Pimpri Chinchwad Education Trust's
[PCET-A trusted brand in Education Since 1990]

Pimpri Chinchwad College of Engineering

(An Autonomous Institute, affiliated to Savitribai Phule Pune University)

MINI PROJECT AND BASICS OF INNOVATION OF

FY B.Tech (Computer Science and Engineering)

Artificial Intelligence and Machine Learning

Department of Applied Sciences & Humanities

(Course 2020-21)

With effect from Academic Year 2022-2023

(Updated with Minor Changes)

A PROJECT REPORT ON

CONTENT RECOMMENDATION BASED ON EMOTIONS IN PYTHON

PIMPRI CHINCHWAD COLLEGE OF ENGINEERING, PUNE [PCCOE]
(AN AUTONOMOUS INSTITUTE)
NBA Accredited | NAAC Accredited with 'A' Grade | An Autonomous Institute | AICTE Approved
| ISO 9001: 2015 Certified | Permanently Affiliated to SPPU, Pune

FOR FULFILLMENT OF THE
MINI PROJECT AND BASICS
OF
INNOVATION OF F.Y.B.TECH

Submitted by
Swetank Raut
Harsh Vahal
Rohan Shivpure
Mrunalini Phutane
Nishka Salunkhe

in

FEBRUARY 2023

CERTIFICATE

Pimpri Chinchwad Education Trust's

Pimpri Chinchwad College of Engineering

Sector No. 26, Pradhikaran,

Nigdi, Pune – 411 044

Certified that this project report " CONTENT RECOMMENDATION BASED ON EMOTIONS IN PYTHON " is the bonafide work of

Swetank Raut	(Roll no. FY22P058)
Harsh Vahal	(Roll no. FY22P068)
Rohan Shivpure	(Roll no. FY22P059)
Mrunalini Phutane	(Roll no. FY22P055)
Nishka Salunkhe	(Roll no. FY22P060)

Submitted to the Pimpri Chinchwad College Of Engineering ,Pune In the fulfillment of the requirement for the MPBI activity of "First Year Bachelor of Technology ".

Date : 15/02/23

Dr. Amol P. Kharche	DR. S. S. Sonar
Guide	**Head of the Department**

Department of Applied Science & Humanities

DECLARATION OF CANDIDATES

We,the undersigned solemnly declare that the project entitled “**CONTENT RECOMMENDATION BASED ON EMOTIONS IN PYTHON “** is an outcome of our efforts under the guidance of **Prof. Dr.Amol P. Kharche**. The project is submitted to the Applied Science and Humanities Department for the partial fulfillment of the FY B.Tech. in Computer Science & Engineering (AI And ML) Department.

We also declare that this project report has not been previously submitted to the Applied Science and Humanities Department.

Date: 15/02/23 Signature of Candidates

Place:Pune

1.Swetank Raut

2.Harsh Vahal

3.Rohan Shivpure

4.Mrunalini Phutane

5.Nishka Salunkhe

ACKNOWLEDGEMENT

We are deeply grateful to our project guide, Dr. Amol P. Kharche Sir, Dept. of Applied Sciences and Humanities, PCCOE, Nigdi, Pune, who gave us this wonderful opportunity to work on the topic **CONTENT RECOMMENDATION BASED ON EMOTIONS IN PYTHON**. In addition to helping us do a lot of research, he also introduced us to so many new things. We are really grateful to him for that. His generous attitude led to the creation of this report.

Secondly, we wish to express our sincere thanks to the Management of PCCOE, the Director of PCCOE, Pune, and the Head of Department. Of Applied Science and Humanities and for their guidance and all our teachers for their constant motivation.

Lastly, we would like to express our deepest gratitude to our parents for their deep-rooted love, blessings, prayers, sacrifices, care and support that paved the way for us.

Swetank Raut

Harsh Vahal

Rohan Shivpure

Mrunalini Phutane

Nishka Salunkhe

ABSTRACT

There are a lot of applications where websites collect data from their users and use that data to predict the likes and dislikes of their users . This allows them to recommend the content that they like. Recommender systems are a way of suggesting or similar items and ideas to a user's specific way of thinking.

Technically nowadays there is a rapid widening when it comes to the field of Data Science, Machine learning, Deep learning and Artificial Intelligence. Data acquisition is more efficient through the recommendation systems. The Recommender engines or recommendation systems are the major wing of Data Analytics, Machine learning algorithms and Artificial intelligence which is conducted by the software engineers to improve the quality of searching results and predicts the users' ratings / rankings on a particular item/ product or commodity, then offers relevant recommendations to the users and returns back the user's preferences *(Shin and Mckay, 1984)*. These recommender systems entice users with enhanced experience and pure joy. In this project work is a Movie recommender system is built by loading the datasets from Kaggle Website, then various filtering algorithms like demographic filtering, Content-based filtering, emotion / mood-based filtering through web scraping methodologies are used which makes recommendations based on the users' preferences, rating, genre, matching the terms, experiences, emotions, popularity and collects information to eventually extract the final movie recommendation systems. In this project both personalized (user specific) and non-personalized (common) recommendations are processed.

In today's world, we have tons of unstructured data/information (mostly web data) available freely. Sometimes the freely available data is easy to read and sometimes not. No matter how your data is available, web scraping is a very useful tool to transform unstructured data into structured data that is easier to read & analyze. In other words, one way to collect, organize and analyze this enormous amount of data is through web scraping.

Table Of Contents

1. Synopsis

1.1 Information on Recommendation System Based on Emotions.

One of the underlying targets of movies is to evoke emotions in their viewers. IMDb offers all the movies for all genre. Therefore the movie titles can be scraped from the IMDb list to recommend to the user. IMDb does not have an API, for accessing information on movies and TV Series. Therefore we have to perform scraping. Scraping is used for accessing information from a website which is usually done with APIs.

Definition of Emotion:

Emotion literally means "disturbance." The word comes from the Latin *emovere*, meaning "to disturb." Characters who reside more in the mind and their thoughts than in their body and their emotions put distance between the story and the audience. Thoughts can lie. Dialogue can lie, too. However, emotions are universal, relatable and humanizing. Emotions always tell the truth.

