MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR

(A Govt. Aided UGC Autonomous & NAAC Accredited Institute Affiliated to RGPV, Bhopal)

Project Report

on

Recommendation System

Submitted By:

Shivansh Sharma

0901CS191116

Shubh Garg

0901CS191118

Faculty Mentor:

Mr. Mir Shahnawaz Ahmad

Assistant Professor, Computer Science and Engineering

Dr. Surbhi Kansal

Assistant Professor, Electronics Engineering

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE

GWALIOR - 474005 (MP) est. 1957

MAY-JUNE 2022

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR

(A Govt. Aided UGC Autonomous & NAAC Accredited Institute Affiliated to RGPV, Bhopal)

Project Report

on

Recommendation System

Submitted By:

Shivansh Sharma

0901CS191116

Shubh Garg

0901CS191118

Faculty Mentor:

Mr. Mir Shahnawaz Ahmad

Assistant Professor, Computer Science and Engineering

Dr. Surbhi Kansal

Assistant Professor, Electronics Engineering

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE

GWALIOR - 474005 (MP) est. 1957

MAY-JUNE 2022

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR

(A Govt. Aided UGC Autonomous & NAAC Accredited Institute Affiliated to RGPV, Bhopal)

CERTIFICATE

This is certified that **Shivansh Sharma** (0901CS191116) has submitted the project report titled **Recommendation System** under the mentorship of **Mr. Mir Shahnawaz Ahmad**, in partial fulfilment of the requirement for the award of degree of Bachelor of Technology in Computer Science and Engineering from Madhav Institute of Technology and Science, Gwalior.

09/05/22

Mr. Mir Shahnawaz Ahamad

Faculty Mentor

Assistant Professor

Computer Science and Engineering

9/5/22

Dr. Shubhi Kansal

Faculty Mentor

Assistant Professor

Electronics Engineering

09/05/2022

Dr. Manish Dixit

Professor and Head

Computer Science and Engineering

Dr. Manish Dixit
Professor & HOD
Department of CSE
M.I.T.S. Gwalior

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR

(A Govt. Aided UGC Autonomous & NAAC Accredited Institute Affiliated to RGPV, Bhopal)

CERTIFICATE

This is certified that **Shubh Garg (0901CS191118)** has submitted the project report titled **Recommendation System** under the mentorship of **Mr. Mir Shahnawaz Ahmad**, in partial fulfilment of the requirement for the award of degree of Bachelor of Technology in Computer Science and Engineering from Madhav Institute of Technology and Science, Gwalior.

09/05/22

Mr. Mir Shahnawaz Ahamad

Faculty Mentor

Assistant Professor

Computer Science and Engineering

9/5/22

Dr. Shubhi Kansal

Faculty Mentor

Assistant Professor

Electronics Engineering

09/05/22

Dr. Manish Dixit

Professor and Head

Computer Science and Engineering

Dr. Manish Dixit
Professor & HOD
Department of CSE
M.I.T.S. Gwalior

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR

(A Govt. Aided UGC Autonomous & NAAC Accredited Institute Affiliated to RGPV, Bhopal)

DECLARATION

I hereby declare that the work being presented in this project report, for the partial fulfilment of requirement for the award of the degree of Bachelor of Technology in Computer Science and Engineering at Madhav Institute of Technology & Science, Gwalior is an authenticated and original record of my work under the mentorship of **Mr. Mir Shahnawaz Ahmad**, Assistant Professor, Computer Science and Engineering and **Dr. Shubhi Kansal**, Assistant Professor, Electronics Engineering.

I declare that I have not submitted the matter embodied in this report for the award of any degree or diploma anywhere else.

