EMOTION BASED MOVIE RECOMMENDATION SYSTEM

Submitted in partial fulfillment of the
requirements for the award of
Bachelor of Engineering degree in Computer Science and Engineering

By

MEGHANA ELISETTI (Reg.No - 39110295)
PULIPATI ANSHU (Reg.No – 39110811)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

SCHOOL OF COMPUTING

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)
Accredited with Grade "A" by NAAC | 12B Status by UGC | Approved by AICTE
JEPPIAAR NAGAR, RAJIV GANDHISALAI,
CHENNAI - 600119

APRIL - 2023

SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)
Accredited with Grade "A" by NAAC | 12B Status by UGC | Approved by AICTE

www.sathyabama.ac.in

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of **MEGHANA ELISETTI (Reg.No - 39110295) and PULIPATI ANSHU (Reg.No - 39110811)** who carried out the Project Phase-2 entitled **"EMOTION BASED MOVIE RECOMMENDATION SYSTEM"** under my supervision from January 2023 to April 2023.

Internal Guide

Dr. J. ALBERT MAYAN, M.E., Ph.D

Head of the Department

Dr. L. LAKSHMANAN, M.E.,Ph.D.

SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY
DEPARTMENT OF CSE

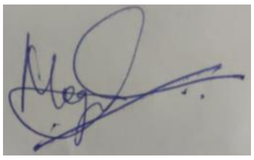
Submitted for Viva-voce Examination held on 19-04-2023

Internal Examiner

External Examiner

DECLARATION

I, **MEGHANA ELISETTI (Reg.No - 39110295),** hereby declare that the Project Phase-2 Report entitled **"EMOTION BASED MOVIE RECOMMENDATION SYSTEM"** done by me under the guidance of **Dr. J. ALBERT MAYAN, M.E., Ph.D.,** is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering degree in **Computer Science and Engineering**.



DATE: 19-04-2023

PLACE: Chennai **SIGNATURE OF THE CANDIDATE**

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to **Board of Management** of **SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T. Sasikala M.E., Ph. D**, **Dean**, School of Computing, **Dr. L. Lakshmanan M.E., Ph.D.,** Head of the Department of Computer Science and Engineering for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide **Dr. J. Albert Mayan M.E., Ph.D,** for his valuable guidance, suggestions and constant encouragement paved way for the successful completion of my phase-1 project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of Computer Science and Engineering** who were helpful in many ways for the completion of the project.

ABSTRACT

The modern world comprises many forms of entertainment, the most common being – movies. Over the past decades, the ways of production, creation and distribution of movies have been greatly advanced. People have a variety of movies to choose from and require a recommendation system to guide them in this process.

Emotion-based movie recommendation system (E-MRS) is one that recommends movies based on the user's emotions. The objective of E-MRS is to provide users with adapted and personalized suggestions by combining collaborative filtering with and content-based approaches. The recommendation technique used is not fully personalized or does not consider the user's current point of interest. Hence the solution to this is a personalized content recommendation based on emotional characteristics.

Emotions are a strong reaction to stimuli and are an intelligent and rational form of behavior. It is difficult to assess the emotional responses to the movies based on the diverse reactions to movies. Color psychology can be used to detect various emotional states such as happiness, sadness, anger, fear, and excitement.

The aim of this project is to develop an user interface where colors are used to represent the user's appropriate emotions and compare the emotion with facial expressions and then recommend a movie based on that. A hybrid approach, combining collaborative filtering and content-based filtering is used to recommend movies.

A convolutional neural network (CNN or convnet) is a subset of machine learning which is a kind of network architecture for deep learning algorithms and is specifically used for image recognition and tasks that involve the processing of pixel data. There are other types of neural networks in deep learning, but for identifying and recognizing objects, CNNs are the network architecture of choice.

INDEX

LIST OF ABBREVIATIONS

ABBREVIATION	EXPANSION
E-MRS	Emotion based Movie Recommendation System
RS	Recommendation System
SVOD	Subscription Video On Demand
AVOD	Advertisement On Demand
TVOD	Transactional Video On Demand
OTT	Over The Top
FER	Facial Emotion Recognition
EEG	Electroencephalogram
BCI	Brain Computer Interface
RMSE	Root Mean Square Error
MAE	Mean Absolute Error
PMC	PubMed Central

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 HISTORY AND INTRODUCTION OF RECOMMENDATION SYSTEM

Many decisions are made in our daily lives, often without our knowledge. Users are often overwhelmed by choice and need help finding what they're looking for. For example, what should we put on in the morning that is suitable for our daily program? Which menu to choose in the dining room? Which task should we perform first? Which school should we enroll our child in? In the past, specialists have attempted to assist users in making judgments, but for a while there have been other options. A website that sells books (as well), like Amazon, can assist individuals in choosing their next book instead of only the librarian or retail employee. The videos that YouTube suggests are all based on prior browsing and provide engaging audiovisual content with a high hit rate. This is where the recommendation system comes into the picture.

Recommendation system is a system that aids in decision making by providing users with suggestions. These suggestions are developed based on past information or domain knowledge. Applications of recommender systems are music on Spotify, recommend products on amazon. The first application of the recommendation system concept appeared in 1979, in the form of Grundy, a computer-based librarian that recommended books to the user. Tapestry, the first commercial RS, was introduced in the early 1990s.

GroupLens, a research lab at the University of Minnesota in the United States, launched another RS implementation in the early 1990s to assist people in finding their preferred articles. In the late 1990s, recommendation systems were further developed with the implementation of Amazon Collaborative Filtering, one of the most well-known RS technologies. Since this time, recommender systems based on Collaborative Filtering have grown in popularity and are used by many e-commerce and online systems.There has been a change in the users' preferences with a change in technology. A most recent example of this is the aftermath of the pandemic. The sudden pandemic created a standstill of a regular lifestyle causing everyone to be indoors. People became more active on the internet and used video streaming platforms regularly for entertainment. The video streaming service in the entertainment industry is now one of the fastest-growing segments in the entertainment industry due to the new OTT (Over the Top) platforms.

An OTT platform is a direct-to-consumer video content platform which provides premium content for their customers to stream on demand. Netflix, Disney+, Hulu, HBOMax and AmazonPrime are such examples of OTT platforms. These platforms are categorized based on their revenue structures such as subscription video on demand (SVOD), advertisement on demand (AVOD) and transactional video on demand (TVOD). A good recommender system is essential for an OTT platform to have an edge in the market.

In the past two decades, e-commerce has surged extensively due to changes in consumer behavior. Several customers are more inclined to shop from the comfort of their home and use online services. The pandemic has also caused more businesses and consumers to turn to digital platforms for sales and purchases of goods online. Recommender systems have the ability to predict whether the user would prefer a certain item or not. This helps the customer by reducing their transaction costs during online shopping, and in turn helps the service provider by creating more traffic for their website / service.

Recommender systems also showed a positive growth in revenue for online services and operations. Recommender systems are further categorized into non-personalized and personalized systems. A non-personalized recommender system is a system which recommends the general most popular items available, for example: The top ten places to visit for a vacation. A personalized recommender system is a system which recommends the most suitable choices for the user based on their past preferences. A good, personalized recommender system is one that is diverse and can recommend different items to the user.

Electronic commerce refers to the transaction of everything and everything pertaining to commercial activity that takes place on digital platforms such as the internet and the world wide web. The proliferation of online commercial activity will one day lead to the streamlining, speeding up, and improvement of the efficiency of transactions in the business environment. Primarily, because customers stand to benefit from an increased variety of goods and a simplified path to information pertinent to their needs as a result of the market's continued expansion, in addition to the services.

Nevertheless, given the ruthless nature of the modern economic environment, it is highly crucial to deliver value to the customer. It is indispensable to the survival of businesses in their current forms. Learning about the customer and what they want is the most effective way to meet their requirements while still giving value. We provide for them in our capacity as humans. It is essential that the customer understand that they possess a unique and personal connection to the organization that sets them apart from other employees. In today's world, the use of recommendation systems is the answer to many of the issues that arise from the need for individualized effort.

1.2 EMOTION BASED RECOMMENDATION SYSTEM

Most of the time, the customer is the one who supplies the recommendation system with data. This may include things like the information on the items that he is looking for, as well as his ratings, numerous demographics, and other data. The recommender system may use a single strategy or several techniques for making recommendations. based on this information, and as a result, provides product suggestions to the customers.

To provide suggestions that can be trusted, the recommender system must be able to grasp exactly what it is that the user is seeking. The requirements of the customer as well as their preferences. On the other hand, when it comes to things that are more subjective and complex, such as movies, music, and novels, aroma, the process of giving a rating to or determining the desirable. There are occasions when consumers have difficulty comprehending the characteristics of a product. In addition, since consumers' preferences for these essentially subjective products change often depending on how they are feeling, the It is not enough to try to comprehend a person based just on the conventional profile that they fit.

In addition to taking these developments into consideration. To uncover answers that would fix these problems, we will It is suggested that you make use of an Emotion-based Recommender System, also known as an E-MRS, which is able to keep track of the preferences of clients. Depending on the emotions that they are experiencing. As a direct consequence of the fact that emotions play a large role in determining both reasonable and intelligent behavior. It is important that the sentiments of other users be included into the idea process.

CHAPTER 2

LITERATURE SURVEY

2.1 REVIEW OF LITERATURE SURVEY

[1] Title: Sharing geotagged pictures for an Emotion-based Recommender System

Author: A. Hitz, S. -A. Naas and S. Sigg

Year: 2021

By considering profiles, previous preferences, and increasingly also other tailored criteria, recommender systems are widely employed in a variety of contexts, the most common of which are the suggestion of movies or apps and online shopping. It is devised and put into operation an emotion-based recommender system for tourists in the city. This system takes into consideration both the user's current mood and their location while making recommendations to the user. A head-to-head comparison of the emotion-based recommender system and the classic measure-based recommender systems was carried out by our research team. Our assessment research included a total of 28 participants, and the results of the trials demonstrated that the emotion-based recommender system resulted in an increase of about 19 percent in the average rating of the suggestion. It is concluded that the use of emotion may greatly enhance the outcomes, particularly the degree to which they are personalized.

[2] Title: Multi-Relational Stacking Ensemble Recommender System Using Cinematic Experience

Author: C. Lee, D. Han, S. Choi, K. Han and M. Yi

Year: 2022

With multiple movie content outlets, customers confront a deluge of material and difficulty picking movies. Much research has gone into establishing successful recommender systems to deliver tailored suggestions based on customers' historical preferences and actions. but not on harnessing users' thoughts and emotions. In this work, the construction of a graph-based movie recommender system is described, where the analysis is done based on the sentiment and emotion data together with user ratings. Fine-tuned BERT extracts feeling and emotion. The Kaggle dataset is being used which is made from Rotten Tomatoes meta-data and reviewed the data.

The research shows that emotion- and sentiment-based models are better than traditional ones. This research supports utilizing sentiment and emotion in movie recommender systems.

[3] Title: Analysis of Intelligent movie recommender system from facial expression

Author: S. Chauhan, R. Mangrola and D. Viji

Year: 2021

Machine learning helps tackle real-time business and research difficulties. Machine learning is a conversion of classical mathematics. Facial recognition uses machine learning models. Real-time facial recognition is utilized in security systems, workplaces, etc. Using facial detection, machine learning models may propose movies. This is done through recording user reaction rather than seeking particular videos. Some relevant research is based on attentional convolutional neural (recognizes facial micro expressions), and a recommender system has been created to suggest movies or songs based on CNN output. Boosting algorithms, another face recognition method employing decision trees, are less effective than CNN. CNN seems more accurate. Combining content-based and collaborative filtering recommendation systems increases their power.

[4] Title: Study on Improvement of Recommendation Algorithm Based on Emotional Polarity Classification

Author: H. Cao and J. Kang

Year: 2020

This study enhances the object-based collaborative filtering recommendation algorithm utilizing user reviews emotional polarity categorization. The algorithm is explained using the film recommendation system as an example. Cnn's emotional polarity categorization technique classifies movie review emotional polarity. Process: (Chinese word segmentation, de-stop words, feature extraction, and CNN classification model). The comment's emotional polarity categorization is added to the user's movie rating, and a suggested movie list is formed.

[5] Title: On the Influence of Shot Scale on Film Mood and Narrative Engagement in Film Viewers

Author: S. Benini, M. Savardi, K. Bálint, A. B. Kovács and A. Signoroni

Year: 2022

Shot scale, the apparent distance of the camera from a scene's subject, has artistic and narrative effects. To quantify how shot scale influences both lower and higher complexity reactions in viewers, we first studied how Close, Medium, and Long Shots connect to viewers' ratings on film mood, measured in terms of hedonic tone, energetic arousal, and tense arousal on 50 film clips. Then, the evaluation of shot scale's influence on violent scene viewers' narrative engagement and its sub-scales: narrative comprehension, attentional focus, emotional engagement, and narrative presence is done. Convolutional Neural Networks trained on the filmographies of six filmmakers, analyze 120 full-length movies at one frame per second to automate shot size categorization. This research examines the link between shot size and viewer emotional engagement using big corpora. Beyond style study, understanding cinema narrative impacts may help with movie recommendations and film therapy.

[6] Title: EEG Based Neuromarketing Recommender System for Video Commercials

Author: S. K. Bandara, U. C. Wijesinghe, B. P. Jayalath, S. K. Bandara, P. S. Haddela and L. M. Wickramasinghe

Year: 2021

Video ads are popular due to technological advancements and corporate competitiveness. Before marketing and promotions begin, it's important to analyze these video advertisements' impact. This research uses EEG data from a Brain-Computer Interface (BCI) to analyze the viability of movie trailers. This helps the firm consider the commercial's impact and worth. The analysis offers movie ads based on user choices. The video's emotion, attentiveness, and pleasure will be assessed. Random Forest prediction algorithm with 91.97% accuracy was used for emotion analysis, while c-Support Vector Classifier method with 91.70% accuracy was utilized for attention analysis. Central Limit theorem and Empirical rule were applied to analyze pleasure. Initial findings validate the suggested framework's promise. This study focuses on the movie and entertainment sector, but it may be applied to other industries.

