

Project Report
on
Drug Recommendation System using Patient Reviews

Submitted as partial fulfillment for the award of

BACHELOR OF TECHNOLOGY
DEGREE

Session 2021-22
in

Computer Science and Engineering

By

Shatmanyu Gupta (1803210135)
Tauheed Shahid (1803210164)
Suraj Singh (1803210155)

Under the guidance of

Ms. Shanu Sharma
(Department of CSE, ABESEC)

ABES ENGINEERING COLLEGE, GHAZIABAD



AFFILIATED TO
DR. A.P.J. ABDUL KALAM TECHNICAL UNIVERSITY, U.P., LUCKNOW
(Formerly UPTU)

Project Report
on
Drug Recommendation System using Patient Reviews

Submitted as partial fulfillment for the award of

BACHELOR OF TECHNOLOGY
DEGREE

Session 2021-22
in

Computer Science and Engineering

By

Shatmanyu Gupta (1803210135)
Tauheed Shahid (1803210164)
Suraj Singh (1803210155)

Under the guidance of

Ms. Shanu Sharma
(Department of CSE, ABESEC)

ABES ENGINEERING COLLEGE, GHAZIABAD



Estd. 2000



AFFILIATED TO
DR. A.P.J. ABDUL KALAM TECHNICAL UNIVERSITY, U.P., LUCKNOW
(Formerly UPTU)

STUDENT'S DECLARATION

We hereby declare that the work being presented in this report entitled **Drug Recommendation using Patient Reviews** is an authentic record of our own work carried out under the supervision of **Ms. Shanu Shama**.

The matter embodied in this report has not been submitted by us for the award of any other degree.

Dated:

Signature of students(s)

Tauheed Shahid

Suraj Singh

Shatmanyu Gupta

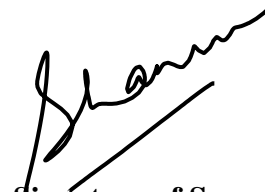
Department: CSE

This is to certify that the above statement made by the candidates is correct to the best of my knowledge.

Signature of HOD

(Prof. (Dr.) Divya Mishra)

**(Computer Science & Engineering
Department)**



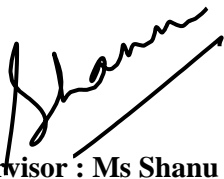
Signature of Supervisor

(Ms Shanu Sharma)

**(Computer Science & Engineering
Department)**

CERTIFICATE

This is to certify that Project Report entitle “Drugs Recommendation System using Patient Reviews” which is submitted by Shatmanyu Gupta, Suraj Singh, Tauheed Shahid in partial fulfillment of the requirement for the award of degree B. Tech. in Department of Computer Science and Engineering of Dr. A.P.J. Abdul Kalam Technical University, formerly Uttar Pradesh Technical University is a record of the candidate own work carried out by him/them under my supervision. The matter embodied in this thesis is original and has not been submitted for the award of any other degree.



Supervisor : Ms Shanu Sharma

Date :

ACKNOWLEDGEMENT

It gives us a great sense of pleasure to present the report of the B. Tech Project undertaken during B. Tech. Final Year. We owe special debt of gratitude to Ms. Shanu Sharma, Department of Computer Science & Engineering, ABESEC Ghaziabad for his constant support and guidance throughout the course of our work. Her sincerity, thoroughness and perseverance have been a constant source of inspiration for us. It is only his cognizant efforts that our endeavors have seen light of the day.

We also take the opportunity to acknowledge the contribution of (Dr.) Divya Mishra Head, Department of Computer Science & Engineering, ABESEC Ghaziabad for his full support and assistance during the development of the project.

We also do not like to miss the opportunity to acknowledge the contribution of all faculty members of the department for their kind assistance and cooperation during the development of our project. Last but not the least, we acknowledge our friends for their contribution in the completion of the project.

Signature:

Name : Shatmanyu Gupta

Roll No.: 1803210135

Date :

Signature:

Name : Tauheed Shahid

Roll No.: 1803210164

Date :

Signature:

Name : Suraj Singh

Roll No.: 1803210155

Date

ABSTRACT

In today's technological world healthcare is one major area of the medical field. A healthcare system requires to study huge data of the patients to conclude significant insights and help in prediction of best possible medications for the disease. But many studies show that majority of population die due to the medical errors caused in terms of taking wrong medications and these errors are caused by doctors, who prescribe medicines based on their limited experiences. As advancements in machine learning, deep learning like technologies that are emerging day by day, these methodologies can assist us to explore the medical history and can deplete medical errors by being doctor friendly. In this context, health recommender systems are a significant tools in speeding up the decision making process in the health care sector. As people use social networks to research their health condition, so the health recommender system is very important to conclude outcomes such as health insurance, clinical pathway-based treatment methods and other drugs based on the patient's health profile. In this project we propose a drug recommendation system, which takes the patient reviews data and performs sentiment analysis on it to find the best drug for a disease by using NGram model. In order to increase the accuracy, a Lightgbm model is also used.. The project consists of collecting dataset, reviewing dataset and cleaning the dataset, visualizing the relations between attributes of dataset and concluding different significant information from the dataset. Then we train the data using sentiment analysis of Natural Language Processing, a application of Convulation Neural Networks.

TABLE OF CONTENTS

Page

DECLARATION	ii
CERTIFICATE... ..	iii
ACKNOWLEDGEMENTS	iv
ABSTRACT.....	v
LIST OF TABLES.....	vii
LIST OF FIGURES	viii
CHAPTER 1 Introduction	10
1.1 Problem Introduction.....	10
1.2 Related Previous Work.....	13
1.3 Organization of Report.....	13
CHAPTER 2 Literature Survey.....	15
CHAPTER System design and methodology.....	19
3.1 System Design.....	19
3.2 Methodology/Algorithm.....	27
CHAPTER 4 Implementation and Results.....	29
4.1 Software and Hardware Requirements.....	29
4.2 Assumptions and Dependencies.....	29
4.3 Results	29
CHAPTER 5 CONCLUSIONS	48
REFERENCES	50

LIST OF TABLES

Table Number	Table Name	Page Number
3.1	Attributes and their types	19
5.1-5.2	Results	48

LIST OF FIGURES

Figure Number	Figure Name	Page Number
2.1	GalenOwl Framework	15
3.1	Dataset Description	20
3.2	Dataset Description	20
3.3	Dataset Description	20
3.4	Missing Values in data	21
3.5	Count of null values	22
3.6	Drugs vs Conditions	23
3.7	Share of Ratings	24
3.8	Common Conditions	24
3.9	Common Drugs	25
3.10	Phases of Recommender System	25
3.11	Architecture of Drug Recommendation system	26
3.12	Data Flow Diagram	26
4.1- 4.37	Results	30

CHAPTER 1

INTRODUCTION

1.1. Problem Introduction

A drug recommendation system is a framework that recommend the best drugs for a particular disease by analyzing patient reviews.

The proposed drug recommendation system uses the current technologies like machine learning, Sentiment Analysis, Neural Networks etc. to find out the fascinating records hidden in the history of drug and diminish the medical errors by the doctors while prescribing medicines for any condition.

1.1.1. Motivation

- Since numerous studies show people die due to the medical errors due to the limited experiences of doctors. Also newly joined doctors are also unable to prescribe the right medicine for the patients.
- One of the most significant and researched topic on web is regarding health Care. In this digital space everyone is looking for quick solutions so majority of the people go online for health related issues to educate them.
- One of the most concerned and searched topic on the internet is about health information. According to Pew Research article 2013, almost 60% of grownups are looking for enough health information on the web with 35% of respondents concentrating on diagnosing ailments online only[1]
- The number of specialists can't be extended rapidly in a brief period of time. A recommender system must be displayed beyond what many would consider possible in this troublesome time of emerging severe diseases [10].

1.1.2. Project Objective

- The prime objective is to build a drug recommendation system that recommend the best drugs for a particular condition by analyzing different patient reviews of the drugs of each type.
- To speed up the decision making process in healthcare system by developing a tool which can assist doctors in recommending best drug for the condition.
- Since numerous studies show people die due to the medical errors due to the limited experiences of doctors. Also newly joined doctors are also unable to prescribe the right medicine for the patients.

1.1.3. Scope of the Project

- System consists of five modules which are
 - (i) Dataset
 - (ii) Data Preprocessing
 - (iii) Recommendation model
 - (iv) Model evaluation
 - (v) Data visualization
- **Dataset:** it contains a patient drug review dataset containing attributes like unique Id, drug name, condition(disease of patient), date , useful count, reviews and ratings given by the patients. The Dataset contains 6 dimension with about 215063 rows.
- **Data Preprocessing:** a) Find count of missing or empty fields for all the attributes
 - b) These vacant fields can be disregarded ,eliminated or filled. So the missing qualities in lines can be taken out since the dataset is exceptionally colossal
 - c) Delete duplicate rows from the dataset to clean the data and efficiently train our model.
 - d) Remove the conditions with only one drug as single drug might not

be sufficient to recommend the best one.

- e) Words like not,don't need, needn't, never etc are significant parts of sentiment analysis, so remove them from stop words('a','the',etc).

Now clean reviews by removing stop words.

- **Model building:** The whole drug review dataset is splitted into two portions where 75% of the data is training data and 25% is used for testing the data. Sentiment analysis is done using N-gram deep learning model and LightBgm learning model.
- **Model Evaluation:** Model evaluation is used to check all the models by considering the properties like efficiency, accuracy and scalability of model and to select the best one.
- **Data Visualization:** It is used to depict some crucial information that comes out during experimenting.

1.2. Related Previous Work

- The research [6] presents Galen ontological writing learning, a semantic-enabled web-based system, to assist experts with finding subtleties on the medications. The paper portrays a structure that recommends drugs for a patient in view of the patient's disease, responsive qualities, and medication associations
- Leilei Sun [2] confirms enormous scope treatment records to recognize the best medication solution for patients
- In this exploration [8], multilingual opinion examination was performed utilizing Naive Bayes and Recurrent Neural Network (RNN). Google interpreter API was utilized to change over multilingual tweets into the English language.
- Sadly, there are a set number of investigations accessible in the field of drugs proposition system utilizing sentiment analysis because the prescription surveys are significantly more nitty gritty to dissect as it comprises clinical expressions like disease names, responses, a manufactured names that utilized in the development of the medication [5]

1.3. Organization of the Report.

The report is divided into 5 chapters –

(i) Introduction

This Chapter discusses the introduction of the problem statement

- a) discusses the problem statement
- b) discusses the objective of problem
- c) discusses the motivation and scope of the project

(ii) Literature Survey

This chapter discusses about the previous work undertaken in this field and how we are currently proposing our method in relation to the works.

- a) References to previous researches related to the topic
- b) How we are proposing our method related to these works

(iii) System Design and Methodology

This chapter discusses about the methodology we are proposing and the architecture behind it

- a) System Architecture or Diagrammatic view of proposed methodology
- b) Flowchart or Data Flow diagrams related to the proposed method

(iv) Implementation and Results

This chapter discusses the implementation status of project and references related to it in the form of figures and code snippets.

- a) Software and Hardware Requirements of the project
- b) Implementation progress of the project
- c) Snapshots of interfaces of the project or code snippets

(v) **Conclusion**

This chapter discusses about the model metrics evaluation and future work related to the project.

- a) Performace Evaluation of the model
- b) Evaluation of different Algorithms
- c) Future Scope of the project

CHAPTER 2

LITERATURE SURVEY

With increase in AI frameworks, there is a command to apply ML and profound learning procedures to proposal models. These days, recommender frameworks are exceptionally ordinary in the travel industry, internet business, and so on Tragically, there are a predetermined number of investigations accessible in the field of medication proposition system utilizing feeling examination because the medicine surveys are significantly more itemized to break down as it comprises clinical expressions like contamination names, responses, a manufactured names that utilized in the development of the medication [5].

The examination [6] presents GalenOWL, a semantic-enabled web-based system, to assist experts with finding subtleties on the medications. The paper portrays a structure that recommends drugs for a patient dependent on the patient's contamination, sensitivities, and medication cooperations.

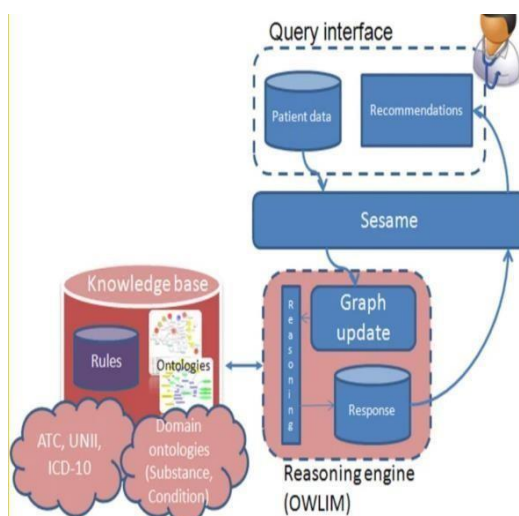


Figure: 2.1 Galen OWL Framework

Leilei Sun [2] verifies enormous scope treatment records to distinguish the best medication remedy for patients. In this examination [8], multilingual inclination examination was performed using Naive Bayes and Recurrent Neural Network (RNN). Google translator API was used to change multilingual tweets into the English language. The outcomes show that RNN with 95.34% beat Naive Bayes, 77.21%.

The review depends on the way that the suggested medication ought to rely on the patient's ability. For instance, assuming the patient's insusceptibility is low, by then, dependable drugs should be suggested. Proposed a danger level arrangement strategy to recognize the patient's invulnerability. For instance, more than 60 danger factors, hypertension, alcohol dependence, etc have been taken on, which choose the patient's ability to protect himself from contamination.

Fundamentally there are 3 types of filtering in Recommender Systems namely :

- **Collaborative Filtering** – It is based on the knowledge collected, it finds similarities between users and items simultaneously to provide recommendations, the model doesn't use features to recommend the item, rather it classify the chunks of clients with similar interests
- **Content-based Filtering** – The model works on the methodology of most overlapping features between and customer history regarding that product. The product which have the most overlapping metrics will be recommended. The model will recommend the product on the basis of information about the customer.
- **Hybrid Filtering** – Mixture of Collaborative Filtering and Content - based filtering. The model works on the drawback of each model and improves the performance of the system.

As the correspondence interface, web has been the fundamental hotspot for clients to get to the wellbeing data [16]. Consequently, Heath recommenders assume a huge part in separating data. A few models being 'HealthyHarlem' and 'MyHeathEducator'

Since wellbeing based recommender models opens another arising field of exploration for the most part rumored at meetings rather than top logical journals. The research technique was based upon an audit convention to survey the writing. The review by Kitchenman's efficient audit technique was utilized [17].

- Determining the topic of research
- Discovery of literature using inclusion-exclusion principle
- Evaluation of quality of studies
- Analysis of data
- Report of the results

On careful Analysis of research, papers related to health recommender systems generally lies in these categories –

- User groups and system design
- Nutrition based recommended frameworks
- Challenges and opportunities

Electronic Health records were also a part of HRS in terms of health marketing, personal recommendations and self-examinations. Also due to Covid Scene , the trending domain in Health recommender frameworks is Telemedicine. The ongoing investigations show that telemedicine and determination applications were principal focus of HRS concentrates as far as overseeing wellbeing undertakings.

In recommender systems, it should be mentioned that analyzing semantics or the context of reviews are a challenging topic which is crucial in predicting user behavior and the efficiency of the medication. After going through several articles of IEEE , IEEE proved to be the most crucial resource covering over 50% of our research. The principle end is that recommender frameworks connected with well being are a promising advancement for medical care administrations. The investigations exhibited that Health recommender frameworks have been spread out in various fields of

wellbeing industry, and wellbeing recommender applications have been progressively installed in the wellbeing administration frameworks.