Shivansh Sharma
0901CS191116
CSE 3rd Year
Computer Science and Engineering

Shubh Garg
0901CS191118
CSE 3rd Year
Computer Science and Engineering

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR

(A Govt. Aided UGC Autonomous & NAAC Accredited Institute Affiliated to RGPV, Bhopal)

DECLARATION

I hereby declare that the work being presented in this project report, for the partial fulfilment of requirement for the award of the degree of Bachelor of Technology in Computer Science and Engineering at Madhav Institute of Technology & Science, Gwalior is an authenticated and original record of my work under the mentorship of **Mr. Mir Shahnawaz Ahmad**, Assistant Professor, Computer Science and Engineering and **Dr. Shubhi Kansal**, Assistant Professor, Electronics Engineering.

I declare that I have not submitted the matter embodied in this report for the award of any degree or diploma anywhere else.

Shivansh Sharma
0901CS191116
CSE 3rd Year
Computer Science and Engineering

Shubh Garg
0901CS191118
CSE 3rd Year
Computer Science and Engineering

Abstract

In this project, we build a recommendation system that suggests movies to you based on what interests you. The recommendation system takes the name of the movie from the website and gives it to the recommendation model which gives back the names of the most likely movies along with their movie posters. In this project, we use the TMDB Api to fetch recommended movie posters that get movies posters from their poster id. With the help of Stream lit framework, we display the recommended movies on the website. We deploy our project on HEROKU CLOUD, so that anyone can access our project anywhere in the world with just one click via internet.

Keywords: Recommend, NLP, ML, Movies, Talkies-talk, Prediction, TMDB API

सार:

इस परियोजना में, हम एक अनुशंसा प्रणाली का निर्माण करते हैं जो आपको आपकी रुचि के आधार पर फिल्मों का सुझाव देती है। सिफारिश प्रणाली वेबसाइट से फिल्म का नाम लेती है और इसे सिफारिश करने वाले मॉडल को देती है जो उनके मूवी पोस्टर के साथ सबसे संभावित फिल्मों के नाम वापस देता है इस परियोजना में, हम अनुशंसित मूवी पोस्टर लाने के लिए tmdb api का उपयोग करते हैं जो पोस्टर को उनके पोस्टर आईडी से प्राप्त करते हैं। स्ट्रीमलाइट फ्रेमवर्क की सहायता से हम वेबसाइट पर अनुशंसित फिल्में प्रदर्शित करते हैं। हम अपने प्रोजेक्ट को HEROKU CLOUD पर तैनात करते हैं, ताकि कोई भी इंटरनेट के माध्यम से सिर्फ एक क्लिक के साथ दुनिया में कहीं भी हमारे प्रोजेक्ट तक पहुंच सके।

LIST OF FIGURES

TABLE OF CONTENTS

CHAPTER 1: INTRODUCTION

1.1 What is a recommender system?

A recommender system is a simple algorithm whose aim is to provide the most relevant information to a user by discovering patterns in a dataset. The algorithm rates the items and shows the user the items that they would rate highly. An example of recommendation in action is when you visit Amazon and you notice that some items are being recommended to you or when Netflix recommends certain movies to you. They are also used by Music streaming applications such as Spotify and Deezer to recommend music that you might like.

Below is a very simple illustration of how recommender systems work in the context of an e-commerce site:

Two users buy the same items A and B from an e-commerce store. When this happens the similarity index of these two users is computed. Depending on the score the system can recommend item C to the other user because it detects that those two users are similar in terms of the items they purchase.

1.2 Need of recommendation system

➢ In the immortal words of Steve Jobs: "A lot of times, people don't know what they want until you show it to them." Customers may love your movie, your product, your job opening- but they may not know it exists. The job of the recommender system is to open the customer/user up to completely new products and possibilities, which they would not think to directly search for themselves.

➢ With the growing amount of information on the internet and with a significant rise in the number of users, it is becoming important for companies to search, map and provide them with the relevant chunk of information according to their preferences and tastes.

