# Machine Learning Based Drug Recommendation from Sentiment Analysis of Drug Rating and Reviews

Kodepogu Koteswara Rao [a], Kona Sravya [b], Kadamati Jaya Phanidra Sai [b], Gummadi Giri Ratna Sai [b], Geetha Ganesan [c]

[a] *Associate Professor, Dept of CSE, PVP Siddhartha Institute of Institute of Technology, Vijayawada, India*
[b] *Bachelor of Technology, Dept of CSE, PVP Siddhartha Institute of Institute of Technology, Vijayawada, India*
[c] *Advanced Computing Research Society, Chennai, Tamilnadu, India*

## Abstract

A suggestion framework can help the client to make an arrangement out of necessities and propose educated choices from a great deal regarding confounded information. Suggestion from an investigation of feelings is by all accounts an incredible test as client created content is addressed involving human language in more ways than one. Many examinations have zeroed in on normal fields like surveys of electrical things, movies, and cafés, yet insufficient on wellbeing and clinical issues. Feeling examination of medical care overall and that of the medication encounters of people, specifically, may reveal extensive insight into how to zero in on working on general wellbeing and arrive at the right choice. In this work, we plan in addition carry out a medication recommender framework scheme that spread on feeling examination advancements taking place drug audits. The target of this examination is to construct a dynamic help stage to assist patients with accomplishing more huge decisions in drug determination. First and foremost, we propose a wistful estimation way to deal with drug surveys and produce evaluations on drugs. Furthermore, we receipts by what means much the medication audits are helpful to clients, patient's situations, and word reference opinion extremity of medication surveys into thought. Then, at that point, we intertwine those factors into the proposal framework to list suitable meds. Tests have been done utilizing Decision Tree, K Nearest Neighbours, and Linear Support Vector Classifier calculation in rating age and Hybrid model in proposal in light of the given open dataset. The investigation is kept out to melody the boundaries for every calculation to accomplish more prominent execution. At long last, Linear Support Vector Classifier is chosen intended for rating age to get a decent compromise in the middle of model exactness, model effectiveness, then model versatility.

**Keywords 1**
Drug rating, Sentiment, machine Learning

## 1. Introduction

With the impact of Web 2.0 stages, there are enormous measures of content made by customers, called internet-based media. Consequently, an excessive number of researchers have been investigating capable calculations for feeling examination of content made by purchasers throughout the most recent ten years. The area of feeling investigation, otherwise called assessment mining, examinations the conclusions, insights, convictions, decisions, perspectives, and feelings of individuals, including items, administrations, associations, characters, occasions and points. Lately, these two spaces of utilization have gotten extraordinary interest. In nostalgic exploration, the investigations are by and large partitioned into two classes, positive and negative. Yet, in the event

---

WAI-2022: Workshop on Artificial Intelligence, January 27 – 28, 2022, Chennai, India.
EMAIL: geetha@advancedcomputingresearchsociety.org (Geetha Ganesan)
ORCID: 0000-0001-7338-973X (Geetha Ganesan)

that every one of up-and-comers' items reflect good or gloomy sentiments it is hard for individuals to decide. To settle on a choice, individuals need not exclusively to know whether the item is great yet in addition how great it is. It is additionally acknowledged that different individuals have various inclinations for nostalgic articulation. Thus, it is additional essential to offer mathematical notches rather than paired choices in numerous useful cases, for example, drug suggestion and fabricates an arrangement of choice help that helps individuals in choosing items. This new application field presents the two difficulties and examination valuable open doors in clinical wellbeing. A proposal system intends to anticipate the inclinations of clients and make ideas that would bear some significance with clients. Cooperative sifting (CF), content based (CB), and information based (KB), and half-breed proposal advancements, all of which have specific limits, are closed by conventional suggestion innovation. CB has overspecialized suggestions and CF dislikes sparsity, adaptability, and cold-start issue. Yet, a few scientists zeroed in on drug proposal framework from client audits, and have demonstrated that the opinion investigation of medical services overall and that of client's medication experience, specifically, could reveal critical insight into the interaction to work on general wellbeing and settle on the best choices, and this framework joins with customary suggestion framework is more successful. In our exploration, we are centered on assessment mining in drug audits, in which patients share their encounters and conclusions about prescriptions and afterward group the suppositions into appraisals, and even suggest a medicine list that would be generally suitable for the patient. Executing the proposed way to deal with feeling examination won't simply be helpful to patients yet in addition to drug specialists and clinicians for significant popular assessment synopses.

## 2. Problem Statement

Proposal from an investigation of opinions is by all accounts an incredible test as client produced content is addressed involving human language in more than one way. Feeling investigation of medical services overall and that of the medication encounters of people, specifically, may reveal significant insight into how to zero in on working on general wellbeing and arrive at the right choice. For our situation, we are executing directed AI calculations which remain utilized to create assessment from drug audit and suggestion model that recommend a suitable prescription to eliminate the particular condition.

### 2.1. Objective

Suggestions procedures expect to give buyers customized labour and products to adapt to the developing issue of over-burdening on the web data. Reads up involved various techniques for feeling examination, and since the mid1990s, recommender model procedures have been suggested. Many early explores focus on report level review and allude to e-business, e-government, e-learning, web based business/e-shopping, e-the travel industry, and so on Notwithstanding, the universe of medication contains uncommon suggesting advances. This task expects to introduce a medication recommender framework that can radically diminish expert's load. AI has been important in in numerous applications, and there is an expansion in inventive work for computerization. In this examination, we fabricate a medication proposal framework that utilizes patient surveys to anticipate the opinion utilizing different vectorization processes like Manual Feature Analysis, which can assist with suggesting the top medication for a given infection by various characterization calculations.

## 3. Proposed Work

Our medication rating age and recommender framework system essentially comprises five modules, should be visible in Figure beneath, which be situated information pre-handling building block (including highlight taking out), rating age module, model assessment module, word reference feeling investigation module, and proposal model module.

