

MULTIREC: THE THREE-IN-ONE RECOMMENDER*

BY

AISHWARYA JENA, ARYAN NAMBIAR, ISHAN BHISE, SIDDHARTH KUMAR, SOHAM
DEO, YASHI SHRIVASTAVA, PROF. VIJAY GAIKE*

Ajeenkya DY Patil University, Charoli Bk. via Lohegaon, District Pune - 412105, Maharashtra

Email: aryan.nambiar@adypu.edu.in, siddharth.kumar@adypu.edu.in

ABSTRACT

We have been assigned to do a small project based on our ideas and knowledge. Our team decided to build a multimedia recommendation system. We aimed to build a movie recommendation system, but we added two more media recommendation systems, they are music and game. Young people nowadays have access to video games, movies, and music. We knew platforms that already have recommendation systems and can play the media. However, our website recommends which type of movie the user can watch, which music they can listen to or the game they can play. Designing the website needs the design concept for the website, the understanding of the front-end which mainly includes the design of the webpage. Understanding the back end is a vital part as the entire recommendation system would work with the backend code. After researching thoroughly, we found that we can build the front-end using CSS and HTML-5, and the back end can be coded using python. After finding that the same layout could be built for more than one type of media, our team decided to include music and games. The database, which contains information about the movies, music, and games, was discovered on the internet. There are certain limitations to the project. We cannot make a recommendation system for every type of media. Artificial intelligence will not be used to its full extent in our recommendation system. The recommendation system will be using the statistics and the information provided in the database to recommend the user of the media they search recommendations for. This recommendation system will help us in understanding how the database is linked to the program that has been created; it will also help in understanding the basics of coding. This building of knowledge will help us in understanding how this recommendation system or another type of programme can be improved in the future.

KEYWORDS

Recommendation System, Statistics, Database, Python.

* Received 22 September 2021, Accepted 09 October 2021, Published 24 October 2021

* Corresponding Author

INTRODUCTION

Our team had been given the opportunity to design a project that could be useful to our society in some way. We as a team brainstormed a few ideas as to what we could build or do that would help society.

From our team, the first idea, from our teammate Aryan Nambiar, was to build a movie recommender, the same technology that many media platforms use, such as Spotify, Netflix, Amazon Prime Video, Pandora, etc. These organisations improve their chances of the user to use their product which is the app or their website that uses a code that is the foundation to the recommender which is linked to the frontends of the apps as well as the website. We thought of building just the recommender, which can help the user to search for a certain genre or a certain movie and the details of the movie will be displayed to the user along with the movies that match the genre, the main character of the movie, etc. The other idea that had risen in our group was from Aishwarya Jena. She had thought of an idea where we would build a system that would help people when needing medical assistance. Her idea was to make an app that would give details of the nearby ambulances available, the number of beds available in each hospital, medical stores that were open at the hour when the user is checking, etc. This idea seems to be equally excellent, but the issue of this idea was the feasibility. The scale of collaboration with hospitals, ambulances, medical stores would need to be very vast. At this point as a student, the capability of building such an app is possible, but many factors would not remain in sync and would make it impossible to finish. Hence, we all continued with the movie recommender.

The movie recommender will be a stepping stone for all of the team members in understanding how innovation, planning, implementation, and final execution plays a great role in making a good project successful.

AIM

Our aim of the ED project is to build a website that serves the purpose of giving options of entertainment media based on their liking. This would be done by making recommender for multiple entertainment media.

POTENTIAL PRODUCT

We decided to build the website with only one recommender, the movie recommender. The backend consists of the linked database of the movies as well as the code for the recommendation system. The frontend will consist of HTML5 and CSS which will manage and

be designed as per our expectations. This potential product we thought of will help users in getting the possible movies that they would prefer to watch, or in the same genre, actor, etc. This product will be free to all and can be accessed by anyone from the website link that we will share. To expect the entire recommender to work without any issue, we will need a domain that will have all strong connections and does not cause any sorts of issues like crashing the website or the user is unable to access the recommender.

FINALISED PRODUCT

The product we finalised is the same as the movie recommender we had planned on building. However, we planned to extend or diversify the media recommender types. We finalised to build a recommender system that will help the users to find their possible liking for the media such as movies, music and games. These three media recommenders will be built on the website, we will make new web pages for the website so that each of the media recommenders has its own web page. This will help the user to distinguish between each of the recommenders, that is the music, the game and the movie recommender.