Thinking about the writing, there were number of studies connected with well being recommender frameworks use, plan and procedures. In this regard, it was found that arising wellbeing recommender studies were likewise in a rising pattern in the writing. Nonetheless, wellbeing recommender area is somewhat new, along these lines it needs an ideal opportunity to introduce mature investigates and to improve separating calculations. What's more, security issues establish a main pressing issue to survive. In this paper, it was intended to contribute writing by [16] offering a perspective with regards to the writing of wellbeing recommender models, [18] underlining the examinations about health recommender models, and [19] giving a bunch of audit techniques for additional investigations in the field. It comprised a starter study, subsequently, further examinations covering more extensive arrangement of rules, also as scholastic diaries, can be directed as the following stage in writing audit of Health recommender frameworks.

CHAPTER 3

SYSTEM DESIGN AND METHODOLOGY

3.1. System Design

3.1.1. DataSet Description

It contains a patient medication survey dataset containing ascribes like unique Id, drug name, condition(disease of patient), date , helpful count, audits and appraisals given by the patients.

- **Drug Name** – It is a categorical attribute which specifies the name of the drug prescribed by the practitioner.
- **Condition** - It is a categorical attribute which states that in which condition or disease the drug is prescribed by the practitioner.
- **Review** - It is the text review given by patient about the drug.
- **Rating** - It is the overall patient satisfaction score.
- **Date** - It is a date attribute which tells the date of entry of review.
- **Useful Count** - It is the number of patients that find the review useful

The Dataset contains around 215063 entries with 6 attributes.

Table : 3.1 Types of Attributes

Attributes	Type of Attributes
DrugName	Categorical
Condition	Categorical
Review	Text
Date	Date

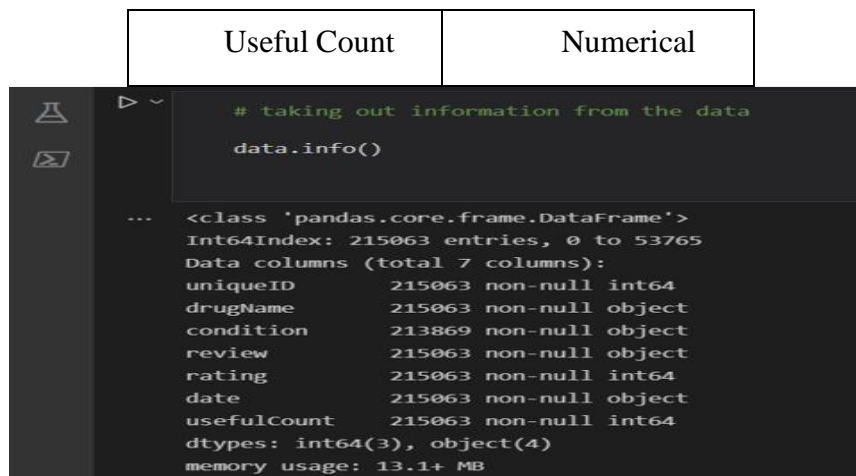


Figure : 3.1 Dataset Description

	uniqueID	drugName	condition
0	206461	Valsartan	Left Ventricular Dysfunction
1	95260	Guanfacine	ADHD
2	92703	Lybrel	Birth Control
3	138000	Ortho Evra	Birth Control
4	35696	Buprenorphine / naloxone	Opiate Dependence

Figure : 3.2 Dataset Description

review	rating	date	usefulCount
"It has no side effect, I take it in combinati...	9	2012-05-20	27
"My son is halfway through his fourth week of ...	8	2010-04-27	192
"I used to take another oral contraceptive, wh...	5	2009-12-14	17
"This is my first time using any form of birth...	8	2015-11-03	10
"Suboxone has completely turned my life around...	9	2016-11-27	37

Figure : 3.3 Dataset Description

3.1.2. Data Preprocessing

It comprises of dataset collection, reviewing, cleaning and dataset preprocessing. The real-world information is raw information which can be splitted and unorganized and is unable to be used for training of the model. So, information cleaning is used to clean information. it consists of null/missing values processing, correlation analysis and removing duplicate data.

- Data Exploration -
 - (i) Analyzing patient ids to check if a patient has written more than one review.
 - (ii) Find count of drugs for each condition by analyzing condition and quantity of drugs.
- Data Cleaning –
 - (i) Find the count of missing or empty fields for all the dimensions.

```
In [92]: # checking if the data contains any NULL values
         data.isnull().any()

Out[92]: uniqueID      False
         drugName      False
         condition     True
         review        False
         rating        False
         date          False
         usefulcount   False
         dtype: bool
```

Figure : 3.4 Checking Null Values

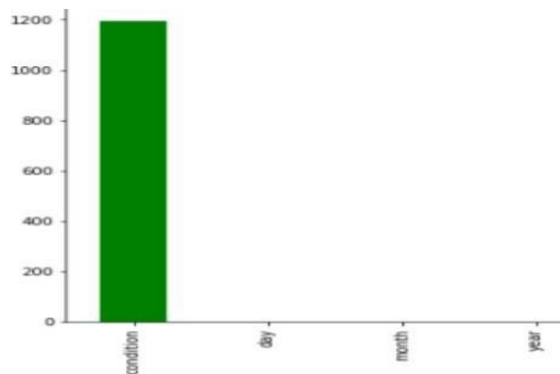


Figure : 3.5 Count of Null Values

- (ii) These none values can be removed, ignored or filled so the rows with missing values can be deleted, so as to efficiently train the data as the dataset huge.
- (iii) Remove the redundancy in the dataset to normalize the data
- (iv) Delete the rows with only drug as only one drug is unable to recommend the best one.
- (v) Words like don't need, never etc should be removed from stop words as they don't possess any specific results about the attitude of review. At last, remove the stop words to clean the reviews

3.1.3. Data Visualization

Data Visualization is the process to visualize the data and relationships among different attributes and how one attribute depends on the other

Some snapshots of relationships of attributes are shown below :-

- (i) Applied standard Data Cleaning techniques like really looking at invalid qualities, copy columns, eliminating anomalies, and text from lines in this examination. Hence, eliminated each of the 1200 invalid qualities lines in the conditions segment as displayed in fig – 3.1.5.

- (ii) Below fig – 3.1.6 shows top conditions with maximum number of drug available. After removing conditions that have no meaning , the dataset reduces to 212141 rows.

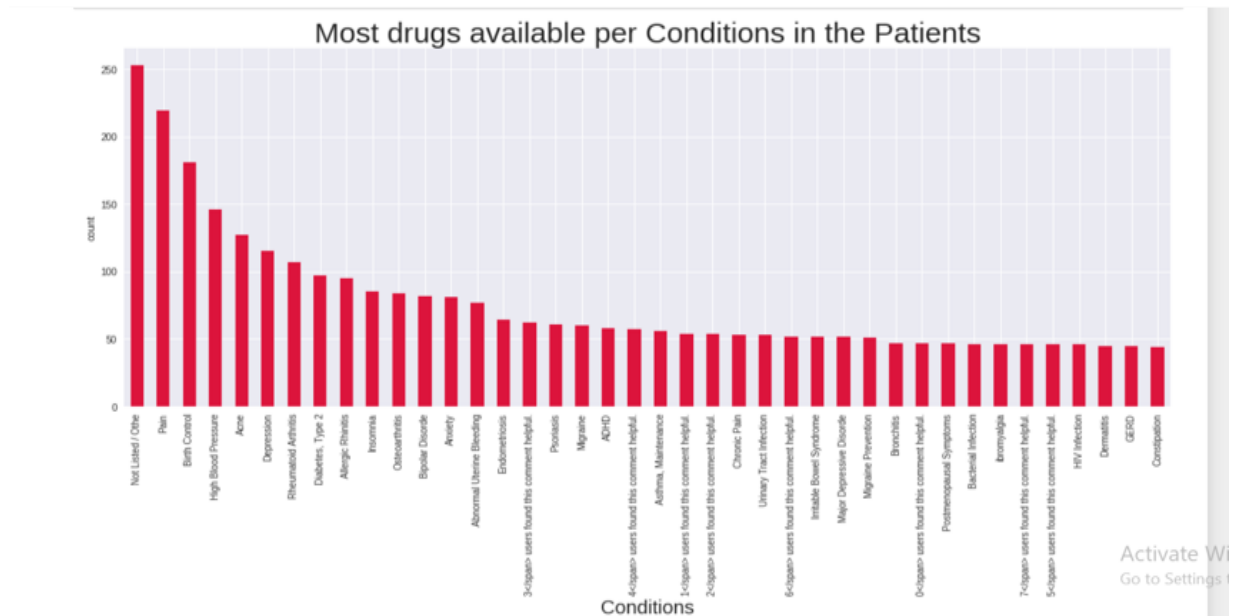


Figure : 3.6 Most Drugs present per Conditions

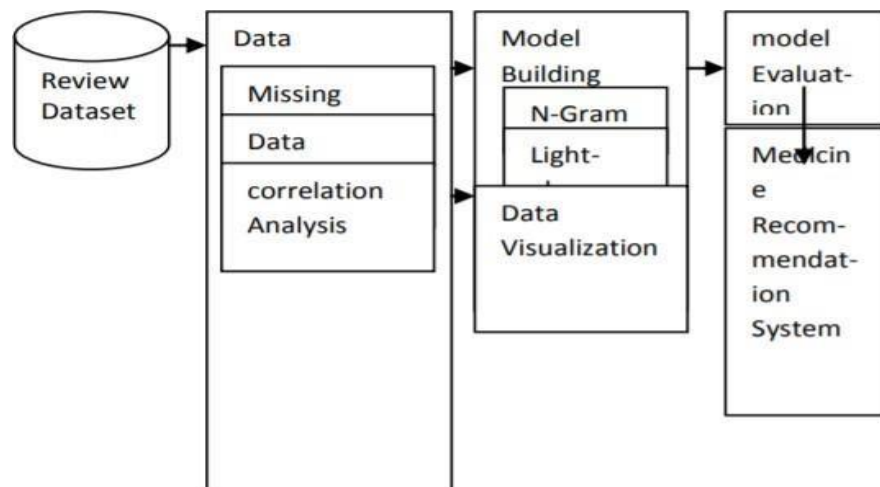


Figure : 3.11 Architecture of Drugs Recommendation System

3.1.5. Data Flow Diagram

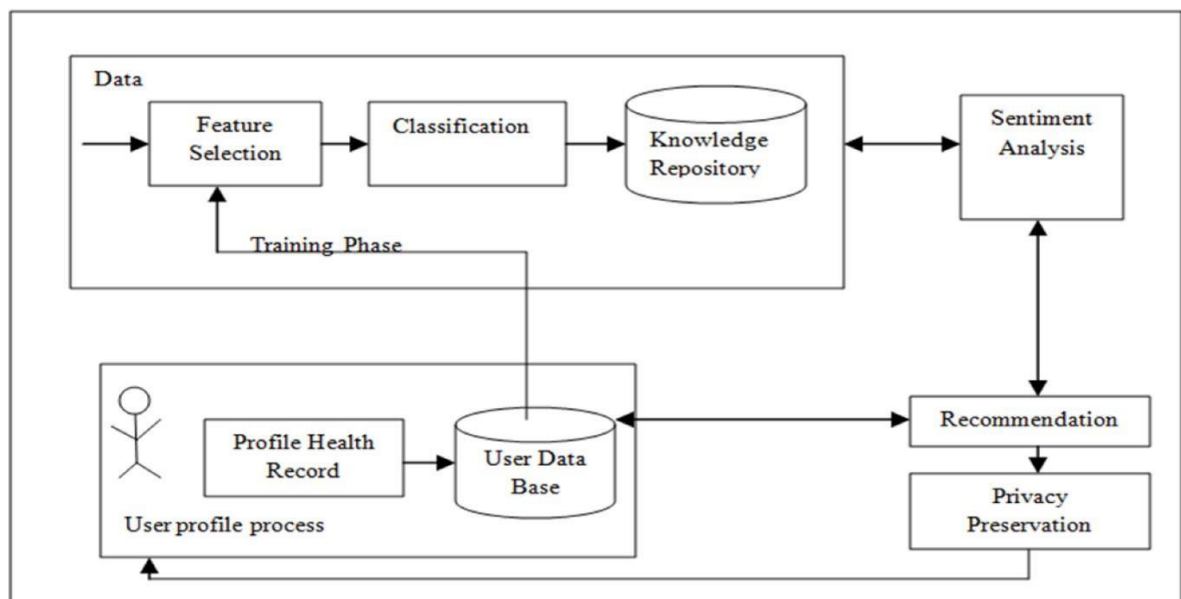


Figure : 3.12 Data Flow Diagram

3.2. Algorithm

3.2.1. Sentiment Analysis

In deep learning , Sentiment Analysis is used in the field of text mining. In Sentiment Analysis , predefined labels positive or negative is given to text document.

Sentiment analysis is a method of natural language processing that finds the emotional tone behind a body of text. This is a popular way for organizations to determine and categorize opinions. It includes the usage of data mining, machine learning and artificial intelligence to mine text for sentiment and subjective information.

3.2.2. N-gram model

N-grams are a chunk of subsequent words, by studying these sequences we can efficiently understand the context in which a particular word is used.

Example – the word ‘book’ can be used in different contexts like – to ‘book’ tickets, read the ‘book’ . Here the word book is used as the verb in the first phrase while as a noun in the latter. So in order to efficiently understand the context of the word , N-grams look at the after the word and before the word and then determine if the word is used as a noun or verb in the sentence or in other context.

N in N-grams denotes the number of words machine will look at before and after the target word. Ex- This ‘book’, A ‘book’, Your ‘book’ are all examples of bi-grams where before word ‘book’ is a noun. Bi-grams are the two pairs of words to look at before and after the target word while sliding over the words. the context can be extended by going to tri-grams which means looking at three pairs of words before and after the target word.

Feeling investigation helps assessing the exhibition of items or administrations from client created substance. Vocabulary based opinion investigation approaches are liked

over learning based ones when preparing information isn't satisfactory. Existing vocabularies contain just unigrams alongside their feeling scores. It is seen that opinion n-grams shaped by joining unigrams with intensifiers or invalidations show further developed outcomes. Such opinion n-gram vocabularies are not openly accessible. This paper presents a procedure to make such a vocabulary called Senti-N-Gram. Proposed rule-based methodology removes the n-grams opinion scores from an irregular corpus containing item surveys and relating numeric rating in 10-point scale. The scores from this computerized system are contrasted and that of the human annotators utilizing t-test and viewed as genuinely same

N-grams can also be used to capture words in positive or negative context or vice-versa. Example – ‘the staff were not friendly,terrible really’. In this sentence ‘Not friendly’ and ‘friendly terrible’ is enough context to elucidate that the word ‘friendly’ is used in a negative context. In isolation the word ‘friendly’ is positive in when we are looking forward ‘terrible’ and backward ‘not’ which cancels out the positive meaning of the word.

CHAPTER 4

IMPLEMENTATION AND RESULTS

4.1. Software and Hardware Requirements

4.1.1. Software Requirements

- Jupyter Notebook
- Anaconda Navigator/Google Colab
- Python libraries

4.1.2. Hardware Requirements

- Any laptop with min 4GB Ram, i5 intel/Ryzen-5 processor and nvidia/integrated graphic card.