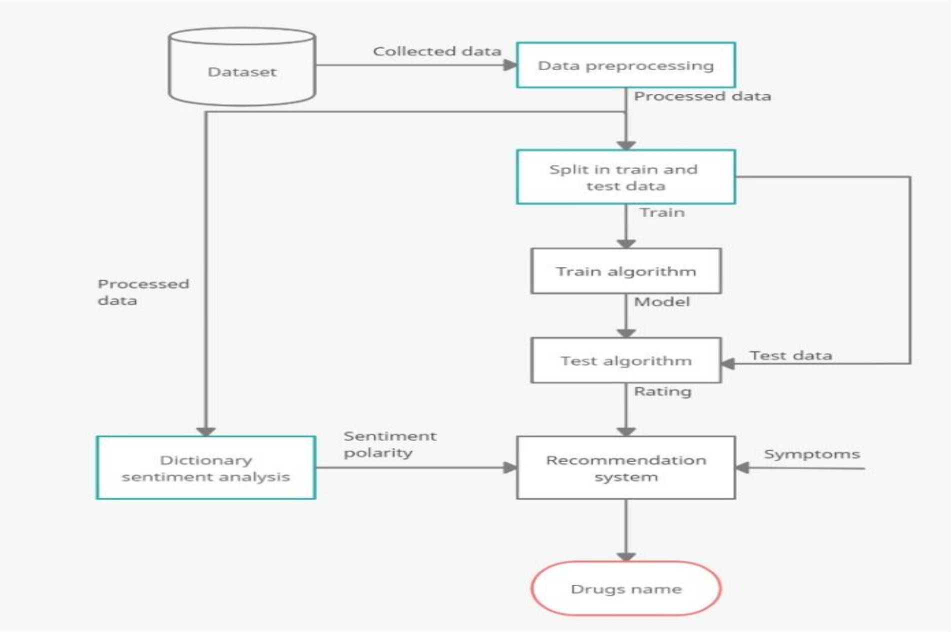


**Figure 1:** Proposed methodologies

## 3.1. Data Pre-processing

Information cleaning is the strategy for finding and fixing (or eliminating) harmed or blemished data from a record set, which alludes to finding absent, mistaken, deficient, or insignificant segments of the information and afterward adding, changing, or erasing filthy or coarse information. Legitimate information planning is a necessary advance, for a substantial trial as well as in any case to permit the mining of a dataset utilizing the method for AI. An assortment of pre-handling steps expected to permit the AI framework and calculations to peruse and investigate the information, just as to diminish the dataset to contain the essential items and qualities for the examination. Essentially, the creation or estimation of extra ascribes from the information could likewise be significant assuming such determined traits may help the examination and in this manner permit better forecasts. At the point when we've utilized online media information, the informational indexes should be cleaned astutely. Basically, online media information can't be handled in a solitary manner. Consequently, we involved our procedures for appropriately investigating opinions to tidy up that information.

These are the subsequent tools we rummage-sale for pre-processing our drug dataset:

- Tokenization
- Stop word
- Handling Negative Adjectives
- Stemming

## 3.2. Feature Extraction

Machines can't get characters and words. So, when managing message information we really want to address it in numbers to be perceived by the machine

**Count Vectorizer**

Count Vectorizer is a technique to change text over to mathematical information. It makes a grid in which every extraordinary word is addressed by a segment of the framework, and every text test from the report is a line in the lattice. The worth of every cell is only the include of the word in that specific message test.

Machines can't get characters and words. So, when managing message information, we really want to address it in numbers to be perceived by the machine. Count vectorizer is a technique to change text over to mathematical information.

CountVectorizer is an exceptional gadget given by the scikit-learn library in Python. It is used to change a given text into a vector in light of the repeat (count) of each word that occurs in the entire text. This is useful when we have different such texts, and we wish to change over each word in every text into vectors (for utilizing in additional text examination). Count Vectorization includes counting the quantity of events each words shows up in a report (unmistakable text like an article, book, even a passage). It likewise empowers the pre-handling of message information preceding creating the vector portrayal. This usefulness makes it a profoundly adaptable component portrayal module for text. Count Vectorizer makes it simple for text information to be utilized straightforwardly in AI and profound learning models like text order.

## 3.3. Method

The 3 administered AI calculations which be situated utilized to create rating from drug audit in addition proposal model that prescribe a suitable prescription to eliminate the particular condition are as per the following:

**Decision Tree (DT)**

Perhaps the most broadly involved progressive models for directed discovering that distinguishes neighborhood districts as series of recursive partition through choice hubs in the test work. The instinct behind the calculation of the choice tree is straightforward, yet at the same time very strong. It segments data into two subsections to keep the information in each fragment extremely homogeneous (all information in the section is of a comparative objective class) than the prior/substitute subsections; the two subsections can then be disconnected again before the homogeneity or later based halting edges are met. In extending the decision tree, a comparable marker limit can be applied to many spots. An authoritative place of parcel is to survey the right element associated with the legitimate edge to assemble subgroup/branch homogeneity.

**Naïve-Bayes (NB)**

It is a characterization strategy in light of Bayes' Theorem with an assumption of opportunity among pointers. In clear terms, a Naive Bayes classifier acknowledges that the presence of a particular part in a class is immaterial to the presence of another component. Gullible Bayes model is easy to gather and particularly significant for very gigantic educational assortments. Close by ease, Naive Bayes is known to outmaneuver even astoundingly present day gathering systems.

1. Gaussian naïve Bayes
2. Multinomial naïve Bayes

Bayes theorem provides a way of calculating posterior probability P(c|x) from P(c), P(x) and P(x|c). Look at the equation below:

$$P(c|x) = (P(x|c)*P(c))/P(x) \qquad (1)$$

Where, P(c|x) is the posterior probability of class (c, target) given predictor (x, attributes). P(c) is the prior probability of class.

P(x|c) is the probability which is the likelihood of indicator given class. P(x) is the earlier likelihood of indicator. Exactly when doubt of independence holds, a Naive Bayes classifier performs better differentiation with various models like determined backslide and you truly need less planning data. It performs well in the event that there ought to emerge an event of full scale input factors diverged from numerical variable(s). For numerical variable, normal allotment is acknowledged (ring twist, which is a strong speculation).

