

Plagiarism - Report

Originality Assessment

19%



Overall Similarity

Date: Mar 13, 2024

Matches: 594 / 3096 words

Sources: 14

Remarks: Low similarity detected, check with your supervisor if changes are required.

Verify Report:

Addressing Challenges in Recommendation Systems: Proposing and Implementing Solutions for Improvement

Abstract: Recommendation systems play a crucial role in today's digital landscape, assisting users in discovering content tailored to their preferences. However, they face challenges such as cold start, data sparsity, scalability, diversity of content, and lack of trust. This paper proposes solutions to these challenges and implements a model combining neural matrix factorization for **1 collaborative filtering and content-based recommendations using user** demographics. By leveraging hybrid approaches, demographic information, trust networks, distributed computing, and incremental data **2 training, the system** aims **to provide accurate**, diverse, and trustworthy recommendations while addressing scalability issues. The implementation is tested on the Movie-Lens dataset, demonstrating improved performance in mitigating **cold start and** sparsity issues. Future enhancements include feature-rich onboarding, feature engineering, and user-controlled filters to further enhance recommendation accuracy **and user experience**.

Introduction: In Today's Digital world we are surrounded by Recommendations systems, be it suggesting us our favourite movie to watch on a streaming platform, or helping us browse products from an e-commerce platform. Ultimately, they have become an integral part of our digital experiences. To define we can think of a recommendation system as an engine that filters out content for us from a pool of content based on our preferences and browsing history. Both producers and customers **2 benefit from the** recommendation system, which lowers transaction costs when it comes to locating and selecting items [1]. **There are two** essential components present in any recommendation a user base and product / item base. We can categorise **a recommendation system** as content-based, collaborative, demographic and hybrid. Collaborative filtering **1 systems are being** used on a large scale, although being this popular, there are some **limitations and challenges** faced by recommendation systems. The common challenges include colds-start, data sparsity **and privacy concerns**. This paper addresses some of these challenges and proposes solutions to overcome these challenges. Further we implement **4 a model that** utilizes these solutions in order to give optimal and more personalized

recommendations.

Background and Methodology

2 Types of Recommendation Systems:

We can categorise a recommender system based on the working of algorithm.

Content based Recommendation Systems:

In content-based recommender systems, recommendations are generated based on users browsing history and preferences. These systems focus on features of items. Consider an example of a 2 music recommendation system. Here the features of songs are Artist, Album, Movie, Genre, Release year etc. The songs recommended to a user can be from the Similar genre, movie or artist based on his listening or browsing history. 13 There are various similarity calculation techniques used by these systems. Cosine Similarity, TF-IDF, Jaccard Similarity and various others to match items according to user's preferences.

2 Collaborative filtering Recommendation Systems:

In collaborative recommender systems, recommendations are generated based on users browsing history as well as other users who have similar browsing history. There are two sub-categories 3 of Collaborative filtering, User-Item and Item-Item.

In User-Item filtering, based on preferences and user history similar users are grouped together and their uncommon preferences are exchanged. Consider an example of a Food recommender system, here it is observed that user A and user B have ordered Burger and fries as common in past, however user B also ordered Coke. The system identifies them as similar users and Coke 1 is recommended to user A.

In Item-Item filtering, items that go hand in hand are grouped together, an 13 example of this could be of an e-commerce recommender system, based on users purchase history the system identifies items that are often purchased together (laptop and mouse, cap and sunglasses). If a user is about to purchase one the other is recommended.

Demographic Based 2 Recommender Systems.

In Demographic Based Recommender Systems, recommendations are generated based on

demographic information of user. These systems focus on characteristics of users like age, gender, location, etc [1] and recommend items. Let's consider an example of Job Recommending system, a user would be recommended jobs based on the locality he resides in.

Literature Review:

Designing [2] a recommendation system is a complex task, it is the engine which lies between users and products. Providing personalized and accurate recommendations to users keep them engaged with the system. If new users are provided the right recommendations they're looking for, it builds user confidence, improves system adoption and increases user value, this altogether evaluates growth potential. Hence [1] there is a need to understand some of the basic functionalities and entities present in a recommendation system.

