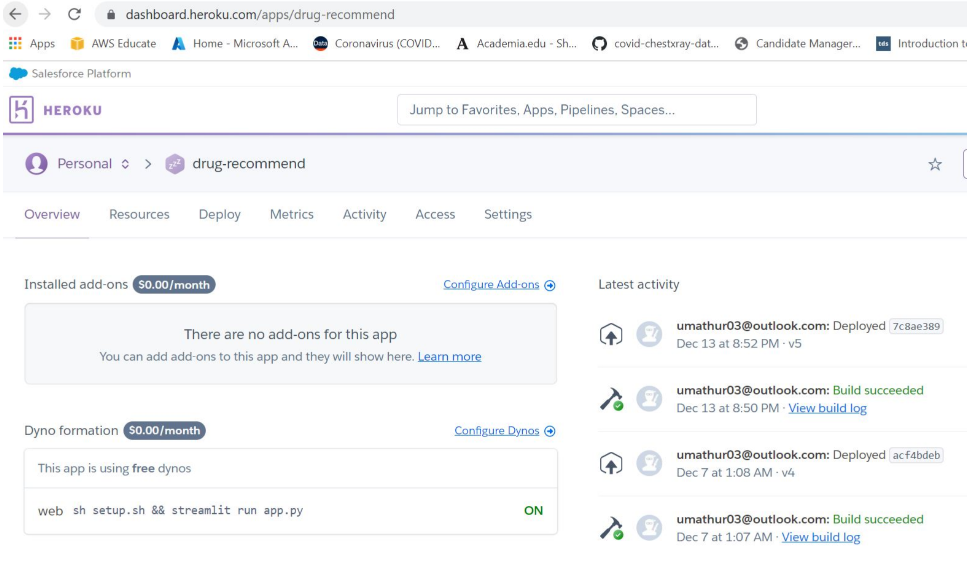
```
mkdir -p ~/.streamlit/

echo "\
[server]\n\
port = $PORT\n\
enableCORS = false\n\
headless = true\n\
\n\
" > ~/.streamlit/config.toml
```

Figure 17: Setup shell file


C:\Users\Utkarsh Mathur\Desktop\Research_Project\drug-recommend\app.py
app.py
.gitignore
Procfile
setup.sh

```python
@author: Utkarsh Mathur
"""

import streamlit as st
import pickle
import pandas as pd

def recommend(condition):
    drug_index = drugs[drugs['condition'] == condition].index[0]
    distances = similarity[drug_index]
    drug_list = sorted((list(enumerate(distances))), reverse=True, key=lambd

    recommended_drugs = []
    for i in drug_list:
        recommended_drugs.append(drugs.iloc[i[0]].drugName)
    return recommended_drugs

drugs_list = pickle.load(open('drugs_dict.pkl','rb'))
drugs = pd.DataFrame(drugs_list)

similarity = pickle.load(open('similarity.pkl','rb'))

st.title("Drug Recommender System")

selected_drug_name = st.selectbox(
"Please select health condition and get medicine recommendation",
drugs['condition'].values)

if st.button('Recommend'):
    recommendation = recommend(selected_drug_name)
    for i in recommendation:
        st.write(i)
```

Figure 18: Code Snippet for Deploying Application



Installed add-ons $0.00/month Configure Add-ons

There are no add-ons for this app

You can add add-ons to this app and they will show here. Learn more

Dyno formation $0.00/month Configure Dynos

This app is using **free** dynos

```
web  sh setup.sh && streamlit run app.py    ON
```

Latest activity

umathur03@outlook.com: Deployed 7c8ae389
Dec 13 at 8:52 PM · v5

umathur03@outlook.com: Build succeeded
Dec 13 at 8:50 PM · View build log

umathur03@outlook.com: Deployed acf4bdeb
Dec 7 at 1:08 AM · v4

umathur03@outlook.com: Build succeeded
Dec 7 at 1:07 AM · View build log

Figure 19: Heroku Application Deployment

Figure 19 shows the Heroku App Deployment Activity on its server Steps for deploying Application is given below:

Step 1: Download and install the Heroku CLI at `https://devcenter.heroku.com/articles/heroku-cli`

Step 2: $ heroku login

Step 3: Use Git to clone drug-recommend's source code to your local machine.

Step 4: $ heroku git:clone -a drug-recommend

Step 5: $ cd drug-recommend

Step 6: Copy all the 7 files from the folder Website_Deploy_File which is shared in Artefacts to current directory which is "$ drug-recommend"

Step 7: Make some changes to the code you just cloned and deploy them to Heroku using Git.

Step 8: $ git add .

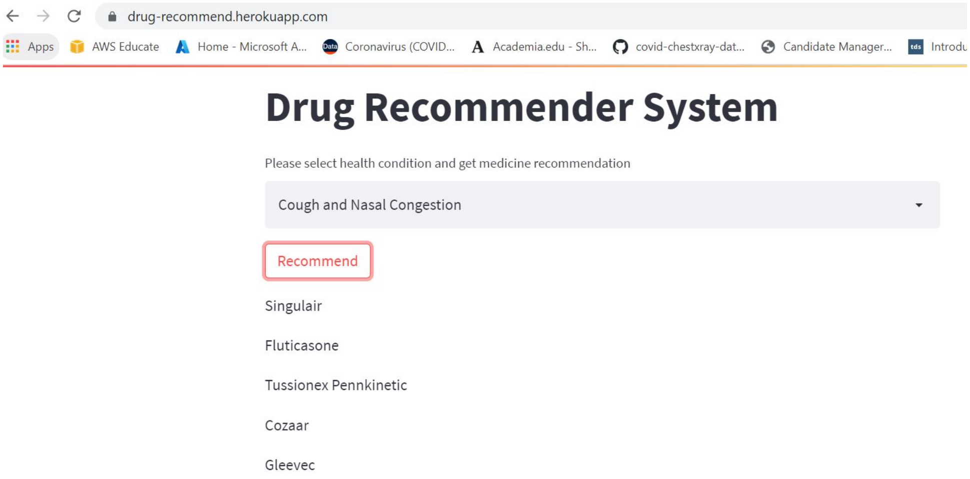
Step 9: $ git commit -am "your comment"

Step 10: $ git push heroku main

The process will take some time to upload and deploy on cloud.

Once it is deployed the website will be accessible at `https://drug-recommend.herokuapp.com/`

Below is the screenshot from main website for Drug Recommender System Figure 20



Drug Recommender System

Please select health condition and get medicine recommendation

Cough and Nasal Congestion

Singulair

Fluticasone

Tussionex Pennkinetic

Cozaar

Gleevec

Figure 20: Webpage built by Streamlit on Heroku Server

References

Dai, Q., Hong, X., Cai, J., Liu, Y., Zhao, H., Luo, J., Lin, Z. and Chen, S. (2018). Deep learning based recommendation algorithm in online medical platform, *in* J. Ren, A. Hussain, J. Zheng, C.-L. Liu, B. Luo, H. Zhao and X. Zhao (eds), *Advances in Brain Inspired Cognitive Systems*, Springer International Publishing, Cham, pp. 34–43.