[7] Title: Evaluation Method for Video Advertisetments Using EEG Signals

Author: A. Dushantha, R. Akalanka, H. Gayan, K. C. Siriwardhana, P. S. Haddela and L. Wickramasinghe

Year : 2020

Video advertisements became very popular in recent past due to the technology advancement and competitiveness of businesses. Therefore, analyzing the impact of commercial video advertisements is important before they launch the marketing campaign. This paper presents a unique method that can evaluate the effectiveness of movie commercials (trailers) using Electroencephalogram (EEG) signals captured from a brain computer interface. Randomly selected fifteen movie lovers participated to capture EEG signals. For a selected set of movie trailers, three different types of classification models were trained and tested. With the help of classification models, it measures attention and enjoyment levels and also emotional status of a viewer to compute effectiveness of an advertisement. It also consists of a recommender system which suggests movie advertisements based on the preferences of the users. From the initial results received, it confirms that the proposed framework is producing promising results. Even though this work focuses on the movie/entertainment industry, it has the potential to be developed and applied for many other industries as well.

[8] Title: Sentiment Analysis using deep learning for use in recommendation systems of various public media applications

Author: K. Arava, R. S. K. Chaitanya, S. Sikindar, S. P. Praveen and S. D

Year: 2022

Sentiment Analysis is a method of analyzing text and extracting opinions from it. It’s also known as emotion or opinion extraction, and it’s part of the machine learning as well as data mining categories. There are numerous ways to convey one’s sentiments. It can be articulated in a variety of ways, such as through feelings, making judgments, or expressing one’s vision or insight. Sentiment investigation is the act of detecting, recognizing, and categorizing a user’s emotion or view for any service, such as movies, product issues, events, or any other attribute that can be good, negative, or neutral. This analysis is based on social communication channels such as websites that include ratings, forum conversations, blogs, micro-blogs, Twitter, and other social media platforms. The important goal of suggested systems is to improvise accuracy and to generate recommendation systems using deep learning algorithms.

[9] Title: A Hybrid Recommender System for Improving Rating Prediction of Movie Recommendation

Author: N. Kannikaklang, S. Wongthanavasu and W. Thamviset

Year: 2022

Because of the COVID-19 pandemic, online movies are now extremely popular. While movie theaters have not been serviced and people are staying quarantined, movies are the best choice for relaxing and treating stress. At present, recommender systems are widely integrated into many platforms of movie applications. A hybrid recommender system is one promising technique to improve system performance, especially for cold-start, data sparsity, and scalability. This paper proposed a hybrid of matrix factorization, biased matrix factorization, and factor-wise matrix factorization to solve all mentioned drawback problems. Simulation shows that the proposed hybrid algorithm can decrease approximately 11.91% and 10.70% for RMSE and MAE, respectively, when compared with the traditional methods. In addition, the proposed algorithm is capable of scalability. While the number of datasets is tremendously increased by 10 times, it are still effectively executed.

2.2 INFERENCES FROM LITERATURE SURVEY

The work of [1] A.Hitz S, A Naas and S.Sigg regarding the paper based on the sharing of geotagged pictures for an Emotion based recommend system shows that an emotion based recommendation system showed a 19% increase in average rating when compared to regular recommendation systems. The benefits of this are that, enhancement is done automatically when an input is given. However, the accuracy is questioned.

Another journal released by Ayush Tanwar and Dinesh Kumar Vishwakarma supports the argument that the traditional recommender systems suffer from low accuracy, linear latent factor and cold-start problems for new users. They employed a deep neural network-based approach which uses user and item vectors to encapsulate users' and items' data to train on High dimensionality non-linear data to provide more accurate recommendations.

2.3 OPEN PROBLEMS IN EXISTING SYSTEM

The existing systems use facial recognition to detect the emotion of the user. A facial recognition system is a technology capable of matching a human face from a digital image or a video frame against a database of faces. Facial recognition can be used in airports, ATM machines, surveillance (monitoring and searching for drug offender and criminals, CCTV control), security (office access, building access control, airports, flight boarding system, email authentication on multimedia workstation).

Facial Emotion Recognition (FER) is a technology that recognizes emotions from photos and static videos. Some applications of this software is for research areas targeting mental disease diagnosis and human interaction detection. This technology is ground-breaking in areas of research; however it is hard to handle in real world applications. This is shown in a journal written by Najmeh Samadiani, published by the PubMed Central (PMC). According to the journal, 97% of the laboratory - controlled FER systems have high accuracy, however while transferring this is to real-world applications, 50% of the results have low accuracy [21].

The reason for the low accuracy is due to the large variety of responses people showcase. Some people are more expressive and react obviously to certain scenarios. while others are unable to do so and react in less obvious manners. According to an IEEE journal written by Chung-Hsien Wu and Jen-Chun Lin, the facial expressions of introverts are significantly different from the distinct expression of an extroverted person [23]. Hence this suggests that it is hard a system to accurately recognize the emotion of the user merely based on their facial recognition, as there is no standard reaction to showcase that the user is feeling a certain way. For example, the user who is an introvert might feel happy but just react with a small smile, while an extrovert would react with a big grin on their face. This does not mean that the introvert was less happy than the extrovert, but it means that there is more than one way to showcase your feeling.

CHAPTER 3
REQUIREMENT ANALYSIS

3.1 SOFTWARE REQUIREMENTS SPECIFICATION DOCUMENT

- **Scope**

Movie Recommender is a movie recommendation system, which provides users with movies which they may like, based on the movies that they previously saw. Every logged-in user should have access to the recommender system. The system will go through the movies that the user previously saw and rated, and then according to that information, it should provide movies to the user. The project's main aim is to provide accurate movie recommendations to the user through emotions. These emotions are detected by using the concept of colours and face recognition. This project is beneficial for the users and the companies. Users may find movies that they may like without consuming time and even they can encounter new movies which they like from the recommendations. For the company, they make the website more attractive, so they draw more users to the website and the system makes the users of the website spend more time online.

- **Operating Environment**

Operating environment for the movie recommendation system is as listed below.
Operating System: Windows
Platform: Chrome/ Edge/ Firefox

- **Design and Implementation Constraints**

Python language for the implementation of a movie recommendation system.
Streamlit is an open-source app framework in Python language. It helps us create web apps for data science and machine learning in a short time. It is compatible with major Python libraries such as scikit-learn, Keras, PyTorch, SymPy(latex), NumPy, pandas, Matplotlib etc.

- **User Interfaces**

The user interface for the software shall be compatible to any browser such as

Internet Explorer, Mozilla or Google Chrome by which user can access to the system. The user interface shall be implemented using any framework or software package like streamlit.

- **Hardware Interfaces**

Since the application must run over the internet, this brings out the requirement of a network interface on the device. User should have a device with valid internet connection, Wi-Fi or 3G/4G/5G.

- **Software Interfaces**

The data of the project is organized in a relational database as it makes it easier to curate data with a large number of attributes. Python 3.9.6 will be used as the predominant programming language for this project. Since the project is based on Machine Learning Python is the clear as it has built in modules for handling and cleansing data eg. Pandas and numpy and for training the recommendation model eg. ScikitLearn.

- **Get_pos_neg () function:**

The calculation of positive and negative emotions is done here. Initially the pos and neg variables are assigned to zero. The zero and one values are mapped to each color by the color selection of the user. Each color represents different emotions.

- **Get_final_sentiment () function:**

Based on the colors that the user selects, the number of ones and zeroes will be calculated with the help of get_pos_neg() function. The emotion detected by the face is also mapped to zeroes and ones which represents emotion. The face emotion is compared with the color emotion using dot operator.

- **Get_poster() function:**

The posters of the movies will be shown based on the genres and the use of api. The genres are also mapped to zeroes and ones. This function returns the path of the poster using api.

- **Suggest_movie() function:**

It can suggest movies as recommendation based on dataset by user's approach. The main function will show movies based on recommendation algorithm. When a user chooses the colors and gives the face input, the system will recommend the movies through their face emotion and color emotion. User can get movie recommendation on the device. Movie information will be collected according to genre and rating.

3.1.1 Hardware requirements

RAM	:	4GB or above 4GB
Processor of Frequency	:	1.5GHz or above
Processor	:	Intel Pentium 4 or higher

3.1.2 Software requirements

Operating System	: Windows 8 and above
Languages	: Machine Learning, Deep Learning
Tools used	: Visual Studio Code, Jupyter Notebook

- **Python**

Python is a scripting language that is high-level, interpreted, interactive, and object-oriented. Python is intended to be extremely readable. It commonly employs English terms rather than punctuation, and it has fewer syntactical structures than other languages.

Python Features

Python's features include −

- **Easy-to-learn** − Python has a small number of keywords, a basic structure, and a well-defined syntax. This enables the pupil to swiftly learn the language.
- **Easy-to-read** − Python code is more clearly defined and visible to the eyes.
- **Easy-to-maintain** − Python's source code is relatively simple to maintain..
- **A broad standard library** − The majority of Python's library is portable and cross-platform compatible on UNIX, Windows, and Macintosh.
- **Interactive Mode** − Python features an interactive mode that enables for interactive testing and debugging of code snippets.

- **Portable −** Python can operate on a broad range of hardware devices and has the same interface across them all.

- **Extendable −** The Python interpreter may be extended using low-level modules. These modules allow programmers to enhance or adapt their tools to make them more efficient.

- **Databases −** Python provides interfaces to all major commercial databases.

- **GUI Programming −** Python can construct and port GUI programs to numerous system calls, libraries, and Windows systems, including Windows MFC, Macintosh, and Unix's X Window system.

- **Scalable −** Python provides a better structure and support for large programs than shell scripting.

Python Libraries

Python libraries include:

- **Scikit-learn**

  It is also known as Sklearn is a free software machine-learning library for the Python programming language. It features various classification, regression and clustering algorithms including support vector machines, random forests, gradient boosting and k-means. It was originally called ***scikits. learn*** and was initially developed by David Cournapeau as a Google summer of code project in 2007. Later, in 2010, Fabian Pedregosa, Gael Varoquaux, Alexandre Gramfort, and Vincent Michel, from FIRCA (French Institute for Research in Computer Science and Automation), took this project to another level and made the first public release (v0.1 beta) on 1st Feb. 2010.v It is a community, where everyone can contribute to it. Various organizations such as Booking.co, JP Morgan, Evernote, Inria, Spotify and much more use Sklearn.

  The prerequisites of using Sklearn, including the installation of Python, NumPy, Scipy, Joblib, Matplotlib and Pandas. There are two ways to install scikit-learn –

  1. Using pip: pip install -U scikit-learn
  2. Using Conda: conda install scikit-learn

Features of Scikit-learn

- **Supervised Learning algorithms −** Almost all the popular supervised learning algorithms, like Linear Regression, Support Vector Machine (SVM), Decision Tree etc., are part of scikit-learn.
- **Unsupervised Learning algorithms −** On the other hand, it also has all the popular unsupervised learning algorithms from clustering, factor analysis, and PCA (Principal Component Analysis) to unsupervised neural networks.
- **Clustering −** This model is used for grouping unlabelled data.
- **Cross Validation −** It is used to check the accuracy of supervised models on unseen data.
- **Dimensionality Reduction −** It is used for reducing the number of attributes in data which can be further used for summarisation, visualisation and feature selection.
- **Ensemble methods −** As name suggests, it is used for combining the predictions of multiple supervised models.
- **Feature extraction −** It is used to extract the features from data to define the attributes in image and text data.
- **Feature selection −** It is used to identify useful attributes to create supervised models.
- **Open Source −** It is an open source library and also commercially usable under a BSD license.

▪ **Keras**

Keras is a deep learning API written in Python, running on top of the machine learning platform TensorFlow. It was developed with a focus on enabling fast experimentation. Being able to go from idea to result as fast as possible is key to doing good research.

- **Simple** -- but not simplistic. Keras reduces developer *cognitive load* to free you to focus on the parts of the problem that really matter.

- **Flexible** -- Keras adopts the principle of *progressive disclosure of complexity*: simple workflows should be quick and easy, while arbitrarily advanced workflows should be *possible* via a clear path that builds upon what you've already learned.
- **Powerful** -- Keras provides industry-strength performance and scalability: it is used by organizations and companies including NASA, YouTube, and Waymo.
- **Highly Flexible** -- Keras provide high flexibility to all of its developers by integrating low-level deep learning languages such as TensorFlow or Theano, which ensures that anything written in the base language can be implemented in Keras.

▪ **PyTorch**

PyTorch is a fully featured framework for building deep learning models, which is a type of machine learning that's commonly used in applications like image recognition and language processing. PyTorch is written in Python, thus making it easier for most machine learning developers to learn and use. PyTorch is distinctive for its excellent support for GPUs and its use of reverse-mode auto-differentiation, which enables computation graphs to be modified on the fly. This makes it a popular choice for fast experimentation and prototyping. PyTorch is known for having three levels of abstraction as given below −

- Tensor − Imperative n-dimensional array which runs on GPU.
- Variable − Node in the computational graph. This stores data and gradient.
- Module − Neural network layer which will store state or learnable weights.

Features

▪ **Easy Interface** − PyTorch offers easy-to-use API; hence it is considered very simple to operate and runs on Python. The code execution in this framework is quite easy.

▪ **Python usage** − This library is Pythonic which smoothly integrates with the Python data science stack. Thus, it can leverage all the services and functionalities offered by the Python environment.

- **Computational graphs** − PyTorch provides an excellent platform which offers dynamic computational graphs. Thus, a user can change them during runtime. This is highly useful when a developer has no idea of how much memory is required for creating a neural network model.