1.3 Types of recommendation systems

There are majorly three important types of recommendation systems:

1. Collaborative filtering
2. Content-Based Filtering

1.3.1 Collaborative filtering

➢ This filtering method is usually based on collecting and analyzing information on user's behaviors, their activities or preferences and predicting what they will like based on the similarity with other users.

➢ A key advantage of the collaborative filtering approach is that it does not rely on machine analyzable content and thus it is capable of accurately recommending complex items such as movies without requiring an "understanding" of the item itself.

➢ Collaborative filtering is based on the assumption that people who agreed in the past will agree in the future, and that they will like similar kinds of items as they liked in the past. ➢ For example, if a person A likes item 1, 2, 3 and B like 2,3,4 then they have similar interests and A should like item 4 and B should like item 1.

1.3.2 Content-based filtering

➢ These filtering methods are based on the description of an item and a profile of the user's preferred choices.

➢ In a content-based recommendation system, keywords are used to describe the items; besides, a user profile is built to state the type of item this user likes. In other words, the algorithms try to recommend products which are similar to the ones that a user has liked in the past.

➢ The idea of content-based filtering is that if a user like an item then he/she will also like a 'similar' item.

➢ For example, when we are recommending the same kind of item like a movie or song recommendation. This approach has its roots in information retrieval and information filtering research.

➢ A major issue with content-based filtering is whether the system is able to learn user preferences from users actions about one content source and replicate them across other different content types.

➢ When the system is limited to recommending the content of the same type as the user is already using, the value from the recommendation system is significantly less when other content types from other services can be recommended. For example, recommending news articles based on the browsing of news is useful, but wouldn't it be much more useful when music, videos from different services can be recommended based on the news browsing.

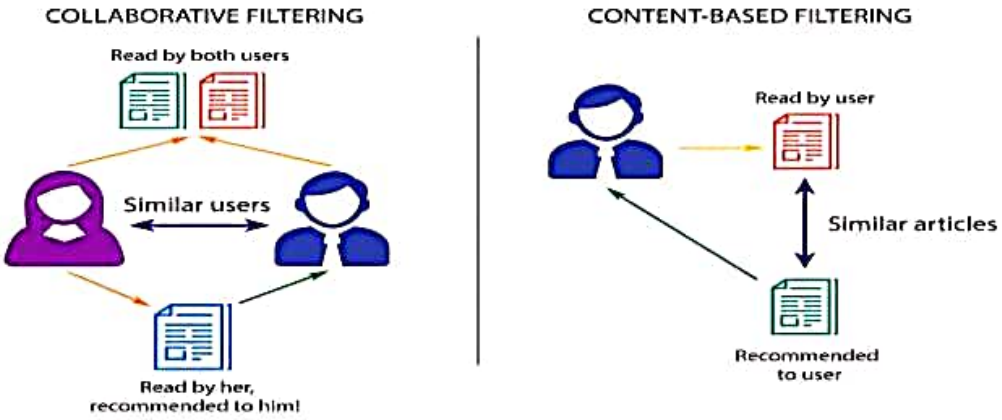

Fig 1.1: Filtering method representation of collaborative and content-based filtering

1.4 Basic terminologies

1.4.1. Labels: A label is a thing we're predicting. For example, the label could be the future price of wheat, the kind of animal shown in a picture, the meaning of an audio clip, or just about anything.

1.4.2. Feature: A feature is an input variable. For example, in spam detector example, the features could include the words in the email text, sender's address etc.

1.4.3 Model: It defines the relationship between features and label. For example, a spam detection model might associate certain features strongly with "spam".

1.4.4 Vectorization: We are dealing with text data . So, to get insights about user interest we need to find the similarities between the movie user see and the rest in dataset. To do this , we convert all the key words into axis and then convert every tag into a vector.

1.4.5 Cosine Similarity : It is a technique used to calculate the cosine distance between points instead of using the regularly used Euclidean distance.

1.4.6 Stem : Stemming is the process of reducing a word to its word stem that affixes to suffixes and prefixes or to the roots of words known as a lemma. Stemming is important in natural language understanding (NLU) and natural language processing (NLP).