### Support Vector Machine (SVM)

The SVM thought relies upon the Structural Risk Minimization rule of computational learning speculation [24] and conceivably the most strong and convincing strategy used in AI. In this speculation, data is evaluated and the restrictions of decisions are portrayed by having hyper planes. By virtue of data that can't be easily disengaged, it utilizes 4 section structures for request tasks including straight, polynomial, outspread based, and sigmoid limits by arranging the information data into high-layered component space to allow the data supportively particular. The hyper plane parcels the text vectors of each class with the end goal that the capability is held as broad as could be anticipated. Straight SVC is undifferentiated from SVC with limit kernel='linear'. Learning the hyper plane in straight SVM occurs by using direct polynomial math to change the issue. Clear information is that, rather than using insights themselves, the direct SVM is generally rephrased using inward thing with any two components. A measure of the expansion of the data regards for each pair is the inward thing between two vectors. The condition for making a gauge for data using the spot thing between the information (X) and each help vector not entirely settled as follows:

$$F(X) = Bo + \sum Ai(X, Xi) \tag{2}$$

The condition no. 1 includes the working out of the internal results of another info vector (X) with all help vectors in preparing information. From the preparation information on the learning calculation, coefficients Bo and Ai (for each information) should be determined. The dab item is known as the piece and can be re-composed as:

$$K(X, Xi) = \sum (X^*Xi) \tag{3}$$

The piece chooses the equivalence or distance of new data from help vectors. The bit thing is an extent of similarity used for straight SVM or an immediate part since the distance between the information sources is immediate show [24]. Proposition Model: We are familiar with anticipating what a purchaser will give the "rate" or "tendency." Recommendation engines are mechanical assemblies for data filtering that use computations and data to enlighten a lone customer in regards to the primary things. Then again they are only a motorized sort of a "shop counterman". For a thing, you ask him, he shows the prescription just as the things you would purchase. They are particularly ready in decisively pitching and up selling. As there is growing data on the Internet and the amount of customers has extended essentially, looking, arranging, and outfitting associations with the information they need, according to their tendencies and tastes, is huge.

### Dictionary Sentiment Analysis

In the examination of word reference feeling, we performed enthusiastic investigation utilizing a passionate word reference to determine the constraints of the bundle worked with the information from motion pictures. To compensate for this, we utilized the Harvard enthusiastic word reference to play out extra passionate evaluations. To begin with, we count the quantity of words remembered for the word reference and determined positive proportion in pre-handled information.

$$\text{Positive Ratio} = n(P)/(n(P)+n(N)) \tag{4}$$

Where:

$n(P)$ is the number of positive words in the review and
$n(N)$ is the number of negative words in the review

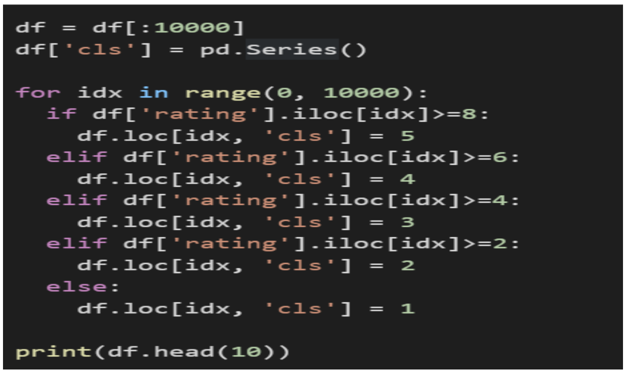
Assuming the proportion is under 0.5, we have ordered it as negative and in the event that it is more noteworthy than 0.5, we have grouped it as certain. We reviewed it as nonpartisan with leftovers, which incorporates the sentence with no sure or negative terms.

## 4. Experimental Results

For building this model, we use the dataset of drug reviews. The dataset contains data like the drug name, the condition the patient is in while using the drug, date the review collected on, useful count which is the number of people found the review helpful, rating given by the user for the drug and finally, the detailed review given by the user.

Since the rating is on the scale 1-10 in the dataset, to reduce the number of classes a review falls in, we brought down the rating to the scale of 1-5 by simple division as shown below:

| Rating in dataset(R) | Converted Rating |
|---|---|
| 10, 9, 8 | 5 |
| 7, 6 | 4 |
| 5, 4 | 3 |
| 3, 2 | 1 |



```
df = df[:10000]
df['cls'] = pd.Series()

for idx in range(0, 10000):
  if df['rating'].iloc[idx]>=8:
    df.loc[idx, 'cls'] = 5
  elif df['rating'].iloc[idx]>=6:
    df.loc[idx, 'cls'] = 4
  elif df['rating'].iloc[idx]>=4:
    df.loc[idx, 'cls'] = 3
  elif df['rating'].iloc[idx]>=2:
    df.loc[idx, 'cls'] = 2
  else:
    df.loc[idx, 'cls'] = 1

print(df.head(10))
```

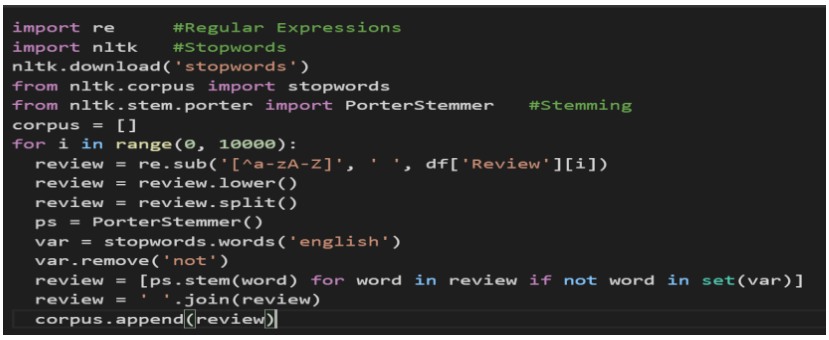
**Figure 2:** Removing Stop words

For removing stop words, we used the Natural Language Toolkit (NLTK) in Python. The NLTK library contains stop words from 16 different languages. Since the reviews are in English, we used the list of English stop words. Since sets in Python provide better Time Complexity for searching, we converted the list into a Python set before searching for a word in stop words.

For Tokenization, we used the Regular Expression(re) module in python. Also, instead of dealing with alphabets of uppercase and lowercase separately, we converted all uppercase alphabets to lowercase.

For Stemming, we used the Porter Stemmer algorithm in the NLTK module. Porter Stemmer is the widely used algorithm for stemming words in English language.