Advantages

- It is easy to debug and understand the code.
- It includes many layers as Torch.
- It includes a lot of loss functions.
- It can be considered as a NumPy extension to GPUs.
- It allows for building networks whose structure is dependent on computation itself.

- **SymPy(latex)**

SymPy is an open-source Python library for symbolic computation. It provides computer algebra capabilities either as a standalone application, as a library to other applications, or live on the web as SymPy Live or SymPy Gamma.

- **NumPy**

NumPy(Numerical Python) is a library for the Python programming language, that adds support for large, multi-dimensional arrays and matrices, along with a large collection of high-level mathematical functions to operate on these arrays. It contains various features including:

- A powerful N-dimensional array object
- Sophisticated functions
- Tools for integrating C/C++ and Fortan code
- Useful linear algebra, Fourier transform and random number capabalitites.

- **Pandas**

pandas is a software library written for the Python programming language for data manipulation and analysis. It offers data structures and operations for manipulating numerical tables and time series. It is free software released under the three-clause BSD license.

Features

- It has a fast and efficient DataFrame object with default and customized indexing.
- Used for reshaping and pivoting data sets.
- Group by data for aggregations and transformations.
- It is used for data alignment and integration of missing data.
- Provide the functionality of Time Series.
- Process a variety of data sets in different formats like matrix data, tabular heterogeneous, and time series.
- Handle multiple operations of the data sets such as sub setting, slicing, filtering, groupBy, re-ordering, and re-shaping.
- It integrates with other libraries such as SciPy, and scikit-learn.
- Provides fast performance, and If you want to speed it, up even more, you can use **Cython**.

- **Matplotlib**

Matplotlib is a plotting library for the Python programming language and its numerical mathematics extension NumPy. It provides an object-oriented API for embedding plots into applications using general-purpose GUI toolkits like Tkinter, wxPython, Qt, or GTK.

Features

- Create publication-quality plots.
- Make interactive figures that can zoom, pan, and update.
- Customize visual style and layout.
- Export to many file formats.
- Embed in JupyterLab and Graphical User Interfaces.

- Use a rich array of third-party packages built on Matplotlib**.**

- **Machine Learning**

Machine learning is defined as "the branch of research that enables computers to learn without being explicitly programmed." Machine learning is the process of programming computers to maximize a performance criterion based on example data or previous experience. The model may be predictive in order to make future forecasts, or descriptive in order to gather information from data.

Definition of Learning: A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P. if its performance at tasks T, as measured by P, improves with experience E.

Examples: Handwriting recognition learning problem

- Task T: Recognizing and classifying handwritten words within images
- Performance P: Percent of words correctly classified
- Training experience E: A dataset of handwritten words with given classifications

Machine Learning comes into the picture when problems cannot be solved using typical approaches. ML algorithms combined with new computing technologies promote scalability and improve efficiency. Modern ML models can forecast everything from disease outbreaks to stock market fluctuations.

- **Visual Studio Code**

Visual Studio Code is a lightweight yet capable source code editor for Windows, macOS, and Linux that runs on your desktop. It has built-in support for JavaScript, TypeScript, and Node.js, as well as a robust ecosystem of extensions for additional languages and runtimes (including C++, C#, Java, Python, PHP, Go, and.NET).

Visual Studio Code has built-in multi-language support that enables programmers to make use of one editor for different languages. VScode can detect if any snippet of code is left incomplete. Resources can also be pulled from online GitHub Repositories that allow cloning of the code and storing it online.

- **Jupyter Notebook**

The original web application for creating and sharing computational documents is Jupyter Notebook. It provides a straightforward, streamlined, document-centric experience. Jupyter Notebook allows users to collect all components of a data project in one location, making it easier to demonstrate the complete project process to your target audience. Users may utilize the web-based application to generate data visualizations and other project components to share with others via the platforms.

- **Deep Learning**

Deep learning is a subset of machine learning, which is essentially a neural network with three or more layers. These neural networks attempt to simulate the behaviour of the human brain, allowing it to learn from large amounts of data. While a neural network with a single layer can still make approximate predictions, additional hidden layers can help to optimize and refine for accuracy.

Definition of Neural Network: Neural networks are comprised of a node layer that contains an input layer, one or more hidden layers, and an output layer. Each node connects to another and has an associated weight and threshold. If the output of any individual node is above the specified threshold value, that node is activated, sending data to the next layer of the network. Otherwise, no data is passed along to the next layer of the network.

Working:

- The first neuron layer i.e. input layer receives the input data and passes it to the first hidden layer.
- The hidden layers perform the computations on the received data. The biggest challenge in neural network creation is to decide the number of neurons and many hidden layers.
- Finally, the output layer produces the required output.

Deep learning can generate new features from the limited available training data sets. One of the major benefits of deep learning is the reduced time required for feature engineering as compared to machine learning. Some fields where deep learning is applied are driverless cars, virtual assistants, and facial recognition.

Virtual assistants such as Siri, Alexa and Cortana use deep learning to translate human speech and language into necessary instructions. Deep learning aids driverless cars like Tesla's to understand different scenarios of the road, speed limits, signals and pedestrian behaviours.

CHAPTER 4

DESCRIPTION OF PROPOSED SYSTEM

4.1 PROPOSED SYSTEM

The proposed system is to create a user interface to recommend movies based on user emotions using color psychology and facial expressions of the user. There are five basic emotions that a human being has: love, joy, anger, sadness, and fear. Some researchers agreed that the colors have a strong impact on our emotions and feelings. Color is a natural form to represent human emotions. The colors represent the positive and negative emotions of a user as shown in the below figure 4.1. Dark colors represent negative emotions and light colors refer to positive emotions.

POSITIVE EMOTIONS	NEGATIVE EMOTIONS
Yellow: Joy, Merry, Good mood	Black:Sadness, expression and loneliness
Green: Happiness, calmness, and feelings of relieving	Grey: Sadness, depression, and loneliness
Blue: Pleasure and Happiness	Brown: Sadness and depression
Red: love, passion, and excitation	Red: Anger
Orange: cheerfulness	Orange: fear and distress

Table 4.1 : Positive and Negative Emotions

The summary of the relation between colors and emotions that will be used in our system is shown below in Table 4.1.

Emotions	Colours
Joy	Yellow Light Orange Blue Green
Love	Light Red
Anger	Dark Red
Sadness	Brown Grey Black

Table 4.2: Emotions based on colors

4.2 SELECTED METHODOLOGY OR PROCESS MODEL

There are two techniques in recommendation system. They are collaborative filtering technique and content-based filtering technique.

Collaborative filtering is best suited to problems with known data on users (age, gender, occupation, etc.) but a lack of data for items of interest or difficult feature extraction. In contrast to the content-based approach, collaborative recommender systems attempt to predict a user's utility for an item based on the utility of other users with the item in the past.

Content-based filtering methods are based on item featurization (as opposed to user featurization) and a profile of a user's utility. It is best suited to problems that have known data on items (e.g., leading actors, year of release, genre for movies) and how the user has historically interacted with the recommender system but lack personal information about the user. Content-based recommenders are essentially a user-specific learning problem that uses item features to quantify the user's utility (likes and dislikes, rating, etc.).

4.3 ARCHITECTURE /OVERALL DESIGN OF PROPOSED SYSTEM

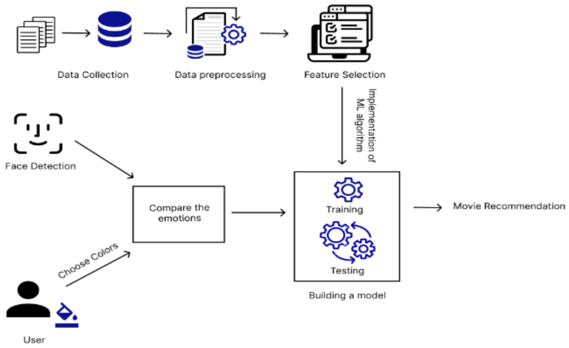


Fig 4.1: Development of the New Methodology

The main objective is to develop a website where a user can get the recommendations of movie by their emotions using color psychology and facial gestures. Color psychology is an effective way of detecting. a user's emotions through color. The TMDB movie dataset is used to train and test a model to detect the appropriate emotions.

Any missing values, errors, or outliers must be corrected. To accomplish this, a preprocessing technique must be used to improve accuracy. The following step is to visualize the data. The hybrid technique is then used to extract features in the following step. The two techniques used in this project to accomplish this are collaborative filtering and content-based filtering.

The collaborative filtering technique deals with the similarities of features between users and recommends personalized movies, whereas content-based filtering is all about filtering a movie's like, dislike, or ratings. The user profiles vector using matrix system and emotions of a user will be extracted with the help of these two techniques.

As a result, a model is developed to recommend movies based on emotions using face detection along with color selection. The user will choose any three distinct colors here. Every color represents a different emotion (from color psychology). The model will recommend movies based on the emotions expressed by the user using colors and facial emotion.

4.4 DESCRIPTION OF SOFTWARE FOR IMPLEMENTATION AND TESTING PLAN OF THE PROPOSED MODEL/SYSTEM

Anaconda is an open-source package manager for Python and R. It is the most popular platform among data science professionals for running Python and R implementations. There are over 300 libraries in data science, so having a robust distribution system for them is a must for any professional in this field. Anaconda simplifies package deployment and management. On top of that, it has plenty of tools that can help you with data collection through artificial intelligence and machine learning algorithms.

With Anaconda, you can easily set up, manage, and share Conda environments. Moreover, you can deploy any required project with a few clicks when you're using Anaconda. There are many advantages to using Anaconda and the following are the most prominent ones among them: Anaconda is free and open source. This means you can use it without spending any money. In the data science sector, Anaconda is an industry staple.

It is open source too, which has made it widely popular. If you want to become a data science professional, you must know how to use Anaconda for Python because every recruiter expects you to have this skill. It is a must-have for data science. It has more than 1500 Python and R data science packages, so you don't face any compatibility issues while collaborating with others.

For example, suppose your colleague sends you a project which requires packages called A and B but you only have package A. Without having package B, you wouldn't be able to run the project. Anaconda mitigates the chances of such errors. You can easily collaborate on projects without worrying about any compatibility issues. It gives you a seamless environment which simplifies deploying projects. You can deploy any project with just a few clicks and commands while managing the rest.

Anaconda has a thriving community of data scientists and machine learning professionals who use it regularly. If you encounter an issue, chances are, the community has already answered the same. On the other hand, you can also ask people in the community about the issues you face there, it's a very helpful community ready to help new learners. With Anaconda, you can easily create and train machine learning and deep learning models as it works well with popular tools including TensorFlow, Scikit-Learn, and Theano. You can create visualizations by using Bokeh, Holoviews, Matplotlib, and Datashader while using Anaconda.

How to Use Anaconda for Python

Now that we have discussed all the basics in our Python Anaconda tutorial, let's discuss some fundamental commands you can use to start using this package manager.

Listing All Environments

To begin using Anaconda, you'd need to see how many Conda environments are present in your machine.

conda env list

It will list all the available Conda environments in your machine.

Creating a New Environment

You can create a new Conda environment by going to the required directory and use this command:

conda create -n <your_environment_name>

You can replace <your_environment_name> with the name of your environment. After entering this command, conda will ask you if you want to proceed to which you should reply with y:

proceed ([y])/n)?

On the other hand, if you want to create an environment with a particular version of Python, you should use the following command:

conda create -n <your_environment_name> python=3.6

Similarly, if you want to create an environment with a particular package, you can use the following command:

conda create -n <your_environment_name>pack_name

Here, you can replace pack_name with the name of the package you want to use.

If you have a .yml file, you can use the following command to create a new Conda environment based on that file:

conda env create -n <your_environment_name> -f <file_name>.yml

We have also discussed how you can export an existing Conda environment to a .yml file later in this article.

Activating an Environment

You can activate a Conda environment by using the following command:

conda activate <environment_name>

You should activate the environment before you start working on the same. Also, replace the term <environment_name> with the environment name you want to activate. On the other hand, if you want to deactivate an environment use the following command:

conda deactivate

Installing Packages in an Environment

Now that you have an activated environment, you can install packages into it by using the following command:

conda install <pack_name>

Replace the term <pack_name> with the name of the package you want to install in your Conda environment while using this command.

Updating Packages in an Environment

If you want to update the packages present in a particular Conda environment, you should use the following command:

conda update

The above command will update all the packages present in the environment. However, if you want to update a package to a certain version, you will need to use the following command:

conda install <package_name>=<version>

Exporting an Environment Configuration

Suppose you want to share your project with someone else (colleague, friend, etc.). While you can share the directory on Github, it would have many Python packages, making the transfer process very challenging. Instead of that, you can create an environment configuration .yml file and share it with that person. Now, they can create an environment like yours by using the .yml file.

For exporting the environment to the .yml file, you'll first have to activate the same and run the following command:

conda env export ><file_name>.yml

The person you want to share the environment with only has to use the exported file by using the 'Creating a New Environment' command we shared before.

Removing a Package from an Environment

If you want to uninstall a package from a specific Conda environment, use the following command:

conda remove -n <env_name><package_name>

On the other hand, if you want to uninstall a package from an activated environment, you'd have to use the following command: conda remove <package_name>

Deleting an Environment

Sometimes, you don't need to add a new environment but remove one. In such cases, you must know how to delete a Conda environment, which you can do so by using the following command:

conda env remove –name <env_name>

The above command would delete the Conda environment right away.

4.5 PROJECT MANAGEMENT PLAN

Project Scope: Define the project's goals, objectives, and requirements. Identify the key stakeholders, team members, and resources needed to complete the project.

Project Planning: Develop a detailed project plan that outlines the tasks, timelines, and deliverables for each phase of the project. Create a work breakdown structure

(WBS) to organize the project tasks into manageable parts.

