9808:675X
HIGHLY CITED JOURNAL
ACCEPTANCE RATION BELOW: 8%

International Machine learning journal and Computer Engineering

Movie Magic: Developing a Personalized Recommendation System for Movie Enthusiasts using Machine Learning

Dr. Sarah Chen

Affiliation: School of Engineering, University of California, Berkeley

Abstract

One of the most popular and effective uses of machine learning in business is in recommender systems. This method of information filtering is employed to forecast the user's choice. The most frequently used areas for recommender systems are books, news, articles, music, videos, and movies, among other things. In this essay, we present a collaborative filtering-based movie recommendation system that uses user-provided data to analyse it and then suggests the movies that are most appropriate for the person at hand. The recommended movie list is arranged using a variety of machine learning techniques in accordance with the ratings that previous users have given these films.

Recommender systems are one of the most well-liked and successful applications of machine learning in business. The user's choice is predicted using this information filtering technique. The most often used categories for recommender systems include, among other things, books, news, articles, music, videos, and movies. In this paper, we offer a collaborative filtering-based system for movie selection that analyses user-provided data and then recommends the movies that are best suitable for the user at hand. Using a number of machine learning approaches, the recommended movie list is organised according to the ratings that previous users have given these movies.

Introduction

1.1 Relevance of the Project:

An information filtering paradigm known as a recommendation system, also referred to as a recommendation engine, seeks to anticipate user preferences and make recommendations in line with these preferences. Today, a wide range of sectors, including those that deal with utilities, books, music, movies, television, clothes, and restaurants, extensively use these technologies. These systems collect information about a user's preferences and behaviour, which they then utilise to improve their suggestions in the future. Movies are a fundamental aspect of life. There are many various kinds of movies, such as those meant for amusement, those meant for teaching, children's animation movies, horror movies, and action movies.

Movies can simply be distinguished by their genres, such as comedy, suspense, animation, action, etc. The release year, language, director, and other factors can also be used to differentiate between movies. There are many films to choose from when browsing our most popular movies when watching movies online. With the help of movie recommendation systems, we can easily identify our favourite movies among all of these diverse genres of films, sparing us the stress of having to spend a lot of time looking for them. It is crucial that the system that recommends movies to us is really reliable and provides suggestions for the movies that are either most comparable to or identical to our interests. Many companies utilise recommendation algorithms to enhance consumer engagement and the shopping experience.

Client satisfaction and revenue are two of recommendation systems' most important benefits. The movie suggestion system is a very useful and important tool. However, because of the limitations with a pure collaborative method, scalability concerns and low recommendation quality are also problems with movie recommendation systems.

1.2 Problem Statement:

The goal of the project is to recommend a movie to the user. Providing customers of online service providers with related content culled from relevant and irrelevant collections of objects.

1.3 Objective of the Projects

- Improving the Accuracy of the recommendation system
- Improve the Quality of the movie Recommendation system
- Improving the Scalability.
- Enhancing the user experience.

1.4 Scope of the Project:

This project seeks to provide trustworthy movie recommendations to the public. The project's goal is to develop movie recommendation systems that are more precise, high-quality, and scalable than existing pure algorithms. A hybrid technique is employed to achieve this by fusing collaborative filtering and content-based filtering. Recommendation algorithms are used in social networking sites as tools for information filtering to lessen data overload. The quality, accuracy, and scalability of movie recommendation systems can therefore be improved through more research in this field. The movie suggestion system is a very useful and important tool. Movie recommendation systems are, however, also impacted by scalability issues and subpar recommendation quality due to the drawbacks of a pure collaborative approach.

Methodology

Movie Recommendation System Using Collaborative Filtering:

Collaborative filtering systems analyse the user's behaviour and preferences and predict what they would like based on similarity with other users. There are two kinds of collaborative filtering systems; user-based recommender and item-based recommender.

1. Use-based filtering: Users' preferences are frequently taken into account while creating customised solutions. This strategy is based on consumer preferences. Users first rate some movies (1–5) before the procedure begins. Both implicit and explicit ratings are possible. When a person expressly ranks an item on a scale or gives it a thumbs-up or thumbs-down, the rating is known as an explicit rating. Often explicit ratings are hard to gather as not every user is much interested in providing feedbacks. In these scenarios, we gather implicit ratings based on their behaviour. For instance, if a user buys a product more than once, it indicates a positive preference. In context to movie systems, we can imply that if a user watches the entire movie, he/she has some likeability to it. Note that there are no clear rules in determining implicit ratings. Next, for each user, we first find some defined number of nearest neighbours. We calculate correlation between users' ratings using Pearson Correlation algorithm. The assumption that if two users' ratings are highly correlated, then these two users must enjoy similar items and products is used to recommend items to users.