The paper, "Recommendation Systems: Principles, Methods, and Evaluation" by F.O. Isinkaye, Y.O. Folajimi, and B.A. Ojokoh [2] highlights the importance of recommendation systems and explains in-depth about the principles and analyses various methods of recommendation approaches along with pros and cons for each approach. It presents various phases of recommendation process how information is collected, how model is trained and how predictions/recommendations are presented to users. The algorithm uses hybrid approach to overcome common problems like cold-start [1] and data sparsity, because using multiple techniques suppress the weakness on an individual technique. It highlights how essential it is to keep user history. The algorithms are evaluated based on MAE, lower MAE better algorithm performance. To summarize the paper, it provides a systematic and resourceful approach about how [2] to develop a recommender system.

While building an effective recommendation system it is important that we overcome the challenges faced.

In their study, [1] "A Systematic Review and Research Perspective on Recommender Systems [3]," Deepjyoti Roy and Mala Dutta look closely at how recommender systems help us find things we might

like online. A tabulation of the features and challenges helps in understanding which algorithm performs well for what data. They explain different types of these systems and talk about the challenges they face, such as not having enough information about users or items. 8 There are various Optimization techniques listed by the author. The paper had identified some research gaps regarding Deployment 1 such as cold start, scalability, sparsity, challenges in collecting implicit user data and real-time user feedback, challenges faced in measuring system performance.

2 In the paper, "A Systematic Review of Recommendation Systems: Applications and Challenges," B. Sunitha and Dr. B. Kranthi Kiran explore how recommendation systems work in our digital world, along with some history about recommender systems. They talk about 8 different types of recommendation systems, like ones based on what you've liked before or what people similar to you have liked. They also mention problems these systems face, like when new users join or when there isn't enough information about items. They suggest ways to solve these problems, like using information about users or making recommendations more transparent. They have included 2 some applications of recommender systems in various fields. To summarize this paper provided an overview of challenges faced by recommender system and their uses in day-to-day life.

While progressing in this 3 literature review and citing more papers like [5], [6], [7], [8] we observe that matrix-factorization and neural collaborative filtering and their variants are regularly utilized and primarily selected. We will further provide a comparative study and implement the one which provides optimal performance. While reviewing these papers there were some challenges discussed in those papers [2] and [3]. Some of them are stated in section given below.

Challenges faced By Recommender Systems.

Cold Start:

1 The problem of cold start is one of the most common challenges observed in recommender systems. It is usually noticed when new users and new items are introduced to system. New users 2 do not have any browsing history / preferences, similarly new items do not have interactions.

Hence the system struggles to provide accurate recommendations to new users and new items take time to gain sufficient data or feedback for accurate recommendations.

Data Sparsity:

For any recommendation algorithm to train, data is input as a matrix in fig. . The columns represent all items and the rows represent all the users present in the system. Usually, the cells represent a rating given by a user to a particular item. There exists only a small number of user-item interactions. In reality it is observed that while training 95-99% of the matrix is sparse.

Scalability:

As a recommender system gets popular, more users and items are added to the system. Since calculations are compute intensive, there comes a point where the existing instance won't have sufficient computing and memory capacity to carry-out the calculations required to train the algorithm.

Diversity of content:

There exists a trade-off between providing accurate and diverse recommendations. The more we focus on provide accurate recommendations, diversity of content decreases and vice versa.

Lack of Trust:

It is observed that users raise privacy and security concerns, since these systems record browsing and purchase history of users. There exist systems who use trust networks and create profile based on the user's personal data. Users often feel a lack of trust because the system isn't able to provide transparent and explainable recommendations.

Solutions:

Hybrid approach:

Design a system that implements two or more recommendation algorithms like content based and collaborative filtering. This will increase accuracy and coverage further leading to cope with challenges like Data sparsity and Diversity of content.

Demographics:

Collecting users demographics like age, gender, location etc and clustering users based on their

demographics helps to deal with **the cold start problem**. Recommending popular and trending items to newly signed up **users based on their location**.

Trust Network:

Allowing users to link their social media accounts and recommending them **1 items which are** rated by or purchased by their social connections. We can modify the training algorithm in such a manner that the user's social connections have a higher priority and strangers have a low priority.

Distributed Computing:

4 Use Distributed computing frameworks like Apache Spark to train the algorithm. The training is distributed among many instances which are available on-demand and coordinated by a master node. This will eliminate scalability issue

Incremental Data Imports and Training:

The algorithm should be incrementally trained over latest data available.

Data privacy and Explainable Recommendations:

Use hashing and encrypt data so as a precaution to data breaches. The recommendations provided by the system should be explainable, this establishes a level of trust in the user's mind towards the system. Labelling promoted content assures **1 to users that** the **system is not** biased.