Emotion associated with Genre of Movie

There are 8 classes of emotion that would be effective to classify a text. These are: *'Anger', 'Anticipation', 'Disgust', 'Fear', 'Joy', 'Sad', 'Surprise', 'Trust'*. Here these are taken as input and the corresponding movies would be displayed for the emotion. *("Our Basic Emotions Infographic | List of Human Emotions | UWA Online" 2019)*

The correspondence of every emotion with genre of movies is listed below:

Sad – Drama

Disgust – Musical

Anger – Family

Anticipation – Thriller

Fear – Sport

Enjoyment – Thriller , Humour.

Trust – Western

Surprise – Film-Noir

Based on the input emotion, the corresponding genre would be selected and all the top 5 movies of that genre would be recommended to the user.

EFFECTS OF MOVIES ON US :

1. Movies inspire us:

   A good movie will entertain, educate, and inspire the viewer in many ways. Think of the impact that songs have on people, for example. They make us think. They make us compassionate. They inspire us to help others and to do good to and for humanity. Romantic movies, on the other hand, remind us why love is important and why it is worth fighting for. They make us cry and laugh at our own romantic flaws, consequently helping us understand our partners and family members more. They make life worth living- they make us feel alive. That's without forgetting how affordable transcription services have made it easy for people to watch and understand movies that teach meditation and mindfulness, most of which are in Asian languages. There are many films whose plots give us reasons to rise up every morning and venture into the world with hope and optimism. They encourage us to conquer personal pains and to impact positively on other people's lives. Movies such as The Pursuit of Happiness (2006) and The Bucket List (2007), for example, have inspired their viewers to work towards making the world better for everyone.

2. Movies can create awareness on multiple aspects of life:

   People need to be reminded about the importance of formal education as well as co-curricular school activities such as art and sports. School-related films underline this importance and give education stakeholders ideas on how to improve education syste ms in different parts of the world. Besides that, film brings us to understand the negative effects of drugs, alcohol, and substance abuse. Crime and action TV shows also warn us about the dangers of criminal activities, terrorism, and war. Speaking of war, movies help people understand the atrocity of living homeless and miserably in refugee camps. Movies awaken the senses of empathy in people who have never experienced civil war firsthand. They help us feel responsible for our brothers and sisters living in war-torn countries even as much as we've never been there ourselves. This is the awareness that has fuelled the growth of so many charity organizations and trust funds.

3. Movies mirror culture:

Every movie is set and developed in a particular culture. They are an integral part of us; they mirror what we believe in and how we coexist as people. It is easier to see our concerns, attitudes, flaws, and strengths in films than it is to decipher them from our daily interactions. When our prevalent beliefs and ideologies are challenged in films, we are able to interrogate ourselves and embrace change. And thanks to audiovisual translations, people from all over the world are able to watch movies and understand the cultures of faraway communities. That has, in turn, helped us to become more united even when our cultures are so different and diverse.

4. Movies shape culture:

Besides mirroring our diverse cultures, the film has for a long time been shaping our beliefs and values. A good example is when people copy fashion trends from movie stars and musicians. It is also common these days to find societies using figures of speech that are inspired by the film industry. At the very least, film solidifies selected cultural beliefs and renders some redundant.

5. Movies teach us history:

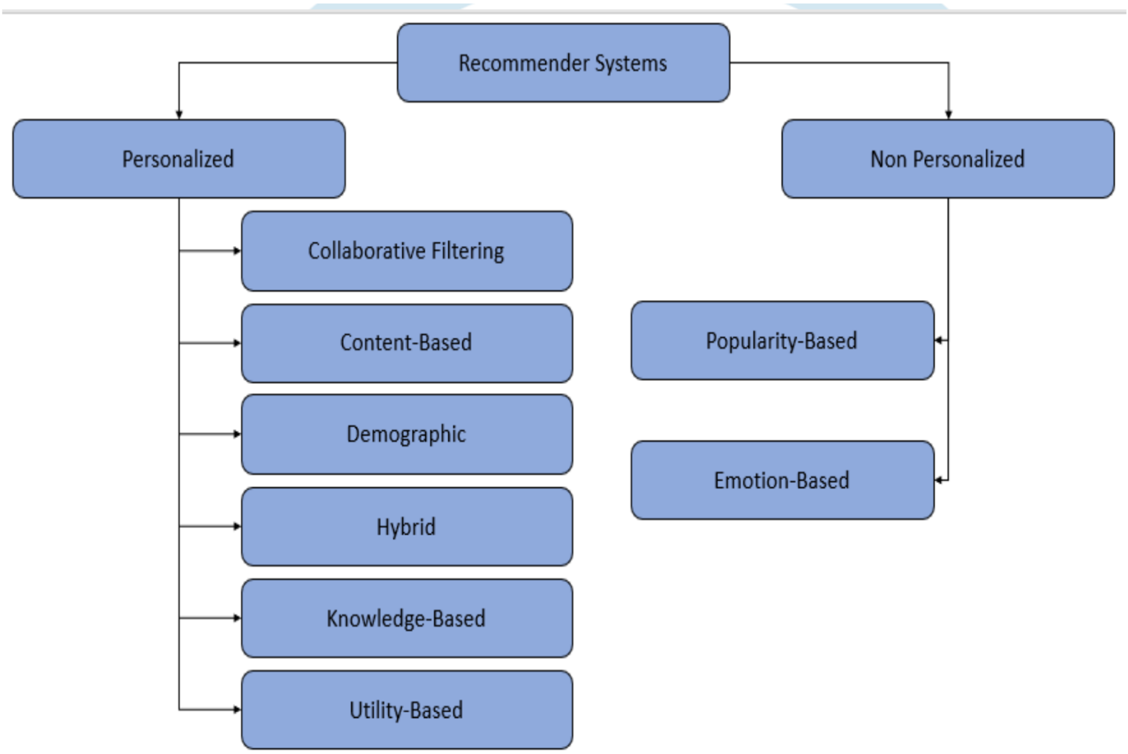
Most history movies are fact-based and, even for the few that are fictional, they still depict a realistic and fantastic picture of how the world was before the invention of the audiovisual devices that we know today. They connect the modern world with past generations. A good example is the Vietnam War movies that explain what transpired back in the day and help today’s generation to appreciate the significances of the war, both positive and negative. Add transcription services to these movies and the history comes out even clearer. Transcribed historical videos support learning by connecting viewers to the proper subject matter without losing the context. And then the negatives

