

# Movie Recommendation Application

Ryan S. Shaw

Northwest Missouri State University, Maryville MO 64468, USA  
S546850@nwmissouri.edu

**Abstract. Keywords:** data analytics · movies · Python · web scraping

## 1 Introduction

I plan to work in the domain of web scraping. Primarily because it is the most interesting to me but also because I feel I did not get to fully explore web scraping in the web scraping course due to time constraints.

I would like to scrape my data from the online website IMDB primarily. I may also source data from Rotten Tomatoes.

I intend to solve the problem of trying to decide which movies to watch. I feel like this is something everyone struggles with at some point. Scrolling through endless movies on Netflix or some other streaming service and not knowing what to choose.

Steps taken would be as follows:

1. Scrape the data from the previously mentioned websites.
2. Clean/organize the data into a usable format.
3. Create a Python script that requests user information (name, birthdate, gender, favorite genre).
4. Create a model that can recommend some movies that the user may like based on their information and the movie rating.

Python will be a huge component for me in this project. I will need it to for just about every aspect of this project.

## 2 Data Sourcing

I used Python and web scraping (specifically the BeautifulSoup Module) to gather my data from the IMDB website. The data is in HTML format being written initially for a webpage and collected using Python. The attributes that will be gathered are: Title, Release Year, Rating, Runtime, Genre, Metascore, Movie Description and Votes. IMDB organizes movies by genre, so moving forward I would like to implement a method for users to search movie recommendations by genre.



```
movie_data.append(data)

with open(r'C:\Users\User\Desktop\Capstone\Module 3\ module3.csv', 'w', newline='') as csv_file:
    for y in movie_data:
        writer = csv.DictWriter(csv_file, fieldnames=y.keys())
        writer.writerow(y)
```

**Fig. 3.** New code to write to CSV file.

I changed the Writer method to DictWriter since it is specifically made to handle writing dictionary objects to CSV which is what I am doing here. This keeps each movie record on a line instead of printing each attribute of every movie on a new line.

I also added the newline argument. This tells Python exactly how I want it to handle new lines. As you can see in Fig. 1, it was skipping a line after each record. It no longer does this thanks to the newline argument being specified.

These were the biggest changes made using Python. The rest I was able to easily accomplish using Microsoft Excel as you can see in Fig. 4. I added a title row which shows the names of each attribute, bolded. I also centered all the data and expanded the cells to see the data better. I also deleted the Year column completely, as it is not important for my project and IMDB includes this information in the movie titles, making this redundant. This left me with 8 attributes and had no missing values that needed cleaning.

Movie Title	Rating	Runtime	Genres	IMDB User Rating	Metascore Rating	Description	Number of Votes
1. Indiana Jones and the Dial of Destiny (2023)	PG-13	154 min	Action, Adventure	6.9	58 Metascore	Archaeologist Indiana Jones races against time	Votes: 71,063
2. Sound of Freedom (2023)	PG-13	131 min	Action, Biography, Drama	8.2	43 Metascore	The incredible true story of a former governm	Votes: 22,491
3. Spider-Man: Across the Spider-Verse (2023)	PG	140 min	Animation, Action, Adventure	8.9	86 Metascore	Miles Morales catapults across the Multiverse,	Votes: 182,518
4. Mission: Impossible - Dead Reckoning Part One (2023)	PG-13	163 min	Action, Adventure, Thriller	8.1	81 Metascore	Ethan Hunt and his IMF team must track down	Votes: 45,855
5. The Flash (2023)	PG-13	144 min	Action, Adventure, Fantasy	7.1	56 Metascore	Barry Allen uses his super speed to change the	Votes: 88,374
6. Nimona (2023)	PG	101 min	Animation, Action, Adventure	7.7	75 Metascore	When a knight in a futuristic medieval world is	Votes: 15,615
7. Extraction II (2023)	R	122 min	Action, Thriller	7.1	57 Metascore	After barely surviving his grievous wounds fro	Votes: 96,544
8. Guardians of the Galaxy Vol. 3	PG-13	150 min	Action, Adventure, Comedy	8.1	64 Metascore	Still reeling from the loss of Gamora, Peter Qu	Votes: 213,698

Fig. 4. New output CSV file after data cleaning.

This leaves me with the following attributes:

1. Movie Title (Dependent Variable): The title of the movies.
2. Rating: (Independent Variable): Movie rating (E.g. G, PG - 13, R, etc.).
3. Runtime: Length of movie.
4. Genres (Independent Variable): Movie genre according to IMDB.
5. IMDB User Rating (Independent Variable): Average rating (out of 10 possible) given by IMDB users.
6. Metascore Rating (Independent Variable): Critic rating (out of 100 possible) assigned by movie critics.
7. Description: Movie description.
8. Number of Votes: Number of votes by IMDB users for that movie.

## 4 Results and Analysis

## 5 Limitations

## 6 Conclusion

□

## References

1. Imdb: Ratings, reviews, and where to watch the best movies and tv, <https://www.imdb.com/>
2. Rotten tomatoes: Movies — tv shows, <https://www.rottentomatoes.com/>