



MOVIELENS: BUILDING A MODEL FOR PERSONALIZED MOVIE RECOMMENDATIONS BASED ON USER RATINGS

Presentation by Kemboi Elly Kiplimo



INTRODUCTION

This project focuses on developing a movie recommendation system for personalized content. Using the MovieLens dataset with 100K data points, the aim is to create a model that suggests the top 5 movies to a user based on their ratings of other films. Such systems are vital in today's era of tailored entertainment experiences.



PROBLEM STATEMENT

The goal of this project is to build a movie recommendation system from the GroupLens research lab at the University of Minnesota that can provide personalized movie recommendations to users based on their ratings of other movies. The system should be able to handle the cold start problem and provide accurate recommendations to users.

METHODOLOGY

COLLABORATIVE FILTERING

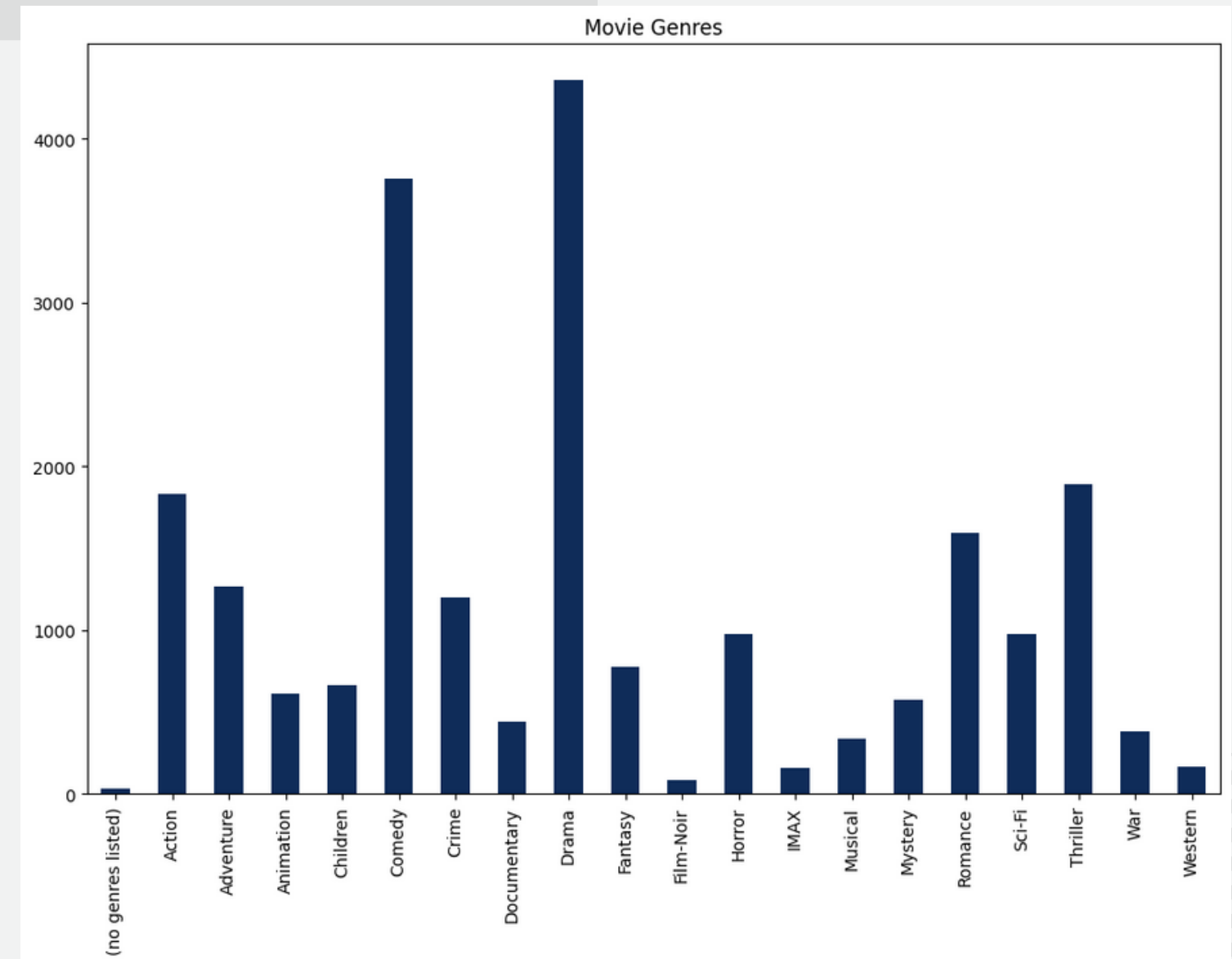
Its a recommendation method that suggests items to a user based on the preferences and behaviors of similar users.

