

PROJECT PROPOSAL

PROJECT NAME: JUST ANOTHER MOVIE

TEAM MEMBERS

Hayden Richard, Robert Jones, Emmett Storey, and Kaitlyn Hardin

ABSTRACT

With so many things in the world, it is difficult for people to choose them. For example, a person decides to watch a new movie. Then, he or she goes to find a movie, but does not know what to search. The person tries to search for some movies but just is not successful. Now overcome with frustration, the person succumbs and chooses a movie he or she has previously watched. Unfortunately, many people experience this scenario daily. People need something to help.

Just Another Movie is an application that combats this problem. Our application allows the user to be prompted and to enter information about what type of movies they like. Users search by their favorite genres and actors. The application takes that information and employs different classification and recommendation techniques. Then, it searches through our database and returns a list of movies that the user should enjoy. The system includes functionalities for the user such as, removing movies from the list if they have either seen it before or do not like the movie.

DESCRIPTION

There are a multitude of items that we can use and consume. These items can range from media to art and to commerce. As a result, many people create various forms of each item that a person can use. Thousands of similar items are available to meet the growing demands of consumers. However, it is difficult for consumers to decide which item best fits their needs. The difficulty consumers are facing produces a new problem. Therefore, new software is being developed to recommend items to consumers based on certain qualities that are favorable to the consumer. For our purposes, we have decided to focus on movies. There is a wide magnitude of movies, and unfortunately, this presents a problem. Consumers do not know what movie to choose for their entertainment.

Just Another Movie is a movie recommendation system that produces a list of recommended movies for a user. The system is a desktop application built in the programming language, C#. Our team chooses to develop the system on the Windows OS and to use Visual Studio 2019 as the IDE to create the backend and frontend components of the system.

The primary objective of the movie recommendation system is to curate a list of movies that the user will likely enjoy, watch, and or rate favorably. This project combines machine learning algorithms, classification techniques, and the content-based recommendation approach to create the system.

The application runs when the user starts the Just Another Movie program. When the user gains access the system, the system displays a list of current movies, and the top movies for each available genre. The user can then choose from the available lists provided by the application, or the user can input a movie. If the user chooses the latter, then the system uses the information provided by the user to output a list. The information relates to a movie's main attributes. These attributes include any actors in the movie, the release year of the movie, the director of the movie, and the genre of the movie.

Next, the system selects all of the movies from the datasets stored in the system. The datasets in the system will be from the IMDb database. The IMDb database is an online database that contains free, current, and available datasets for any customer's non-commercial usage. The datasets include movie information such as titles, languages, release year, genres, directors, writers, ratings, characters, and actors. These datasets are in the form of a .csv file. The system imports these files and parses the datasets based on the queries made by the user. From the queries, the machine learning algorithm chooses the movies that best match the interests of the user based on similarity and ratings.

We plan to use machine learning by building a classifier to properly recommend which movies each user will like. This algorithm will allow the system to efficiently cluster movies based off the user's information, the movie's attributes, and any filters the user applies. To prepare the system to recommend movies, we plan to use the ML.NET Model Builder in Visual Studio. With the model, we plan to determine the best machine learning algorithms that best fits this application. This tool in Visual Studio allows the team to create a trained model that will make predictions. Using our desired machine learning algorithm, the system decides the movie cluster and lists these movies for the user.

Once the system outputs the list of recommended movies, the user can review each movie in the list. If the user dislikes a movie, then the user can remove the movie from the list. If the user thinks of another movie to add to the list, then the user can manually enter the movie's title and attributes to the list. Users are allowed to create lists and can edit the list after they are created. Additionally, the user can select each movie within the list and create a new list of movies based on the selected movie. Users can continue to select movies and receive new lists until they find a suitable movie.

If given enough time, our team will implement additional features to improve the Just Another Movie application. We will enhance existing features or create new features. For example, our system can allow users to rate a set number of movies to develop a specialized list of movies for them. Then, this list can consistently update based on future ratings made by the user.

FEATURE LIST

Features that will be implemented

System

System creates a curated list of recommended movies for the user

System bases movie recommendations on movie attributes and content

System provides user with general lists of best movies in each genre

User

User can add list of movies

User can delete list of movies

User can add a movie to a list

User can delete a movie from a list

User can dislike or like movies

User can lookup movies based on their attributes

Users can search for movies based on attributes and filtering

Users can create their own lists

Features that will be completed if given time

System

System will base movie recommendations on similar users' information

System stores user information to recommend movies for the user

User

Users can save movie lists in the system

Features that we want but cannot be implemented

System

System allows similar users (friends) to refer movies or suggest movies
System shows users where to stream and find movies

User

User can share lists
Users can send friend requests to other users
Users can watch movies in the application
Users can share screens with friends to watch movies in a group

TECHNOLOGY

- Operating System:** Windows OS
- IDE:** Visual Studio 2019
- Data:** IMDb database
- Programming Languages:** C#
- Communication:** Discord
- Platform:** Desktop application
- 7zip for file compression
- GitHub (for project organization and maintenance)

SERVER INFORMATION

N/A

DATA SOURCES

IMDb database with datasets for various movie attributes
Link: <https://www.IMDb.com/interfaces/>

TEAM MEMBERS' BACKGROUND

Hayden

I have experience programming in HTML, CSS, JavaScript, Java, C++, C#, and SQL. My part is to help develop the backend of the system.

Emmett

I have experience in programming in C++, C, C#, and Java. I also have experience in photoshopping. My part is to work on the backend of the program.

Robert

My skills consist of C++, Java, Linux, and Cobol. I assist with the frontend part of the system and system design.

Kaitlyn

My background experience includes HTML, CSS, JavaScript, C#, C, Java, C++, Python, and SQL. I am working on the frontend part of the system and system design.

DEPENDENCIES LIMITATIONS AND RISKS

Dependencies

We are using one data source: IMDb database.

We are dependent on the tools provided by Visual Studio 2019.

We are developing on Windows OS.

Limitations

Hayden

My laptop is a mac and will not allow me to work on the project in class.

I will only be able to work on the project at home from my desktop.

Emmett

I can work in the afternoon on most weekdays except Tuesday, and in the future Friday.

I will not be free on weekdays in the future until around late evening.

Robert

I am only free Tuesday, Thursday and weekends.

Kaitlyn

I work throughout the weekday and on weekends.

I am in another class with a group project.

I have never built a recommendation system before.

Risks

There is the risk of the new delta variant causing another shutdown.

Greater risk for Covid exposure due the lax measures in the county where Emmett lives.

Robert's greatest risk would be Covid exposure due to kids and working.

TIMELINE

Weekly

We will meet every Saturday. (may meet up to 2x a week)

- discuss updates, issues, concerns, and progress

Monthly

September - Project Design

- we all discuss the design aspects of the system

October – Development

- load data
 - successfully import and parse data
- backend development
 - Hayden and Emmett will schedule meeting times to start and complete the backend part of the application.

November - Frontend

Kaitlyn and Robert will schedule time to start and complete the backend part of the application.

November/December - Testing

- make sure the application works together with backend and frontend
- test simulating different users
- test for failure
- test for success
- test all features of the application