The following image includes all the steps we used for pre-processing

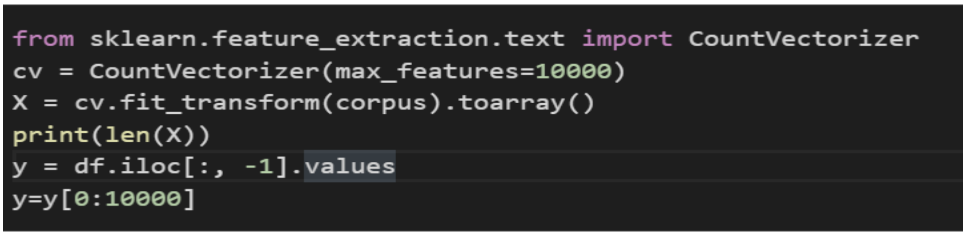


```
import re      #Regular Expressions
import nltk    #Stopwords
nltk.download('stopwords')
from nltk.corpus import stopwords
from nltk.stem.porter import PorterStemmer   #Stemming
corpus = []
for i in range(0, 10000):
  review = re.sub('[^a-zA-Z]', ' ', df['Review'][i])
  review = review.lower()
  review = review.split()
  ps = PorterStemmer()
  var = stopwords.words('english')
  var.remove('not')
  review = [ps.stem(word) for word in review if not word in set(var)]
  review = ' '.join(review)
  corpus.append(review)
```

**Figure 3:** Pre Processing

We stored the result of the pre-processing steps in a Python list “corpus”. The ‘corpus’ list contains the refined reviews, which is then used for the further steps.

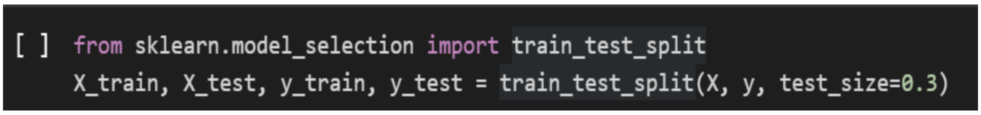
After Pre-processing, we extracted features using the CountVectorizer from the Scikit-learn library. We limited the maximum allowed features to 10,000 which is the nearest round figure to eliminate the less-frequent words that are likely to be un-useful. Also, we used only the first 10,400 reviews, considering the size of the dataset, which approximately contains 53,000 reviews.



```
from sklearn.feature_extraction.text import CountVectorizer
cv = CountVectorizer(max_features=10000)
X = cv.fit_transform(corpus).toarray()
print(len(X))
y = df.iloc[:, -1].values
y=y[0:10000]
```

**Figure 4:** Features Extraction

Also, we split the data into train data and test data in the ratio 7:3. The train data is used to build the model and the test data is used to test the accuracy of the model.

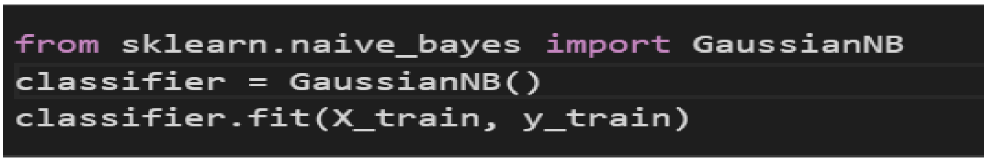


```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3)
```

**Figure 5**: Train data and test data

We used 3 classification algorithms namely- Gaussian Naïve Bayes Classifier, Decision Tree Classier and Support Vector Classifier for generating rating.

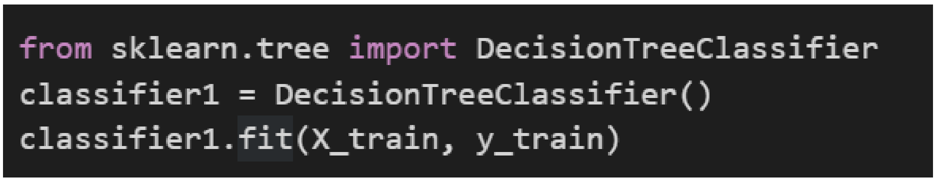
### Gaussian Naïve Bayes Classifier



```
from sklearn.naive_bayes import GaussianNB
classifier = GaussianNB()
classifier.fit(X_train, y_train)
```

**Figure 6:** Gaussian Naïve Bayes Classifier

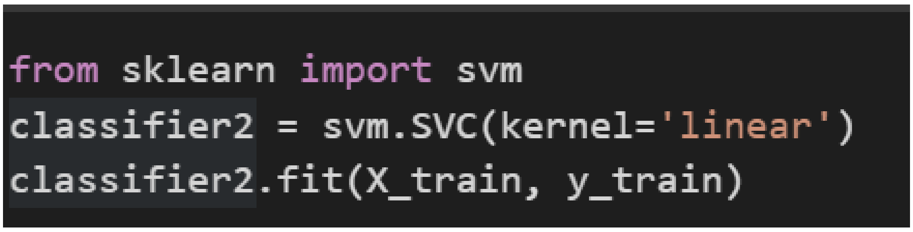
### Decision Tree Classifier



```
from sklearn.tree import DecisionTreeClassifier
classifier1 = DecisionTreeClassifier()
classifier1.fit(X_train, y_train)
```

**Figure 7:** Decision Tree Classifier

### Support Vector Classifier



```
from sklearn import svm
classifier2 = svm.SVC(kernel='linear')
classifier2.fit(X_train, y_train)
```

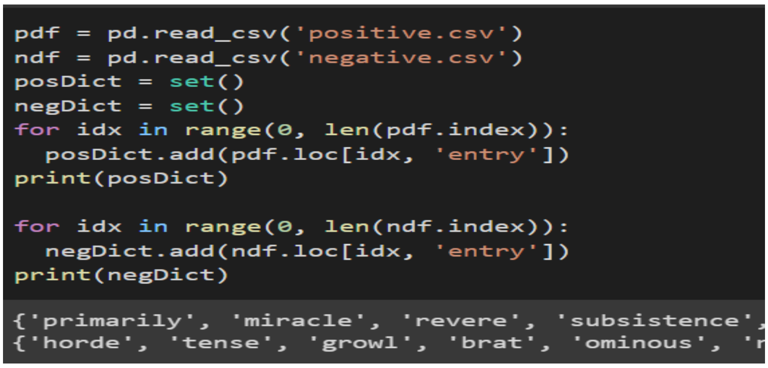
**Figure 8:** Support Vector Machine

Of all 3 classifiers, the Naïve Bayes classifier gave the best accuracy of 60.41%, followed by Support Vector Classifier and Decision Tree Classifier.