4.2. Assumptions and dependencies

4.2.1. Assumptions

- Know about Python Language and its libraries
- Know about Machine Learning Algos used

4.2.2. Dependencies

- Python version – 3.9

4.3. Implementation Details

4.3.1. Snapshots Of Interfaces

```
import numpy as np
import pandas as pd

import matplotlib.pyplot as plt
import seaborn as sns
```

Figure : 4.1 Importing packages

```
# reading the data

train = pd.read_csv('drive/My Drive/Projects/practice/Drugs/drugsComTrain_raw.csv')
test = pd.read_csv('drive/My Drive/Projects/practice/Drugs/drugsComTest_raw.csv')

# getting the shapes
print("Shape of train :", train.shape)
print("Shape of test :", test.shape)
```

```
Shape of train : (161297, 7)
Shape of test : (53766, 7)
```

Figure : 4.2 Size of Training and Testing Data

```
# checking the head of the train

train.head()
```

	uniqueID	drugName	condition	review	rating	date	usefulCount
0	206461	Valsartan	Left Ventricular Dysfunction	"It has no side effect, I take it in combinati...	9	20-May-12	27
1	95260	Guanfacine	ADHD	"My son is halfway through his fourth week of ...	8	27-Apr-10	192
2	92703	Lybrel	Birth Control	"I used to take another oral contraceptive, wh...	5	14-Dec-09	17
3	138000	Ortho Evra	Birth Control	"This is my first time using any form of birth...	8	3-Nov-15	10
4	35696	Buprenorphine / naloxone	Opiate Dependence	"Suboxone has completely turned my life around...	9	27-Nov-16	37

Figure : 4.3 First five rows of Dataset

```
# checking the head of the test
```

```
test.head()
```

	uniqueID	drugName	condition	review	rating	date	usefulCount
0	163740	Mirtazapine	Depression	"I've tried a few antidepressants over th...	10	28-Feb-12	22
1	206473	Mesalamine	Crohn's Disease, Maintenance	"My son has Crohn's disease and has done ...	8	17-May-09	17
2	159672	Bactrim	Urinary Tract Infection	"Quick reduction of symptoms"	9	29-Sep-17	3
3	39293	Contrave	Weight Loss	"Contrave combines drugs that were used for al...	9	5-Mar-17	35
4	97768	Cyclafem 1 / 35	Birth Control	"I have been on this birth control for one cyc...	9	22-Oct-15	4

Figure: 4.4 First five rows of Dataset

```
# as both the dataset contains same columns we can combine them for better analysis
```

```
data = pd.concat([train, test])
```

```
# checking the shape
```

```
data.shape
```

```
(215063, 7)
```

Figure : 4.5 Size of Dataset

```
# checing the sample of new dataset
```

```
data.sample(5)
```

	uniqueID	drugName	condition	review	rating	date	usefulCount
112607	170298	Quetiapine	Generalized Anxiety Disorde	"\n\n please tell the ones who is sufferin...	10	25-Jul-16	45
141096	136119	Acamprosate	Alcohol Dependence	"I was a two bottle plus red wine drinker ever...	9	28-Jan-14	97
35338	170518	Quetiapine	Bipolar Disorde	"I have been taking seroquel for a little over...	10	8-Oct-15	16
26858	209685	Lupron Depot	Endometriosis	"I started this about 6 -7 weeks ago after lap...	8	11-Oct-16	10
121226	122848	Linaclotide	Constipation, Chronic	"I've had chronic constipation for as lon...	9	23-Mar-14	113

Figure : 4.6 Dataset Description

```
# describing the data
data.describe()
```

	uniqueID	rating	usefulCount
count	215063.000000	215063.000000	215063.000000
mean	116039.364814	6.990008	28.001004
std	67007.913366	3.275554	36.346069
min	0.000000	1.000000	0.000000
25%	58115.500000	5.000000	6.000000
50%	115867.000000	8.000000	16.000000
75%	173963.500000	10.000000	36.000000
max	232291.000000	10.000000	1291.000000

Figure: 4.7 Dataset Description

```
# most common condition

from wordcloud import WordCloud
from wordcloud import STOPWORDS

stopwords = set(STOPWORDS)

wordcloud = WordCloud(background_color = 'lightblue', stopwords = stopwords, max_words = 100, width = 1200, height = 800).generate(str(data['condition']))

plt.rcParams['figure.figsize'] = (15, 15)
plt.title('Most Common Conditions among the Patients', fontsize = 30)
print(wordcloud)
plt.axis('off')
plt.imshow(wordcloud)
plt.show()
```

<wordcloud.wordcloud.WordCloud object at 0x7fd3241c5278>

Figure : 4.8 Most Common Conditions


```

# most popular drugs

from wordcloud import WordCloud
from wordcloud import STOPWORDS

stopwords = set(STOPWORDS)

wordcloud = WordCloud(background_color = 'white', stopwords = stopwords, width = 1200, height = 800).generate(str(data['drugName']))

plt.rcParams['figure.figsize'] = (15, 15)
plt.title('Most Popular Drugs', fontsize = 30)
print(wordcloud)
plt.axis('off')
plt.imshow(wordcloud)
plt.show()

<wordcloud.wordcloud.WordCloud object at 0x7fd3215c2160>

```

Figure : 4.9 Most Popular Drugs

```

# checking the most popular drugs per conditions

data.groupby(['drugName'])['condition'].nunique().sort_values(ascending = False).head(40).plot.bar(figsize = (19, 7), color = 'violet')
plt.title('Drugs which can be used for many Conditions in the Patients', fontsize = 30)
plt.xlabel('Drug Names', fontsize = 20)
plt.ylabel('count')
plt.show()

```

Figure : 4.10 Grouping drugs per conditions

```

# checking the most popular drugs per conditions

data.groupby(['condition'])['drugName'].nunique().sort_values(ascending = False).head(40).plot.bar(figsize = (19, 7), color = 'violet')
plt.title('Most drugs available per Conditions in the Patients', fontsize = 30)
plt.xlabel('Conditions', fontsize = 20)
plt.ylabel('count')
plt.show()

```

Figure : 4.11 Popular Drugs per Condition

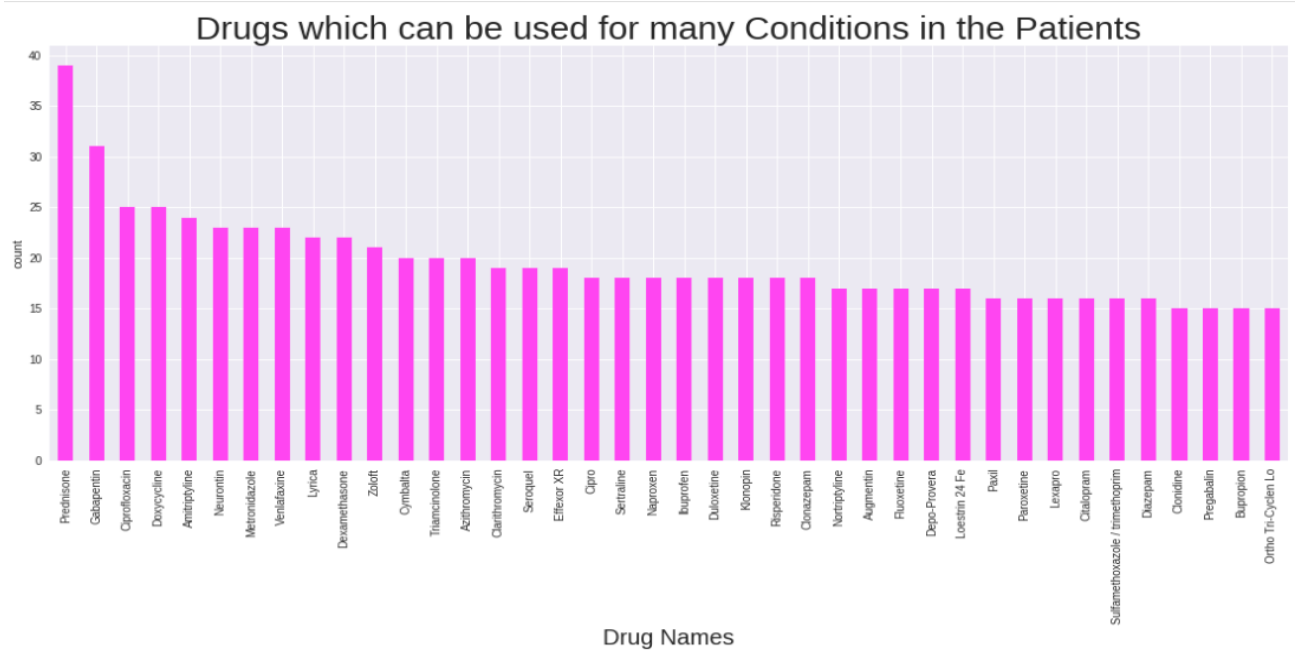


Figure : 4.12 Popular Drugs

```
# checking the different types of conditions patients

data['condition'].value_counts().head(40).plot.bar(figsize = (19, 7), color = 'purple')
plt.title('Most Common Conditions in the Patients', fontsize = 30)
plt.xlabel('Conditions', fontsize = 20)
plt.ylabel('count')
plt.show()
```

Figure : 4.13 Types of Condition Patients

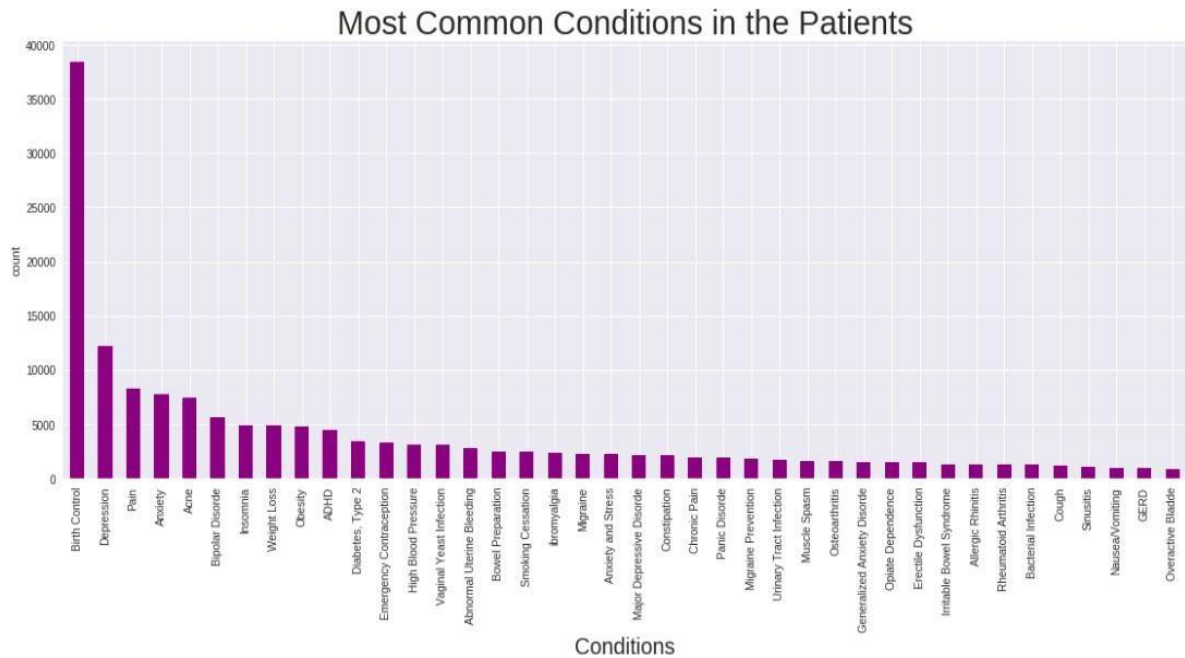


Figure : 4.14 Common Conditions for Patients

```
# let's see the words cloud for the reviews

# most popular drugs

from wordcloud import WordCloud
from wordcloud import STOPWORDS

stopwords = set(STOPWORDS)

wordcloud = WordCloud(background_color = 'yellow', stopwords = stopwords, width = 1200, height = 800).generate(str(data['review']))

plt.rcParams['figure.figsize'] = (15, 15)
plt.title('Most Popular Drugs', fontsize = 30)
print(wordcloud)
plt.axis('off')
plt.imshow(wordcloud)
plt.show()
```

Figure : 4.15 Word Cloud for Reviews

```
# let's read some reviews

train['review'][2]
```

"I used to take another oral contraceptive, which had 21 pill cycle, and was very happy- very light periods, max 5 days, no other side effects. But it contained hormone gestodene, which is not available in US, so I switched to Lybrel, because the ingredients are similar. When my other pills ended, I started Lybrel immediately, on my first day of period, as the instructions said. And the period lasted for two weeks. When taking the second pack- same two weeks. And now, with third pack things got even worse- my third period lasted for two weeks and now it's the end of the third week- I still have daily brown discharge.\n\nThe positive side is that I didn't have any other side effects. The idea of being period free was so tempting... Alas."

Figure : 4.16 Examples of Reviews

```
data['rating'].value_counts()
```

```
10    68005
9     36708
1     28918
8     25046
7     12547
5     10723
2      9265
3      8718
6      8462
4      6671
Name: rating, dtype: int64
```

Figure : 4.17 Count of Ratings per score

```
# making a donut chart to represent share of each ratings
size = [68005, 46901, 36708, 25046, 12547, 10723, 8462, 6671]
colors = ['pink', 'cyan', 'maroon', 'magenta', 'orange', 'lightblue', 'lightgreen', 'yellow']
labels = "10", "1", "9", "8", "7", "5", "6", "4"

my_circle = plt.Circle((0, 0), 0.7, color = 'white')

plt.rcParams['figure.figsize'] = (10, 10)
plt.pie(size, colors = colors, labels = labels, autopct = '%.2f%%')
plt.axis('off')
plt.title('A Pie Chart Representing the Share of Ratings', fontsize = 30)
p = plt.gcf()
plt.gca().add_artist(my_circle)
plt.legend()
plt.show()
```

Figure : 4.18 Data Visualization

```
# feature engineering
# let's make a new column review sentiment

data.loc[(data['rating'] >= 5), 'Review_Sentiment'] = 1
data.loc[(data['rating'] < 5), 'Review_Sentiment'] = 0

data['Review_Sentiment'].value_counts()
```

```
1.0    161491
0.0     53572
Name: Review_Sentiment, dtype: int64
```

Figure : 4.19 Feature Engineering

```
# a pie chart to represent the sentiments of the patients

size = [161491, 53572]
colors = ['yellow', 'skyblue']
labels = "Positive Sentiment", "Negative Sentiment"
explode = [0, 0.1]

plt.rcParams['figure.figsize'] = (10, 10)
plt.pie(size, colors = colors, labels = labels, explode = explode, autopct = '%.2f%%')
plt.axis('off')
plt.title('A Pie Chart Representing the Sentiments of Patients', fontsize = 30)
plt.legend()
plt.show()
```

Figure : 4.20 pie chart

A Pie Chart Representing the Sentiments of Patients

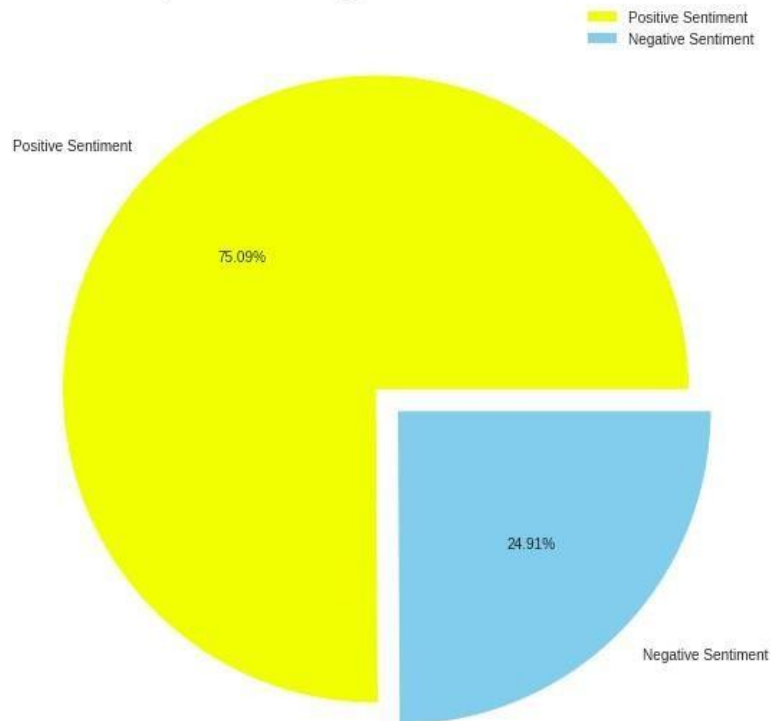


Figure : 4.21 Pie Chart and Code of Sentiments of Patients


```

# converting the date into datetime format
data['date'] = pd.to_datetime(data['date'], errors = 'coerce')

# now extracting year from date
data['Year'] = data['date'].dt.year

# extracting the month from the date
data['month'] = data['date'].dt.month

# extracting the days from the date
data['day'] = data['date'].dt.day

# looking at the no. of reviews in each of the year

plt.rcParams['figure.figsize'] = (19, 8)
sns.countplot(data['Year'], palette = 'dark')
plt.title('The No. of Reviews in each year', fontsize = 30)
plt.xlabel('Year', fontsize = 15)
plt.ylabel('Count of Reviews', fontsize = 15)
plt.show()