BACKGROUND INFORMATION

Recommendation Systems: It is a strong software that works based on user's preferences. This software recommends products (material, media, etc.) to the user that might be of the user's liking. They are used for increasing the chances for sale or increasing the usage of the recommender. They make the process of searching easier for the user to check the content the user is interested in. There are two methods by which the recommendation system checks the information. They are characteristic information and user-item interaction. Characteristic information is for the keywords that have been entered, while user-item interaction is based on the number of views, ratings, number of purchases, etc. Then there are also different types of algorithms used in the recommender. Among them are collaborative, content-based, and hybrid systems. The content-based system uses the characteristic information, meaning it hypothesizes what the user likes, and displays it to the user. Collaborative filtering uses the data of the interaction of the user with the recommender. The collaborative interaction analyses which kind of music the user tends to listen to; based on that the recommender starts to give suggestions. In collaborative interaction, there are two ways the recommender analyses the data, first is memory and the second is model. The memory-based recommendation suggests another type of media that is similar to the media that the user likes. On the other hand, the model-based suggestions can predict the user's top 5 items that they might like the most and based on that they recommend new items. The hybrid system uses both content-based and collaborative filtering to give possibly the results that the user might find interesting.

Python: It is a programming language that can be used for the development of the web, data science, creating new software prototypes, etc. It can be run on multiple operating systems like Windows, Linux, Mac. It is a programming language that can be used for multiple purposes. They are used by data analysts, engineers, scientists, children, etc. It is simpler than JavaScript and C#.

HTML-5: It is also known as hypertext markup language revision 5, it is used to build the structure and presentation of the world wide web contents. This revision is compatible with HTML and XHTML- style syntax and a few new features in its markup, new API (application programming interface), XHTML and error handling. It increases the functionality of the markup language (HTML), the style (CSS) and better interactivity as well.

CSS: It is also known as cascading style sheets. This helps in describing how the elements present on the displayed screen, newspaper, or other media should be. It helps in controlling multiple web pages at one time. It helps in designing and placing the elements to produce a more aesthetic looking webpage, which can attract the attention of the users.

PROPOSED DESIGNS AND IDEAS

REQUIREMENT SPECIFICATIONS

Hardware requirements

- Personal computer
- Internet to keep the website working
- Storage space to compile and store the programme data
- Minimum RAM space 8 Gb for reduced time for compiling codes and programming
- Processor: Intel i3 or better for faster processing
- Minimum internet speed: 10 Mbps

Software requirements

- Web browser (e.g.: google chrome)
- Python
- Jupyter notebook
- HTML-5
- CSS
- Draw.io

System Interaction

- The website and recommendation system should be able to run properly on any browser that is available, such as Google Chrome, Firefox, or Internet Explorer.
- The website should be able to function with the software that was used to programme and design it. With the help of the ISP, under the condition that the website is taken down for maintenance, the internet connection should not break down.
- Webspaces- to store the database in which all the details of all the different media are stored.
- Smooth working of hyperlinks
- Consistency in the layout of the website
- Best possible viewing option- Google Chrome
- Images/ GIFs on the website should load properly
- The recommender should be working without any issues faced

Input requirements

- Name of all the different recommending systems on the top of the website
- Information of each recommending system on each of the webpage allocated
- Images/ Gifs
- Database of the recommending system
- Front end design using HTML-5 and CSS
- Linking of the backend (recommending system) and the front end (website layout)

Output requirements

- Sections of the website include:
 - Multirec
 - Bingehub
 - Musichub
 - Gamehub
- Consistent page layout with theme-based colour and images
- Images and text should be displayed in various web browsers
- Image (JPEG), video (GIF)
- Analysing and giving the results matching the data put into the search section by the user

Processing

- Clicking on the headings would lead the users to another page that would be having different details and would be based on the recommendation system that matches the heading.

- The user may get to the top of the screen by clicking the button in the bottom right corner.
- The recommender will analyse the details put in by the user for the media that they are searching for, based on that the recommender would display the results.

Security

- Images will not be watermarked as they are stock images and gifs and can be used without any legal notices
- Will not need any other means of security as the website free for all

Performance requirements

To measure the effectiveness of the website:

- Provide an easily navigable system to display the details of the recommenders and the website.
- Create a website with a consistent style based on the colours chosen.
- Create a website with the function of the users to go and search all different types of the media recommenders
- All tabs, hyperlinks, web page layout and input boxes will be tested on different web browsers and platforms.
- All buttons should send the user to the respective webpages
- The go-to top button on the bottom right of every webpage should function properly and take the user to the top of the page
- The recommenders should be able to display to the user all the various types of movies, music or games based on the type they have chosen.

JUSTIFICATION OF CHOSEN PRODUCT

The Final solution discussed together as the team was to build a website. It is a very efficient method for publicity of the type of recommenders that our team has built. The website will stay online all the time. This makes it more efficient for our team as well as the users of the website.

After having a proper discussion of the benefits and drawbacks of using a website, this option seemed the most preferable and appropriate. The benefit of it over an app would be that the users would not need to download the recommenders, it is a hassle as it fills up a certain amount of memory within the device they are using. They are better than offline recommenders because we as programmers can update the list of movies anytime we wish to. Making an app

would be a bigger issue for us as we would have to integrate many languages just to create the front end of the app.