**Table 1:** Accuracy

| Classifier | Accuracy (%) |
|---|---|
| Naïve Bayes | 60.41 |
| Decision Tree Classifier | 56.63 |
| Support Vector Classifier | 56.83 |

For Dictionary Sentiment Analysis, we refined the Harvard emotional dictionary csv file into 2 files- one consisting of positive words and the other consisting of negative words. We imported the csv files and stored in 2 Python sets. Later, we had added some code to count the number of positive and negative words in each review and to calculate Dictionary Sentiment polarity of the review.

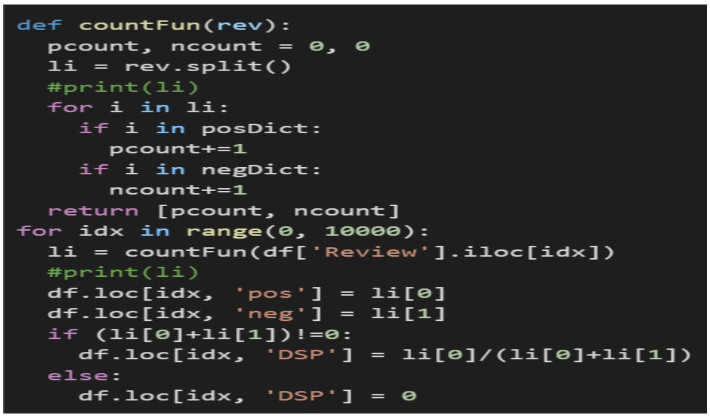


```
pdf = pd.read_csv('positive.csv')
ndf = pd.read_csv('negative.csv')
posDict = set()
negDict = set()
for idx in range(0, len(pdf.index)):
  posDict.add(pdf.loc[idx, 'entry'])
print(posDict)

for idx in range(0, len(ndf.index)):
  negDict.add(ndf.loc[idx, 'entry'])
print(negDict)
```

```
{'primarily', 'miracle', 'revere', 'subsistence',
{'horde', 'tense', 'growl', 'brat', 'ominous', '
```

**Figure 9:** Calculate Dictionary Sentiment polarity



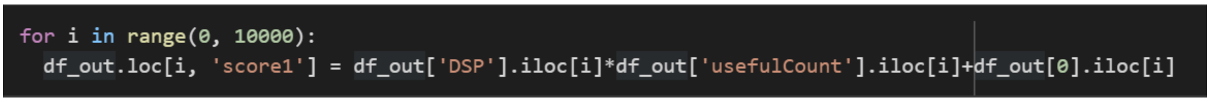
```
def countFun(rev):
  pcount, ncount = 0, 0
  li = rev.split()
  #print(li)
  for i in li:
    if i in posDict:
      pcount+=1
    if i in negDict:
      ncount+=1
  return [pcount, ncount]
for idx in range(0, 10000):
  li = countFun(df['Review'].iloc[idx])
  #print(li)
  df.loc[idx, 'pos'] = li[0]
  df.loc[idx, 'neg'] = li[1]
  if (li[0]+li[1])!=0:
    df.loc[idx, 'DSP'] = li[0]/(li[0]+li[1])
  else:
    df.loc[idx, 'DSP'] = 0
```

**Figure 10:** Dictionary Sentiment polarity calculation

In the next step, we used the formula shown below to generate the score of each drug for a particular condition.

$$\text{Score} = (\text{Dictionary Sentiment Polarity} * \text{usefulCount}) + \text{generatedRating}$$

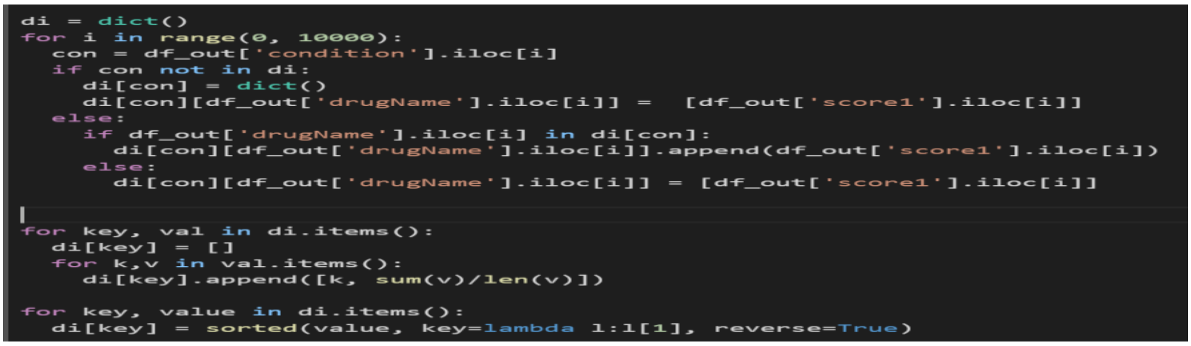
The implementation s shown below:



```
for i in range(0, 10000):
  df_out.loc[i, 'score1'] = df_out['DSP'].iloc[i]*df_out['usefulCount'].iloc[i]+df_out[0].iloc[i]
```

**Figure 11:** Score Calculation

The next step we did is to create a dictionary containing conditions and drug names and the mean of scores of each drug for the specified condition.



```
di = dict()
for i in range(0, 10000):
  con = df_out['condition'].iloc[i]
  if con not in di:
    di[con] = dict()
    di[con][df_out['drugName'].iloc[i]] =  [df_out['score1'].iloc[i]]
  else:
    if df_out['drugName'].iloc[i] in di[con]:
      di[con][df_out['drugName'].iloc[i]].append(df_out['score1'].iloc[i])
    else:
      di[con][df_out['drugName'].iloc[i]] = [df_out['score1'].iloc[i]]

for key, val in di.items():
  di[key] = []
  for k,v in val.items():
    di[key].append([k, sum(v)/len(v)])

for key, value in di.items():
  di[key] = sorted(value, key=lambda l:l[1], reverse=True)
```

**Figure 12:** Creating dictionary

The last step is to prescribe take the list of conditions the patient is suffering from and to recommend the top-3 drugs for each condition along with the recommended score for each drug.