CHAPTER 2: LITERATURE REVIEW

Recommender systems have grown to be a critical research subject after the emergence of the first paper on collaborative filtering in the Nineties. Despite the fact that educational studies on recommender systems, has extended extensively over the last 10 years, there are deficiencies in the complete literature evaluation and classification of that research. Because of this, we reviewed articles on recommender structures, and then classified those based on sentiment analysis. The articles are categorized into three techniques of recommender system, i.e.; collaborative filtering (CF), content based and context based. We have tried to find out the research papers related to sentimental analysis based recommender system. To classify research done by authors in this field, we have shown different approaches of recommender system based on sentimental analysis with the help of tables. Our studies give statistics, approximately trends in recommender structures research, and gives practitioners and researchers with perception and destiny route on the recommender system using sentimental analysis. We hope that this paper enables all and sundry who is interested in recommender systems research with insight for destiny.

We used NLP (Natural Language Processing) in our project. Natural language processing (NLP) refers to the branch of computer science—and more specifically, the branch of artificial intelligence or AI—concerned with giving computers the ability to understand text and spoken words in much the same way human beings can.

NLP combines computational linguistics—rule-based modeling of human language—with statistical, machine learning, and deep learning models. Together, these technologies enable computers to process human language in the form of text or voice data and to 'understand' its full meaning, complete with the speaker or writer's intent and sentiment.

CHAPTER 3: PRELEMINARY DESIGN

3.1 Software Development Life Cycle Model

3.1.1 Rapid Application Development

Reason: Since the software size was not much large and there was a time-bound and the project was made in modules therefore in this project, I used Rapid Application Development. A software project can be implemented using this model if the project can be broken down into small modules wherein each module can be assigned independently to separate teams. These modules can finally be combined to form the final product.

3.2 Data Flow Diagram

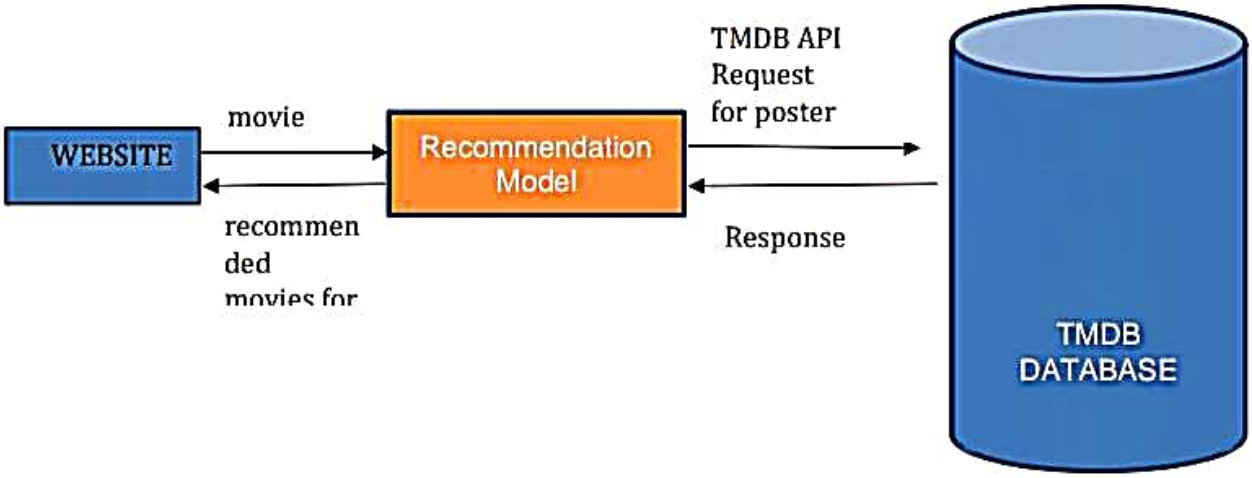

fig 3.1 data flow diagram for the recommender system

3.3 Tools & Technologies

3.3.1 Python

Python is a high-level, interpreted, interactive, and object-oriented scripting language. Python is designed to be highly readable. It uses English keywords frequently whereas other languages use punctuation, and it has fewer syntactical constructions than other languages.