1.2 LITERATURE REVIEW:

Recommendation systems are one sort of information gathering and filtering mechanism which makes the search engine to improve and provide more relevant / related items to the user's preferences. These systems are designed to predict the user's opinion they can have on any product or commodity based on previous searches. Previous works on Recommendation Systems:

1. **Recommendation System with Hybrid systems:** In this paper the system uses hybrid recommender systems to use contextual information and ratings of the users. It makes improvements on user's ratings. *(de Compos and Fernández-Luna, 2010)*
2. **Video Games Recommendation Systems:** It recommends the games for users based on their searches and likes which they can browse and purchase the games they like and play. *(IJRTI © 2022)*
3. **Diet Recommendation System:** This system is designed for recommending the proper diet/ food for patients based on their age, weight, calories, proteins, sodium, fat by the implementation of machine learning and deep learning concepts like RNN, MLP etc *(Al-Shamri and Bharadwaj, 2008)*
4. **Articles Recommendation System:** It recommends the articles for the readers based on their content, reading habits, genre and categories based on the content and collaborative filtering methods of machine learning concepts.
5. **Tinder:** An Application which gives recommendations for users about the similarities, likes, dislikes, common grounds and preferences in different sectors based on machine learning and deep learning concepts to get user's perfect match for dating.
6. **Netflix/ Amazon prime/ Hayu:** All these applications give the users with different recommendations on Movies, Television shows, Documentaries and web series etc based hybrid recommendation systems.
7. **Make my Trip:** The Tourist places recommendation guide which plans the tours, makes recommendations based on the budget, interests, previous places visited. These Apps will also recommend local places and food to new travelers. *(Hu Jinning, 2010)*
8. **Facebook:** Facebook recommends the mutual friends, posts and similar interests that people like and want to expand their business through social media to reach out to many people

1.2.1 Different Types of Recommended Systems:



2. Introduction

2.1 Overview:

It's a python project that suggest movies based on your present emotion/mood on real time without using any Database.

2.2 Problem Domain:

1. Large number of unsorted movies are available online.
2. Viewer is unable to find the movies that he/she wants to watch.
3. Movies are not available on the basis of mood/emotion of the user.
4. Existing software doesn't suggest movies to improve the viewer's mood.

2.3 Solution Domain:

1. We are making software that is based on the present mood of the user.
2. This software will give movie suggestions to the viewer according to their present mood.
3. This software will suggest the latest and trending movies based on their mood from a large variety of movies online by "Web Scraping".

2.4 Objective OR Purpose:

1. The main purpose of the software is to suggest a list of movies to the viewer, from the large amount of content available online, based on his/her emotion.
2. Another purpose is to suggest a list of latest and trending movies.

2.5 An improvement over the existing system:

1. We are suggesting movies based on 'Human Emotion' or their present mood, which is not present in the existing system.
2. Rather than storing movies on database we are using "Web Scraping" to suggest latest featured films using internet.

   Expected Outcomes/ Benefits: -

3. Outcome: - "List of latest featured movies based on the user's input i.e. mood of user/emotion."

3. SOFTWARE REQUIREMENTS SPECIFICATION

A Software requirements specification document describes the intended purpose, requirements, and nature of software to be developed. It also includes the yield and cost of the software.

3.1 EXTERNAL INTERFACE REQUIREMENTS

3.1.1 User Interface

Front-End Software: Python version 3.8 or above

3.1.2 Hardware Interface

• Memory: 4Gb or above

• Graphics Card: NVIDIA GeForce GTX 970 or above CPU: Intel Core i5-4590 or above

3.1.3 Software Interfaces

• Python 3
• Operating System: Windows 7 or above

3.2 NON-FUNCTIONAL REQUIREMENTS

3.2.1 Performance Requirements

The system should be able to detect and classify the raw visual data into

different categories accurately.

The output should be quick and accurate.

3.3 SOFTWARE AND QUALITY ATTRIBUTES

- **Performance efficiency:** the capability of the software product to provide appropriate performance, relative to the amount of resources used, under stated conditions.

- **Compatibility:** The system must be compatible with all browsers
- **Maintainability:** The system should be fault-tolerant & easily maintainable.
- **Availability:** The system should be readily available to take input provided by the user.
- **Usability:** The system should be user-friendly & easy to use.

3.4 SYSTEM REQUIREMENTS

3.4.1 Database Requirements:

External Links as user desired from trusted and user friendly websites like IMDB.

3.4.2 Hardware requirement:

Memory: 4Gb or above

Graphics Card: NVIDIA GeForce GTX 970 or above CPU: Intel Core i5-4590 or above

3.4.3 Software Requirement:

- Front End - Google Collab Python Notebook/ Visual Studio IDE
- Database - Google Drive
- Database Connectivity - BeautifulSoup4 Module to Access Dataset from websites on different browsers.
- IDE-Google Colab/ Visual Studio
- Server-Google's Cloud Server
- Browser-Google / Firefox/Microsoft edge

❖ Packages:

- Beautiful Soup
- Requests
- Urllib
- Path from pathlib *(Hug, 2020)*

4. Methodology

4.1 Project Working:

- **Firstly we created a Virtual Environment (Optional) on Windows.**
- **Then we installed the library called as BeautifulSoup for that we used BeautifulSoup4 package (known as bs4)**
- **Also the Requests library was installed in the system.**
- **Then we debugged and solved the problems faced during actual handling of HTML files for that we installed a Parser**
- **Then Testing and Running of Beautiful Soup is done for satisfactorily working of program**
- **Installed latest version of Python (till 08-02-23) i.e. python 3.11.2**
- **Performed Actual Programming in Visual studio code IDE**
- **Next we performed some tests on our program by putting some specific human emotion based on instructions which were already stored in the program.**
- **At the end, the actual links of movies based on different categories were stored in the program and the program was executed succefully.**

4.2 Tools And Technology Used;

4.2.1 Tools:

- Visual Studio Code/ Google Colab: For the back-end programming
- Google Drive: For storing the database and results
- Google Chrome: For the graphical interface
- Google Docs: For creating and working on the report

4.2.2 Technologies:

Web Scraping:

Scraping is simply a process of extracting (from various means), copying and screening of data. *(Amos 2022)*

When we do scraping or extracting data or feeds from the web (like from web-pages or websites), it is termed as web-scraping.