```

Figure : 4.25 Converting dataset to datetime format

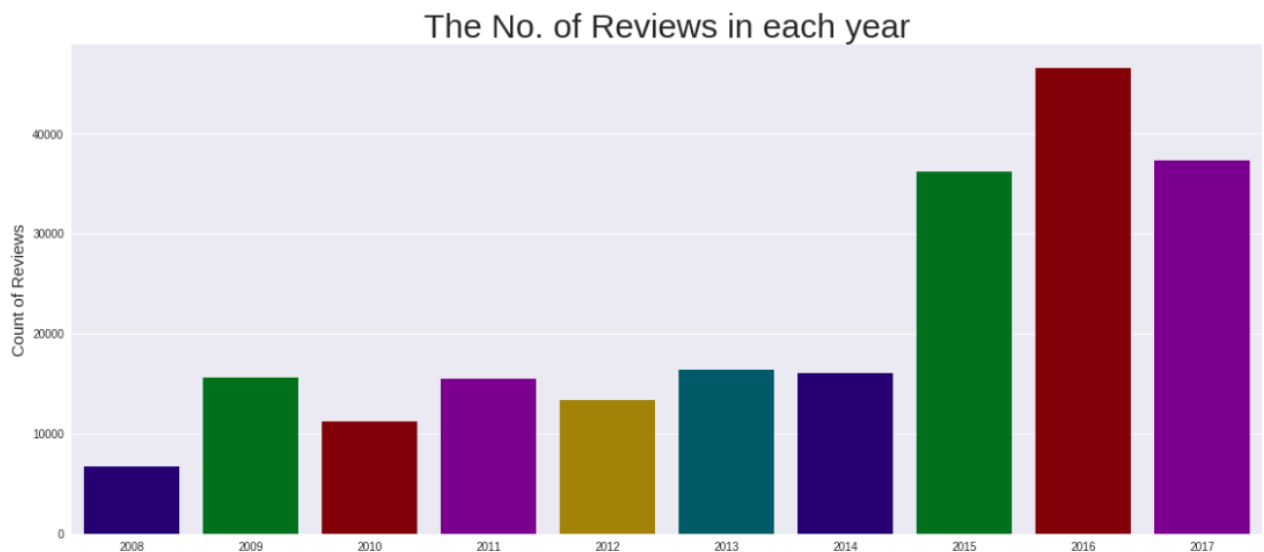


Figure : 4.26 Count of Reviews Each year


```
# looking at the no. of ratings in each of the year
```

```
plt.rcParams['figure.figsize'] = (19, 8)  
sns.boxplot(x = data['Year'], y = data['rating'], palette = 'dark')  
plt.title('The Distribution of Ratings in each Year', fontsize = 30)  
plt.xlabel('Year', fontsize = 15)  
plt.ylabel('Count of Reviews', fontsize = 15)  
plt.show()
```

/usr/local/lib/python3.6/dist-packages/seaborn/categorical.py:454: FutureWarning: remove_na is deprecated and is a private function. Do not use.

```
box_data = remove_na(group_data)
```

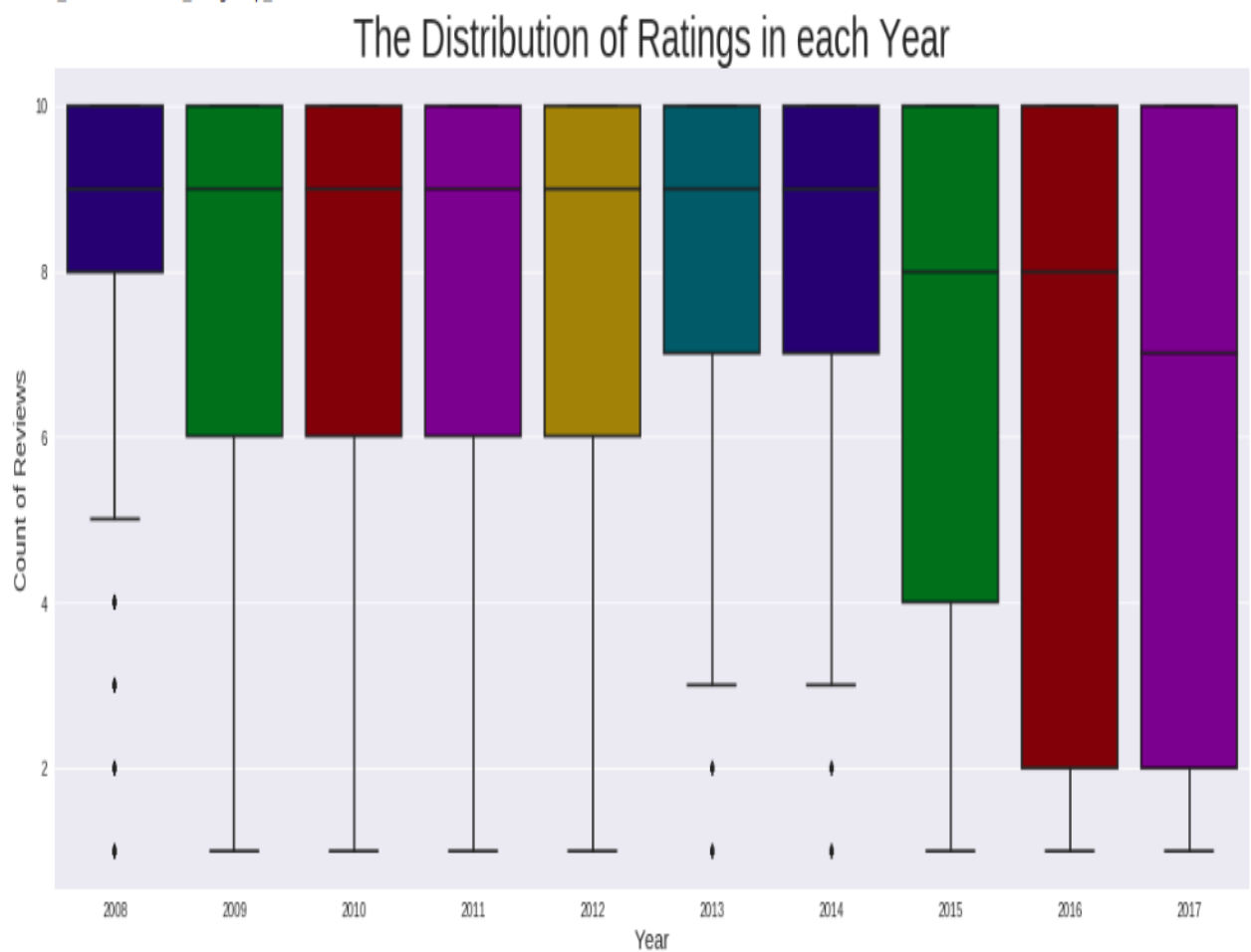


Figure : 4.27 Distribution of Ratings per year

```
# looking at the no. of ratings in each of the year
```

```
plt.rcParams['figure.figsize'] = (19, 8)  
sns.violinplot(x = data['Year'], y = data['Review Sentiment'])  
plt.title('The Distribution of Sentiments in each Year', fontsize = 30)  
plt.xlabel('Year', fontsize = 15)  
plt.ylabel('Variation of Sentimens', fontsize = 15)  
plt.show()
```

```
/usr/local/lib/python3.6/dist-packages/seaborn/categorical.py:588: FutureWarning: remove_na is deprecated and is a private function. Do not use.  
kde_data = remove_na(group_data)
```

```
/usr/local/lib/python3.6/dist-packages/seaborn/categorical.py:816: FutureWarning: remove_na is deprecated and is a private function. Do not use.  
violin_data = remove_na(group_data)
```

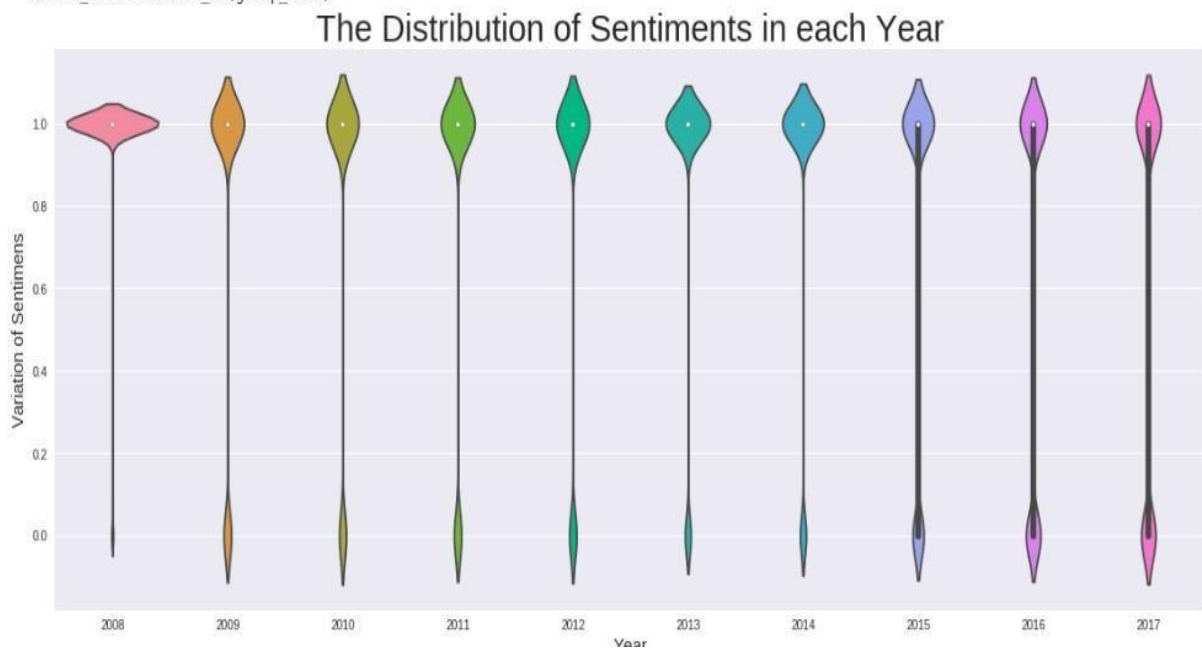


Figure : 4.28 Distribution of Sentiments per year

```
# looking at the no. of reviews in each of the months
plt.rcParams['figure.figsize'] = (19, 8)
sns.countplot(data['month'], palette = 'pastel')
plt.title('The No. of Reviews in each Month', fontsize = 30)
plt.xlabel('Months', fontsize = 15)
plt.ylabel('Ratings', fontsize = 15)
plt.show()
```

/usr/local/lib/python3.6/dist-packages/seaborn/categorical.py:1428: FutureWarning: remove_na is deprecated and is a private function. Do not use.
stat_data = remove_na(group_data)

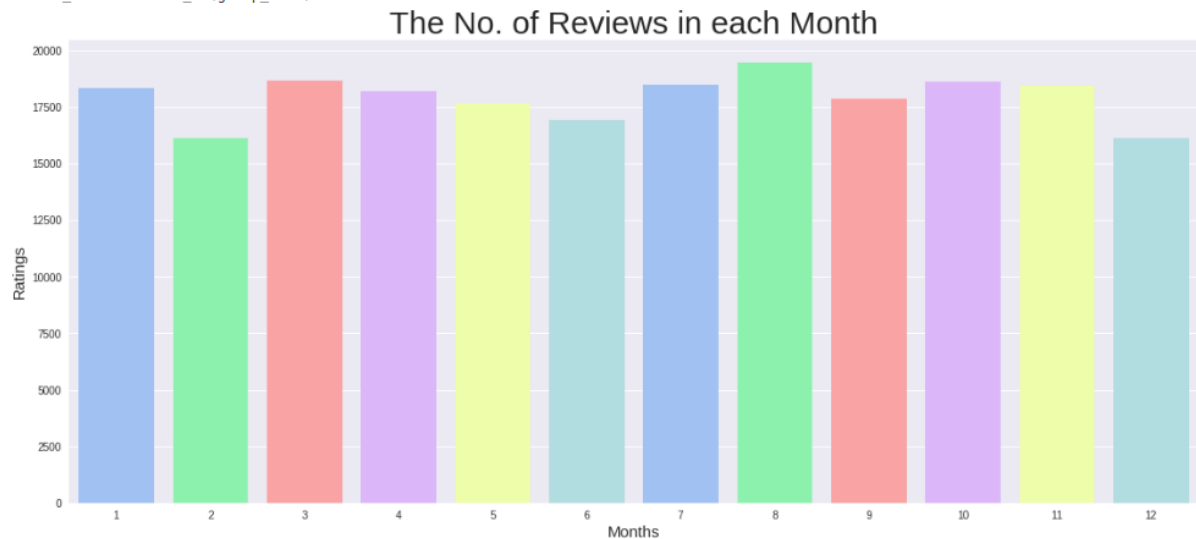


Figure : 4.29 Count of Reviews per month

```
# looking at the no. of reviews in each of the day
plt.rcParams['figure.figsize'] = (19, 8)
sns.countplot(data['day'], palette = 'colorblind')
plt.title('The No. of Reviews in each day', fontsize = 30)
plt.xlabel('Days', fontsize = 15)
plt.ylabel('Count of Reviews', fontsize = 15)
plt.show()
```

/usr/local/lib/python3.6/dist-packages/seaborn/categorical.py:1428: FutureWarning: remove_na is deprecated and is a private function. Do not use.
stat_data = remove_na(group_data)