Resource Allocation: Identify the resources needed for the project, including personnel, hardware, and software. Allocate resources based on project requirements, timelines, and budget.

Risk Management: Identify potential risks to the project and develop a risk management plan to mitigate or avoid them. Create contingency plans to handle unexpected events that may affect the project timeline or budget.

Project Execution: Implement the project plan by assigning tasks to team members, monitoring progress, and addressing any issues that arise. Regularly communicate project status to stakeholders and make adjustments as needed.

Project Monitoring and Control: Monitor project progress against the project plan and make adjustments as necessary. Track project costs, timelines, and quality to ensure the project is completed within budget, on time, and to the required quality standards.

Project Closure: Evaluate the project's success against the project goals and objectives. Close out the project by delivering the final product, documenting project outcomes, and conducting lessons learned review to identify areas for improvement in future projects.

4.6 PROJECT TRANSITION / SOFTWARE TO OPERATION PLAN

Testing and Refinement: Before transitioning to operation, the recommendation system needs to be thoroughly tested to ensure its accuracy and reliability. The system should be tested with a diverse set of users, and feedback should be collected to refine the system and improve its performance.

Deployment and Integration: Once the recommendation system has been tested and refined, it can be deployed to the production environment. The system should be integrated with the existing infrastructure and made accessible to users through an easy-to-use interface.

User Training: Users need to be trained on how to use the recommendation system to ensure they get the most out of it. Training materials, such as user guides, tutorials, and videos, should be provided to help users navigate and utilize the system effectively.

Monitoring and Maintenance: Once the system is in operation, it needs to be monitored and maintained to ensure its ongoing performance and reliability. Monitoring tools should be set up to detect and alert any potential issues, and maintenance schedules should be established to ensure the system stays up-to-date and operates at peak performance.

Continuous Improvement: The recommendation system should be continuously improved based on user feedback and new data. Regular updates should be made to the system to ensure it remains relevant and useful to users.

CHAPTER - 5

IMPLEMENTATION DETAILS

5.1 DEVELOPMENT AND DEPLOYMENT SETUP

The following steps should be followed to set up the development and deployment environment for this project:

Install Python:

Download and install python from the official Python website. Make sure to select the latest version of the Python. Click the appropriate link for your system to download the executable file.

Install required libraries:

Open the command prompt or terminal and navigate to the project directory. Install the required libraries by running the following command: “pip install -r requirements.txt”. This command will install all the required libraries mentioned in the requirements.txt file.

Develop the Streamlit Application:

Build a streamlit application to create a framework for the machine learning and deep learning algorithms. The code is saved in the python file named webapp.py in the project directory.

Run the Streamlit Application locally:

To run the application locally, execute the following command in the command prompt/ terminal : “streamlit run webapp.py” or “python -m streamlit run webapp.py”. This command will start the streamlit application, and it will be accessible at http://localhost:8501.

Deployment Setup:

Web Server: The system can be deployed on a web server, such as Apache or Nginx, to provide a web interface for users to access the recommendation system.

Application Server: The web server can be connected to an application server, such as Tomcat or JBoss, to run the recommendation system application.

Database Server: The recommendation system can be connected to a database server, such as MySQL or PostgreSQL, to store and retrieve movie data, user profiles, and recommendation history.

Load Balancer: To ensure scalability and high availability, a load balancer, such as HAProxy or Nginx, can be used to distribute the traffic across multiple web servers.

Cloud Platform: The system can be deployed on a cloud platform, such as Amazon Web Services (AWS) or Google Cloud Platform (GCP), to take advantage of their scalability and flexibility features. Cloud platforms provide infrastructure-as-a-service (IaaS) and platform-as-a-service (PaaS) options to deploy, manage and scale the system.

Security: Security measures should be taken into consideration to ensure the system is protected against attacks. For instance, deploying an SSL/TLS certificate to encrypt communication between the server and clients, restricting access to sensitive data, and implementing firewalls to prevent unauthorized access.

Monitoring and Logging: Monitoring and logging tools, such as Prometheus and ELK stack, can be used to monitor system performance, detect and alert any potential issues, and log system events and user activities.

Steps for Deployment:

Deploying an emotion-based movie recommendation system (E-MRS) involves several steps, including setting up the environment, building the system, and deploying it to a server. Here is a possible deployment setup for the project:

1. Environment setup:
    - Choose a cloud services provider such as Amazon Web Services (AWS), Microsoft Azure, or Google Cloud Platform (GCP).
    - Create a virtual machine (VM) or container to host the application.
    - Install necessary software dependencies such as Python, Flask, and other packages required for the project.
    - Configure the network settings and firewall rules to allow traffic to and from the application.
2. Building the system:

- Train the convolutional neural network (CNN) model on a dataset of movie posters and emotions using a deep learning framework such as TensorFlow or PyTorch.
- Build the user interface using a web development framework such as React, Angular, or Vue.js.
- Integrate the CNN model and emotion detection algorithm into the user interface, so that users can upload a picture of themselves, and the system can detect their emotion and recommend a movie.

3. Deployment:
    - Package the application and its dependencies into a Docker container.
    - Push the container to a container registry such as Docker Hub, AWS ECR, or GCP Container Registry.
    - Deploy the container to the virtual machine or container service.
    - Set up a load balancer and configure it to distribute traffic across multiple instances of the application for high availability and scalability.
    - Test the application to ensure it is functioning correctly.
4. Maintenance:
    - Monitor the application and infrastructure for performance and security issues.
    - Regularly update the application and its dependencies to patch security vulnerabilities and add new features.
    - Scale up or down the application as needed to accommodate changes in traffic volume or user demand.

Deploying an E-MRS involves many steps, and there may be variations depending on the specific requirements of the project. However, this deployment setup should provide a solid foundation for building and deploying an emotion-based movie recommendation system.

5.2 ALGORITHMS

- **Data Acquisition and Pre-processing**

The TMDB film database, known for its extensive marking, was utilized to narrow the pool of possible outcomes. As the primary identifier for clustering, the use of genre was chosen. This brief examination has resulted in the identification of 24 distinct genres. These films span the gamut from action to history, and many fall into multiple categories. There are many of products that may not even come close to summarizing a complete film story. To make meaningful discoveries, the need of more information is necessary. Obtaining as much information as possible is mainly limited by the time period. As the data model becomes more refined, less of it is considered. There may be one significant advantage depending on how recommendations are made. Structures for sharing information with social media, movie forums, machine learning forums, and other online channels are required. Our data collection for the recommender system took roughly eight weeks. Users of the sentiment-based recommender contributed thousands of individual opinions.

The datasets used in this project were obtained from the Kaggle website. The datasets are “tmdb_5000_movies.csv” and “tmdb_5000_credits.csv” as shown below. In the “tmdb_5000_movies.csv” dataset, there are 4803 rows and 20 columns. It comprises budget, genre, homepage, original language, id, original title, companies, overview, popularity, keywords, release date, production countries, revenue, spoken languages, runtime, tagline, title, vote average, status, and vote count as fields. There are 4 columns and 4803 rows in the "tmdb 5000 credits.csv" dataset, with fields like crew, cast, title, and movie id. The two datasets contain 24 genres. The datasets are joined based on the column shared by both CSV files. To improve results, null or missing values will be checked while training the model. If there are any null values or missing values, drop those values. Among all the columns in the merged dataset, the selected features are genres, keywords, cast, crew, title, overview, and movie id.

movie_id	title	cast	crew
19995	Avatar	[{"cast_id": 242, "character": "Jake Sully", "...	[{"credit_id": "52fe48009251416c750aca23", "de...
285	Pirates of the Caribbean: At World's End	[{"cast_id": 4, "character": "Captain Jack Spa...	[{"credit_id": "52fe4232c3a36847f800b579", "de...
206647	Spectre	[{"cast_id": 1, "character": "James Bond", "cr...	[{"credit_id": "54805967c3a36829b5002c41", "de...
49026	The Dark Knight Rises	[{"cast_id": 2, "character": "Bruce Wayne / Ba...	[{"credit_id": "52fe4781c3a36847f81398c3", "de...
49529	John Carter	[{"cast_id": 5, "character": "John Carter", "c...	[{"credit_id": "52fe479ac3a36847f813eaa3", "de...

Table 5.1: Credits TMDB dataset

budget	genres	homepag	id	keywords	original_l	original_t	overview	popularity	productio	productio	release_d	revenue	runtime	spoken_l	status	tagline	title	vote_ave	vote_count
2.37E+08	[{"id": 28,	http://ww	19995	[{"id": 14	en	Avatar	In the 22n	150.4376	[{"name":	[{"iso_31	#######	2.79E+09	162	[{"iso_63	Released	Enter the	Avatar	7.2	11800
3E+08	[{"id": 12,	http://dis	285	[{"id": 27	en	Pirates of	Captain Ba	139.0826	[{"name":	[{"iso_31	#######	9.61E+08	169	[{"iso_63	Released	At the en	Pirates of	6.9	4500
2.45E+08	[{"id": 28,	http://ww	206647	[{"id": 47	en	Spectre	A cryptic r	107.3768	[{"name":	[{"iso_31	#######	8.81E+08	148	[{"iso_63	Released	A Plan No	Spectre	6.3	4466
2.5E+08	[{"id": 28,	http://ww	49026	[{"id": 84	en	The Dark I	Following	112.313	[{"name":	[{"iso_31	#######	1.08E+09	165	[{"iso_63	Released	The Leger	The Dark I	7.6	9106
2.6E+08	[{"id": 28,	http://mc	49529	[{"id": 81	en	John Carte	John Carte	43.927	[{"name":	[{"iso_31	3/7/2012	2.84E+08	132	[{"iso_63	Released	Lost in ou	John Carte	6.1	2124
2.58E+08	[{"id": 14,	http://ww	559	[{"id": 851	en	Spider-Ma	The seem	115.6998	[{"name":	[{"iso_31	5/1/2007	8.91E+08	139	[{"iso_63	Released	The battle	Spider-Ma	5.9	3576
2.6E+08	[{"id": 16,	http://dis	38757	[{"id": 15	en	Tangled	When the	48.68197	[{"name":	[{"iso_31	#######	5.92E+08	100	[{"iso_63	Released	They're ta	Tangled	7.4	3330
2.8E+08	[{"id": 28,	http://ma	99861	[{"id": 88	en	Avengers:	When Tor	134.2792	[{"name":	[{"iso_31	#######	1.41E+09	141	[{"iso_63	Released	A New Ag	Avengers:	7.3	6767
2.5E+08	[{"id": 12,	http://har	767	[{"id": 61	en	Harry Pott	As Harry b	98.88564	[{"name":	[{"iso_31	7/7/2009	9.34E+08	153	[{"iso_63	Released	Dark Secr	Harry Pott	7.4	5293
2.5E+08	[{"id": 28,	http://ww	209112	[{"id": 84	en	Batman v	Fearing th	155.7905	[{"name":	[{"iso_31	#######	8.73E+08	151	[{"iso_63	Released	Justice or	Batman v	5.7	7004
2.7E+08	[{"id": 12,	http://ww	1452	[{"id": 83,	en	Supermar	Supermar	57.92562	[{"name":	[{"iso_31	#######	3.91E+08	154	[{"iso_63	Released		Supermar	5.4	1400
2E+08	[{"id": 12,	http://ww	10764	[{"id": 62	en	Quantum	Quantum	107.9288	[{"name":	[{"iso_31	#######	5.86E+08	106	[{"iso_63	Released	For love, f	Quantum	6.1	2965

Table 5.2: Movies TMDB dataset

- **Building Emotion detector**

This algorithm's job is to deduce the user's emotional state from a palette of three colors. So that it can achieve that, it will use the following reasoning to examine the color sequence: If the user selects three colors and at least two of them have the same emotional connotation, then that feeling is the one they are experiencing at the moment. If the user selects the hues yellow, blue, and green or yellow, blue, and black, for instance, they are experiencing elation. An alternative scenario exists where the first hue chosen represents happiness, the second hue signifies love, and the third represents either sadness or wrath (negative emotion).

The user's current emotional state is "joy-love" in this case. If, on the other hand, two of the colors signify melancholy and one suggests fury, then sadness and anger represent the user's current emotional state. If a person selects the positive feelings of "joy-love". For instance, the result will be a combination of the following colors: yellow, light red, and black. Movie evaluations and associated feelings are recorded in user accounts. All of this data may be found in the User Profiles repository. The user will be asked to fill out a survey during registration to share his thoughts on which kind of movies best represent certain moods. A user's choice of movie is treated as an implicit 5-star rating. Love, rage, joy, and sadness are the four categories of user emotions that make up the profile vector.

The user's preferred genres and titles of films to view when experiencing an emotion E are subsequently catalogued under each emotion group E. If user u selects films A and B for the emotion "love," films C and D for "anger," and film D for "joy," then user u's profile vector would look like this: u = {{A: 5, B: 5}, {C: 5}, {D: 5}, {}}. To provide input on the system's film suggestion, the user can rate the degree to which the film is in line with his present may suggest film F; nevertheless, the user may not agree that he would like to see this film while he is in love, and so he may give it a rating of 1 out of 5. This transforms the letter u into a profile vector: u = {{A: 5, B: 5, F: 1}, {C: 5}, {D: 5}, {}}.

The colors used in this project are black, white, red, yellow, and blue. The user is required to select three colors among the colors mentioned. Each color is assigned 1 and 0 values. Negative and positive emotions are represented by numbers 1 and 0. The colors are mapped by 1 and 0 if the user selects black, white, and blue. The mapped values are 0, 1, and 1. The resulting emotion is positive if the mapped value of positive emotions exceeds the assigned value of negative emotions. Otherwise, the outcome would be negative.

- **Training the Model**

The recommendation algorithm is in fact a cascade hybrid of two techniques: a collaborative filtering and a content-based recommendation. Collaborative filtering: The detailed description of how our collaborative filtering approach can be applied in the recommendation process.