3.3.2 Natural Language Processing

Natural language processing is a subfield of linguistics, computer science, and artificial intelligence concerned with the interactions between computers and human language, in particular how to program computers to process and analyze large amounts of natural language data

3.3.3 Jupyter Notebook

Jupyter Notebook is a web-based interactive computational environment for creating

notebook documents. A Jupyter Notebook document is a browser-based REPL containing an ordered list of input/output cells which can contain code, text, mathematics, plots and rich media. Underneath the interface, a notebook is a JSON document, following a versioned schema, usually ending with the ".ipynb" extension.

3.3.4 Visual Studio Code

Visual Studio Code is a source-code editor made by Microsoft for Windows, Linux and macOS. Features include support for debugging, syntax highlighting, intelligent code completion, snippets, code refactoring, and embedded Git. Users can change the theme, keyboard shortcuts, preferences, and install extensions that add additional functionality.

3.3.5 Anaconda

Anaconda is a distribution of the Python and R programming languages for scientific computing that aims to simplify package management and deployment. The distribution includes data-science packages suitable for Windows, Linux, and macOS. We used anaconda to get access to Anaconda Navigator and Anaconda Prompt.

3.3.6 Git

Git is a software for tracking changes in any set of files, usually used for coordinating work among programmers collaboratively developing source code during software development. Its goals include speed, data integrity, and support for distributed, non-linear workflows

3.4 Libraries Used

3.4.1 Numpy

NumPy is a library for the Python programming language, adding support for large, multi-dimensional arrays and matrices, along with a large collection of high-level mathematical functions to operate on these arrays. NumPy is open-source software and has many contributors.

3.4.2 Pandas

Pandas is a software library written for the Python programming language for data manipulation and analysis. In particular, it offers data structures and operations for manipulating numerical tables and time series. It is free software released under the three-clause BSD license. The name is derived from the term "panel data", an econometrics term for data sets that include observations over multiple time periods for the same individuals.

3.4.3 Scikit-Learn

Scikit-learn 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, k-means and DBSCAN, and is designed to interoperate with the Python numerical and scientific libraries NumPy and SciPy.

3.4.4 NLTK (Natural Language Toolkit)

The Natural Language Toolkit, or more commonly NLTK, is a suite of libraries and programs for symbolic and statistical natural language processing for English written in the Python programming language.

3.4.5 Pickle

Pickle module is used for serializing and de-serializing a Python object structure. Any object in Python can be pickled so that it can be saved on disk. What pickle does is that it "serializes" the object first before writing it to file. Pickling is a way to convert a python object into a character stream. The idea is that this character stream contains all the information necessary to reconstruct the object in another python script.

3.4.6 Requests

Python requests module has several built-in methods to make Http requests to specified URI using GET, POST, PUT, PATCH or HEAD requests. A Http request is meant to either retrieve data from a specified URI or to push data to a server. It works as a request-response protocol between a client and a server.

3.5 Dataset

For the project, TMDB dataset is used . The TMDB dataset is a big CSV file that has features like title , movie id , overview , keywords , genres ,popularity etc. for around 5000 movies.