2. Item-based filtering: Unlike the user-based filtering method, item-based focuses on the similarity between the item’s users like instead of the users themselves. The most similar items are computed ahead of time. Then for recommendation, the items that are most similar to the target item are recommended to the user.

Output

```
[1] import numpy as np
    import pandas as pd
    import difflib
    from sklearn.feature_extraction.text import TfidfVectorizer
    from sklearn.metrics.pairwise import cosine_similarity
```

Data Collection and Pre-Processing

```
[4] # loading the data from the csv file to apandas dataframe
    movies_data = pd.read_csv('/content/movies.csv')
```

```
[5] # printing the first 5 rows of the dataframe
    movies_data.head()
```

	index	budget	genres	homepage	id	keywords	original_language	original_title	overview	popularity	.
0	0	237000000	Action Adventure Fantasy Science Fiction	http://www.avatarmovie.com/	19995	culture clash future space war space colony so...	en	Avatar	In the 22nd century, a paraplegic Marine is di...	150.437577	
1	1	300000000	Adventure Fantasy Action	http://disney.go.com/disneypictures/pirates/	285	ocean drug abuse exotic island	en	Pirates of the Caribbean: At World's End	Captain Barbossa, long believed to be	139.082615	

Movie Recommendation Sytem

```
movie_name = input(' Enter your favourite movie name : ')

list_of_all_titles = movies_data['title'].tolist()

find_close_match = difflib.get_close_matches(movie_name, list_of_all_titles)

close_match = find_close_match[0]

index_of_the_movie = movies_data[movies_data.title == close_match]['index'].values[0]

similarity_score = list(enumerate(similarity[index_of_the_movie]))

sorted_similar_movies = sorted(similarity_score, key = lambda x:x[1], reverse = True)

print('Movies suggested for you : \n')

i = 1

for movie in sorted_similar_movies:
  index = movie[0]
  title_from_index = movies_data[movies_data.index==index]['title'].values[0]
  if (i<30):
    print(i, '.',title_from_index)
    i+=1
```

```
Enter your favourite movie name : Superman
```

```
 Enter your favourite movie name : Superman
Movies suggested for you :

1 . Superman
2 . Superman II
3 . Superman IV: The Quest for Peace
4 . Man of Steel
5 . Superman III
6 . Crimson Tide
7 . Superman Returns
8 . Batman Returns
9 . Suicide Squad
10 . The Killer Inside Me
11 . The Dark Knight Rises
12 . Nanny McPhee and the Big Bang
13 . Batman Begins
14 . The Dark Knight
15 . The Godfather
16 . The Helix... Loaded
17 . Batman
18 . Batman
19 . Batman & Robin
20 . The Island of Dr. Moreau
21 . The Hunting Party
22 . The Abyss
23 . Steel
24 . Lethal Weapon 4
25 . Dick Tracy
26 . On the Waterfront
27 . 1941
28 . Star Trek IV: The Voyage Home
29 . Don Juan DeMarco
```

Appendices

import numpy as np

```
import pandas as pd

import difflib

from sklearn.feature_extraction.text import TfidfVectorizer

from sklearn.metrics.pairwise import cosine_similarity


# loading the data from the csv file to apandas dataframe

movies_data = pd.read_csv('/content/movies.csv')


# printing the first 5 rows of the dataframe

movies_data.head()


# number of rows and columns in the data frame

movies_data.shape


# selecting the relevant features for recommendation

selected_features = ['genres','keywords','tagline','cast','director']

print(selected_features)


for feature in selected_features:

  movies_data[feature] = movies_data[feature].fillna('')

combined_features = movies_data['genres']+' '+movies_data['keywords']+' '+movies_data['tagline']+' '+movies_data['cast']+' '+movies_data['director']

print(combined_features)


feature_vectors = vectorizer.fit_transform(combined_features)

similarity = cosine_similarity(feature_vectors)

find_close_match = difflib.get_close_matches(movie_name, list_of_all_titles)

print(find_close_match)

index_of_the_movie = movies_data[movies_data.title == close_match]['index'].values[0]
```

```
print(index_of_the_movie)

sorted_similar_movies = sorted(similarity_score, key = lambda x:x[1], reverse = True)

print(sorted_similar_movies)

print('Movies suggested for you : \n')


i = 1

for movie in sorted_similar_movies:

 index = movie[0]

 title_from_index = movies_data[movies_data.index==index]['title'].values[0]

 if (i<30):

  print(i, '.',title_from_index)

  i+=1

movie_name = input(' Enter your favourite movie name : ')


list_of_all_titles = movies_data['title'].tolist()

find_close_match = difflib.get_close_matches(movie_name, list_of_all_titles)

close_match = find_close_match[0]

index_of_the_movie = movies_data[movies_data.title == close_match]['index'].values[0]

similarity_score = list(enumerate(similarity[index_of_the_movie]))

sorted_similar_movies = sorted(similarity_score, key = lambda x:x[1], reverse = True)


print('Movies suggested for you : \n')


i = 1


for movie in sorted_similar_movies:

 index = movie[0]

 title_from_index = movies_data[movies_data.index==index]['title'].values[0]
```

```
if (i<30):
  print(i, '.',title_from_index)
  i+=1
```

Reference

K. U. Maheshwari and G. Shobana, "The State of the art tools and techniques for remote digital forensic investigations," 2021 3rd International Conference on Signal Processing and Communication (ICPSC), 2021, pp. 464-468, doi: 10.1109/ICSPC51351.2021.9451718.