## 4.1. Input and Output Parameters

**Reducing the Scale of Ratings:**

INPUT : Data frame with ratings on the scale of 1-10
OUTPUT : Data frame with ratings on the scale of 1-5

**Pre-Processing**

INPUT : Data frame with unprocessed data
OUTPUT : A Python list containing the reviews that are tokenized, stemmed and free from stop words.

**Count Vectorizer:**

INPUT : The Python list containing reviews.
OUTPUT : A matrix of with each cell containing the number of occurrences of a word(column) in each review(row)

**Naïve Bayes Classifier:**

INPUT : The count vectorizer matrix with 7000 rows
OUTPUT : Naïve Bayes classifier

**Decision Tree Classifier:**

INPUT : The count vectorizer matrix with 7000 rows
OUTPUT : Decision Tree classifier

**Support Vector Classifier:**

INPUT : The count vectorizer matrix with 7000 rows
OUTPUT : Support Vector classifier

**Dictionary Sentiment Polarity:**

INPUT : Data frame containing reviews
OUTPUT : Data frame containing Dictionary Sentiment Polarity of each review.

### Calculating Score of Each Review:

INPUT : Data frame containing Dictionary Sentiment Polarity and ratings
OUTPUT : Data frame containing score for each review.

### Grouping Conditions and Drugs:

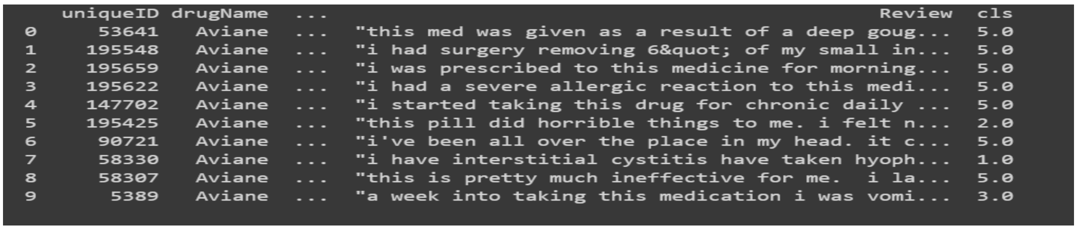
INPUT : Data frame containing score or each review
OUTPUT : A Python Dictionary containing conditions, drugs and the mean score of drugs

### Recommending Drugs:

INPUT : A Python list containing the conditions patients has.
OUTPUT : Recommended Drugs in decreasing order of their scores

## 4.2. Implementation Results

### Reducing the Scale of Ratings



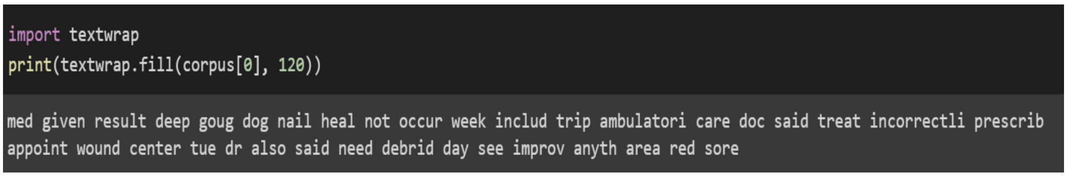
```
   uniqueID drugName  ...                                             Review  cls
0     53641   Aviane  ...  "this med was given as a result of a deep goug...  5.0
1    195548   Aviane  ...  "i had surgery removing 6&quot; of my small in...  5.0
2    195659   Aviane  ...  "i was prescribed to this medicine for morning...  5.0
3    195622   Aviane  ...  "i had a severe allergic reaction to this medi...  5.0
4    147702   Aviane  ...  "i started taking this drug for chronic daily ...  5.0
5    195425   Aviane  ...  "this pill did horrible things to me. i felt n...  2.0
6     90721   Aviane  ...  "i've been all over the place in my head. it c...  5.0
7     58330   Aviane  ...  "i have interstitial cystitis have taken hyoph...  1.0
8     58307   Aviane  ...  "this is pretty much ineffective for me.  i la...  5.0
9      5389   Aviane  ...  "a week into taking this medication i was vomi...  3.0
```

**Figure 13:** Reducing the scale of ratings

### Pre-Processing:

After pre-processing the following review, the refined review generated is shown below:

“This med was given as a result of a deep gouge from a dog nail. healing was not occurring after 4 weeks including a trip to an ambulatory care. my doc said they treated it incorrectly. he prescribed this. i have an appointment at the wound center Tues. dr also said it needed debriding. after 5 days i see no improvement. if anything, the area is more red and sore.”

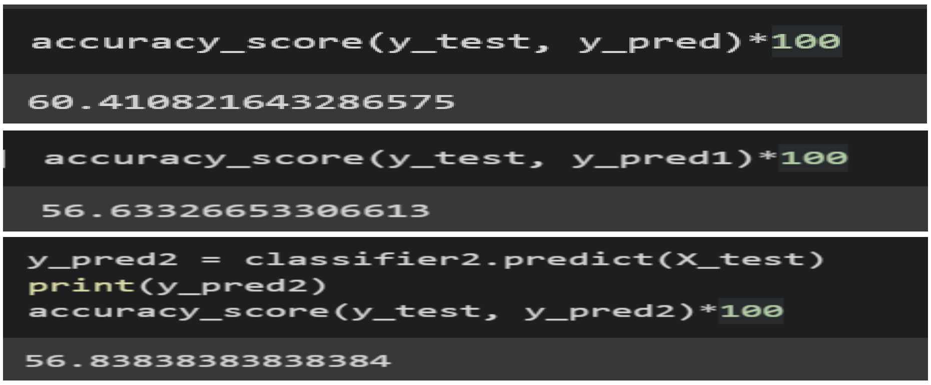


```
import textwrap
print(textwrap.fill(corpus[0], 120))
```

```
med given result deep goug dog nail heal not occur week includ trip ambulatori care doc said treat incorrectli prescrib
appoint wound center tue dr also said need debrid day see improv anyth area red sore
```

**Figure 14:** Import text wrap

### Naïve Bayes Classifier, Decision Tree Classifier, Support Vector Classifier:



```
accuracy_score(y_test, y_pred)*100
```

```
60.410821643286575
```

```
accuracy_score(y_test, y_pred1)*100
```