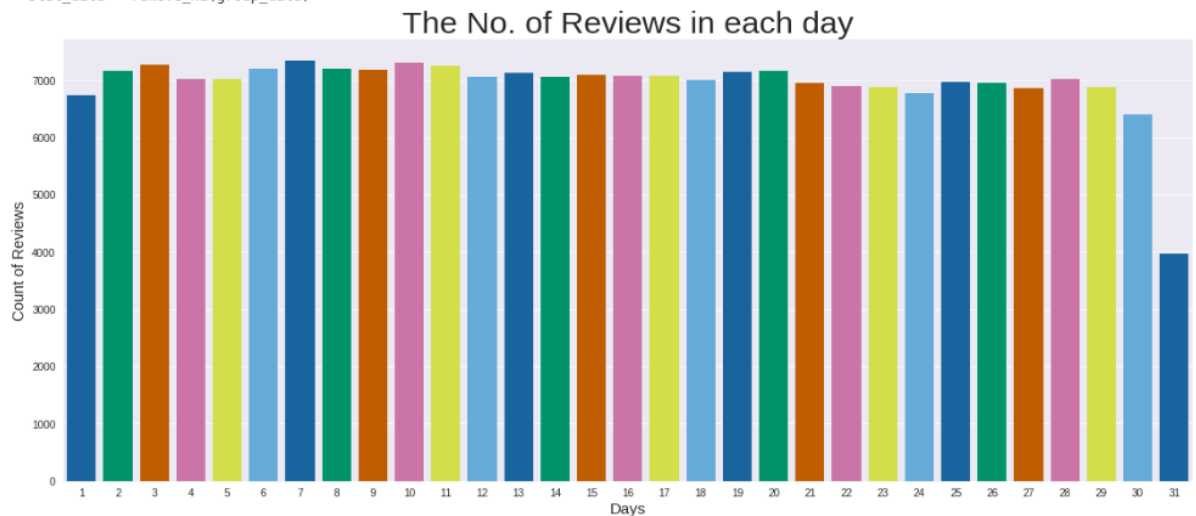


Figure : 4.30 Count of Reviews per day

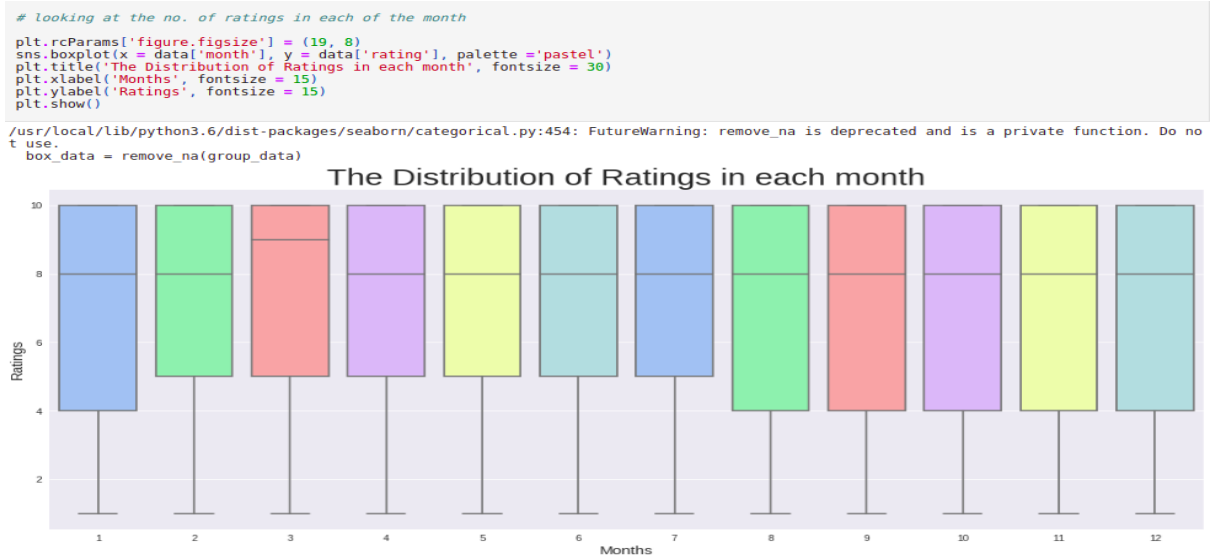


Figure : 4.31 Distribution of Ratings per month

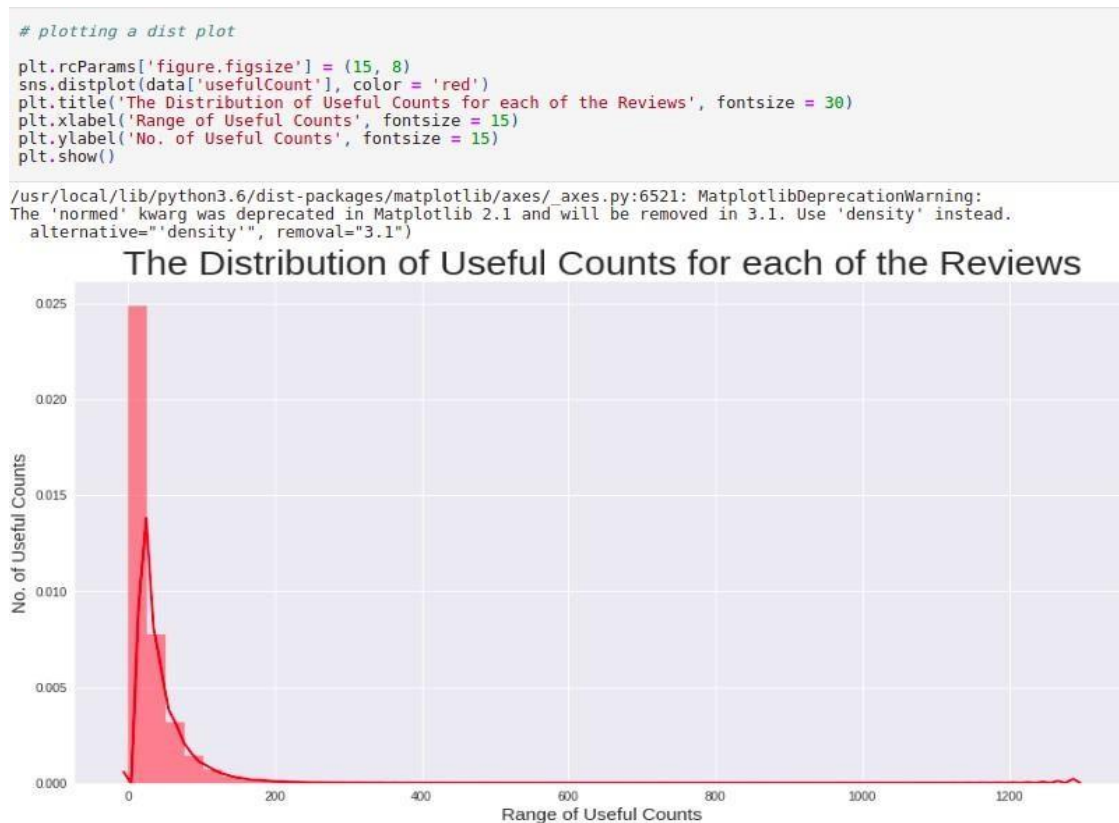


Figure : 4.32 Distribution of Useful Counts per review

```
# looking at the no. of ratings in each of the month

plt.rcParams['figure.figsize'] = (19, 8)
sns.violinplot(x = data['month'], y = data['rating'], palette = 'pastel')
plt.title('The Distribution of Sentiments in each month', fontsize = 30)
plt.xlabel('Months', fontsize = 15)
plt.ylabel('Sentiments', fontsize = 15)
plt.show()

/usr/local/lib/python3.6/dist-packages/seaborn/categorical.py:588: FutureWarning: remove_na is deprecated and is a private function. Do not use.
  kde_data = remove_na(group_data)
/usr/local/lib/python3.6/dist-packages/seaborn/categorical.py:816: FutureWarning: remove_na is deprecated and is a private function. Do not use.
  violin_data = remove_na(group_data)
```

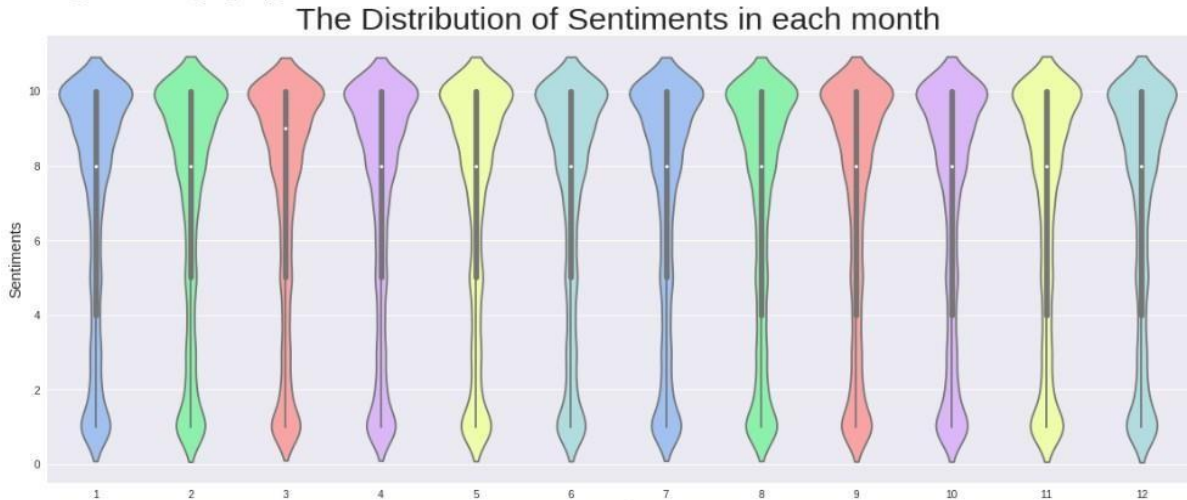


Figure : 4.33 Distribution of Sentiments per month

```
# plotting a stacked bar to see in which year what were the sentiments

df = pd.crosstab(data['Year'], data['Review Sentiment'])
df.div(df.sum(1).astype(float), axis = 0).plot(kind = 'bar', stacked = True, figsize = (19, 8), color = ['red', 'orange', 'pink'])
plt.title('The Distribution of Sentiments for each of the Reviews Year-wise', fontsize = 30)
plt.xlabel('Year', fontsize = 15)
plt.show()
```

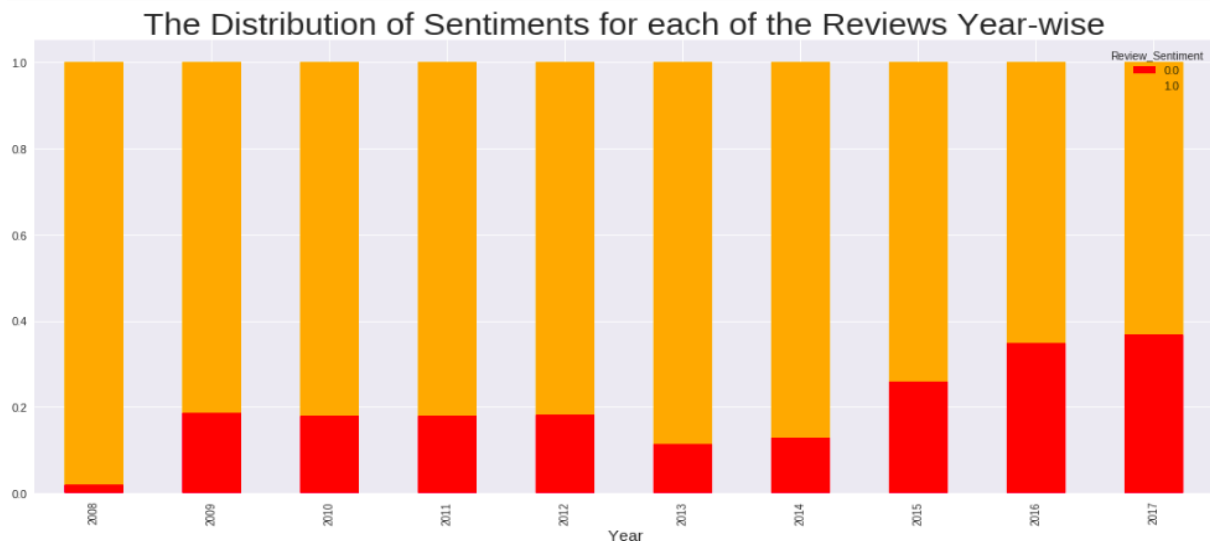


Figure : 4.34 Distribution of Sentiments per review yearwise

```
# plotting a stacked bar to see in which year what were the sentiments
df = pd.crosstab(data['month'], data['Review_Sentiment'])
df.div(df.sum(1).astype(float), axis = 0).plot(kind = 'bar', stacked = True, figsize = (19, 8), color = ['darkblue', 'purple', 'violet'])
plt.title('The Distribution of Sentiments for each of the Reviews month-wise', fontsize = 30)
plt.xlabel('Month', fontsize = 15)
plt.show()
```

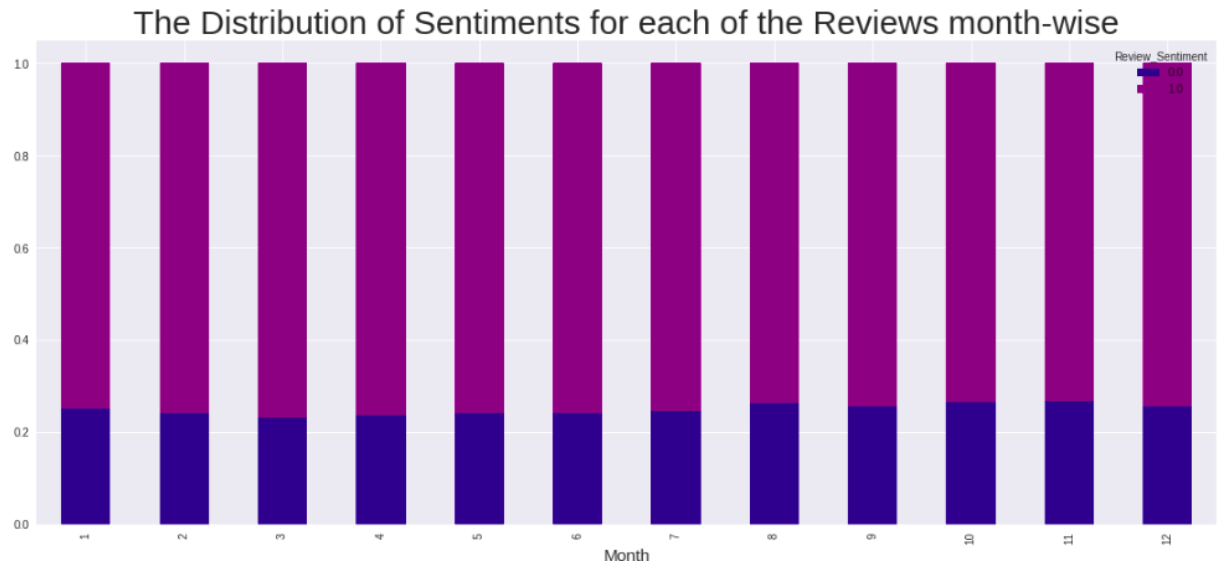


Figure : 4.35 Distribution of Sentiments per review month-wise

```
# plotting a stacked bar to see in which year what were the sentiments
df = pd.crosstab(data['day'], data['Review_Sentiment'])
df.div(df.sum(1).astype(float), axis = 0).plot(kind = 'bar', stacked = True, figsize = (19, 8), color = ['lightblue', 'yellow', 'lightgreen'])
plt.title('The Distribution of Sentiments for each of the Reviews day-wise', fontsize = 30)
plt.xlabel('Days', fontsize = 15)
plt.legend(loc = 2)
plt.show()
```

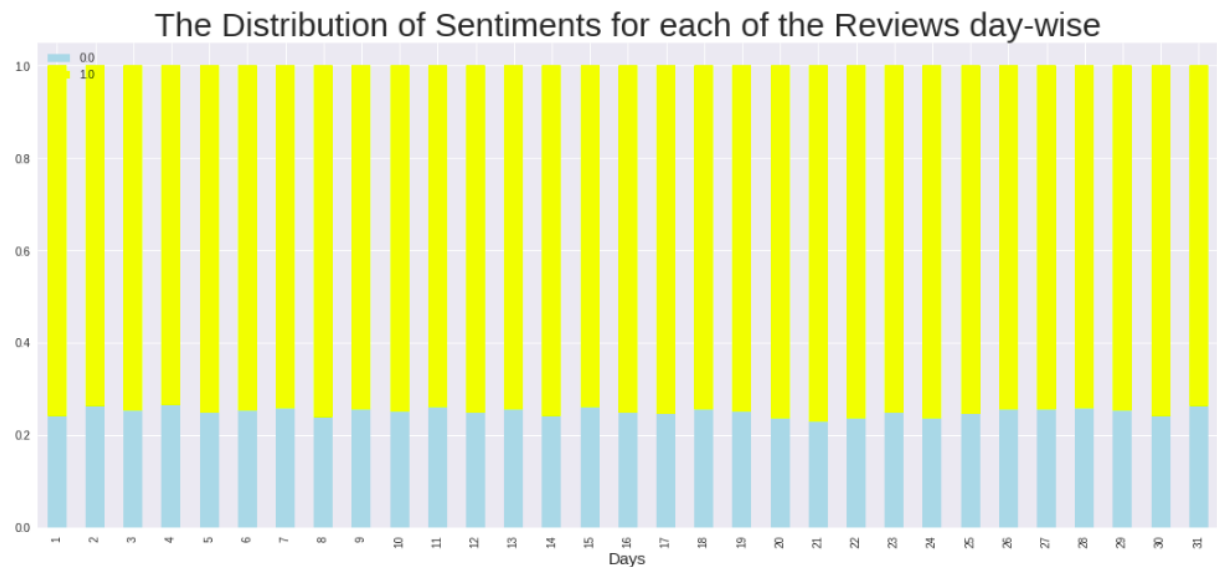


Figure : 4.36 Distribution of Sentiments per review day-wise

```

data['condition'].isnull().sum()

1194

# we will delete the rows so that the data does not overfits
data = data.dropna(axis = 0)

# checking the new shape of the data
data.shape

(213869, 11)

# importing the important libraries
import re
from bs4 import BeautifulSoup
import nltk
nltk.download('stopwords')
from nltk.corpus import stopwords
from nltk.stem.snowball import SnowballStemmer
from nltk.stem.porter import PorterStemmer

[nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data] Package stopwords is already up-to-date!

# removing some stopwords from the list of stopwords as they are important for drug recommendation
stops = set(stopwords.words('english'))

not_stop = ["aren't", "couldn't", "didn't", "doesn't", "don't", "hadn't", "hasn't", "haven't", "isn't", "mightn't",
            "mustn't", "needn't", "no", "nor", "not", "shan't", "shouldn't", "wasn't", "weren't", "wouldn't"]
for i in not_stop:
    stops.remove(i)

```

Figure : 4.37 New size of data and Importing packages

CHAPTER 5

CONCLUSION

5.1. Performance Evaluation

Baseline models: Evaluation of classification models using confusion matrix and accuracy as shown:-

Table : 5.1

Vectorizer	Model	Accuracy
TF-IDF	Linear Regression	78.2%
	Naive Bayes	75.2%
	Random Forest	83.1%

Deep learning models: Evaluation of deep learning models using metrics like loss and model accuracy.