Implementation:

The goal is to implement a new system that would sweep off the challenges like cold-start, Data **sparsity and scalability** effectively. It will provide solutions to improve performance **2 of recommendation systems**, making them more adaptable to new **users and items** and prioritizing users privacy, security and trust.

Selection of algorithm:

11 Matrix factorization and Neural Collaborative Filtering using Multilayer Perceptron are primarily used **in collaborative filtering**. However, there were many studies that used various modified versions

of these, like Neural Matrix Factorization. We will have **3** a comparative study on which will perform better, and further implement it in our system.

Matrix Factorization

Matrix factorization is a technique used to decompose a large sparse matrix into lower-dimensional matrices, typically to identify latent features that represent users and items. This technique is employed to overcome challenges such as data sparsity and cold start by capturing underlying patterns in user-item interactions. In the implementation described, matrix factorization is utilized to extract meaningful representations **2** of users and items from the Movie-Lens dataset, facilitating collaborative filtering-based recommendations.

3 Neural Collaborative Filtering using MLP

Neural Collaborative Filtering (NCF) is an extension of traditional collaborative filtering techniques that incorporates neural networks to capture nonlinear patterns in user-item interactions. In this context, using **9** a Multi-Layer Perceptron (MLP) as the neural network architecture, NCF learns intricate relationships between users and items to enhance recommendation accuracy. By leveraging the expressive power of neural networks, NCF can capture complex user-item interactions and generate more accurate recommendations compared to traditional methods.

5 Neural Matrix Factorization

Neural Matrix Factorization (NeuMF) combines the principles of matrix factorization and neural networks to improve recommendation performance. In NeuMF, neural networks are integrated with traditional matrix factorization techniques to capture both linear and nonlinear patterns in user-item interactions. By jointly learning from latent features and high-level representations extracted by neural networks, NeuMF offers enhanced recommendation accuracy and can better handle challenges such as **2** data sparsity and cold start. In the described implementation, NeuMF **1** is employed to provide more accurate and personalized recommendations to users based on their preferences and demographic information.

Dataset: We will use Movie-lens 1M Dataset for model selection and training. It consists of approx. 1000000 **7** ratings given by users to movies. The ratings are explicitly given and are on a scale of 1-

5.

MovieLens 1M

Users

3706

Movies

6040

Interactions

1000209

Dimension of Matrix

3706 X 6040

Cells

2,23,84,240

Percentage of Cells Occupied

4.46%

Sparsity

95.54%

Dataset Statistics

As we can see the dataset has very high sparsity. It is logical because there exists no user who watched all movies, 4 and there is no movie which is watched by every user.

Training:

Epoch

MF_loss

MLP_loss

NeuMF_loss

1

2.0773

0.3902

0.4038

2

1.4394

0.3472

0.3395

3

0.9157

0.3245

0.3147

4

0.7401

0.3100

0.3036

5

0.6511

0.2987

0.2967

6

0.5894

0.2890

0.2919

7

0.5401

0.2819

0.2878

8

0.4995

0.2769

0.2839

9

0.4669

0.2728

0.2804

10

0.4408

0.2695

0.2772

From experimenting we observe that 5 Neural Matrix factorization performs optimally.

System Desing:

Hence, we develop system that uses Hybrid 3 approach to recommend movies using Neural Matrix Factorization for collaborative filtering, content-based recommendations using demographic of users.

Gathering user preferences while onboarding users and use this information to create user profile. All of this together will help in solving Problem of cold start.

To have a balance between accuracy and diversity, for every user we will record the latest 10 movies he watched and based on these 10 movies we will recommend him 10 more movies 9 for each of the 10 movies.

To enhance transparency, we will provide explainable recommendations. Example A movie is suggested 2 to a user based on the genre he prefers, or suggesting popular and trending movies from his location.

We collect additional meta-data for movies in order to generate similar recommendations, briefly explained in dataset modification

Dataset: We used Movie-latest, Movies Dataset for modification and implementation.

Dataset Modification:

Cleaned and extracted metadata for each movie present in movie-lens dataset and extracted producer, language, keywords, production house and country of origin. This will help in generating similar movies with more accuracy.

Added generated data like age, gender, location essential for user profiling.

Added user-history **8** based on the latest timestamp of user.

Preprocessing:

Performed Exploratory Data analysis. Encoded **7 ratings given by users** in a range of [1-5] with the following function

Rating >3.5 user liked the movie

Now instead of [1-5] **the ratings are** labelled to -1 if user liked the movie and 1 [-1:1]

Added generated data like age, gender, location for user profiling.