So, web scraping which is also known as web data extraction or web harvesting is the extraction of data from the web. In short, web scraping provides a way to the developers to collect and analyze data from the internet.

Python:

Python is one of the most popular languages for web scraping as it can handle most of the web crawling related tasks very easily. *("Install python PIP, Requests and Beautiful soup for WINDOWS (in 5 minutes)" 2018)*

Below are some of the points on why to choose python for web scraping:

- Ease To Use

  As most of the developers agree that python is very easy to code. We don't have to use any curly braces "{ }" or semi -colons ";" anywhere, which makes it more readable and easy-to-use while developing web scrapers.

- Huge Library Support

  Python provides a huge set of libraries for different requirements, so it is appropriate for web scraping as well as for data visualization, machine learning, etc.

- Easily Explicable Syntax

  Python is a very readable programming language as python syntax are easy to understand. Python is very expressive and code indentation helps the users to differentiate different blocks or scopes in the code.

- Dynamically-typed language

  Python is a dynamically-typed language, which means the data assigned to a variable tells what type of variable it is. It saves a lot of time and makes work faster. *(Shin and Mckay, 1984)*

- Huge Community

  The Python community is huge which helps you wherever you are stuck while writing code.

EXAMPLE ON WEB SCRAPING :

INPUT :

```
import requests
# Making a GET request
r = requests.get('https://www.geeksforgeeks.org/python-programming-language/')
# print request object
print(r.url)
# print status code
print(r.status_code)
```

OUTPUT

```
https://www.geeksforgeeks.org/python-programming-language/
200
```

Introduction to Beautiful Soup

The Beautiful Soup is a python library which is named after a Lewis Carroll poem of the same name in “Alice’s Adventures in Wonderland”. Beautiful Soup is a python package and as the name suggests, parses the unwanted data and helps to organize and format the messy web data by fixing bad HTML and present to us in easily-traversable XML structures. *(“Beautiful Soup - Installation”, n.d.)*

In short, Beautiful Soup is a python package which allows us to pull data out of HTML and XML documents.

5. EXPERIMENTATION

5.1 Emotion associated with Genre of Movie

There are 8 classes of emotion that would be effective to classify a text. These are: *'Anger', 'Anticipation', 'Disgust', 'Fear', 'Joy', 'Sad', 'Surprise', 'Trust'*. Here these are taken as input and the corresponding movies would be displayed for the emotion. *("Our Basic Emotions Infographic | List of Human Emotions | UWA Online" 2019)*

The correspondence of every emotion with genre of movies is listed below:

Sad – Drama

Disgust – Musical

Anger – Family

Anticipation – Thriller

Fear – Sport

Enjoyment – Thriller

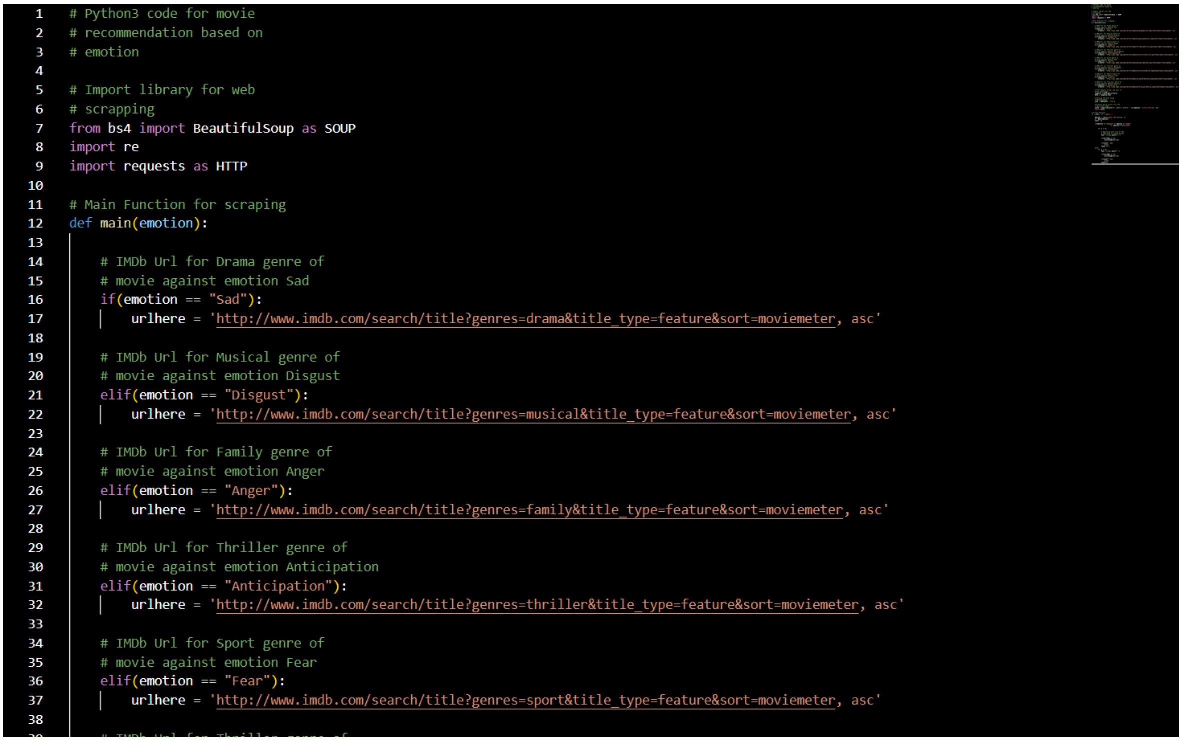
Trust – Western

Surprise – Film-Noir

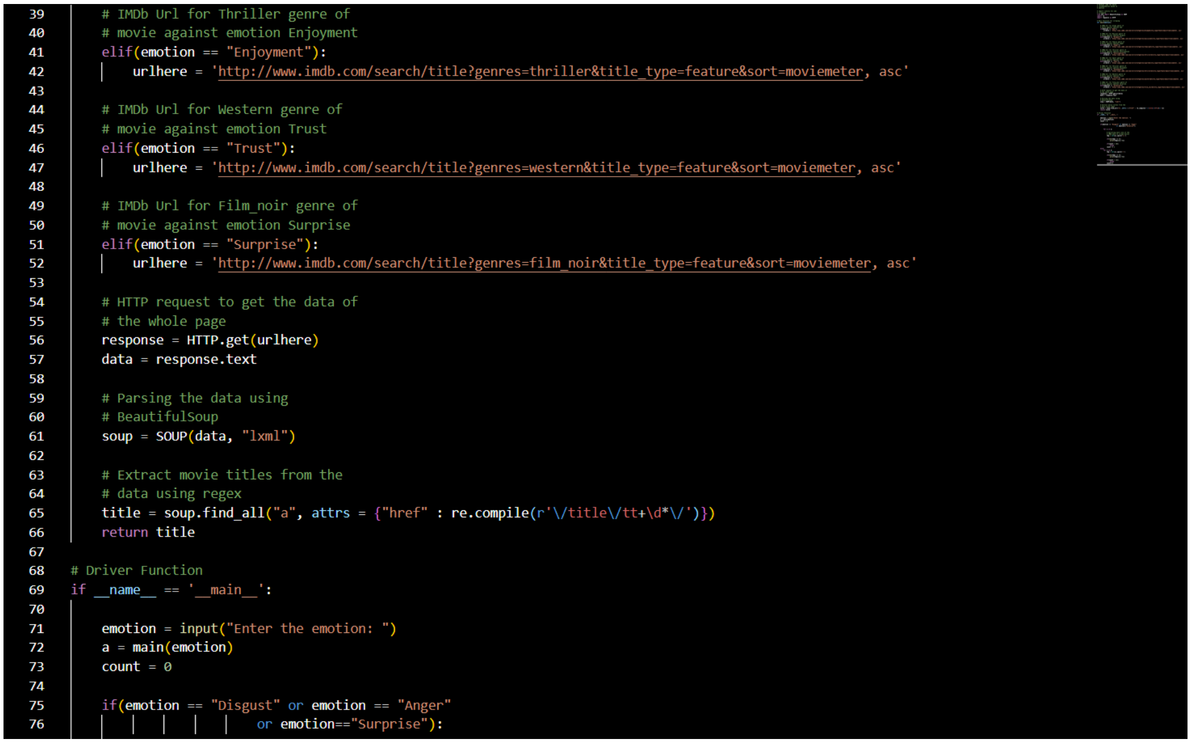
Based on the input emotion, the corresponding genre would be selected and all the top 5 movies of that genre would be recommended to the user.

5.2 SOURCE CODE :

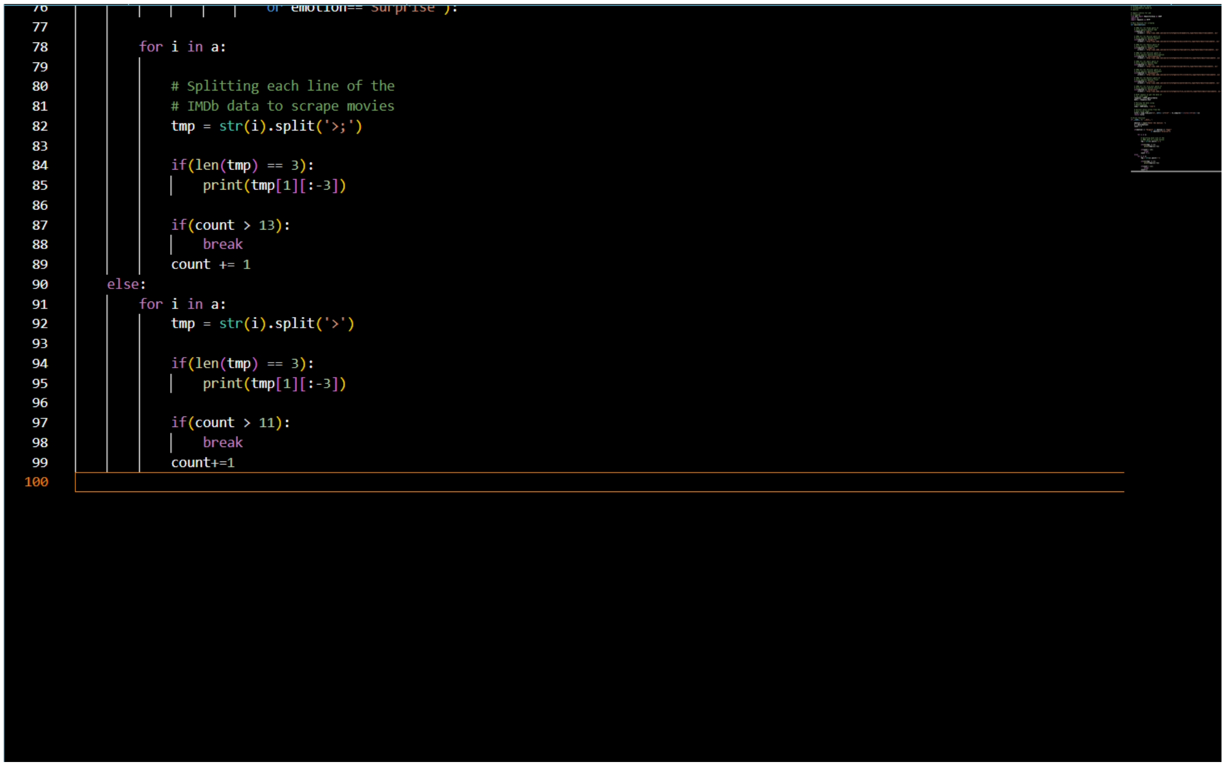
5.2.1 AS AN IMAGE :



```
# Python3 code for movie
# recommendation based on
# emotion

# Import library for web
# scrapping
from bs4 import BeautifulSoup as SOUP
import re
import requests as HTTP

# Main Function for scraping
def main(emotion):

    # IMDb Url for Drama genre of
    # movie against emotion Sad
    if(emotion == "Sad"):
        urlhere = 'http://www.imdb.com/search/title?genres=drama&title_type=feature&sort=moviemeter, asc'

    # IMDb Url for Musical genre of
    # movie against emotion Disgust
    elif(emotion == "Disgust"):
        urlhere = 'http://www.imdb.com/search/title?genres=musical&title_type=feature&sort=moviemeter, asc'

    # IMDb Url for Family genre of
    # movie against emotion Anger
    elif(emotion == "Anger"):
        urlhere = 'http://www.imdb.com/search/title?genres=family&title_type=feature&sort=moviemeter, asc'

    # IMDb Url for Thriller genre of
    # movie against emotion Anticipation
    elif(emotion == "Anticipation"):
        urlhere = 'http://www.imdb.com/search/title?genres=thriller&title_type=feature&sort=moviemeter, asc'

    # IMDb Url for Sport genre of
    # movie against emotion Fear
    elif(emotion == "Fear"):
        urlhere = 'http://www.imdb.com/search/title?genres=sport&title_type=feature&sort=moviemeter, asc'