CONTENT BASED FILTERING

Its a recommendation technique that suggests items to users based on the features and characteristics of the items themselves.

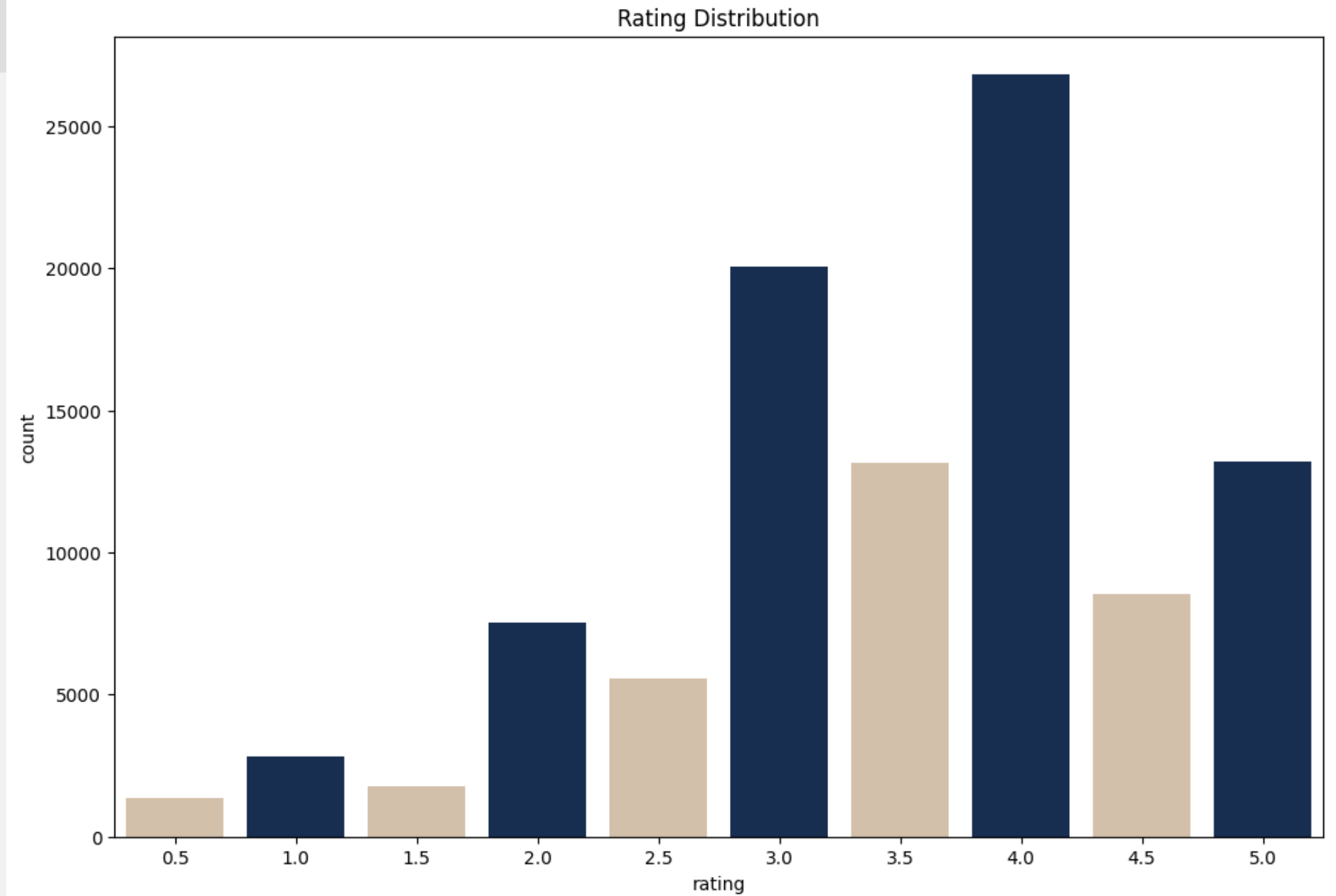
DATA ANALYSIS

From analysis, the drama genre is the most common genre while film noir is the least common type of genre