For a given database U of user profiles, and a target user u, a movie m, an emotion e, the executive steps of the CF algorithm can be outlined as following:

- Step 1: Extract the profile vector of the target user u from the database. We consider the information of user preferences available from the User Profiles database U. For example, for the target user u, his profile vector is u = {{A : 5, B : 5, F : 1}, {C : 5}, {D : 5}, {}}
- Step 2: Search for other users who have rated at least one movie in common with the target user u.

In order to reduce the computing user-film matrix, the system will consider only the users who have rated at least one movie-emotion in common with the target user u. For example: Let Un is a sub collection of U, containing only users who have at least one movie-emotion in common with u. Un =< u1, u2, u3 > There are 3 users in Un with: u1 = {{A : 5, F : 2}, {C : 5}, {E : 5}, {I : 5}} u2 = {{B : 5}, {}, {G : 5}, {H : 4}}

However, users do not provide consistent ratings when asked to rate the same movie at different times. As a result, it is impossible to create an algorithm more accurate than the variance in a user's ratings for the same item. Even when accuracy differences between recommender systems are measurable, they are usually too small to be noticed by the users.

Another problem is that some recommender systems produce highly accurate but totally useless recommendations to users. For example, Lord of the Rings is a very famous movie and nearly all of the movie lovers have already seen it. So, it is useless to recommend it again to the users.

Therefore, to make the evaluation results to be more reliable, we propose to use the precision and mean absolute error with consideration to the novelty factor.

- Mean absolute error (MAE) measures the average absolute deviation between a predicted rating and the user's true rating. Thus, it can measure how close the recommender system's predicted ratings are to the true user ratings.
- Precision is defined as the ratio between the number of well-recommended items and the total number of rated recommendations. Well recommended movies are movies which the users have never seen before and are rated at least 4 out of 5.

The main goal of our tests is to measure how close our system's recommended products are to the true user's needs and preferences. Another goal is to determine the precision of the Emotion detector. After providing a list of movie recommendations to the users, the system will ask them to evaluate each movie and to specify whether they have already seen that movie. The users are also invited to answer a survey at the end of the recommendation process:

- Questions about the overall evaluation of the user about the system,
- Questions about the user interface and recommendation explanations, and
- Questions about the quality and the precision of the Emotion detector.

All responses from this survey will be stored and calculated to determine the user satisfaction after using the system.

- **Comparison of face emotion and color emotion**

Six user emotions are considered in this project, like anger, sadness, happiness, disgust, surprise, and fear, and these emotions will be recognized by the user's face. Angry, fear, sadness, and disgust come under negative emotions. Surprised and happy come under positive emotions. If the face-detected emotion is negative and the three-color choice produces a positive emotion, then the movies which result in positive emotions to the user will be recommended. The Dot operator is used to compare the face emotion and color emotion.

CHAPTER - 6

RESULTS AND DISCUSSION

In this paper, A recommendation system is designed to help you find the best movies to watch based on the genre you like. The simple reason that viewers either love or hated the movies in our system is, it only considers ratings of 1 to 5. This approach provides far superior recommendations to consumers since it lets them comprehend the connection between their feelings and the suggested content. When a user gives a film's genre a high rating, similar film genres aresuggested.

What makes this system unique is that it considers the user's feelings and compares those emotions with that of emotion detected by the face. By using this information to make informed product recommendations. From these findings, we can infer that using more dense data will improve the recommender's performance and that include additional genres will allow us to provide more targeted suggestions. In addition to this, we have carried out studies to demonstrate the accuracy of our predictions when employing emotional feature data.

Movies that have received ratings of 3 or above are presumed to be loved by the user, whilst movies that have received ratings of 1 or 2 are supposed to be despised by the user. We do not consider unclear 3 rating. The result matrix is produced by making use of the ratings matrix and the genres matrix. The result matrix is the dot product of the two matrices that came before it. The outcome then undergoes an additional conversion into a binary format. If the result of the dot product is greater than 0, then the value 1 is given to the cell inquestion; otherwise, the value 0 is used.

Emotions are the undeniable reliance of information in bridging human and machine intercommunication. Machines can recommend better when they can comprehend an individual’s emotions. Producing emotions in the users is conventionally recognized as the fundamental goal of movies. Hence, movie recommendations based on one’s emotional trajectory is key as it allows them to map movie recommendations based on their emotional stage.

CHAPTER – 7

CONCLUSION

7.1 CONCLUSION

In conclusion, the emotion-based movie recommendation system represents an exciting application of machine learning and natural language processing technologies. The system generates movie recommendations based on a user's current emotional state, allowing users to discover movies that match their mood and elicit positive emotional responses. However, the system also has several research issues that must be explored to ensure its accuracy, fairness, transparency, and user satisfaction.

The research issues related to the emotion-based movie recommendation system include explain ability and transparency, ethical and social implications, integration of multiple modalities for emotion recognition, personalization of recommendations, and evaluation of recommendation quality. Exploring these issues can help improve the system's performance, mitigate potential ethical and social concerns, and enhance user satisfaction.

To address these research issues, researchers can develop new methods and techniques, such as user-based evaluations, surveys, and interviews, to evaluate the system's performance and identify areas for improvement. Researchers can also explore the integration of multiple modalities for emotion recognition and develop personalized recommendation algorithms that consider a user's historical viewing preferences and emotional responses to movies.

In summary, the emotion-based movie recommendation system has the potential to revolutionize the way we discover and watch movies. By addressing the research issues related to the system, researchers can ensure that it is accurate, reliable, and aligned with users' values and expectations. Ultimately, this can lead to a more satisfying and enjoyable movie-watching experience for users.

7.2 FUTURE WORK

The emotion-based movie recommendation system project has several opportunities for future work, including personalization, multi-modal emotion recognition, integration with social media and streaming platforms, and user-based evaluation of recommendation quality. These avenues for improvement and expansion would enable the system to provide a more comprehensive and personalized movie recommendation experience for users, leveraging social signals and multi-modal emotion recognition, and integrating seamlessly with streaming platforms. Furthermore, user-based evaluations would provide a more comprehensive and user-centric evaluation of the system's performance.

7.3 RESEARCH ISSUES

The emotion-based movie recommendation system is an exciting application of machine learning and natural language processing technologies. The system generates movie recommendations based on a user's current emotional state, allowing users to discover movies that match their mood and elicit positive emotional responses. However, like any machine learning application, the emotion-based movie recommendation system has several research issues that can be explored to improve its accuracy, fairness, transparency, and user satisfaction.

One of the primary research issues related to the emotion-based movie recommendation system is explain ability and transparency. As the system relies on machine learning algorithms to generate recommendations, it may be challenging to explain how the system arrived at a particular recommendation. This issue is compounded when the system uses a combination of different modalities for emotion recognition, such as text, speech, facial expressions, and physiological signals. Research could explore methods to increase the transparency and interpretability of the system's recommendations, such as generating user-friendly explanations of the recommendation process. This would not only improve users' understanding of how the system works but also build trust and confidence in the system.

Another research issue is the ethical and social implications of the system's recommendations. The recommendation system has the potential to influence user behavior and shape their movie preferences. As such, research could explore the ethical and social implications of the system's recommendations. For example, what are the potential biases in the recommendation system, and how can they be mitigated? How can the system's recommendations avoid reinforcing stereotypes or perpetuating discrimination? Exploring these issues can help ensure that the system's recommendations are fair, unbiased, and aligned with the values and expectations of its users.

The integration of multiple modalities for emotion recognition is another research issue that can be explored. While these modalities may provide a more comprehensive view of the user's emotional state, integrating them effectively and accurately can be challenging. Research could explore methods to integrate multiple modalities for emotion recognition and determine the optimal combination of modalities for generating accurate recommendations. This would require developing sophisticated machine learning algorithms that can effectively process and analyze different types of data.

Personalization of recommendations is another research issue that can be explored to enhance the system's performance. The current recommendation system generates recommendations based on a user's current emotional state. However, in the future, the system could be personalized to recommend movies based on a user's historical viewing preferences and emotional responses to those movies. Research could explore the impact of personalization on user satisfaction and determine the most effective methods for generating personalized recommendations. This would require developing machine learning algorithms that can analyze user behavior and generate personalized recommendations based on that behavior.

Finally, evaluating recommendation quality is a research issue that can be explored to ensure that the system's recommendations are accurate and effective. The current recommendation system's performance can be evaluated using metrics such as accuracy, precision, recall, and F1 score. However, user-based evaluations, surveys, and interviews can provide a more comprehensive and user-centric evaluation of the system's performance. These methods can help researchers understand the factors that influence user satisfaction and identify areas for improvement.

In conclusion, the emotion-based movie recommendation system has several potential research issues that can be explored to improve its accuracy, fairness, transparency, and user satisfaction. These research issues represent exciting opportunities for innovation and advancement in machine learning and natural language processing technologies. By exploring these issues, researchers can develop new methods and techniques to enhance the recommendation system's performance, improve user satisfaction, and mitigate potential ethical and social concerns.

7.4 IMPLEMENTATION ISSUES

There are several implementation issues that need to be considered while developing an emotion-based movie recommendation system:

Data collection and pre-processing: The first and foremost task is to collect and pre-process data related to movies, including their genres, plot summary, cast, ratings, and reviews. This data needs to be processed and cleaned to extract relevant features that can be used to build an emotion-based recommendation system.

Emotion classification: One of the key challenges is to classify emotions from user reviews or ratings. Various techniques such as sentiment analysis, emotion detection using machine learning algorithms, and deep learning models can be used to classify emotions.

Recommendation engine: Once the emotions are classified, the next step is to build a recommendation engine that can suggest movies based on a user's emotional state. The recommendation engine can use techniques such as collaborative filtering, content-based filtering, or a hybrid of both.

Integration with user interface: The recommendation system needs to be integrated with a user interface that can accept input from the user regarding their emotional state and display movie recommendations accordingly.

Scalability: As the system grows, the scalability of the recommendation engine needs to be considered. The system should be able to handle a large number of users and provide real-time recommendations.

Data privacy and security: The recommendation system may collect sensitive data such as user preferences, which needs to be handled with care. Proper data privacy and security measures need to be implemented to protect user data.

Evaluation: The performance of the recommendation system needs to be evaluated regularly to ensure its effectiveness. Various metrics such as precision, recall, and accuracy can be used to measure the system's performance.

REFERENCES

[1] A. Hitz, S. -A. Naas and S. Sigg, "Sharing geotagged pictures for an Emotion-based Recommender System," *2021*

[2] C. Lee, D. Han, S. Choi, K. Han and M. Yi, "Multi-Relational Stacking Ensemble Recommender System Using Cinematic Experience," *2022*

[3] S. Chauhan, R. Mangrola and D. Viji, "Analysis of Intelligent movie recommender system from facial expression," *2021*

[4] A. Dushantha, R. Akalanka, H. Gayan, K. C. Siriwardhana, P. S. Haddela and L. Wickramasinghe, "Evaluation Method for Video Advertisetments Using EEG Signals," *2020*

[5] S. Benini, M. Savardi, K. Bálint, A. B. Kovács and A. Signoroni, "On the Influence of Shot Scale on Film Mood and Narrative Engagement in Film Viewers,"2022

[6] H. Cao and J. Kang, "Study on Improvement of Recommendation Algorithm Based on Emotional Polarity Classification," *2020*

[7] S. K. Bandara, U. C. Wijesinghe, B. P. Jayalath, S. K. Bandara, P. S. Haddela and L. M. Wickramasinghe, "EEG Based Neuromarketing Recommender System for Video Commercials," *2021*

[8] K. Arava, R. S. K. Chaitanya, S. Sikindar, S. P. Praveen and S. D, "Sentiment Analysis using deep learning for use in recommendation systems of various public media applications," *2022*

[9] N. Kannikaklang, S. Wongthanavasu and W. Thamviset, "A Hybrid Recommender System for Improving Rating Prediction of Movie Recommendation," *2022*

[10] https://towardsdatascience.com/recommendation-systems-a-review-d4592b6caf4b

[11] https://www.onespire.net/news/history-of-recommender-systems/

[12] https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6514576/

[13] https://www.ptidej.net/courses/ift6251/fall06/article/Projet%20Ai%20-%20Ilusca%20-%20Yousra.doc.pdf

[14] https://www.verywellmind.com/color-psychology-2795824

[15] https://www.tutorialspoint.com/python/python_overview.htm#:~:text=Python%20is%20a%20high%2Dlevel,syntactical%20constructions%20than%20other%20languages

[16] https://www.verywellmind.com/color-psychology-2795824M.

[17] Borgaonkar, M. Katta, P. Kudale, V. Deshpande, Prof. V. Babanne, "PVR

System: Personalized Video Recommendation", IJRECE, Vol. 7, Issue 4, pp. 302-305, Oct-Dec2019.