```



```
    # IMDb Url for Thriller genre of
    # movie against emotion Enjoyment
    elif(emotion == "Enjoyment"):
        urlhere = 'http://www.imdb.com/search/title?genres=thriller&title_type=feature&sort=moviemeter, asc'

    # IMDb Url for Western genre of
    # movie against emotion Trust
    elif(emotion == "Trust"):
        urlhere = 'http://www.imdb.com/search/title?genres=western&title_type=feature&sort=moviemeter, asc'

    # IMDb Url for Film_noir genre of
    # movie against emotion Surprise
    elif(emotion == "Surprise"):
        urlhere = 'http://www.imdb.com/search/title?genres=film_noir&title_type=feature&sort=moviemeter, asc'

    # HTTP request to get the data of
    # the whole page
    response = HTTP.get(urlhere)
    data = response.text

    # Parsing the data using
    # BeautifulSoup
    soup = SOUP(data, "lxml")

    # Extract movie titles from the
    # data using regex
    title = soup.find_all("a", attrs = {"href" : re.compile(r'\/title\/tt+\d*\/')})
    return title

# Driver Function
if __name__ == '__main__':

    emotion = input("Enter the emotion: ")
    a = main(emotion)
    count = 0

    if(emotion == "Disgust" or emotion == "Anger"
                        or emotion=="Surprise"):
```



```
                        or emotion=="Surprise"):

        for i in a:

            # Splitting each line of the
            # IMDb data to scrape movies
            tmp = str(i).split('>;')

            if(len(tmp) == 3):
                print(tmp[1][:-3])

            if(count > 13):
                break
            count += 1
    else:
        for i in a:
            tmp = str(i).split('>')

            if(len(tmp) == 3):
                print(tmp[1][:-3])

            if(count > 11):
                break
            count+=1
```

5.2.2 AS A TEXT :

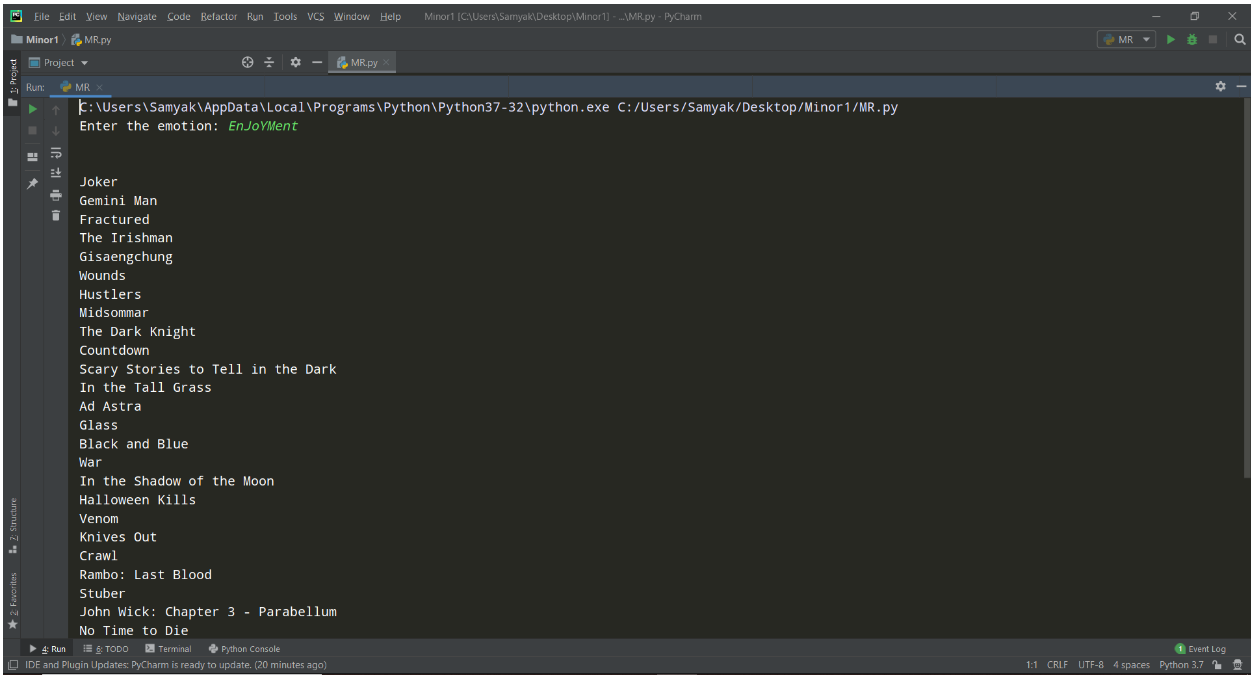
```
# Python3 code for movie
# recommendation based on
# emotion

# Import library for web
# scrapping

from bs4 import BeautifulSoup as SOUP
import re
import requests as HTTP

# Main Function for scraping
def main(emotion):

        # IMDb Url for Drama genre of
        # movie against emotion Sad
        if(emotion == "Sad"):
```

```
urlhere='http://www.imdb.com/search/title?genres=drama&title_type=feature&sort=moviemeter, asc'

                # IMDb Url for Musical genre of
                # movie against emotion Disgust
                elif(emotion == "Disgust"):

urlhere='http://www.imdb.com/search/title?genres=musical&title_type=feature&sort=moviemeter, asc'

                # IMDb Url for Family genre of
                # movie against emotion Anger
                elif(emotion == "Anger"):

urlhere='http://www.imdb.com/search/title?genres=family&title_type=feature&sort=moviemeter, asc'

                # IMDb Url for Thriller genre of
                # movie against emotion Anticipation
                elif(emotion == "Anticipation"):

urlhere='http://www.imdb.com/search/title?genres=thriller&title_type=feature&sort=moviemeter, asc'

                # IMDb Url for Sport genre of
                # movie against emotion Fear
                elif(emotion == "Fear"):

urlhere='http://www.imdb.com/search/title?genres=sport&title_type=feature&sort=moviemeter, asc'

                # IMDb Url for Thriller genre of
                # movie against emotion Enjoyment
                elif(emotion == "Enjoyment"):

urlhere='http://www.imdb.com/search/title?genres=thriller&title_type=feature&sort=moviemeter, asc'

                # IMDb Url for Western genre of
                # movie against emotion Trust
```