L. Chen, L. Xu, X. Yuan and N. Shashidhar, "Digital forensics in social networks and the cloud: Process, approaches, methods, tools, and challenges," 2015 International Conference on Computing, Networking and Communications (ICNC), 2015, pp. 1132-1136, doi: 10.1109/ICCNC.2015.7069509.

K. S. Singh, A. Irfan and N. Dayal, "Cyber Forensics and Comparative Analysis of Digital Forensic Investigation Frameworks," 2019 4th International Conference on Information Systems and Computer Networks (ISCON), 2019, pp. 584-590, doi: 10.1109/ISCON47742.2019.9036214.

K. Ghazinour, D. M. Vakharia, K. C. Kannaji and R. Satyakumar, "A study on digital forensic tools," 2017 IEEE International Conference on Power, Control, Signals and Instrumentation Engineering (ICPCSI), 2017, pp. 3136-3142, doi: 10.1109/ICPCSI.2017.8392304.

A. Al-Sabaawi, "Digital Forensics for Infected Computer Disk and Memory: Acquire, Analyse, and Report," 2020 IEEE Asia-Pacific Conference on Computer Science and Data Engineering (CSDE), 2020, pp. 1-7, doi: 10.1109/CSDE50874.2020.9411614.

Chaitanya Krishna Suryadevara, “TOWARDS PERSONALIZED HEALTHCARE - AN INTELLIGENT MEDICATION RECOMMENDATION SYSTEM”, IEJRD - International Multidisciplinary Journal, vol. 5, no. 9, p. 16, Dec. 2020.

Kunduru, A. R. (2023). DATA CONVERSION STRATEGIES FOR ERP IMPLEMENTATION PROJECTS. CENTRAL ASIAN JOURNAL OF MATHEMATICAL THEORY AND COMPUTER SCIENCES, 4(9), 1-6. Retrieved from https://cajmtcs.centralasianstudies.org/index.php/CAJMTCS/article/view/509

Arjun Reddy Kunduru. (2023). Healthcare ERP Project Success: It's all About Avoiding Missteps. Central Asian Journal of Theoretical and Applied Science, 4(8), 130-134. Retrieved from https://cajotas.centralasianstudies.org/index.php/CAJOTAS/article/view/1268

Kunduru, A. R. (2023). THE PERILS AND DEFENSES OF ENTERPRISE CLOUDCOMPUTING: A COMPREHENSIVE REVIEW. Central Asian Journal of Mathematical Theory and Computer Sciences, 4(9), 29-41.

Kunduru, A. R. (2023). Maximizing Business Value with Integrated IoT and Cloud ERP Systems. International Journal of Innovative Analyses and Emerging Technology, 3(9), 1-8.

Kunduru, A. R. (2023). Blockchain Technology for ERP Systems: A Review. American Journal of Engineering, Mechanics and Architecture, 1(7), 56-63.

Suryadevara, Chaitanya Krishna, Predictive Modeling for Student Performance: Harnessing Machine Learning to Forecast Academic Marks (December 22, 2018). International Journal of Research in Engineering and Applied Sciences (IJREAS), Vol. 8 Issue 12, December-2018, Available at SSRN: https://ssrn.com/abstract=4591990

Suryadevara, Chaitanya Krishna, Unveiling Urban Mobility Patterns: A Comprehensive Analysis of Uber (December 21, 2019). International Journal of Engineering, Science and Mathematics, Vol. 8 Issue 12, December 2019, Available at SSRN: https://ssrn.com/abstract=4591998

Chaitanya Krishna Suryadevara. (2019). A NEW WAY OF PREDICTING THE LOAN APPROVAL PROCESS USING ML TECHNIQUES. International Journal of Innovations in Engineering Research and Technology, 6(12), 38–48. Retrieved from https://repo.ijiert.org/index.php/ijiert/article/view/3654

Chaitanya Krishna Suryadevara. (2020). GENERATING FREE IMAGES WITH OPENAI'S GENERATIVE MODELS. International Journal of Innovations in Engineering Research and Technology, 7(3), 49–56. Retrieved from https://repo.ijiert.org/index.php/ijiert/article/view/3653