S.No.	Feature	Description
1	title	Title of the movie
2	genres	Genre of the movie
3	keywords	Informative words that define movie
4	overview	Overview of the movie
5	cast	Details of the cast in the movie
6	crew	Details of the production team and direction team of the movie

Table 1: Brief description of the TMDB dataset

3.6 What is API?

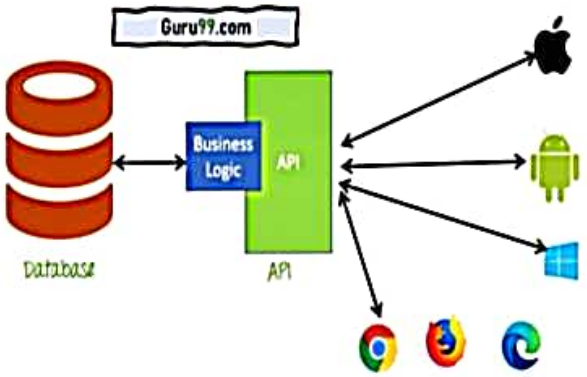

Fig 3.2 how API works

API is an acronym for Application Programming Interface that software uses to access data, server software or other applications and have been around for quite some time. APIs communicate through a set of rules that define how computers, applications or machines can talk to each other. The API acts as a middleman between any two machines that want to connect with each other for a specified task

3.6.1 TMDB API

The API service is for those of you interested in using our movie, TV show or actor images and/or data in your application. Our API is a system we provide for you and your team to programmatically fetch and use our data and/or images.

CHAPTER 4: FINAL ANALYSIS & DESIGN

4.1 Result

After training the model, it predicts the recommended movies for the user id given as an input. The figure below shows the known choices and recommended movies for the user

User Searched for movie TITANIC

fig 4.1 , user searching for movies of his interest

User will get 12 similar movies as a result of similar interest (TITANIC)

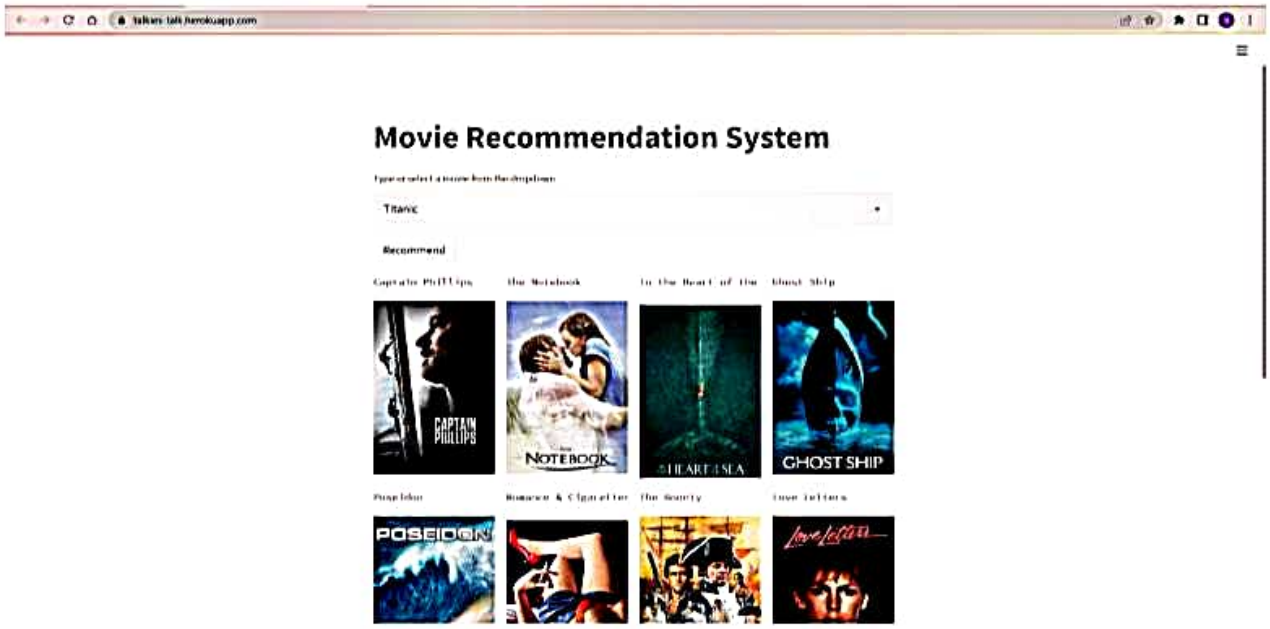

fig 4.2 , movie recommendations given to user by the system

4.2 Applications

The following are the application of Recommendation System:

➢ Recommender systems have become increasingly popular in recent years, and are utilized in a variety of areas including movies, music, news, books, research The following are the application of Recommendation System:

➢ Recommender systems have become increasingly popular in recent years, and are utilized in a variety of areas including movies, music, news, books, research articles, search queries, social tags, and products in general.

➢ Mostly used in the digital domain, the majority of today's E-Commerce sites like eBay, Amazon, Alibaba etc., make use of their proprietary recommendation algorithms in order to better serve the customers with the products they are bound to like.

Popular recommendation system examples:

1.. YouTube

2. Netflix

3. Spotify Music App

4. Amazon

4.3 Advantages

Below are some of the various potential benefits of recommendation systems in business, and the companies that use them:

1. "Improving with use" (retention): One of the core potential benefits of recommendation systems is their ability to continuously calibrate to the preferences of the user. This makes products that become more and more "sticky" in their customer retention as time goes on:

2. Improving cart value: A company with an inventory of thousands and thousands of items would be hard pressed to hard-code product suggestions for all of its products, and it's obvious that such static suggestions would quickly be out-of-date or irrelevant for many customers. By using various means of "filtering", eCommerce giants can find opportune times to suggest (on their site, via email, or through other means) new products that you're likely to buy.

3. Improved engagement and delight: Sometimes seeing an ROI doesn't involve explicitly asking for payment. Many companies use these systems to simply encourage engagement and activity on their product or platform.

That being said, there are some sectors (most notably digital media, eCommerce) where such systems seem to be borderline inevitable. Let's consider examples to better understand the concept of a recommendation engine.

1. According to McKinsey & Company, 35% of Amazon. Com's revenue is generated by its recommendation engine.
2. According to YouTube after implementation of the RS for more than a year, it has been successful in terms of their stated goals, with recommendations accounting for around 60 percent of video clicks from the homepage.

Moreover, they can have positive effects on the user experience as well, which translates into metrics that are harder to measure but are nonetheless of much importance to online businesses, such as customer satisfaction and retention.

CHAPTER 5: CONCLUSION & FUTURE SCOPE

5.1 Conclusion

1. Recommendation engines basically are data filtering tools that make use of algorithms and data to recommend the most relevant items to a particular user.

2. The recommendation system made in this project is able to recommend movies for any movie is given. To recommend movies of similar interest ,we use cosine similarity to find the most similar movie of similar interest.

3. Application of recommendation system: Almost nowadays all web service based business uses recommendation system. Examples of popular recommendation systems are that of Netflix, Amazon, YouTube, Gaana Music App, Flipkart, eBay etc.

4. Advantages of using recommendation system: Recommendation systems can significantly boost revenues, CTRs, conversions, and other important metrics. Moreover, they can have positive effects on the user experience as well, which translates into metrics that are harder to measure but are nonetheless of much importance to online businesses, such as customer satisfaction and retention

5.2 Future Scope

The future scope of this project, the Recommendation System is very wide. There are many additional features, which are planned to be incorporated during the future enhancements of this project. Although all the main objectives have been achieved still there is room for enhancement.

- This system can be easily upgraded in the future. And also include many more features for the existing system.

- The recommendation system can be generalized or changed so that it can give recommendations for other things also like music, books, video recommendation provided appropriate dataset is a available to create and train our model.

- With media files associated with every movie , we can also build a OTT platform

References

1. https://developers.google.com/machine-learning/crash-course

2. https://themoviedb.org/

3. https://www.kaggle.com/datasets/tmdb/tmdb-movie-metadata/

4. https://docs.streamlit.io/