Therefore, the website is the best possible given solution based on the team's suggestions. The website can be viewed by everyone, the website does not need much effort as an app or offline recommender. Editing the website is less time consuming as well. Very few bugs can be found in website code. The design can be made unique compared to many other websites or apps. This website will be an efficient way of recommending the movie, music or game based on user's searches. Compared to rest solutions, this is also one of the cost and time-efficient as well.

WEBPAGES

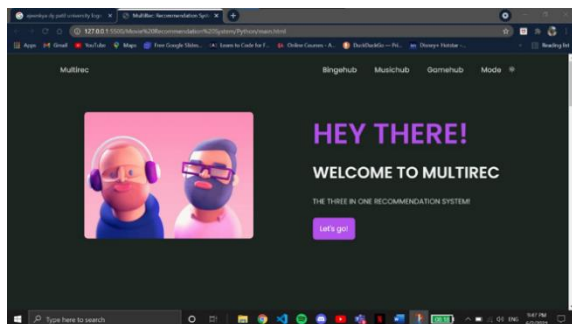
Multirec - In this page of the website there will be details of the recommenders that we have built, then there will be three different links for each of these recommenders. Then there is a widget that will allow the user to go to the top of the page.

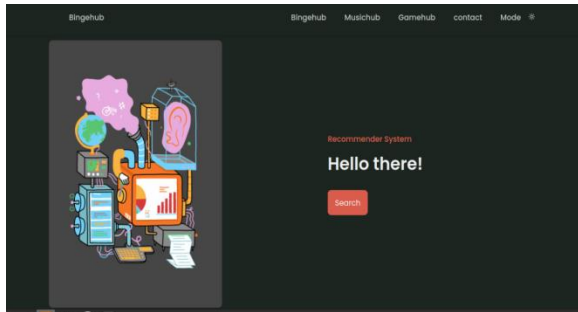
Bingehub - This page of the website consists of the details of what the bingehub as a recommender does. The page will have the recommender running on it when the user tries to use the recommender. The recommender will operate using the backend code as well as the database of movies we've watched.

Musichub - This page of the website consists of the details of what Musichub is and how it works. This page will have the recommender for music enabled with the help of the spotify database that we are using, this database updates as time passes and so the genre and coding will suggest new songs based on the user's liking.

Gamehub - This page of the website has the recommender system that suggests the user games as per what the user searches. This recommender has a more complex system than the Bingehub and the Musichub. This recommender will be using the collaborative as well as content-based filtering, basically the hybrid filtering system.

SNAPSHOTS





Please rate each genre based on your preference:

What is your rating to **Action** genre?

Example: Die Hard

hate this genre!
 Tolerable
 Average
 Satisfactory
 Very Good
 Blockbuster

What is your rating to **Adventure** genre?

Example: Mission: Impossible

hate this genre!
 Tolerable
 Average

Here are the top-20 movies that you'll most probably like:

Recommendation List
I Will Fight No More Forever (1975)
Man Who Shot Liberty Bells, The (1962)
Australia (2008)
3:10 To Yuma (2007)
Legends Of The Fall (1994)
Alamo, The (1960)
Redemption: The Stan Tookie Williams Story (2004)
Shamanshooh (1963)
Blood Diamond (2006)
Lord Of War (2005)
Assassination Of Jesse James By The Coward Robert Ford, The (2007)
Hang 'Em High (1968)
Joint Security Area (Gongdong Gyeonggi Guyeok Jsa) (2000)
They Died With Their Boots On (1941)
800 Bullets (800 Balas) (2002)
Night Of The Generals, The (1967)
Proposition, The (2005)
Night And Fog (Nuit Et Brouillard) (1955)
Jesse James (1939)
Redacted (2007)

CONCLUSION

Recommender systems are a powerful new technology that allows businesses to extract additional value from their user datasets. These systems assist consumers in locating products they wish to purchase from a business. Users gain from recommender systems because they can locate products they enjoy. They, on the other hand, benefit the company by increasing sales. Recommender systems are quickly becoming a critical element in online E-commerce. The large number of user data in existing corporate databases puts recommender systems under strain, and the rising volume of user data available on the Web will put them even more

under strain. New technologies are required to enhance the scalability of recommender systems substantially.

REFERENCES

- 1) "Learn Python Programming." Programiz, www.programiz.com/python-programming.
- 2) Techopedia. "What Is HTML5? - Definition from Techopedia." Techopedia.com, Techopedia, 11 Jan. 2017,
- 3) www.techopedia.com/definition/1891/html5. Tryolabs. "Introduction to Recommender Systems." Tryolabs Blog, Tryolabs, 9 May 2018, tryolabs.com/blog/introduction-to-recommender-systems/.
- 4) Underwood, Corinna. "Use Cases of Recommendation Systems in Business - Current Applications and Methods." Emerj, Emerj, 4 Mar. 2020,