[18] Najmeh Samadiani, Guangyan Huang, Borui Cai, Wei Luo, Yong Xiong, Jing He, A Review on Automatic Facial Expression Recognition Systems Assisted by Multimodal Sensor Data

[19] https://www.geeksforgeeks.org/introduction-machine-learning/

[20] Chung-Hsien Wu, Jen-Chun Lin, Wen-Li Wei, Two-Level Hierarchical Alignment for Semi-Coupled HMM-Based Audiovisual Emotion Recognition with Temporal Course

APPENDIX

A. SOURCE CODE

Webapp.py file:

```
import streamlit as st
from PIL import Image
import cv2
import numpy as np
from keras.models import model_from_json
from support import *
from PIL import Image


def detect_emotion(img):
    json_file = open('model/emotion_model.json', 'r')
    loaded_model_json = json_file.read()
    json_file.close()

    emotion_model = model_from_json(loaded_model_json)

    emotion_model.load_weights('model/emotion_model.h5')

    frame = cv2.resize(img, (1280, 720))
    face_detector =
cv2.CascadeClassifier('haarcascades/haarcascade_frontalface_default.xml')
    gray_frame = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)

    num_faces = face_detector.detectMultiScale(gray_frame, scaleFactor=1.3,
minNeighbors=5)

    for (x,y,w,h) in num_faces:
        cv2.rectangle(frame, (x,y-50), (x+w, y+h+10), (0,255,0), 4)
        roi_gray_frame = gray_frame[y:y + h, x:x + w]
        cropped_img = np.expand_dims(np.expand_dims(cv2.resize(roi_gray_frame,
(48, 48)), -1), 0)

        emotion_prediction = emotion_model.predict(cropped_img)
        maxindex = int(np.argmax(emotion_prediction))
        cv2.putText(frame, emotion_dict[maxindex], (x+5, y-20),
cv2.FONT_HERSHEY_SIMPLEX, 1, (255, 0, 0), 2, cv2.LINE_AA)


    return frame, maxindex


if __name__=='__main__':

    st.title("Movie Recommender System")
```

```
    emotion_dict = {0: "Angry", 1: "Disgusted", 2: "Fearful", 4: "Happy", 5:
"Sad", 6: "Surprised"}
    genres = {"Action":28, "Comedy":35, "Crime":80, "Fantasy":14, "Horror":27,
"Thriller":53}
    color_select1 = st.selectbox('Select a Color', ['Black', 'White', 'Red',
"Yellow", 'Blue'], key=1)
    color_select2 = st.selectbox('Select a Color', ['Black', 'White', 'Red',
'Yellow', 'Blue'], key=2)
    color_select3 = st.selectbox('Select a Color', ['Black', 'White', 'Red',
'Yellow', 'Blue'], key=3)
    img_file_buffer = st.camera_input("Capture")
    suggest = st.button('Suggest')
    if suggest:
            if img_file_buffer is not None:
                image = Image.open(img_file_buffer)
                cv2_img = np.array(image)
                try:
                    img, id = detect_emotion(cv2_img)
                    st.write("Detected Emotion from the face:
"+emotion_dict[id])
                    all_colors = [color_select1, color_select2, color_select3]
                    all_movie_names, all_poster_links =
get_all_recom(all_colors, id)
                    for name, poster in zip(all_movie_names,
all_poster_links):
                        st.write(name)
                        st.image(poster)
                except:
                    id = 1
                    all_colors = [color_select1, color_select2, color_select3]
                    all_movie_names, all_poster_links =
get_all_recom(all_colors, id)
                    for name, poster in zip(all_movie_names,
all_poster_links):
                        st.write(name)
                        st.image(poster)
            else:
                all_colors = [color_select1, color_select2, color_select3]
                all_movie_names, all_poster_links = get_all_recom(all_colors,
1)
                for name, poster in zip(all_movie_names, all_poster_links):
                    st.write(name)
                    st.image(poster)
```

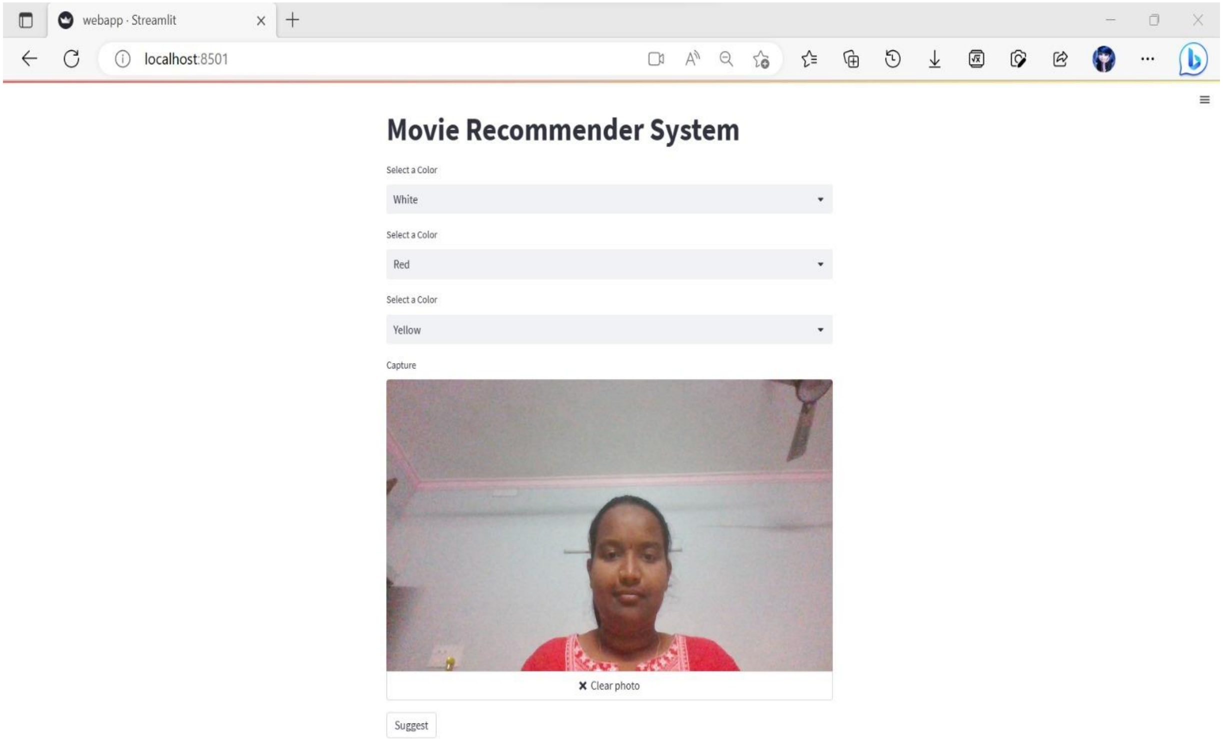
support.py file:

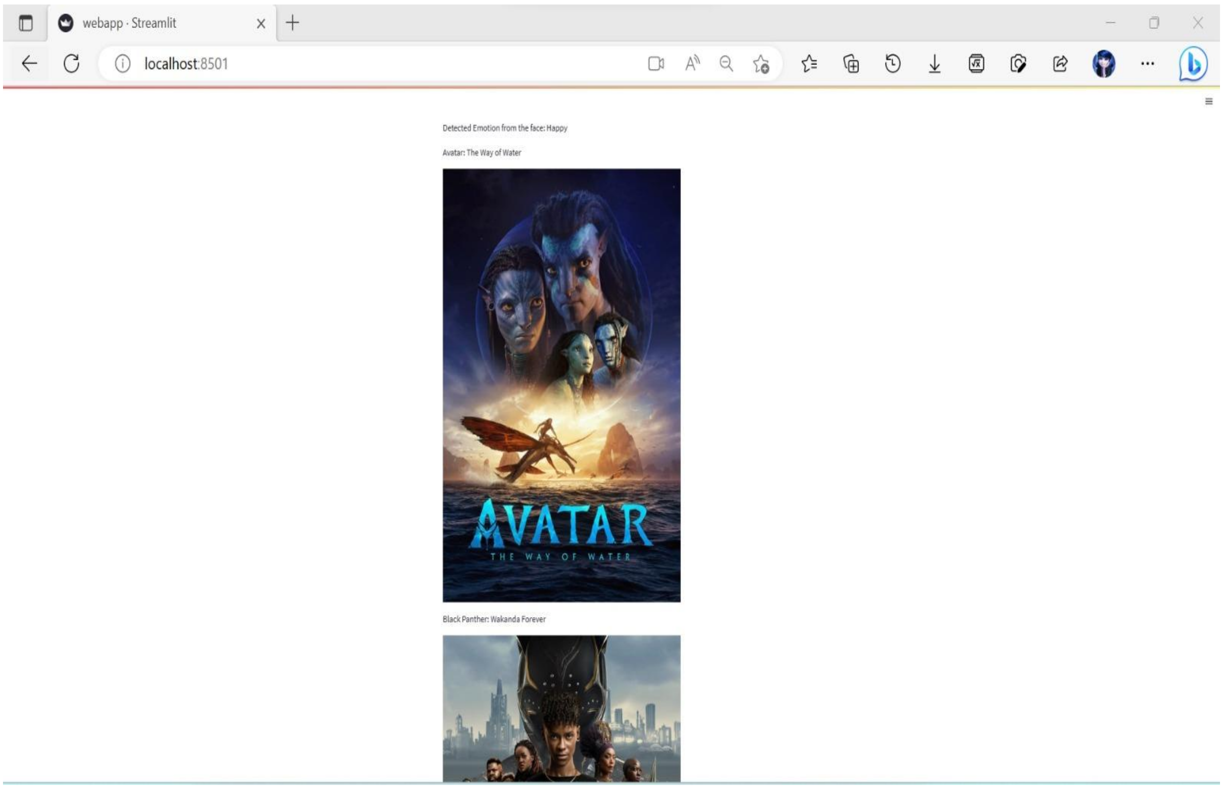
```
import requests
import random as r

color_map = {
```

```
        'Black':0,
        'White': 1,
        'Red':0,
        'Yellow':1,
        'Blue': 1
        }
face_map = {
        'Angry': 0,
        'Disgusted': 0,
        'Fearful': 0,
        'Happy': 1,
        'Sad': 0,
        'Surprised': 1,
        }
emotion_dict = {
        0: "Angry",
        1: "Disgusted",
        2: "Fearful",
        4: "Happy",
        5: "Sad",
        6: "Surprised"
            }
apiKey = 'bdfe31153439600a352617c3ca93d2e1'
a=r.sample([14,27,28,35,878,11749],2)
b=r.sample([28,35,53,80],2)
genre_map = {
        'pos' : a,
        'neg' : b
        }

def get_pos_neg(sentiments):
    pos, neg = 0,0
    for sentiment in sentiments:
        if color_map[sentiment] == 1:
            pos += 1
        else:
            neg += 1

    return (pos,neg)

def get_final_sentiment(color_sentiment, face_id=None):
    if face_id is not None:
        face_val = face_map[emotion_dict[face_id]]
        pos, neg = color_sentiment
        if face_val == 1:
            if pos + 1 > neg:
                return 'pos'
            else:
                return 'neg'
        if face_val == 0:
```

```
            if neg + 1 > pos:
                return 'neg'
            else:
                return 'pos'
    else:
        pos, neg = color_sentiment
        if pos > neg:
            return 'pos'
        return 'neg'

def get_poster(movie_id):
    url =
"https://api.themoviedb.org/3/movie/{}?api_key=bdfe31153439600a352617c3ca93d2e
1&language=en-US".format(movie_id)
    data = requests.get(url).json()
    poster_path = data["poster_path"]
    full_path = "https://image.tmdb.org/t/p/w500/" + poster_path
    return full_path

def suggest_movie(final_emotion):
    genre_list = genre_map[final_emotion]
    movie_names = []
    movie_poster_links = []
    for genre in genre_list:
        response =
requests.get(f"https://api.themoviedb.org/3/discover/movie?api_key={apiKey}&la
nguage=en-
US&sort_by=popularity.desc&include_adult=false&include_video=false&page=1&with
_genres={genre}")

        movies = response.json()['results']
        for movie in movies:
            movie_names.append(movie['original_title'])
            movie_poster_links.append(get_poster(movie['id']))
    return movie_names, movie_poster_links

def get_all_recom(all_color, face_id):
    return suggest_movie(get_final_sentiment(get_pos_neg(all_color), face_id))
```

B. SCREENSHOTS





C. RESEARCH PAPER

Emotionally Driven Film Referral System Using Color Psychology

Pulipati Anshu [1] , Meghana Elisetti [2] , Albert Mayan J [3], Ashok Kumar K [4]

[1,2]U.G Student, Department of CSE, Sathyabama Institute of Science and Technology, Chennai

[3,4] Professor, School of Computing, Sathyabama Institute of Science and Technology, Chennai.

[1]anshudiasyjas257@gmail.com, [2]meghana.elisetii@gmail.com , [3]albert.cse@sathyabama.ac.in, [4]ashokkumar.cse@sathyabama.ac.in

Abstract - The modern world comprises many forms of entertainment, the most common being movies. Over the past decades, the methods of production, creation, and distribution of movies have greatly advanced. People have a variety of movies to choose from and require a recommendation system to guide them in this process. Emotion-based movie recommendation systems (E-MRS) are ones that recommend movies based on the user's emotions. Emotions are a strong reaction to stimuli and are an intelligent and rational form of behaviour. It is difficult to assess the emotional responses to movies based on the diverse reactions to movies. Colour psychology can be used to detect various emotional states such as happiness, sadness, anger, fear, and excitement. In this paper, an user interface is created where colours are used to represent the user's appropriate emotions, and then a movie is recommended based on that. A hybrid approach, combining collaborative filtering and content-based filtering, is used to recommend movies.

Keywords - Movie Recommendation, Color Psychology, Emotion Recognition, Collaborative filtering technique,Content-based filtering technique**.**

I. INTRODUCTION

Recommended systems can forecast user preferences. It reduces transaction costs for online shoppers and boosts traffic for service providers. The use of recommended system increases online service and operating revenue. Non-personalized and personalized recommended systems exist. A non-personalized recommendation system suggests the top 10 holiday spots. A customized recommendation system uses user preferences to recommend the best options. A diverse, individualized recommendation system can recommend a variety of goods. Technology changes user preferences. After the pandemic, this was evident. The pandemic forced everyone indoors. Internet users use video streaming services more often for entertainment.

Direct-to-consumer OTT platforms stream premium content on demand. Netflix, Disney+, Hulu, HBO Now, and Amazon Prime are OTT platforms. Color psychology, facial expressions, movie ratings, user needs, and emojis can recommend movies. Emotions affect ideas, feelings, and actions. Personal physiological arousal causes emotions. Colors are a major source of these feelings. The majority of people enjoy watching movies on a daily basis. Emotion-based Recommender System, also known as an E-MRS, which can keep track of the preferences of clients. Depending on the emotions that they are experiencing. As a direct consequence of the fact that emotions play a large role in determining both reasonable and intelligent behavior. It is important that the sentiments of other users be included into the idea process.