Table : 5.2

Model	Loss	Accuracy
Deep NN (Sequential Model)	0.17	89.4%
Deep Learning model using N-gram	0.39	80.1%

Deep Neural Network Sequential model gave the best accuracy for this problem of 89.4%.

5.2. Comparison with existing State-of-the-Art Technologies

Reviews are becoming daily part of life ex- E-commerce, Restaurants etc. They Form a major trust opinion among masses. Inspired from this, Sentiment Analysis on the drug reviews was studied using various types of Machine Learning Algorithms like SVM, Logistic regression, naïve bayes and classifiers such as Decision Trees were applied.

With Linear SVC outperformed all(91%) accuracy,Decision Tree(88%),Word2vec(78%).

We will be applying CNN – Sentiment Analysis,N-gram model and light-gbm booster for the training purpose.

5.3. Future Directions

- Work efficiency of a recommendation system can be increased by increasing dimensions of dataset or by taking more detailed data like age of person, demographic information during Training Phase
- We can also add attributes for the quality and efficiency of drugs like brand of drug and chemical contents present in it to improve the recommended medicines.
- Future work includes examination of various oversampling methods utilizing various upsides of n-grams and advancement of calculations to work on the presentation of the recommender framework.

References

1. Isinkaye, F.O.; Folajimi, Y.O.; Ojokoh, B.A. Recommender systems: Principles, methods and evaluation. *Egypt. Inform. J.* 2015, 16, 261–273, ISSN 1110-8665..
2. Leilei Sun, Chuanren Liu, Chonghui Guo, Hui Xiong, and Yanming Xie. 2016. Data-driven Automatic Treatment Regimen Development and Recommendation. In *Procedures of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'16)*. ACM, NewYork, NY, USA, 1865–1874. DOI: <https://doi.org/10.1145/2939672.2939866>.
3. Subhash C. Pandey, —'Data Science methodologies for medical data': 'A Review',—IEEE, 2016.
4. Shimada K, Takada H, Mitsuyama S, et al. Drug-recommendation system for patients with infectious diseases. *AMIA Annu Symp Proc.* 2005;2005:1112.
5. T. N. Tekade and M. Emmanuel, "Probabilistic aspect mining approach for understanding and assessment of medication surveys," 2016 International Conference on Signal Processing, Communication, Power and Embedded System (SCOPEs), Paralakhemundi, 2016, pp. 1471-1476, doi: 10.1109/SCOPEs.2016.7955684.
6. Doulaverakis, C., Nikolaidis, G., Kleontas, A. et al. GalenOWL: Ontology-based drug recommendations discovery. *J Biomed Semant* 3, 14 (2012). <https://doi.org/10.1186/2041-1480-3-14>.
7. Mu, R.; Zeng, X.; Han, L. A Survey of Recommender Methodologies related to Deep Learning. *IEEE Access* 2018, 6, 69009–69022
8. V. Goel, A. K. Gupta and N. Kumar, "Sentiment Analysis of Multilingual Twitter Data using Natural Language Processing," 2018 8th International Conference on Communication.p- Systems and Network Technologies (CSNT), Bhopal, India, 2018, pp. 208-212, doi: 10.1109/CSNT.2018.8820254.
9. Y. Bao and X. Jiang, "A canny medication recommender structure," 2016 IEEE eleventh Conference on Industrial Electronics and Applications (ICIEA), Hefei, 2016, pp. 1383-1388, doi: 10.1109/ICIEA.2016.7603801
10. Telemedicine, <https://www.mohfw.gov.in/pdf/Telemedicine.pdf>
11. Sentiment Analysis, <https://towardsdatascience.com/leveraging-n-grams-to-extract-context-from-text-bdc576b47049>

12. Pew Research, <https://www.pewresearch.org/internet/2013/01/15/health-online-2013/>
13. Popescu AM, Etzioni O (2005) Extracting product features and opinions from reviews. Procedures of the gathering on human language innovation and experimental techniques in regular language handling. Relationship for Computational Linguistics, pp: 339-346
14. Sentiment Analysis using N-grams, <https://www.knime.com/blog/sentiment-analysis-with-n-grams>
15. Calero Valdez, A.; Ziefle, M.; Verbert, K.; Felfernig, A.; Holzinger, A. Recommender Systems for Health Informatics. In AI for Health Informatics; Holzinger, A., Ed.; Lecture Notes in Computer Science LNCS 9605; Springer: Cham, Switzerland, 2016
16. L. Fernandez-luque, R. Karlsen and L.K. Vognild, "Troubles and Opportunities of including Recommender Systems for Personalized Health Education" Studies. Prosperity Technology, 150(903), pp. 903-7, 2009
17. B.Kitchenham, "Procedures for Performing Systematic Reviews", Technical Report TR/SE-0401, Keele University, NICTA, 2004
18. F. Ricci, L. Rokach, B. Shapira and P. B. Kantor, Introduction to Recommender Systems Handbook, pp. 257-297, Springer, Berlin, 2011.
19. D. H. park, H. K. Kim, I. Y. Choi and J. K. Kim, J. Expert System Applications, 39 (11), pp.10059-10072, 2012

Appendix - I

NLP and Deep Learning-based Model for Recommending Drugs using Patient Reviews

Tauheed Shahid , Suraj Singh, Shatmanyu Gupta, Shanu Sharma

tauheed.18bcs1108@abes.ac.in, shatmanyu.18bcs1030@abes.ac.in, suraj.18bcs1028@abes.ac.in

Department of Computer Science and Engineering, ABES Engineering College, Ghaziabad

Abstract: Nowadays technological advancement can be seen in the medical field starting from medical devices, data collection, and analysis to diagnosis and treatment recommendations of diseases. Drug and treatment recommendation is one of the most popular applications, which is now observed and used by everyone in this digital era. These type of recommendation systems usually require a huge set of data from the patients and an efficient Machine learning-based model to conclude significant insights that can help in the prediction of the best possible medications for a particular disease. The health-related recommendation systems can be proved as a significant tool in the healthcare sector for speeding up various decision-making processes such as health insurance, clinical pathway-based treatment methods, and assisting doctors by recommending drugs using a patient's health profile. Usually, people use social media to do research about their health conditions, thus a lot of data is nowadays available on social networks which can be used to generate different health-related recommendation systems. In this paper, a drug recommendation system is proposed which takes the patient review data as input and performs sentiment analysis on it to find the best drug for disease by using N-Gram model.

Keywords: Recommender System, Drugs Recommendation, Deep Learning, N-grams

Introduction

In today's era, the most significant and researched topic on the internet is health care or health-related information. In this digital space, everyone is looking for quick solutions, so the majority of people go online for health-related issues to educate themselves [1]. According to a Pew Research article [2], almost 60% of grownups are looking for enough health information on the web with 35% of respondents concentrating on diagnosing ailments online only. Previous studies have also shown that users are often looking for stories from "patients like them" on the Internet, which is hard to find among their friends and family [19]. This type of patient-related data available on social media can be proved as a significant tool in the healthcare sector such as suggesting health insurance, clinical pathway-based treatment methods, and drug recommendations based on the patient's health profile or assisting doctors.

A healthcare system requires to study huge data of the patients to conclude significant insights and help in the prediction of the best possible medications for the disease. Over the past decade, researchers have been analyzing the emotional impact of user experience and the severity of adverse drug reactions by extracting sentiment and semantic information from patient data [20]. With the advancements in machine learning (ML) and deep learning (DL) technologies, this huge amount of patient data can be used to extract meaningful insights for the betterment of the healthcare sector.

A drug recommendation system is a framework that recommends the best drugs for a particular disease by analyzing data related to patients such as their background, reviews, other diseases, etc. In this paper, a drug recommendation system is proposed using deep learning and Natural language processing to find out the fascinating records hidden in the history of patients. In this paper, a drug recommendation system is proposed and its working is depicted, which uses the current technologies like machine learning, natural language processing, sentimental analysis, etc. to find out the interesting records hidden in the data and reduce the medical errors by the doctors while prescribing medicines. The system consists of modules such as a database module, data preparation, data visualization, recommendation and model evaluation section.

The proposed prescription recommender framework is implemented using Machine learning N-Gram and Lightgbm calculations by utilizing information from the publicly available dataset on Kaggle []. The main objective of the proposed work is to provide an optimized model for the medication suggestion framework to achieve the measurements like great exactness, adaptability, and proficiency.

With the goal of offering a better platform for the Drug Recommendation System, the work on the proposed system is presented as: Section II represents the background and related work present on the Drug Recommendation System. Section III represents the adopted methodologies for the proposed system. Section IV represents the outcome of our proposed system and the last section i.e., Section V represents the conclusion and future scope of the presented work.

Background and Related Work

In today's era, the most searched topic on the internet is health care or health-related information. In this virtual space, all people are looking for brief solutions, so the majority of people log on for health-associated troubles to educate themselves. According to a few research articles [2], nearly 60% of grownups are searching out enough fitness facts on the web with 35% of respondents focusing on diagnosing ailments online most effectively. Previous research has also proven stats that customers are frequently searching out stories from "sufferers like them" on the Internet, that are hard to find among their mates and family [19]. This kind of affected person-related facts available on social media may be proved as a substantial tool in the healthcare area from a specific point of perspective including health insurance, clinical pathway-based remedy strategies, and other drugs primarily based on the affected person's fitness profile or assisting doctors.

A healthcare platform demands examining huge information related to the patients to finish big insights and to assist in the prediction of the high-quality possible medicinal drugs for any specific disease. For over a decade, researchers were also studying the emotional impact of a person's reveal in and the harshness of harmful drug reactions by way of extracting sentiment and semantic data from the affected person's information [20]. With the advancements in Machine Learning (ML) and Deep Learning (DL) technologies, researchers are now trying to develop more advanced and effective model by using this massive amount of affected person data to extract meaningful insights for the betterment of the healthcare region.

For as long as decade, analysts have been examining the enthusiastic effect of client experience and the seriousness of antagonistic medication responses by extricating feeling and semantic data (Sharif, Zaffar, Abbasi, and Zimbra (2014).

Past examinations have shown that wellbeing-related client-produced content is helpful according to various perspectives. One of the benchmark papers in this space was composed by Jane Sarasohn-Kahn []. It expresses that clients are regularly searching for stories from "patients like them" on the Internet, which is elusive among their loved ones. For as long as decade, analysts have been examining the enthusiastic effect of client experience and the seriousness of antagonistic medication responses by extricating feeling and semantic data (Sharif, Zaffar, Abbasi, and Zimbra (2014).

Because of an absence of trust and nature of client communicated clinical language, broad examination in the clinical and wellbeing area has not been finished. Along these lines, we expect to construct a stage where patients and clinicians can look by side effects and get drug proposals, symptoms of medications and acquire bits of knowledge into patients' portfolio.

Leilei Sun [1] inspected enormous scope treatment records to find the best treatment solution for patients. The thought was to utilize a productive semantic bunching calculation assessing the similitudes between treatment records. In like manner, the creator made a system to evaluate the sufficiency of the proposed treatment. This construction can endorse the best treatment regimens to new patients according to their segment areas what's more, unexpected problems. An Electronic Medical Record (EMR) of patients assembled from various centers for testing. The outcome shows that this system further develops the fix rate.

Xiaohong Jiang et al. [1] analyzed three unmistakable calculations, choice tree calculation, support vector machine (SVM), and backpropagation brain network on treatment information. SVM was picked for the drug proposition module as it performed really well in every one of the three novel limits - model precision, model capability, model adaptability. Furthermore, proposed

the slip-up actually take a look at framework to guarantee investigation, accuracy and organization quality

Previous Studies	Algorithms	Future Work/Predicted Output
GalenOWL [4]	Ontologies Semantic web technologies Rules	Extension of semantic standards Prioritization of collaborations medications and infections since not all cooperations have a similar significance. Performance streamlining (for example setting extraction from clinical information)
An Intelligent Medicine Recommender System Framework [1]	SVM (Support Vector Machine) Back-propagation neural network ID3 decision tree	Assemble the suggestion model Apply MapReduce to broaden the capacity of handling large determination information
LOD Cloud Mining for Prognosis Model [10]	Semantic Web techniques Data mining algorithm (Decision trees usin C4.5 algorithm and bagging)	Update information on medications, illnesses and collaborations depending on the situation Extract more significant elements like poisonousness, food association and so forth
CADRE [17]	Vector space model (VSM) K-means clustering Collaborative Filtering	Research how to work on the precision of CADRE by thinking about client's age, topography, and different elements
Na and Kyaing (2015)	Clause-wise Lexicon approach and Classification using SVM	Sentiment Polarity
Kallumadi et al. (2018) D	Classification	Drug Rating