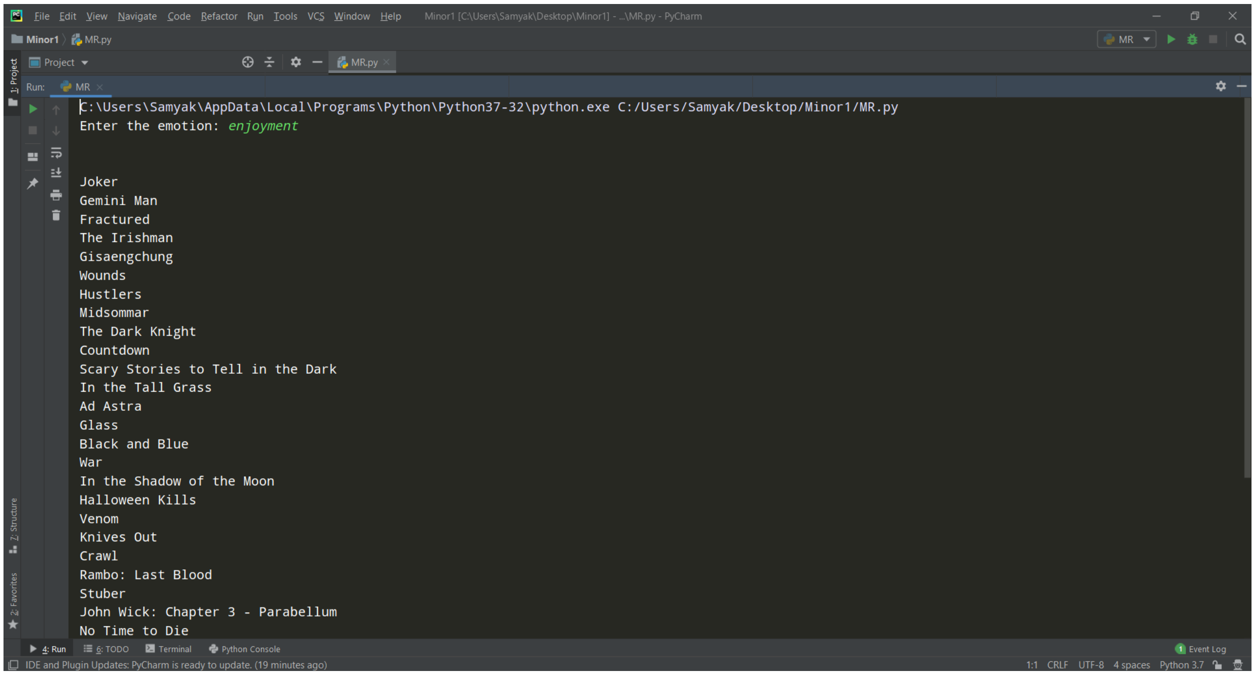
```
        elif(emotion == "Trust"):

urlhere='http://www.imdb.com/search/title?genres=western&title_type=feature&sort=mov
iemeter, asc'

        # IMDb Url for Film_noir genre of
        # movie against emotion Surprise
        elif(emotion == "Surprise"):

urlhere='http://www.imdb.com/search/title?genres=film_noir&title_type=feature&sort=mo
viemeter, asc'

        # HTTP request to get the data of
        # the whole page
        response = HTTP.get(urlhere)
        data = response.text

        # Parsing the data using
        # BeautifulSoup
        soup = SOUP(data, "lxml")

        # Extract movie titles from the
        # data using regex
        title = soup.find_all("a", attrs = {"href" : re.compile(r'\/title\/tt+\d*\/')})
        return title

    # Driver Function
    if __name__ == '__main__':

        emotion = input("Enter the emotion: ")
        a = main(emotion)
        count = 0

        if(emotion == "Disgust" or emotion == "Anger"
                                    or emotion=="Surprise"):

            for i in a:

                # Splitting each line of the
                # IMDb data to scrape movies
                tmp = str(i).split('>;')

                if(len(tmp) == 3):
                    print(tmp[1][:-3])

                if(count > 13):
                    break
                count += 1
        else:
            for i in a:
                tmp = str(i).split('>')

                if(len(tmp) == 3):
                    print(tmp[1][:-3])

                if(count > 11):
                    break
                count+=1
```

6. RESULT AND DISCUSSION

6.1 Snap Shots of Code Output:


File Edit View Navigate Code Refactor Run Tools VCS Window Help
Minor1 [C:\Users\Samyak\Desktop\Minor1] - ...\MR.py - PyCharm
Minor1
MR.py
Project
Run: MR
4: Run
6: TODO
Terminal
Python Console
Event Log
IDE and Plugin Updates: PyCharm is ready to update. (20 minutes ago)
1:1 CRLF UTF-8 4 spaces Python 3.7

```
C:\Users\Samyak\AppData\Local\Programs\Python\Python37-32\python.exe C:/Users/Samyak/Desktop/Minor1/MR.py
Enter the emotion: EnJoYMent


Joker
Gemini Man
Fractured
The Irishman
Gisaengchung
Wounds
Hustlers
Midsommar
The Dark Knight
Countdown
Scary Stories to Tell in the Dark
In the Tall Grass
Ad Astra
Glass
Black and Blue
War
In the Shadow of the Moon
Halloween Kills
Venom
Knives Out
Crawl
Rambo: Last Blood
Stuber
John Wick: Chapter 3 - Parabellum
No Time to Die
```