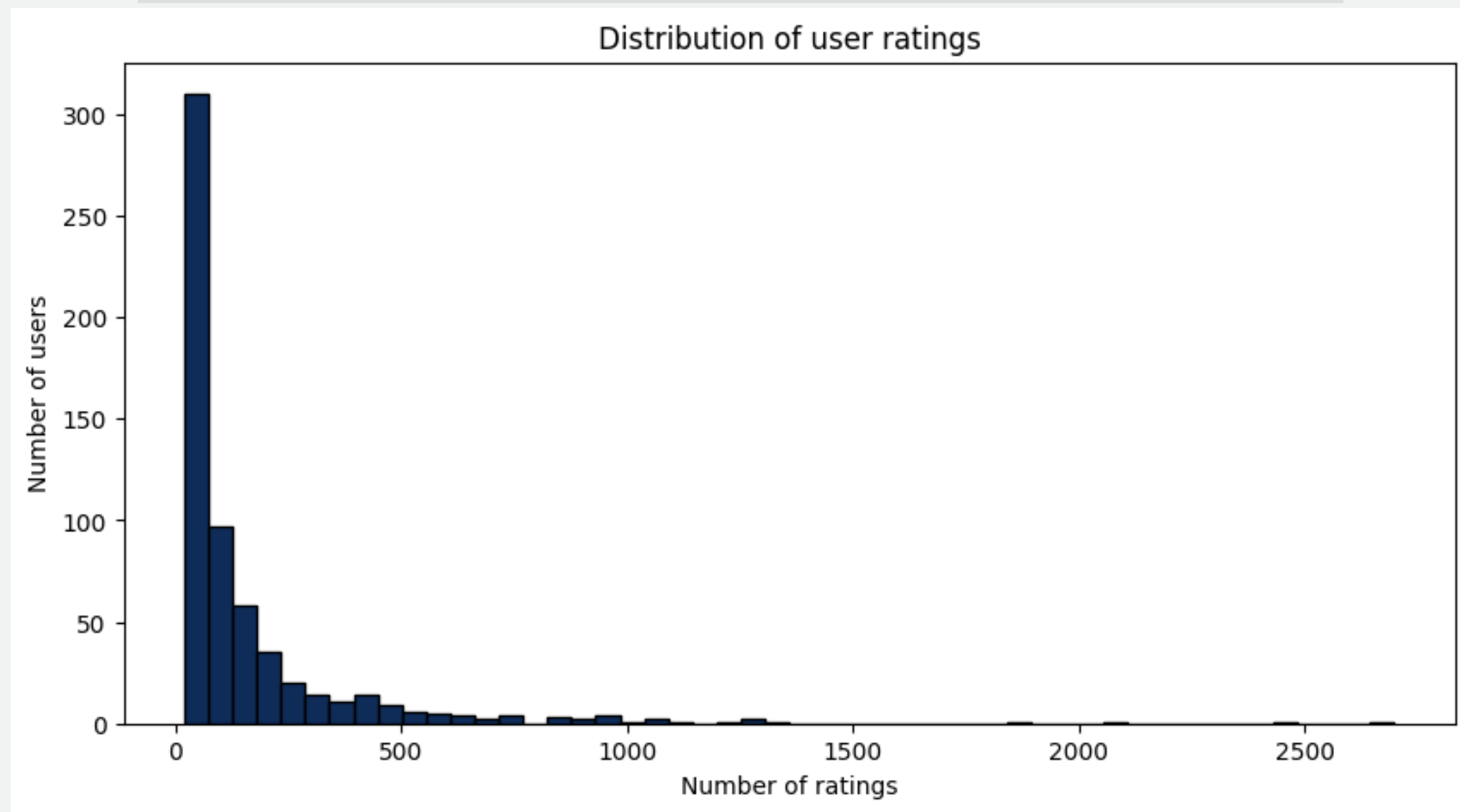


DATA ANALYSIS

From analysis, most movies are rated 4



DATA ANALYSIS



RESULT



7. Testing the Recommendation System

```
user_id = 2
title = "Flint (2017)"
n = 5

# Call the 'combined_collab_and_content_based_recs' function with the specified user ID, movie title, and the number of recommendations.
rec = combined_collab_and_content_based_recs(user_id, title, n)

#Convert the recommendations 'rec' to a list of recommended movie titles and store them in 'recommended_movies.'
recommended_movies = rec.tolist()

#Print a message indicating the top 5 recommended movies for user
print(f"Top {n} recommended movies for user {user_id} are {title}:")

#Iterate through the 'recommended_movies' list, numbering each recommendation, and print them in the specified format with their titles.
for i, movie_title in enumerate(recommended_movies, start=1):
    print(f"{i}. {movie_title}")
```

Python


```
Top 5 recommended movies for user 2 are Flint (2017):
1. Cry, the Beloved Country (1995)
2. Georgia (1995)
3. Othello (1995)
4. Restoration (1995)
5. Dangerous Minds (1995)
```

We can see that the system recommends 5 movies for the user



CONCLUSION

The project successfully tackles the challenge of delivering personalized movie recommendations. By combining collaborative filtering and content-based filtering techniques, it cleverly enhances the precision and relevance of suggestions, ultimately enhancing the user's movie-watching experience. Through meticulous data analysis, algorithm development, and rigorous evaluation using metrics like RMSE and MAE, this project not only meets its goals but also showcases its potential in the realm of recommendation systems. It promises to enhance user satisfaction and engagement in today's ever-changing landscape of personalized content delivery.



THANK YOU

