Collins-Kiptoo / **Movie-Recommendation-System** Public

0 stars 2 forks

Star Watch

Code Issues Pull requests Actions Projects Wiki Security Insig

main

Gorreti Update README.md ... 8 minutes ago 43

View code

README.md

Movie-Recommendation-System

Authors

- Collins Kiptoo
- Amos Pride
- Faith Makokha
- Felix Nyagah
- Gorreti Muthoni
- Jamleck Mathenge

Overview

Anime has become an increasingly popular genre of entertainment in recent years, brought about by nerd culture gaining popularity in the mainstream pop culture areas. Some people like it just for the subtitles The explosive growth in the amount of available digital information and the number of visitors to the Internet have created a big challenge where consumers have a wide variety of choices but yet very few choices at their disposal and producers have a difficult time figuring out their potential market. Information retrieval systems, such as Google, DevilFinder, and Altavista have partially solved this problem but the personalization of this data to make relevant recommendations to consumers and producers were absent. Recommender systems are information filtering systems that deal with the problem of information overload by filtering vital information fragments out of large amounts of dynamically generated information according to the user's preferences, interests, or observed behavior about an item. The recommender system has the ability to predict whether a particular user would prefer an item or not based on the user's profile. Recommender systems are beneficial to both service providers and users. They reduce the transaction costs of finding and selecting items in an online shopping environment. Recommendation systems have also proved to improve the decision-making process and quality. In an e-commerce setting, recommender systems enhance revenues, for the fact that they are effective means of selling more products. In scientific libraries, recommender systems support users by allowing them to move beyond catalog searches. Therefore, the need to use efficient and accurate recommendation techniques within a system that will provide relevant and dependable recommendations for users cannot be over-emphasized.

Problem Statement

In the past, people used to shop in a physical store, in which the items available were limited. For instance, the number of movies/music CDs/vinyl that can be placed in a local shop depends on the size of that shop. By contrast, nowadays, the Internet allows people to access abundant resources online. Netflix, for example, has an enormous collection of movies. Although the amount of available information increased, a new problem arose as people had a hard time selecting the items they actually wanted to see and the production companies had a difficult time locating their target audience. Due to the prevalence of the Internet, we need recommender systems in modern society that will assist people in finding products that are in their tastes and preference from the vast options. Moreover, recommendation systems can be deployed on commercial websites to help producers market their products to the right consumers.

Proposed Solution

The most appropriate solution to deal with our problem is to come up with a system that would recommend movies to our consumers based on their preferences and tastes in order to maximize consumer utility and increase profits for producer companies.

Justification of the Study

The objective of recommender systems is to provide recommendations based on recorded information on the users' preferences. If recommendations are engineered and deployed effectively they can have a direct impact on the world. Users would be able to access movies in line with their interests thereby maximizing their utility whereas movie producers will be able to know their target market thereby maximizing profits.

Specific Objectives

- To build a model that will recommend the most likely anime a user may watch.
- To find out the most watched anime genre
- To determine the highest rated genre
- To determine the anime source with the most members

Research Questions

- Which is the model that provides recommendations?
- What is the most-watched anime genre?
- What is the highest-rated anime genre?
- Which anime sources have the most members?

Success Criteria

Create a model that can recommend movies to users based on their preferences

DATA UNDERSTANDING

Overview

In this project, a dataset containing a list of anime movies was collected from kaggle.
Exploring Data

The dataset has the following features.

● Data contains 17002 entries.

● There are 15 columns.

The following are the columns in the dataset.

- Anime_id :anime Id (as per myanimelist.net)
- Title : Name of anime
- Genre :Main genres in the movie
- Synopsis :Brief Description
- Type
- Producer: Production company of the anime
- Studio: The studio that produced the anime
- Rating: Rating of anime as per myanimelist. net/(on a scale of 1-10)
- ScoredBy: Total no user scored given anime
- Popularity: Rank of anime based on popularity
- Members: No of members added given anime on their list
- Episodes: No. of episodes
- Source
- Aired
- Link: Link to the anime on myanimelist

DATA PREPARATION

This was done to ensure the Validity, Accuracy, Completeness, Consistency, and Uniformity of the Data. These are the steps followed in preparing the data:

> - TASK1: The first task was to check if the column names were uniform and readable.
> - TASK2: The second task was to check for duplicated rows.
> - TASK3: The next task was to check if there were missing values.
> - TASK4:The next task was to decide how to deal with the missing values. (Either drop if they are unnecessary or replace the missing values with the best fit.)
> - TASK5: The last task was to check if the columns had the correct data types.

DATA ANALYSIS

To better comprehend the data patterns and even develop some hypotheses, let's first begin by studying our dataset. In order to find any significant trends, we will first examine the distribution of the various variables.

Univariate analysis

This involves the plotting of data into histograms and bar charts in order to have a better understanding of the data.

Bivariate analysis

This involves the use of two columns on our dataset in order to plot charts that will show the relationship between various features in our dataset.

Multi-variate analysis

This involves using more than two columns of data to create charts that will give us a superior understanding of how the various features affect and relate to each other.

MODELLING

Pre-processing

We create a function that will be able to clean the anime titles using REGEX. Cleaning the title name is crucial because it can often lead to errors whereby the title name is not in a machine-readable format Then we use the term frequency-inverse document frequency to create a search engine so that it is easy to find an anime title and its anime id. The tfdif Vectorizer will enable the computer to find the title that is most similar to the title we enter. We then create a function that will use cosine similarity to compute the similarity between a term that we will enter in our search box and the anime titles in our dataset. We then build an interactive search box that will enable one to type in the name of an anime and have the results.

Modeling Techniques

We used various modeling techniques to create a recommendation system which are;

- Collaborative filtering recommendation system
- User-based recommendation system

- Hybrid recommendation system
- Collaborative filtering Recommendation system

Under collaborative filtering we made use of the KNN machine learning algorithm and SVD. KNN is a machine learning algorithm to find clusters of similar users based on common book ratings and makes predictions using the average rating of top-k nearest neighbors. Once the model was correctly engineered we made a function that takes in a movie and gives recommendations of other movies based on the movie the user watched.

User-based Recommendation system

A user-based recommendation systems predicts the user preferences or ratings that users would give to items. We made use of pair_wise distances that computes the distances between corresponding elements of two arrays. The next step is to make predictions based on these similarities which are done by a function we created that takes in the movie ratings, the user similarity scores and type of user and gives recommendations.

Hybrid Recommendation System

We create a function that takes in a user, ratings based on collaborative filtering and predictions based on user-based recommendations to make recommendations on what movie a user should view based on their tastes and preferences.

Model Validation.

We employ Root mean squared error to validate our hybrid model. The model has an RMSE which is 1.707

Deployment

We made use of Streamlit which is an open source python-based framework for developing and deploying interactive data science dashboards and machine learning models. Link

Releases

No releases published
Create a new release

Packages

No packages published
Publish your first package

Contributors 5

Languages

- **Jupyter Notebook** 100.0%