II. RELATED WORK

In [1], shot scale, the camera's apparent distance from a scene's subject, has artistic and narrative functions in any film. The first step is to examine how the distribution and rotation of close, medium, and long shots affect viewers' ratings of film mood to determine the impact of shot scale on responses of lower and higher complexity. The next step is analyzing how shot scale affects violent scene viewers' narrative engagement and its sub-scales. The shot scale and viewer emotional involvement are being further investigated in this study using large corpora.

In [2], machine learning helps tackle real-time commercial and research problems. Machine learning is simply the expansion of mathematical applications. Facial recognition uses machine-learning models. Security, automatic attendance, and offices employ facial recognition in real-time. Facial detection-based movie recommendation is an important machine learning application. Capturing emotion instead of browsing movies saves time. Compared to CNN, decision tree-based facial recognition employing boosting methods is inefficient. CNN is better for accuracy. Social filtering and content-based systems enhance the recommended systems' power.

The scheme developed in [3] comprises sub-emotional states that indicate the temporal phases of apex, offset and onset. This is done in order to align the hidden Markov model with the audio and visual HMM sequences (SC-HMM). This is achieved using a sub-emotional language model that considers the temporal transition between sub-emotional states. The results of the experiment demonstrates that the proposed method can produce satisfactory results in both the posed MHMC and the naturalistic SEMAINE databases.

In [4], the connotative qualities of movies make it hard to quantify their emotional impact. Connotation represents audio-visual descriptor emotions to predict user emotions. Connotative features can recommend movies. Movie suggestion methods vary. This study compares some techniques. This study includes audio elements that can help analyze movie-scene emotions. Emotions match video features. Multi-stream fused the hidden Markov model can identify interest, boredom, annoyance, puzzlement, surprise, etc. A movie soundtrack may also recognize emotions. This study compares all emotional movie recommendation algorithms.

In [5], this study describes the research and enhances the object-based collaborative filtering recommendation system using user reviews and emotional polarity categorization. The algorithm for movie recommendations is presented. Movie review data is categorized using a model developed by CNN based on emotional polarity. The classification of the emotional polarity of the comment is combined with each user's movie rating to create a list of suggested films.

In [6], analyzing the filmmaking and editing techniques that will help audiences understand the message of a movie - has lately addressed the seeming difficulties in judging movie-induced emotions and the indisputable significant heterogeneity in participants' emotional responses to film content. Connotation uses audio-visual descriptors' objectivity to anticipate user emotions. No physiological signs are needed. It uses connotative concepts and user reactions to similar stimuli, not other people's widely varying emotional rates. This study extracts audio-visual and cinematic language descriptors and uses users' connotative ratings to place, compare, and recommend movie scenes.

In [7], machine learning helps tackle real-time commercial and research problems. Machine learning is simply the expansion of mathematical applications. Facial recognition uses machine learning models. Real-time security, automated attendance, and workplace applications use facial recognition. Facial detection-based movie recommendation is an important machine learning application. Capturing emotion instead of browsing movies saves time. Some studies are based on attentional convolutional neural networks and recommended systems to suggest movies or songs based on CNN output. Compared to CNN, decision tree-based facial recognition employing boosting methods is inefficient. CNN is better for accuracy. Content-based and collaborative filtering recommended systems work better together.

In [8], one recent software recommends things based on customer needs. It cannot generalize the recommender system since user needs vary. Connotation predicts viewers' emotions using audio-visual descriptors' objectivity. Connotative space is formed by extracting audio-visual descriptors and the user's emotional state. Then the connotatively closest movies are suggested. Finally, the framework is evaluated subjectively by asking consumers to validate the film components that match their affective needs.

In [9], mind Frame, the planned online app, recommends music and movies based on mood. The suggested system detects and recommends music and movies based on user moods like joy, sadness, and tranquility. The Python OpenCV library extracts facial features. A neural network trained with labeled photos of the user's emotional states predicts their mood. Django, a web framework, is used here to store, view, and process user data and requests.

Finally, in [10], researchers can create context-aware apps that adjust to users' emotions by recognizing facial expressions. Computer vision researchers studied face recognition. Face Fetch, a novel context based multimedia content recommendation system, analyses a user's facial expressions to determine their emotional state (joy, despair, panic, displeasure, amaze, and wrath) and provides multimedia content accordingly.

The existing systems use facial recognition to detect the emotion of the user. A facial recognition system can match a human face from an image or video to a database of faces. It can be useful in airports, ATMs, surveillance (watching and looking for drug users and criminals, controlling CCTV), and security(building access control, flight boarding system, airports, office access, email authentication on multimedia workstations).

Facial Emotion Recognition (FER) is a technology that recognizes emotions from photos and static videos. Some applications of this software are for research areas targeting mental disease diagnosis and human interaction detection. This technology is ground-breaking in areas of research. However, it is challenging to use in practical settings, as demonstrated in a journal by Najmeh Samadiani and distributed by PubMed Central (PMC). According to the journal, 97% of laboratory-controlled FER systems have high accuracy. However, when these results are transferred to real-world applications, 50% of the systems have low accuracy [11].

The low accuracy is due to the large variety of responses people showcase. Some people are more expressive and react obviously to peculiar scenarios, while others are unable to do so and react in less obvious manners. According to an IEEE journal by According to Chung-Hsien Wu and Jen-Chun Lin, introverts' facial expressions differ significantly from those of extroverted people [12]. Hence this suggests that it is hard for a system to accurately recognize the emotions of the user merely based on their facial recognition, as there is no standard reaction to showcase that the user is feeling a certain way. While an introvert might feel happy but only smile modestly in response, an extrovert would respond with a broad grin. However, it does not imply that the introvert is least happy than the extrovert, but it means that there is more than one way to showcase the feeling.

III. METHODOLOGY

The proposed system is to create a user interface to recommend movies based on user emotions using color psychology and the user's facial expressions. There are five basic emotions that a human being has: love, joy, anger, sadness, and fear. Some researchers discovered that colors significantly influence users' emotions and feelings. Color is a natural form to represent human emotions. According to figure 1 below, the colors represent a user's positive and negative emotions.

POSITIVE EMOTIONS	NEGATIVE EMOTIONS
Yellow: Joy, Merry, Good mood	**Black:Sadness, expression and loneliness**
Green: Happiness, calmness, and feelings of relieving	**Grey: Sadness, depression, and loneliness**
Blue: Pleasure and Happiness	**Brown: Sadness and depression**
Red: love, passion, and excitation	**Red: Anger**
Orange: cheerfulness	**Orange: fear and distress**

Fig. 1: Emotions Varying with Colors

A. Data acquisition and pre-processing

The TMDB film database, known for its extensive marking, was utilized to narrow the pool of possible outcomes. As the primary identifier for clustering, the use of genre was choosen. This brief examination has resulted in the identification of 24 distinct genres. These films span the gamut from action to history, and many fall into multiple categories. There are many of products that may not even come close to summarizing a complete film story. To make meaningful discoveries, the need of more information is necessary. Obtaining as much information as possible is mainly limited by the time period. As the data model becomes more refined, less of it is considered. There may be one significant advantage depending on how recommendations are made. Structures for sharing information with social media, movie forums, machine learning forums, and other online channels are required. Our data collection for the recommender system took roughly eight weeks. Users of the sentiment-based recommender contributed thousands of individual opinions.

The datasets used in this project were obtained from the Kaggle website. The datasets are “tmdb_5000_movies.csv” and “tmdb_5000_credits.csv” as shown below. In the “tmdb_5000_movies.csv” dataset, there are 4803 rows and 20 columns. It comprises budget, genre, homepage, original language, id, original title, companies, overview, popularity, keywords, release date, production countries, revenue, spoken languages, runtime, tagline, title, vote average, status, and vote count as fields. There are 4 columns and 4803 rows in the "tmdb 5000 credits.csv" dataset, with fields like crew, cast, title, and movie id. The two datasets contain 24 genres. The datasets are joined based on the column shared by both CSV files. To improve results, null or missing values will be checked while training the model. If there are any null values or missing values, drop those values. Among all the columns in the merged dataset, the selected features are genres, keywords, cast, crew, title, overview, and movie id.

Table 1: Credits TMDB dataset

movie_id	title	cast	crew
19995	Avatar	[{"cast_id": 242, "character": "Jake Sully", "...	[{"credit_id": "52fe48009251416c750aca23", "de...
285	Pirates of the Caribbean: At World's End	[{"cast_id": 4, "character": "Captain Jack Spa...	[{"credit_id": "52fe4232c3a36847f800b579", "de...
206647	Spectre	[{"cast_id": 1, "character": "James Bond", "cr...	[{"credit_id": "54805967c3a36829b5002c41", "de...
49026	The Dark Knight Rises	[{"cast_id": 2, "character": "Bruce Wayne / Ba...	[{"credit_id": "52fe4781c3a36847f81398c3", "de...
49529	John Carter	[{"cast_id": 5, "character": "John Carter", "c...	[{"credit_id": "52fe479ac3a36847f813eaa3", "de...

Table 2: Movies TMDB dataset

budget	genres	homepag	id	keywords	original_l	original_t	overview	popularity	productio	productio	release_d	revenue	runtime	spoken_la	status	tagline	title	vote_aver	vote_count
2.37E+08	[{"id": 28,	http://ww	19995	[{"id": 146	en	Avatar	In the 22n	150.4376	[{"name":	[{"iso_316	########	2.79E+09	162	[{"iso_639	Released	Enter the	Avatar	7.2	11800
3E+08	[{"id": 12,	http://dis	285	[{"id": 270	en	Pirates of	Captain B	139.0826	[{"name":	[{"iso_316	########	9.61E+08	169	[{"iso_639	Released	At the en	Pirates of	6.9	4500
2.45E+08	[{"id": 28,	http://ww	206647	[{"id": 470	en	Spectre	A cryptic r	107.3768	[{"name":	[{"iso_316	########	8.81E+08	148	[{"iso_639	Released	A Plan No	Spectre	6.3	4466
2.5E+08	[{"id": 28,	http://ww	49026	[{"id": 849	en	The Dark I	Following	112.313	[{"name":	[{"iso_316	########	1.08E+09	165	[{"iso_639	Released	The Leger	The Dark I	7.6	9106
2.6E+08	[{"id": 28,	http://mc	49529	[{"id": 818	en	John Carte	John Carte	43.927	[{"name":	[{"iso_316	3/7/2012	2.84E+08	132	[{"iso_639	Released	Lost in ou	John Carte	6.1	2124
2.58E+08	[{"id": 14,	http://ww	559	[{"id": 851	en	Spider-Ma	The seem	115.6998	[{"name":	[{"iso_316	5/1/2007	8.91E+08	139	[{"iso_639	Released	The battle	Spider-Ma	5.9	3576
2.6E+08	[{"id": 16,	http://dis	38757	[{"id": 156	en	Tangled	When the	48.68197	[{"name":	[{"iso_316	########	5.92E+08	100	[{"iso_639	Released	They're ta	Tangled	7.4	3330
2.8E+08	[{"id": 28,	http://ma	99861	[{"id": 882	en	Avengers:	When Tor	134.2792	[{"name":	[{"iso_316	########	1.41E+09	141	[{"iso_639	Released	A New Ag	Avengers:	7.3	6767
2.5E+08	[{"id": 12,	http://har	767	[{"id": 616	en	Harry Pott	As Harry b	98.88564	[{"name":	[{"iso_316	7/7/2009	9.34E+08	153	[{"iso_639	Released	Dark Secr	Harry Pott	7.4	5293
2.5E+08	[{"id": 28,	http://ww	209112	[{"id": 849	en	Batman v	Fearing th	155.7905	[{"name":	[{"iso_316	########	8.73E+08	151	[{"iso_639	Released	Justice or	Batman v	5.7	7004
2.7E+08	[{"id": 12,	http://ww	1452	[{"id": 83,	en	Supermar	Supermar	57.92562	[{"name":	[{"iso_316	########	3.91E+08	154	[{"iso_639	Released		Supermar	5.4	1400
2E+08	[{"id": 12,	http://ww	10764	[{"id": 627	en	Quantum	Quantum	107.9288	[{"name":	[{"iso_316	########	5.86E+08	106	[{"iso_639	Released	For love, f	Quantum	6.1	2965

B. Building Emotion Detector

This algorithm's job is to deduce the user's emotional state from a palette of three colors. So that it can achieve that, it will use the following reasoning to examine the color sequence: If the user selects three colors and at least two of them have the same emotional connotation, then that feeling is the one they are experiencing at the moment. If the user selects the hues yellow, blue, and green or yellow, blue, and black, for instance, they are experiencing elation. An alternative scenario exists where the first hue chosen represents happiness, the second hue signifies love, and the third represents either sadness or wrath (negative emotion).

The user's current emotional state is "joy-love" in this case. If, on the other hand, two of the colors signify melancholy and one suggests fury, then sadness and anger represent the user's current emotional state. If a person selects the positive feelings of "joy-love". For instance, the result will be a combination of the following colors: yellow, light red, and black. Movie evaluations and associated feelings are recorded in user accounts. All of this data may be found in the User Profiles repository. The user will be asked to fill out a survey during registration to share his thoughts on which kind of movies best represent certain moods. A user's choice of movie is treated as an implicit 5-star rating.