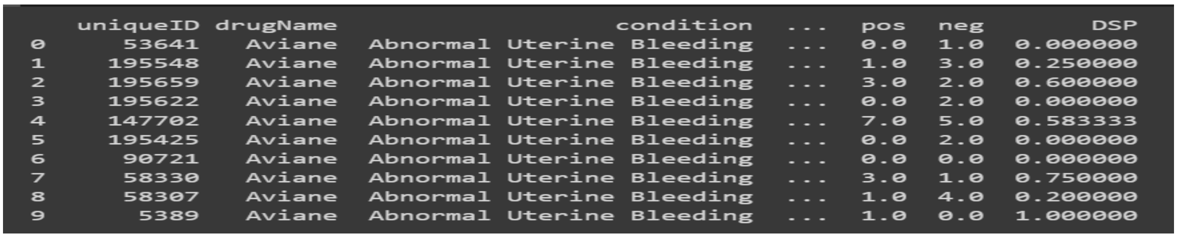
```
56.63326653306613
```

```
y_pred2 = classifier2.predict(X_test)
print(y_pred2)
accuracy_score(y_test, y_pred2)*100
```

```
56.83838383838384
```

**Figure 15:** Accuracy measures

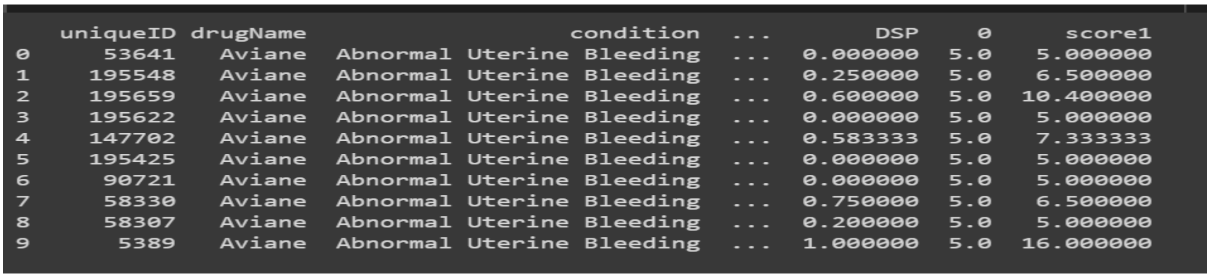
### Dictionary Sentiment Polarity:



```
   uniqueID drugName                     condition  ...  pos  neg       DSP
0     53641   Aviane  Abnormal Uterine Bleeding  ...  0.0  1.0  0.000000
1    195548   Aviane  Abnormal Uterine Bleeding  ...  1.0  3.0  0.250000
2    195659   Aviane  Abnormal Uterine Bleeding  ...  3.0  2.0  0.600000
3    195622   Aviane  Abnormal Uterine Bleeding  ...  0.0  2.0  0.000000
4    147702   Aviane  Abnormal Uterine Bleeding  ...  7.0  5.0  0.583333
5    195425   Aviane  Abnormal Uterine Bleeding  ...  0.0  2.0  0.000000
6     90721   Aviane  Abnormal Uterine Bleeding  ...  0.0  0.0  0.000000
7     58330   Aviane  Abnormal Uterine Bleeding  ...  3.0  1.0  0.750000
8     58307   Aviane  Abnormal Uterine Bleeding  ...  1.0  4.0  0.200000
9      5389   Aviane  Abnormal Uterine Bleeding  ...  1.0  0.0  1.000000
```

**Figure 16:** Dictionary sentiment Polarity

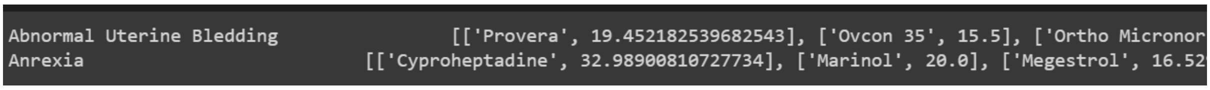
### Calculating Score of Each Review:



```
   uniqueID drugName                  condition  ...       DSP    0     score1
0     53641   Aviane  Abnormal Uterine Bleeding  ...  0.000000  5.0   5.000000
1    195548   Aviane  Abnormal Uterine Bleeding  ...  0.250000  5.0   6.500000
2    195659   Aviane  Abnormal Uterine Bleeding  ...  0.600000  5.0  10.400000
3    195622   Aviane  Abnormal Uterine Bleeding  ...  0.000000  5.0   5.000000
4    147702   Aviane  Abnormal Uterine Bleeding  ...  0.583333  5.0   7.333333
5    195425   Aviane  Abnormal Uterine Bleeding  ...  0.000000  5.0   5.000000
6     90721   Aviane  Abnormal Uterine Bleeding  ...  0.000000  5.0   5.000000
7     58330   Aviane  Abnormal Uterine Bleeding  ...  0.750000  5.0   6.500000
8     58307   Aviane  Abnormal Uterine Bleeding  ...  0.200000  5.0   5.000000
9      5389   Aviane  Abnormal Uterine Bleeding  ...  1.000000  5.0  16.000000
```

**Figure 17:** Score calculation

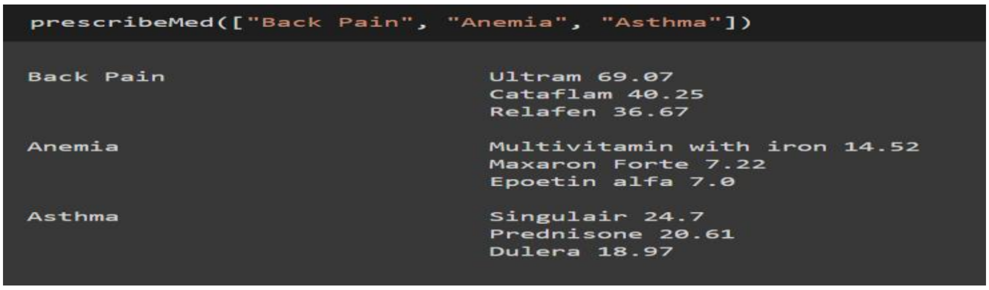
### Grouping Conditions and Drugs:



```
Abnormal Uterine Bledding             [['Provera', 19.452182539682543], ['Ovcon 35', 15.5], ['Ortho Micronor
Anrexia                         [['Cyproheptadine', 32.98900810727734], ['Marinol', 20.0], ['Megestrol', 16.52
```