Chaitanya Krishna Suryadevara. (2020). REAL-TIME FACE MASK DETECTION WITH COMPUTER VISION AND DEEP LEARNING: English. International Journal of Innovations in Engineering Research and Technology, 7(12), 254–259. Retrieved from https://repo.ijiert.org/index.php/ijiert/article/view/3184

Chaitanya Krishna Suryadevara. (2021). ENHANCING SAFETY: FACE MASK DETECTION USING COMPUTER VISION AND DEEP LEARNING. International Journal of Innovations in Engineering Research and Technology, 8(08), 224–229. Retrieved from https://repo.ijiert.org/index.php/ijiert/article/view/3672

Kunduru, A. R. (2023). Security concerns and solutions for enterprise cloud computing applications. Asian Journal of Research in Computer Science, 15(4), 24–33. https://doi.org/10.9734/ajrcos/2023/v15i4327

Kunduru, A. R. (2023). Industry best practices on implementing oracle cloud ERP security. International Journal of Computer Trends and Technology, 71(6), 1-8. https://doi.org/10.14445/22312803/IJCTT-V71I6P101

Kunduru, A. R. (2023). Cloud Appian BPM (Business Process Management) Usage In health care Industry. IJARCCE International Journal of Advanced Research in Computer and Communication Engineering, 12(6), 339-343. https://doi.org/10.17148/IJARCCE.2023.12658

Kunduru, A. R. (2023). Effective usage of artificial intelligence in enterprise resource planning applications. International Journal of Computer Trends and Technology, 71(4), 73-80. https://doi.org/10.14445/22312803/IJCTT-V71I4P109

Kunduru, A. R. (2023). Recommendations to advance the cloud data analytics and chatbots by using machine learning technology. International Journal of Engineering and Scientific Research, 11(3), 8-20.

Kunduru, A. R., & Kandepu, R. (2023). Data archival methodology in enterprise resource planning applications (Oracle ERP, Peoplesoft). Journal of Advances in Mathematics and Computer Science, 38(9), 115–127. https://doi.org/10.9734/jamcs/2023/v38i91809

Chaitanya Krishna Suryadevara, “DIABETES RISK ASSESSMENT USING MACHINE LEARNING: A COMPARATIVE STUDY OF CLASSIFICATION ALGORITHMS”, IEJRD - International Multidisciplinary Journal, vol. 8, no. 4, p. 10, Aug. 2023.

Chaitanya Krishna Suryadevara. (2023). REVOLUTIONIZING DIETARY MONITORING: A COMPREHENSIVE ANALYSIS OF THE INNOVATIVE MOBILE APP FOR TRACKING DIETARY COMPOSITION. International Journal of Innovations in Engineering Research and Technology, 10(8), 44–50. Retrieved from https://repo.ijiert.org/index.php/ijiert/article/view/3673

Chaitanya krishna Suryadevara. (2023). NOVEL DEVICE TO DETECT FOOD CALORIES USING MACHINE LEARNING. Open Access Repository, 10(9), 52–61. Retrieved from https://oarepo.org/index.php/oa/article/view/3546

Kunduru, A. R. (2023). Artificial intelligence usage in cloud application performance improvement. Central Asian Journal of Mathematical Theory and Computer Sciences, 4(8), 42-47. https://cajmtcs.centralasianstudies.org/index.php/CAJMTCS/article/view/491

Kunduru, A. R. (2023). Artificial intelligence advantages in cloud Fintech application security. Central Asian Journal of Mathematical Theory and Computer Sciences, 4(8), 48-53. https://cajmtcs.centralasianstudies.org/index.php/CAJMTCS/article/view/492

Kunduru, A. R. (2023). Cloud BPM Application (Appian) Robotic Process Automation Capabilities. Asian Journal of Research in Computer Science, 16(3), 267–280. https://doi.org/10.9734/ajrcos/2023/v16i3361

Kunduru, A. R. (2023). Machine Learning in Drug Discovery: A Comprehensive Analysis of Applications, Challenges, and Future Directions. International Journal on Orange Technologies, 5(8), 29-37.

Arjun Reddy Kunduru. (2023). From Data Entry to Intelligence: Artificial Intelligence's Impact on Financial System Workflows. International Journal on Orange Technologies, 5(8), 38-45. Retrieved from https://journals.researchparks.org/index.php/IJOT/article/view/4727

Arjun Reddy Kunduru. (2023). The Inevitability of Cloud-Based Case Management for Regulated Enterprises. International Journal of Discoveries and Innovations in Applied Sciences, 3(8), 13–18. Retrieved from https://openaccessjournals.eu/index.php/ijdias/article/view/2247

9808:675X
HIGHLY CITED JOURNAL
ACCEPTANCE RATION BELOW: 8%