Love, rage, joy, and sadness are the four categories of user emotions that make up the profile vector. The user's preferred genres and titles of films to view when experiencing an emotion E are subsequently catalogued under each emotion group E. If user u selects films A and B for the emotion "love," films C and D for "anger," and film D for "joy," then user u's profile vector would look like this: u = {{A: 5, B: 5}, {C: 5}, {D: 5}, {}}. To provide input on the system's film suggestion, the user can rate the degree to which the film is in line with his present may suggest film F; nevertheless, the user may not agree that he would like to see this film while he is in love, and so he may give it a rating of 1 out of 5. This transforms the letter u into a profile vector: u = {{A: 5, B: 5, F: 1}, {C: 5}, {D: 5}, {}}.

The colors used in this project are black, white, red, yellow, and blue. The user is required to select three colors among the colors mentioned. Each color is assigned 1 and 0 values. Negative and positive emotions are represented by numbers 1 and 0. The colors are mapped by 1 and 0 if the user selects black, white, and blue. The mapped values are 0, 1, and 1. The resulting emotion is positive if the mapped value of positive emotions exceeds the assigned value of negative emotions. Otherwise, the outcome would be negative.

C. Training Model

Collaborative filtering and content-based recommendation cascade in the recommendation algorithm. Collaborative filtering: below is the explanation for how our collaborative filtering approach can be used in recommendation.

The executive phases of the CF algorithm for a database U of user profiles, a target user u, a movie m, and an emotion e are:

Step 1: Get u's database profile vector. User Profiles database U preferences are considered. For the target user u, his profile vector is u = {{A: 5, B: 5, F : 1}, {C : 5}, {D : 5}, {}}.

Step 2: Find users who have rated at least one movie with u. The system will only consider users who share at least one movie-emotion with the target user u to decrease the computing user-film matrix. e.g., Let U_n is a subcollection of U with just movie-emotion-sharing participants. u_1, u_2, u_3 3 U_n users: u1 = {{A: 5, F: 2}, {C: 5}, {E: 5}, {I: 5}} and u2 = {{B: 5}, {}, {G : 5}, {H : 4}}.

Users rate the same movie differently. Thus, no algorithm can be more accurate than the variance in user ratings for the same item. Users rarely detect recommender system accuracy changes. Some recommender systems give users accurate but pointless recommendations. Most movie fans have seen Lord of the Rings. As a result, advising users again is worthless. Accuracy, mean absolute error, and novelty factor should be used to improve evaluation findings. The mean absolute error is defined as the average absolute difference between a predicted rating and the actual rating provided by the user.

Thus, it may assess how well the recommender system predicts user ratings. Precision is the ratio of well-recommended things to rated recommendations. Well-recommended movies are new releases with a minimum 4/5 rating. Our tests determine how well our system's recommended products match user needs and preferences. Emotion detector precision is another goal.

D. Comparison of Face Emotion and Color Emotion

Six user emotions are taken into account in this project, like anger, sadness, happiness, disgust, surprise, and fear, and these emotions will be recognized by the user's face. Angry, fear, sadness, disgusted comes under negative emotion. Surprised and happy come under positive emotion. If the face-detected emotion is negative and the three color choice produces a positive emotion, then the movies which results in positive emotions to the user will be recommended. Dot operator is used to compare the face emotion and color emotion.

E. Structure of the Developed Framework

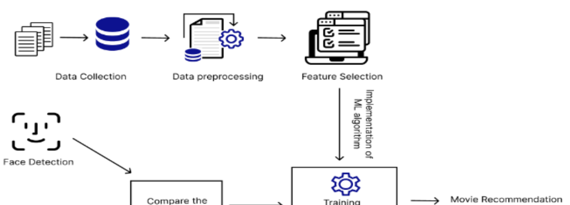


Fig 2: Development of the New Methodology

The primary goal is to create a website where a user can get recommendations of movie by their emotions using color psychology and facial gestures. Color psychology is an effective way of detecting. a user's emotions through color. The TMDB movie dataset is used to train and test a model to detect the appropriate emotions. Any missing values, errors, or outliers must be corrected. To accomplish this, a preprocessing technique must be used to improve accuracy. The following step is to visualize the data.

The hybrid technique is then used to extract features in the following step. The two techniques used in this project to accomplish this are content-based filtering and social filtering. The collaborative filtering technique deals with the similarities of features between users and recommends personalized movies, whereas content-based filtering is all about filtering a movie's like, dislike, or ratings. The user profiles vector using matrix system and emotions of a user will be extracted with the help of these two techniques.

As a result, a model is developed to recommend movies based on emotions. The user will create his profile and choose any three distinct colors here. Following that, the user must complete a survey. Every color represents a different emotion (from color psychology). As a result of the survey results, the model will recommend movies based on the emotions expressed by the user through the use of colors and facial emotion.

IV. RESULTS AND DISCUSSION

In this paper, a recommendation system is designed to help the user find the best movies to watch based on the genre the user like. For the simple reason that viewers either loved or hated the movies, our system only considered ratings of 1 and 5. This approach provides far superior recommendations to consumers since it lets them comprehend the connection between their feelings and the suggested content. When a user gives a film's genre a high rating, similar film genres are suggested. What makes this system unique is that it takes into account the user's feelings and uses that information to make informed product recommendations. From these findings, it can be infer that using more dense data will improve the recommender's performance and that include additional genres will allow us to provide more targeted suggestions.

In addition, some studies have been carried out to demonstrate the accuracy of our predictions when employing emotional feature data. Movies that have received ratings of 3 or above are presumed to be loved by the user, whilst movies that have received ratings of 1 or 2 are supposed to be despised by the user. Unclear 3 rating is not considered. The result matrix is produced by making use of the ratings matrix and the genres matrix. The result matrix is the dot product of the two matrices that came before it. The outcome then undergoes an additional conversion into a binary format.

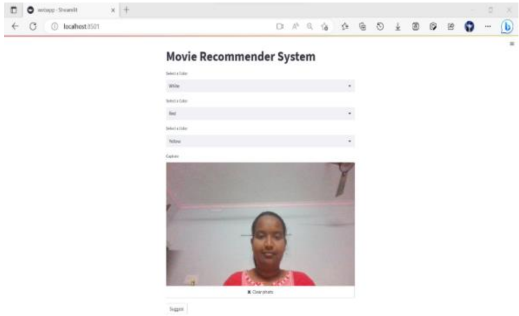


Fig 3: Detection of Emotions

If the result of the dot product is greater than 0, then the value 1 is given to the cell inquestion; otherwise, the value 0 is used. During 1918–19 by completely closing their borders. Other studies suggested that travel restrictions could delay the start of local transmission and slow global spread. According to the scant evidence available, NPIs related to international travel would be ineffective in containing a pandemic of influenza and would require a significant investment in terms of funding.

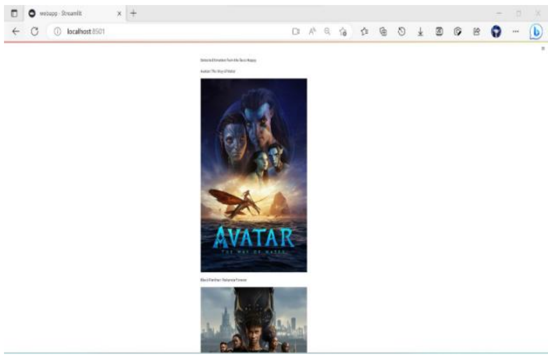


Fig 4: Films Referred According to Emotions

V. CONCLUSION

The study's findings show how emotion-based movie recommendation algorithms might improve user interaction and engagement with movie streaming services. The system can improve user satisfaction and loyalty by making personalised recommendations that are in line with users' emotional states and preferences. The experience of watching a movie is heavily influenced by emotions, and neglecting them might result in lower user engagement. This would ultimately promote the business growth of these services. Future studies in this field might examine the usefulness of emotion-based movie recommendation systems in various cultural and demographic contexts, as well as the feasibility of combining other user data (such as social media usage and browsing history) into the algorithm for making recommendations.

REFERENCES

1. Savardi, M., Benini, S., Kovacs, A. B., Balint, K., & Signoroni, A., "On the influence of shot scale on film mood and narrative engagement in film viewers", IEEE Transactions on Affective Computing, 13(2), 592-603, 2022.
2. Chauhan, Mangrola, Shavak, Rajdeep & D. Viji., "Analysis of Intelligent movie recommender system from facial expression", 2021.
3. Mayan J.A., Ravi. T ,"Test Case Optimization Using Hybrid Search Technique", ACM International Conference Proceeding Series ,ACM New York, 2014, ,doi:10.1145/2660859.2660906.

4. N. G. Meshram and A. P. Bhagat, "Connotative features based affective movie recommendation system", International Conference on Information Communication and Embedded Systems (ICICES2014), pp. 1-7, 2014.
5. H. Cao and J. Kang, "Study on Improvement of Recommendation Algorithm Based on Emotional Polarity Classification", 5th International Conference on Computer and Communication Systems (ICCCS), 2020, pp. 182-186, doi: 10.1109/ICCCS49078.2020.9118414.
6. S. Benini, L. Canini, and R. Leonardi, "Affective Recommendation of Movies Based on Selected Connotative Features" in IEEE Transactions on Circuits and Systems for Video Technology, vol. 23, no. 4, pp. 636-647, April 2013.
7. S. Chauhan, R. Mangrola and D. Viji (2021), "Analysis of Intelligent movie recommender system from facial expression", 5th International Conference on Computing Methodologies and Communication , pp. 1454-1461.
8. N. Balaganesh, K. Muneeswaran and N. Sivakumar, "Feature selection for recommendation of movies" in Global Conference on Communication Technologies (GCCT), pp. 250-255, 2015.
9. Mayan J.A, Arifa S, Pavithra R,"Semantic based multi lexical ranking technique for an effective search in protected cloud,2016 International Conference on Control, Instrumentation, Communication and Computational Technologies, 2016, pp. 570-576.
10. M. Gupta, B. Nithin, S. V. Amirishetty, and H. Kurakula(2021),"Mind Frame Music and Movie Recommendations to uplift the current mood using Deep Learning", International Conference on System, Computation, Automation and Networking , pp. 1-6.
11. M. B. Mariappan, M. Suk and B. Prabhakaran, "FaceFetch: A User Emotion Driven Multimedia Content Recommendation System Based on Facial Expression Recognition", IEEE International Symposium on Multimedia, 2012, pp. 84-87, doi:10.1109/ISM.2012.24.
12. Albert Mayan J, Karthikeyan S, Nikhil C, Bharat M and Padmavathy J, "Facial attendance system technology using Microsoft Cognitive Services" ,International Journal of Engineering Systems Modelling and Simulation, Vol. 12, Nos. 2/3, pp. 180-187 ,2021
13. Mary Posonia A, S. Vigneshwari, Albert Mayan J, D. Jamunarani,"Service Direct : Platform that Incorporates Service Providers and Consumers Directly", International Journal of Engineering and Advanced Technology (IJEAT) , Vol.8 ,No.6, 2019

Gmail

Anshu Pulipati <anshudiasyjas257@gmail.com>

Fwd:

meghana elisetti <meghana.elisetii@gmail.com> 12 February 2023 at 14:48
To: anshudiasyjas257@gmail.com

Sent from my iPhone

Begin forwarded message:

From: Albert John <albertmayan@gmail.com>
Date: 10 February 2023 at 10:12:47 PM IST
To: meghana elisetti <meghana.elisetii@gmail.com>
Subject: Fwd:

---------- Forwarded message ---------
From: **ICONSTEM 2023** <iconstem2023@gmail.com>
Date: Mon, 6 Feb 2023 at 22:26
Subject:
To: <albertmayan@gmail.com>

Dear Author,
Congratulations!!! We are pleased to inform you that your paper with

Paper-id: ICON23T113

Title of the paper : Download Manager and Media Fetcher with Multi-Connection Downloading

has been accepted for **ORAL** presentation in the Eighth International Conference on Science Technology Engineering and Mathematics (ICONSTEM2023) to be held during, 6th & 7th, April, 2023.

You are requested to register for the conference so that the paper will be recommended for possible publication in the IEEE Digital Xplore (Scopus Indexed) on or before 15.02.2023. Ensure that the camera ready copy of the paper in the prescribed format shall reach us in time.

Deadline for registration : 15th Feb 2023
Deadline for camera ready paper submission: 1st March 2023

Each accepted paper has to be registered separately.

Only registered and presented paper will be considered for IEEE publication.

Registration fee for the Conference:

Category	IEEE Member		Non IEEE Member	
	Indian (INR)	Other than India	Indian (INR)	Other than India
Industry / R & D	8500 7000	$175	9000 7500	$200
Research Scholar / Students	6000	$150	6500	$180
Academic(Professor)	1000	$175	500	$200
Listener with certificate		$60	1	$75

BANK DETAILS FOR REGISTRATION:

Account No	**729143907**
IFSC CODE	**IDIB000J037**

Beneficiary	**IEEE Student Branch Counsellor , Jeppiaar Engineering College , Chennai -600119**
Bank Name	**Indian Bank**
Branch	**Semmencherry Branch**

After the payment procedure, please fill the registration form using the following link and upload the required documents.

https://iconstem.jeppiaarcollege.org/register.html

For preparing the camera ready copy, standard templates that are readily available in the following IEEE link can be used (Authors who have already used proper IEEE templates, please ignore):
https://www.ieee.org/conferences_events/conferences/publishing/templates.html

For any queries please feel free to contact:

Dr. J.Jebastine , Professor/ECE – iconstem2023@gmail.com +91 9566143828
Mrs. R. Gracelin Sheeba , Assistant Professor /ECE - +91 9486249231

Regards,
ICON STEM 2023 -Organizing Team

Reviewer(s)' Comments to Author:

OVERALL EVALUATION: Accept

Here with I enclose the plagiarism report.
Regards,
Dr. J.Jebastine., M.E., Ph.D.,
Convenor & IEEE Student Branch Counsellor
Professor and Head, Department of ECE,
Contact No:9566143828,
Jeppiaar Engineering College,
Jeppiaar Nagar, OMR, Chennai-600 119.

ICON23T113.pdf
1755K