**Figure 18:** Grouping Conditions

### RECOMMENDING DRUGS:



```
prescribeMed(["Back Pain", "Anemia", "Asthma"])

Back Pain                          Ultram 69.07
                                   Cataflam 40.25
                                   Relafen 36.67

Anemia                             Multivitamin with iron 14.52
                                   Maxaron Forte 7.22
                                   Epoetin alfa 7.0

Asthma                             Singulair 24.7
                                   Prednisone 20.61
                                   Dulera 18.97
```

**Figure 19:** Recommending drugs

## 5. Conclusion

At last, the Naïve Bayes model is chosen for rating age to get a decent compromise among model exactness (60.0%), model productivity, and model versatility where this outcome is utilized in Hybrid Recommendation Model to list proper meds.

- Notwithstanding it, we directed the passionate investigation utilizing an enthusiastic word reference to defeat constraints of the medication information utilized.
- In the last investigation this study shows that the wistful qualities contribute significantly to the expectation of medication rating, just as suggestions. It additionally shows huge enhancements for a genuine world dataset contrasted with current techniques.

## 6. Future Scope

The scope of this task is that while assessing the unique circumstance, we can track down more phonetic standards, and to fuse state level opinion examination, we might adjust or fabricate half and half factorization models like tensor factorization, or profound learning strategies. The venture can likewise be stretched out to improve the exactness and unwavering quality of the proposal model further.

## 7. References

[1] B. Liu, Sentiment Analysis (Introduction and Survey) and Opinion Mining. 2012.

[2] X. Lei, X. Qian, and G. Zhao, "Rating Prediction Based on Social Sentiment from Textual Reviews," IEEE Trans. Multimed., vol. 18, no. 9, pp. 1910–1921, Sep. 2016, doi: 10.1109/TMM.2016.2575738.

[3] Y. Bao and X. Jiang, "An intelligent medicine recommender system framework," in Proceedings of the 2016 IEEE 11th Conference on Industrial Electronics and Applications, ICIEA 2016, Oct. 2016, pp. 1383–1388, doi: 10.1109/ICIEA.2016.7603801.

[4] R. Majethia, V. Mishra, A. Singhal, K. Lakshmi Manasa, K. Sahiti, and V. Nandwani, "PeopleSave: Recommending effective drugs through web crowdsourcing," in 2016 8th International Conference on Communication Systems and Networks, COMSNETS 2016, Mar. 2016, doi: 10.1109/COMSNETS.2016.7440000.

[5] R. C. Chen, Y. H. Huang, C. T. Bau, and S. M. Chen, "A recommendation system based on domain ontology and SWRL for anti-diabetic drugs selection," Expert Syst. Appl., vol. 39, no. 4, pp. 3995–4006, Mar. 2012, doi: 10.1016/j.eswa.2011.09.061.

[6] J.-C. Na and W. Y. M. Kyaing, "Sentiment Analysis of User-Generated Content on Drug Review Websites," J. Inf. Sci. Theory Pract., vol. 3, no. 1, pp. 6–23, Mar. 2015, doi: 10.1633/jistap.2015.3.1.1.

[7] M. E. Basiri, M. Abdar, M. A. Cifci, S. Nemati, and U. R. Acharya, "A novel method for sentiment classification of drug reviews using fusion of deep and machine learning techniques," Knowledge-Based Syst., vol. 198, p. 105949, Jun. 2020, doi: 10.1016/j.knosys.2020.105949.

[8] S. Vijayaraghavan and D. Basu, "Sentiment Analysis in Drug Reviews using Supervised Machine Learning Algorithms," arXiv, Mar. 2020, Accessed: Nov. 20, 2020. [Online]. Available: http://arxiv.org/abs/2003.11643.

[9] V. Doma et al., "Automated Drug Suggestion Using Machine Learning," in Advances in Intelligent Systems and Computing, Mar. 2020, vol. 1130 AISC, pp. 571–589, doi: 10.1007/978-3-030-39442-4_42.

[10] A. A. Hamed, R. Roose, M. Branicki, and A. Rubin, "TRecs: Time-aware twitter-based drug recommender system," in Proceedings of the 2012 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, ASONAM 2012, 2012, pp. 1027– 1031, doi: 10.1109/ASONAM.2012.178.

[11] C. Chen, L. Zhang, X. Fan, Y. Wang, C. Xu, and R. Liu, "A epilepsy drug recommendation system by implicit feedback and crossing recommendation," in Proceedings - 2018 IEEE SmartWorld, Ubiquitous Intelligence and Computing, Advanced and Trusted Computing, Scalable Computing and Communications, Cloud and Big Data Computing, Internet of People and Smart City Innovations, SmartWorld/UIC/ATC/ScalCom/CBDCo, 2018, doi: 10.1109/SmartWorld.2018.00197.

[12] D. Chen, D. Jin, T. T. Goh, N. Li, and L. Wei, "ContextAwareness Based Personalized Recommendation of AntiHypertension Drugs," J. Med. Syst., vol. 40, no. 9, pp. 1– 10, Sep. 2016, doi: 10.1007/s10916-016-0560-z.

[13] A. Gottlieb, G. Y. Stein, E. Ruppin, R. B. Altman, and R. Sharan, "A method for inferring medical diagnoses from patient similarities," BMC Med., vol. 11, no. 1, p. 194, Sep. 2013, doi: 10.1186/1741-7015-11-194.

[14] K. Shimada, K. Fujikawa, K. Yahara, and T. Nakamura, “Antioxidative Properties of Xanthan on the Autoxidation of Soybean Oil in Cyclodextrin Emulsion,” 1992. Accessed: Jul. 29, 2020. [Online]. Available: https://pubs.acs.org/sharingguidelines.

[15] Q. Zhang, G. Zhang, J. Lu, and D. Wu, “A framework of hybrid recommender system for personalized clinical prescription,” in Proceedings - The 2015 10th International Conference on Intelligent Systems and Knowledge Engineering, ISKE 2015, Jan. 2016, pp. 189–195, doi: 10.1109/ISKE.2015.98.