Table1 :- Previous Works

3. Proposed Methodology

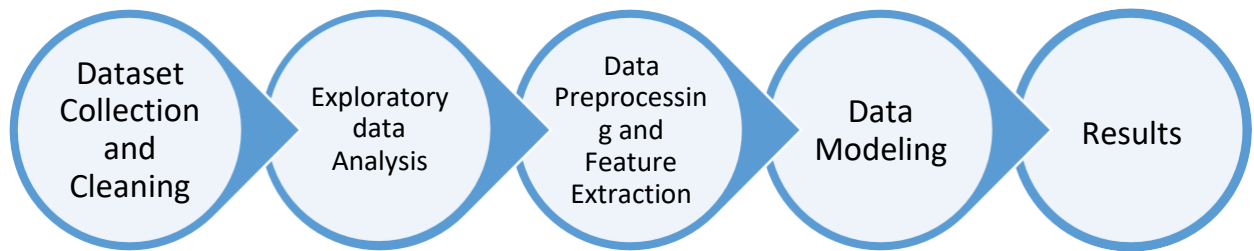


Figure :- Proposed method

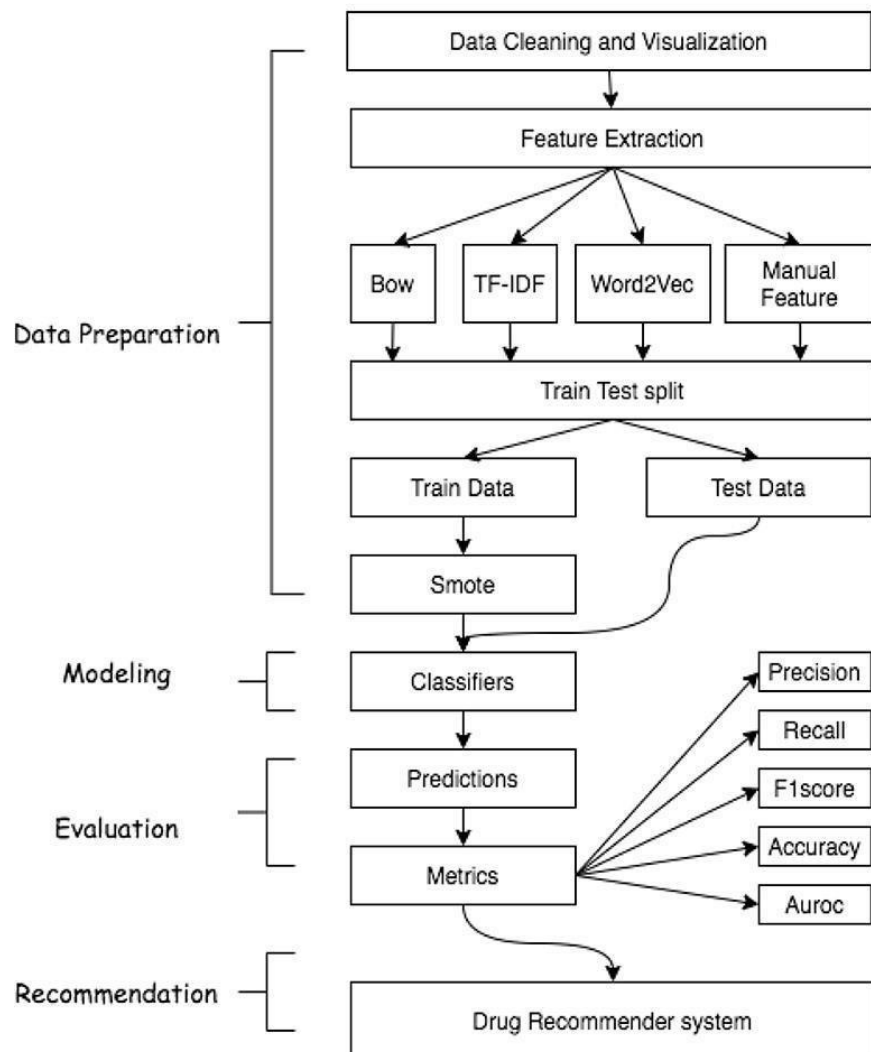


Figure :- Flowchart of Proposed Model

Data Understanding and Preprocessing

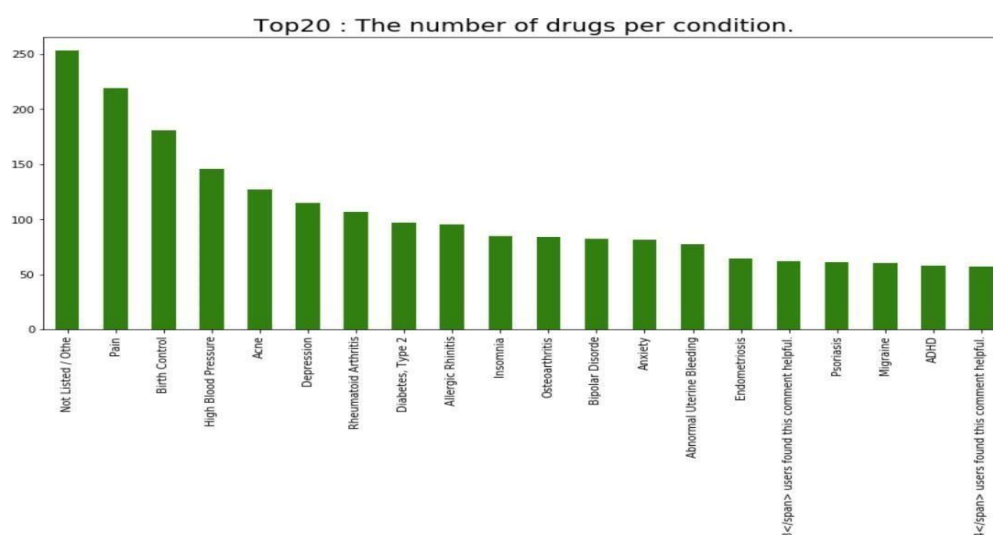
In the data exploration part, we will look at data types with visualization techniques and statistical techniques. Through this process, we can set the topic, preprocess the data to fit the objective, and create various variables to fit model.

Attributes	Type of Attributes
DrugName	Categorical
Condition	Categorical
Review	Text
Rating	Numerical
Date	Date
Useful Count	Numerical

Table:- Datatypes of Attributes

Dataset contains DrugName , Condition , Review , Date , Useful Count. The dataset contains 215063 rows and 6 columns.

The consequence of taking a glance at the information through the head () order shows us that there are six factors with the exception of the novel ID that distinguishes the individual, and survey is the key variable. The construction of the information is that a patient with a distinct ID buys a medication that meets his condition and composes a survey and rating for the medication he/she bought on the date. Subsequently, assuming the others read that survey and think that it is useful, they will click usefulCount, which will add 1 for the variable. In the first place, we will begin investigating factors, beginning from uniqueID. We looked at the one of a kind number of remarkable IDs and the length of the train information to check whether a similar client has composed numerous audits, and there weren't more than one surveys for one client. unique values count of train: 215063 length of train: 215063

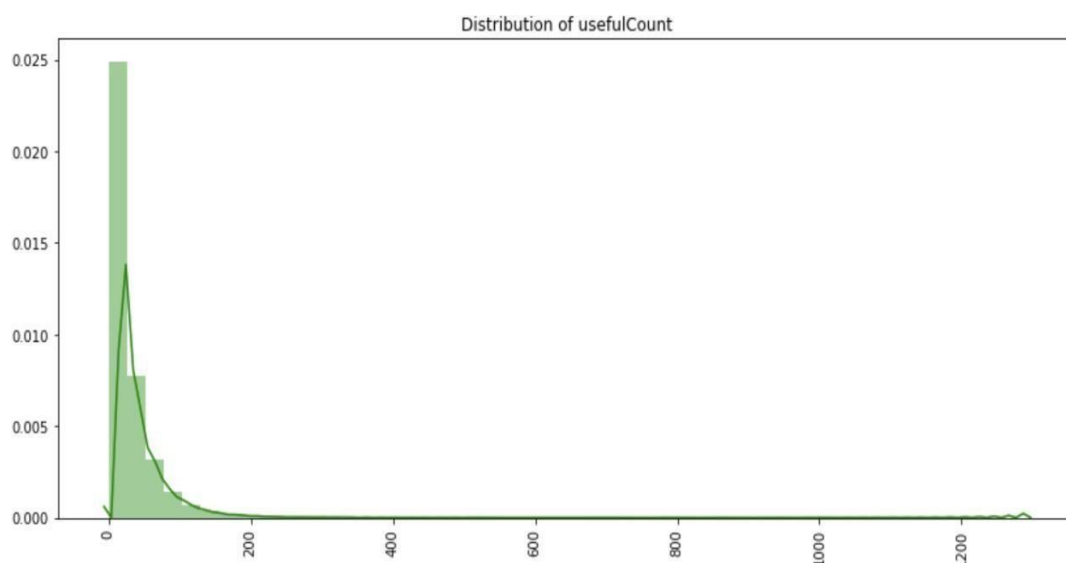


Considering the recommendation system, it is not feasible to recommend with that when there is only one product. Therefore, I will analyze only the conditions that have at least 2 drugs per condition

Let's see how word cloud of review looks like.



Next, I will classify 1 ~ 5 as negative, and 6 ~ 10 as positive, and we will check through 1 ~ 4 grams which corpus best classifies emotions. When I use 1-gram, I can see that the best 5 words have similar items, albeit the request for left (negative) and right (positive) are unique. This implies when I break down the text with a solitary corpus, it doesn't group the inclination well. Thus, I will grow the corpus. In like manner, in 2-gram, the items in the best five corpus are comparable, and arranging positive and negative is difficult. Likewise, 'aftereffects' and 'incidental effects.' are deciphered in an unexpected way, and that implies preprocessing of audit information is essential. Notwithstanding, I can see that this is smarter to order feelings instead of past 1-grams, similar to aftereffects, weight gain, and strongly suggest. From 3-gram I can see that there is a distinction among positive and negative corpus. Terrible aftereffects, conception prevention pills, negative incidental effects are corpus that arrange positive and negative. However, both positive and negative parts can be thought that it has missing parts that reverses the context, such as 'not' in front of a corpus. Clearly, 4-gram classifies emotions much better than other grams. Therefore, we will use 4-gram to build deep learning model. (I will add the figures in final report). I also checked if any day of the month affects the rating like salary day, but it seems like it does not make much difference. Next, I looked for relationship between rating and weather. Interestingly, I found that the average rating differs by year, but it is similar by month



From the distribution of useful Count shown above, it can be observed that the contrast among least and greatest is 1291, which is high. Likewise, the deviation is gigantic, which is 36. The justification behind this is that the more medications individuals search for, the more individuals read the survey regardless of their items are fortunate or unfortunate, which makes the most of the helpful exceptionally high. Thus, when I make the model, I will standardize it by conditions, thinking about individuals' availability. I preprocessed data by removing the missing values and also deleting conditions with only one drug. Removed the stop words from the reviews.

At the model part, for sentiment analysis of reviews I will apply random forests. The obvious starting question for such an approach is how I can convert the raw text of the review into a data representation that can be used by a numerical classifier. To this end, We will use the process of vectorization. By vectorizing the 'review' column, I can allow widely varying lengths of text to be converted into a numerical format which can be processed by the classifier. This can be achieved via the TF-IDF method which involves creating tokens (i.e. individual words or groups of words extracted from the text). The main limitation of this approach can be that it does not consider the relative position of words within the document. Hence, We are planning on only using the frequency of occurrence. For emotion analysis using word dictionary, deep learning with n-gram, etc. can also be used. For the emotional analysis I am planning to use Harvard emotional dictionary. We can compare both these and use the one with good results. For classification I am planning to use Neural network with Keras. Keras gives us lots of freedom to play with hyperparameters and design a network that would be best suited for this data. For feature analysis idea of attempting to derive importance out of the vectorized features by k-clustering by similarity, as keras has no built-in function to check for feature importance. To compensate the limitation of natural language processing, Lightgbm machine learning model can be used, and reliability can be further secured through useful count

Classification with Sklearn and Random Forests:-

As mentioned in the project proposal, We am vectorizing the 'review' column and tokenizing them using TF-IDF. Once the list of tokens is created, they are assigned an index integer identifier which allows them to be listed. I can then count the number of words in the document and normalize them in such a way that de-emphasizes words that appear frequently (like "a", "the", etc.). This creates what is known as a bag (multiset) of words. Such a representation associates a real-valued vector to each review representing the importance of the tokens (words) in the review. This represents the entire corpus of reviews as a large matrix where each row of the matrix represents one of the reviews and each column represents a token occurrence. Term-Frequency Inverse Document-Frequency (TF-IDF) is a way of handling the excessive noise due to words such as "a", "the", "he", "she", etc. Clearly such common words will appear in many reviews, but do not provide much insight into the sentiment of the text and their high frequency tends to obfuscate words that provide significant insight into sentiment. More details about the method can be found on wikipedia. The simplest type of model we can attempt to fit on this data is the Naive Bayes classifier. We will first test Naive Bayes on a binarized version of the rating column which attempts to identify which reviews are favorable. We define a favorable review as one which received a rating above 5. Given the sample size involved for the data set, We choose Naive Bayes over other classifiers due to its scalability.

It can be observed that a more complicated classifier gets us a roughly 8% increase in the accuracy of my classifier

Classification with Keras:-

Neural networks usually scale very well with lots of data, and that is exactly what I have here (150,000 training examples, 50,000 test examples). I could not run all of these through sk-learn as the kernel would just crash after a couple of minutes. However, keras plays with this many data very nicely. While I am sticking to a simple NN in this kernel, other types of NN architecture can deal with natural language processing problems very well (long short-term memory models, other types of recurrent networks). Keras gives me lots of freedom to play with hyperparameters and design a network that would be best suited for the data. For this I did some preprocessing by removing all symbols from the reviews. I use a CountVectorizer to vectorize each of the different reviews into 5000 feature row vectors. This is a very similar approach to vectorization as the TDIF explained above. This vectorizer is also set to not add in very common words such as "and" and "the" as features. This is specified in stop_words. Then we Created a helper function to easily create the y labels. we need to transform the review column to a m number of reviews by 3 columns, where index 0 is negative, 1 is positive and 2 is neutral. we experimented a lot with different kinds of simple NN architecture and this is what we concluded:

- Softmax was the best activation function for the output layer.
- Vectorizing with the TFIDF vectorizer took longer to converge but had roughly the same accuracy as when I vectorized with the CountVectorizer. Changing binary and lowercase didn't result in any changes.
- 256 units was the sweet spot for accuracy without overfitting. a. Any units above would give me less accuracy as the model continued to train as well as overall performing worse, and less units just gave me worse accuracy overall. we also discovered that adding any additional layers to the model resulted in a loss of accuracy, even when I kept the number of layer units small.
- We experimented with different numbers of epochs and batch sizes as well, and found roughly 6 epochs and a batch size of 128 to be optimal.
- Anything more than 6 epochs resulted in the model beginning to overtrain and lose accuracy.
- Before running preprocessing on the reviews by eliminating symbols, the validation accuracy stayed around 84%. Now it scores a perfect 100% on the 100 examples.

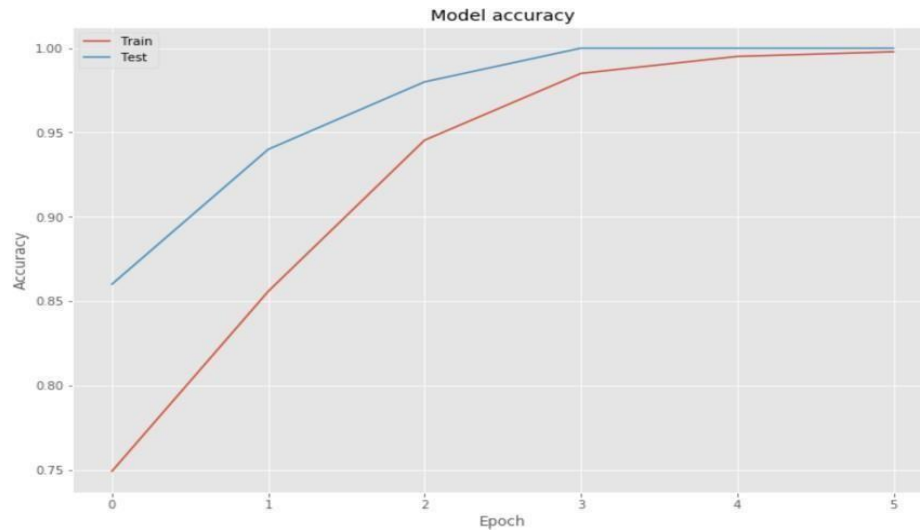


Figure :- Deep learning using NN

This model predicts the if the review is positive, negative or neutral with a ~89.4% accuracy, according to this test set output score.

Deep Learning Using N-gram :-

Here we are using sentiment feature for N-gram analysis. As mentioned in the project proposal we are using Keras as it gives us lots of freedom to play with hyperparameters and design a network that would be best suited for this data. We are using activation function relu, and adam optimizer [4]

N-Grams :-

N-grams are a chunk of subsequent words, by studying these sequences we can efficiently understand the context in which a particular word is used.

Example – the word ‘book’ can be used in different contexts like –

to ‘book’ tickets, read the ‘book’. Here the word book is used as the verb in the first phrase while as a noun in the latter. So in order to efficiently understand the context of the word, N-grams look at the after the word and before the word and then determine if the word is used as a noun or verb in the sentence or in other context.