File Edit View Navigate Code Refactor Run Tools VCS Window Help
Minor1 [C:\Users\Samyak\Desktop\Minor1] - ...\MR.py - PyCharm
Minor1
MR.py
Project
Run: MR
4: Run
6: TODO
Terminal
Python Console
Event Log
IDE and Plugin Updates: PyCharm is ready to update. (19 minutes ago)
1:1 CRLF UTF-8 4 spaces Python 3.7

```
C:\Users\Samyak\AppData\Local\Programs\Python\Python37-32\python.exe C:/Users/Samyak/Desktop/Minor1/MR.py
Enter the emotion: enjoyment


Joker
Gemini Man
Fractured
The Irishman
Gisaengchung
Wounds
Hustlers
Midsommar
The Dark Knight
Countdown
Scary Stories to Tell in the Dark
In the Tall Grass
Ad Astra
Glass
Black and Blue
War
In the Shadow of the Moon
Halloween Kills
Venom
Knives Out
Crawl
Rambo: Last Blood
Stuber
John Wick: Chapter 3 - Parabellum
No Time to Die
```

7. SUMMARY AND CONCLUSION

7.1 Conclusion:

A Recommender System is an information provider system which connects the users and products. It basically helps the users to get the recommendation based on broad category of choices. The deliverer of the system needs the app to be used by many users so that it can be popular. Recommendation systems have their resources in a various area of research, including information retrieval, information extraction, data manipulation, text classification, etc. These systems use various methodologies such as machine learning, deep learning, data mining and other wide range of techniques including algorithms, collaborative and hybrid methods, and precision methods.

Our project work is a Movie recommender system is built by loading the datasets from IMDB Website, then various filtering algorithms like demographic filtering through weighted rating calculation and the quantile function then, we used Content-based filtering for user preferences through the cosine similarity checks for cast, crew, genre and unique keywords then we moved towards the emotion / mood-based filtering through web scraping methodologies are used which makes recommendations based on the users' preferences, rating, genre, matching the terms, experiences, emotions, popularity and collects information to eventually extract the final movie recommendation systems.

7.2 Future Scope:

1. Recommender systems may help in increasing their sales through E-commerce sites.
2. Facial Mood Detection of the users can be added for the Application through image processing and deep learning.
3. We can increase the precision and make our recommendation more relative and personalized.
4. Multiple factors can be taken as an entity at a single time to predict the similarities. *(Kużelewska, 2014)*
5. Relevant movies/ related information and remainders can be updated through the app and Remind to stream at a particular date and time.
6. We can send emails, messages and update flashes through social media.
7. Restrictions on some contents can be automatically added based on users' information like age, usage timing, and interested category.
8. Maintainability can be increased to hold a lot of movie data as recommended through the app.
9. New gestures can be installed on the Application.
10. More contents can be added which can be helpful for students to explore and learn different fields. *(Davidson and Mortiz, 2019)*
11. Security and privacy of the users should be the main concern while we are updating the new features and contents on the app.
12. The memory capacity should be increased so that the system loads the data and its rating for a longer period without any data loss.

7.3 Applications:

The application of recommender systems is wide ranging with them being used by companies to recommend movies, music, learning materials, television programs, books, documents, houses, business partners, websites, conferences, tourism scenic spots, products, financial services, learning materials,etc.

REFERENCES

1. Shin, K.G. and Mckay, N.D. (1984) 'Open Loop Minimum Time Control of Mechanical Manipulations and its Applications', Proc.Amer.Contr.Conf., San Diego, CA, pp. 1231-1236. (Shin and Mckay, 1984)

2. Hu Jinming, "Application and research of collaborative filtering in e-commerce recommendation system," in2010 3rd International Conference on Computer Science and Information Technology, vol. 4, pp. 686–689,2010. (Hu Jinning, 2010)

3. "Our Basic Emotions Infographic | List of Human Emotions | UWA Online." 2019. University of West Alabama Online. https://online.uwa.edu/infographics/basic-emotions/.

4. M. Y. H. Al-Shamri and K. K. Bharadwaj, "Fuzzy-genetic approach to recommender systems based on a novel hybrid user model," Expert Systems with Applications, vol. 35, no. 3, pp. 1386 – 1399, 2008.

5. © 2022 IJRTI | Volume 7, Issue 7 | ISSN: 2456-3315 IJRTI2207198 International Journal for Research Trends and Innovation (www.ijrti.org) 1264

6. Amos, David. 2022. "A Practical Introduction to Web Scraping in Python – RealPython."RealPython.https://realpython.com/python-web-scraping-practical-introduction/.
7. "Install python PIP, Requests and Beautiful soup for WINDOWS (in 5 minutes)." 2018. YouTube. https://youtu.be/cGW70Hc0W4M.

8. "Beautiful Soup - Installation." n.d. Tutorialspoint. Accessed March 5, 2023. https://www.tutorialspoint.com/beautiful_soup/beautiful_soup_installation.htm

9. L. M. de Campos, J. M. Fernández-Luna, J. F. Huete, and M. A. Rueda-Morales, "Combining content-based and collaborative recommendations: A hybrid approach based on Bayesian networks," International Journal Of Approximate Reasoning, vol. 51, no. 7, pp. 785 – 799, 2010. (de Compos and Fernández-Luna, 2010)

10. N. Hug, "Surprise: A python library for recommender systems," Journal of Open-Source Software, vol. 5, p. 2174, 08 2020. (Hug, 2020)

11. U. Kużelewska, "Clustering algorithms in hybrid recommender system on movielens data," Studies in Logic,Grammar and Rhetoric, vol. 37, 01 2014. (Kużelewska, 2014)

12. C. Davidsson and S. Moritz, "Utilizing implicit feedback and context to recommend mobile applications. 2019 (Davidson and Mortiz, 2019)