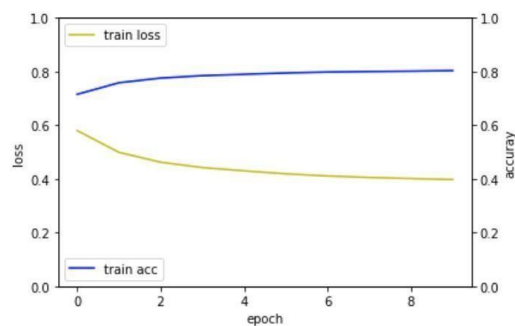
N in N-grams denotes the number of words machine will look at before and after the target word. Ex- This ‘book’, A ‘book’, Your ‘book’ are all examples of bi-grams where before word ‘book’ is a noun. Bi-grams are the two pairs of words to look at before and after the target word while sliding over the words. the context can be extended by going to tri-grams which means looking at three pairs of words before and after the target word.

Layer (type)	Output Shape	Param #
dense_1 (Dense)	(None, 200)	4000200
batch_normalization_1 (Batch Normalization)	(None, 200)	800
activation_1 (Activation)	(None, 200)	0
dropout_1 (Dropout)	(None, 200)	0
dense_2 (Dense)	(None, 300)	60300
batch_normalization_2 (Batch Normalization)	(None, 300)	1200
activation_2 (Activation)	(None, 300)	0
dropout_2 (Dropout)	(None, 300)	0
dense_3 (Dense)	(None, 100)	30100
dense_4 (Dense)	(None, 1)	101
Total params: 4,092,701		
Trainable params: 4,091,701		
Non-trainable params: 1,000		

```

Epoch 1/10
142075/142075 [=====] - 39s 272us/step - loss: 0.5802 - acc: 0.7152
Epoch 2/10
142075/142075 [=====] - 43s 302us/step - loss: 0.4990 - acc: 0.7582
Epoch 3/10
142075/142075 [=====] - 35s 246us/step - loss: 0.4621 - acc: 0.7758
Epoch 4/10
142075/142075 [=====] - 42s 299us/step - loss: 0.4422 - acc: 0.7848
Epoch 5/10
142075/142075 [=====] - 43s 301us/step - loss: 0.4300 - acc: 0.7899
Epoch 6/10
142075/142075 [=====] - 43s 300us/step - loss: 0.4189 - acc: 0.7948
Epoch 7/10
142075/142075 [=====] - 43s 301us/step - loss: 0.4109 - acc: 0.7983
Epoch 8/10
142075/142075 [=====] - 43s 301us/step - loss: 0.4053 - acc: 0.8000
Epoch 9/10
142075/142075 [=====] - 43s 305us/step - loss: 0.4013 - acc: 0.8015
Epoch 10/10
142075/142075 [=====] - 43s 302us/step - loss: 0.3976 - acc: 0.8036

```



```

69978/69978 [=====] - 12s 165us/step
loss_and_metrics : [1.0738699619418919, 0.6469318928772125]

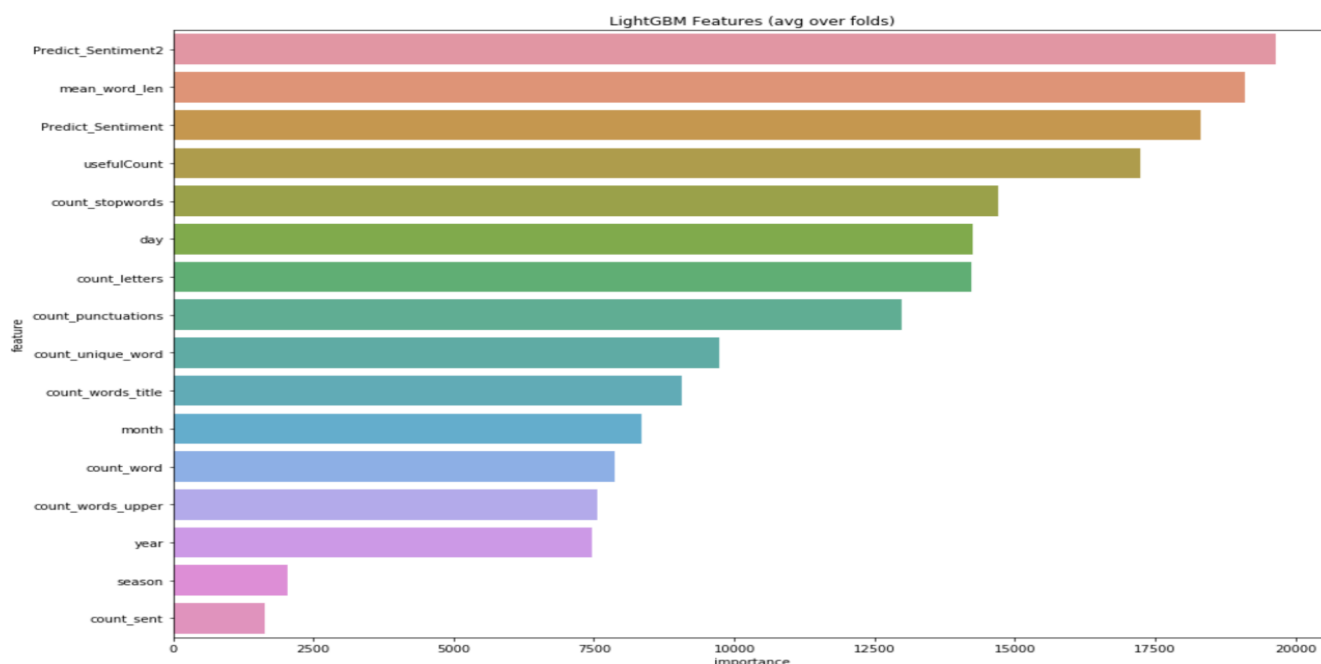
```

From the above graph and loss matrices it can be observed that I will have to improve the low accuracy.

Light Gradient Boosting Machine :-

To improve the low accuracy, we will use machine learning. First, this is the sentiment analysis model using only usefulCount. We will add variables for higher accuracy. We will add factors for higher exactness

Light GBM, short for Light Gradient Boosting Machine, is a free and open source dispersed tendency supporting framework for AI at first made by Microsoft. It relies upon decision tree estimations and used for situating, portrayal and other AI tasks. The improvement center is around execution and adaptability.



Dictionary Sentiment Analysis : -

As the package used for prediction of 'Predict value' is formed with movie review data, it can be unsuitable for this project which analyzes reviews for drugs. To make up for this, we conducted additional emotional analysis using the Harvard emotional dictionary.

I counted the number of words in review_clean which are included in dictionary and found that there are 2006. I defined $\text{Positiv_ratio} = \frac{\text{quantity of positive words}}{\text{quantity of positive words} + \text{quantity of negative words}}$. If the proportion is lower than 0.5, I delegated negative and assuming that it's higher than 0.5, I named positive. With leftovers, I delegated unbiased, which incorporates the sentence without one or the other positive or negative words.

As mentioned earlier, we have normalized usefulCount by condition to solve the problem that usefulCount shows bias depending on condition. You can then add three predicted emotion values and multiply them by the normalized usefulCount to get the predicted value. The underneath table as the determined last anticipated esteem and suggest the proper medication for each condition as indicated by the request for the worth.

		total_pred
		mean
condition	drugName	
ADHD	Adderall	0.070960
	Adderall XR	0.042328
	Adzenys XR-ODT	0.010250
	Amantadine	0.011098
	Amphetamine	0.013925
	Amphetamine / dextroamphetamine	0.046908
	Aptensio XR	0.005885
	Armodafinil	0.028856
	Atomoxetine	0.047597
	Bupropion	0.083736
	Catapres	0.044449
	Clonidine	0.059211
	Concerta	0.059579
	Cylert	0.014713
	Daytrana	0.031764
	Desoxyn	0.133611
	Desvenlafaxine	0.006131
	Dexedrine	0.064658
	Dexmethylphenidate	0.041450
	Dextroamphetamine	0.052630
	Dextrostat	0.045610
	Dyanavel XR	0.016457
	Evekeo	0.008692
	Focalin	0.046282
	Focalin XR	0.044685
	Guanfacine	0.070408
	Intuniv	0.078066
	Kapvay	0.127024
	Lisdexamfetamine	0.045679
	Metadate CD	0.037710
...
fibromyalgia	Nuvigil	0.255291
	Prednisone	0.082950
	Pregabalin	0.131091
	Pristiq	0.076931
	Prozac	0.186982
	Savella	0.114989

Results

Baseline models: Evaluation of classification models using confusion matrix and accuracy as shown below

Vectorizer	Model	Accuracy
TF-IDF	Linear Regression	78.2%
	Naive Bayes	75.2%
	Random Forest	83.1%

Deep learning models: Evaluation of deep learning models using metrics like loss and model accuracy.

Model	Loss	Accuracy
Deep NN (Sequential Model)	0.17	89.4%
Deep Learning model using N-gram	0.39	80.1%

Deep Neural Network Sequential model gave the best accuracy for this problem of 89.4%.

Conclusion and Future Scope

In conclusion, these are the limitations we had during the project and the future scope:

- Sentiment analysis using sentiment word dictionary has low reliability when the number of positive and negative words is small. For instance, assuming there are 0 positive words and 1 negative word, it is delegated negative. So, if the quantity of feeling words is 5 or less, we could reject the perceptions.
- To ensure the reliability of the predicted values, we normalized useful Count and multiplied it to the predicted values. However, usefulCount may tend to be higher for older reviews as the number of cumulated site visitors increases. Therefore, we should have also considered time when normalizing usefulCount.
- If the emotion is positive, the reliability should be increased to the positive side, and if it is negative, the reliability should be increased toward the negative side. However, we simply multiplied the usefulCount for reliability and did not consider this part. So we should have multiplied considering the sign of usefulCount according to different kinds of emotion.
- As future work productivity of proposal framework can be expanded by including age of the individual, segment data during the preparation stage. Additionally the brand and the substance contents accessible in the medication can work on the suggested prescriptions.

References

1. Leilei Sun, Chuanren Liu, Chonghui Guo, Hui Xiong, and Yanming Xie. 2016. Data-driven Automatic Treatment Regimen Development and Recommendation. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '16). Association for Computing Machinery, New York, NY, USA, 1865–1874. DOI: <https://doi.org/10.1145/2939672.2939866>.
2. Subhash C. Pandey, —'Data Mining techniques for medical data': 'A Review', —IEEE, 2016.
3. Shimada K, Takada H, Mitsuyama S, et al. Drug-recommendation system for patients with infectious diseases. AMIA Annu Symp Proc. 2005;2005:1112.
4. T. N. Tekade and M. Emmanuel, "Probabilistic aspect mining approach for interpretation and evaluation of drug reviews," 2016 International Conference on Signal Processing, Communication, Power and Embedded System (SCOPEs), Paralakhemundi, 2016, pp. 1471-1476, doi: 10.1109/SCOPEs.2016.7955684.
5. Doulaverakis, C., Nikolaidis, G., Kleontas, A. et al. GalenOWL: Ontology-based drug recommendations discovery. J Biomed Semant 3, 14 (2012). <https://doi.org/10.1186/2041-1480-3-14>.
6. Mu, R.; Zeng, X.; Han, L. A Survey of Recommender Systems Based on Deep Learning. IEEE Access 2018, 6, 69009–69022
7. V. Goel, A. K. Gupta and N. Kumar, "Sentiment Analysis of Multilingual Twitter Data using Natural Language Processing," 2018 8th International Conference on Communication.p- Systems and Network Technologies (CSNT), Bhopal, India, 2018, pp. 208-212, doi: 10.1109/CSNT.2018.8820254
8. Y. Bao and X. Jiang, "An intelligent medicine recommender system framework," 2016 IEEE 11th Conference on Industrial Electronics and Applications (ICIEA), Hefei, 2016, pp. 1383-1388, doi: 10.1109/ICIEA.2016.7603801
9. Telemedicine, <https://www.mohfw.gov.in/pdf/Telemedicine.pdf>
10. Sentiment Analysis, <https://towardsdatascience.com/leveraging-n-grams-to-extract-context-from-text-bdc576b47049>
11. Pew Research, <https://www.pewresearch.org/internet/2013/01/15/health-online-2013/>
12. Popescu AM, Etzioni O (2005) Extracting product features and opinions from reviews. Proceedings of the conference on human language technology and empirical methods in natural language processing. Association for Computational Linguistics, pp: 339-346
13. Sentiment Analysis using N-grams, <https://www.knime.com/blog/sentiment-analysis-with-n-grams>
14. Calero Valdez, A.; Ziefle, M.; Verbert, K.; Felfernig, A.; Holzinger, A. Recommender Systems for Health Informatics: State-of-the-Art and Future Perspective. In Machine Learning for Health Informatics; Holzinger, A., Ed.; Lecture Notes in Computer Science LNCS 9605; Springer: Cham, Switzerland, 2016
15. L. Fernandez-luque, R. Karlsen and L.K. Vognild, "Challenges and Opportunities of using Recommender Systems for Personalized Health Education" Stud. Health Technol. Inform., 150(903), pp. 903-7, 2009
16. B.Kitchenham, "Procedures for Performing Systematic Reviews", Technical Report TR/SE-0401, Keele University, NICTA, 2004

17. F. Ricci, L. Rokach, B. Shapira and P. B. Kantor, Introduction to Recommender Systems Handbook, pp. 257-297, Springer, Berlin, 2011.
18. D. H. park, H. K. Kim, I. Y. Choi and J. K. Kim, "A literature review and classification of recommender systems research", J. Expert Syst. Appl., 39 (11), pp.10059-10072, 2012
19. Sarasohn-Kahn, Jane. "The wisdom of patients: Health care meets online social media." (2008).
20. Gopalakrishnan, Vinodhini, and Chandrasekaran Ramaswamy. "Patient opinion mining to analyze drugs satisfaction using supervised learning." Journal of applied research and technology 15.4 (2017): 311-319.
21. <https://archive.ics.uci.edu/ml/datasets/Drug+Review+Dataset+%28Drugs.com%29>
22. 4. Felix Gräßer, Surya Kallumadi, Hagen Malberg, and Sebastian Zaunseder. 2018. Aspect-Based Sentiment Analysis of Drug Reviews Applying Cross-Domain and Cross-Data Learning. In Proceedings of the 2018 International Conference on Digital Health (DH '18). ACM, New York, NY, USA, 121-125. DOI: [Web Link]
23. 5. Ozsoy, Makbule Gulcin et al. "Realizing drug repositioning by adapting a recommendation system to handle the process." BMC bioinformatics vol. 19,1 136. 12 Apr. 2018, doi:10.1186/s12859-018-2142-1

Appendix - II

ORIGINALITY REPORT

11%

SIMILARITY INDEX

8%

INTERNET SOURCES

9%

PUBLICATIONS

3%

STUDENT PAPERS

PRIMARY SOURCES

1	www.researchgate.net Internet Source	3%
2	Satvik Garg. "Drug Recommendation System based on Sentiment Analysis of Drug Reviews using Machine Learning", 2021 11th International Conference on Cloud Computing, Data Science & Engineering (Confluence), 2021 Publication	2%
3	mafiadoc.com Internet Source	2%
4	searchbusinessanalytics.techtarget.com Internet Source	1%
5	Submitted to University of London External System Student Paper	1%
6	Emre Sezgin, Sevgi Ozkan. "A systematic literature review on Health Recommender Systems", 2013 E-Health and Bioengineering Conference (EHB), 2013 Publication	<1%



Submitted to Liverpool John Moores
University

Student Paper

<1 %



docs.huihoo.com

Internet Source

<1 %



Sisi Liu, Ickjai Lee. " Email Sentiment Analysis
Through -Means Labeling and Support Vector
Machine Classification ", Cybernetics and
Systems, 2018

Publication

<1 %



studentsrepo.um.edu.my

Internet Source

<1 %

Exclude quotes Off

Exclude matches Off

Exclude bibliography On