Calculating user similarity based on common movies liked by users.

Content based similarity generating text embedding from title overview tagline and keywords tfidf vector

Collaborative filtering

Loaded **5 Neural Matrix Factorization** Into the system

Training:

System Configuration: Amazon Sagemaker notebook instance ml.m5.2xlarge

vCPUs

8

Memory (GiB)

32.0

Memory per vCPU (GiB)

4.0

Physical Processor

Intel Xeon Platinum 8175

Trained the model for 30 epochs and saved the model.

Results:

We Designed our system based on the embeddings generated. The system provides both content based and collaborative **2 recommendations to users**. The recommendation provided are explainable i.e.(recommended x, y, z movies because you watched a). The system provides a balance between Accuracy and Diversity because the recommendations are based on users last 10 movies watched and his set preferences.

Future Scope:

Feature Rich Onboarding

Currently the user preferences are generated using scripts. In future, we can introduce an onboarding process where **12 user signs up for the first time**, we take preferences from users about what all they like (genres, languages, etc.). This will help in profiling and suggesting better recommendations

Feature Engineering

1 It was observed that 95-99 % of the matrix was sparse. Instead of focusing on ratings given, do some feature engineering to introduce a score based on some designed metrics that track **user's interaction with** items **in the system, and** train **model based on this** Score.

User-Controlled Filters

Let the user choose what he wants to be recommended, introduce active feedback filters and introduce incremental data imports to train the model **3 so that the** algorithm stays adapted.

Conclusion:

We **1 employed a hybrid** approach, **combining content-based and collaborative filtering techniques**, along with leveraging demographic information and trust networks, recommendation systems can enhance accuracy, coverage, and user satisfaction. Additionally, employing **14 distributed computing frameworks** like **Apache Spark and** incremental data imports can overcome scalability issues and ensure continuous adaptation to evolving user preferences.

3 The implementation of these solutions in a model utilizing Neural **Matrix Factorization for**

collaborative filtering and demographic-based recommendations has shown promising results. By incorporating user preferences, explainable recommendations, and a balance between accuracy and diversity, the system 9 aims to provide an enriched user experience.

Looking ahead, there are opportunities for further refinement and innovation 2 in recommendation systems. Future developments may include feature-rich onboarding processes, advanced feature engineering techniques, and user-controlled filters to empower users and enhance algorithm adaptability. By embracing these advancements, recommendation systems can continue to evolve, meeting the diverse needs 4 of users and fostering trust and engagement in the digital landscape.

References:

[1] Avis, P., Arif, D. L., & Sismoro, H. (2022). The Comparison Study of Matrix Factorization 6 on Collaborative Filtering Recommender System. In 2022 5th International Conference on Information and Communications Technology (ICOIACT). IEEE

[2] Isinkaye, F. O., Folajimi, Y. O., & Ojokoh, B. A. (2015). 1 Recommendation Systems: Principles, Methods, and Evaluation. Egyptian Informatics Journal, 16(3), 261-273.

[3] Roy, D., & Dutta, M. (2022). A Systematic Review and Research Perspective on Recommender Systems. Journal of Big Data, 9, Article number: 59.

[4] Luo, X., Xie, Y., Zhao, W. X., Wen, J. R., & Zhang, H. J. (2021). 3 A Comparative Study of Deep Learning-based Recommender Systems. Proceedings of the 14th ACM International Conference on Web Search and Data Mining (WSDM), 730-738.

[5] Wang, J., Zhang, R., & Vucetic, S. (2020). A comparative study of collaborative filtering algorithms. Information Sciences, 512, 322-335.

[6] Lee, H., & Oh, H. J. (2021). A comparative study of matrix factorization and deep learning-based

recommendation systems. Information Processing & Management, 58(2), 102451.

[7] Zhang, S., Yao, L., & Sun, A. (2021). [A Comparative Study of Neural Collaborative Filtering and Matrix Factorization for Recommendation](#). IEEE Transactions on Knowledge and Data Engineering, 33(4), 1542-1556.

[8] Wang, S., Wang, W., Li, Y., & Zhang, J. (2021). [A comparative study of graph neural networks and matrix factorization for recommendation](#). Information Sciences, 550, 219-234.

[9] S. F. Ramadhan, Z. K. A. Baizal, and R. Rismala. [10 \(2020\) Lodging Recommendations Using the SparkML Engine ALS and Surprise SVD](#), Volume 4, 889–897

Sources

1	https://journalofbigdata.springeropen.com/articles/10.1186/s40537-022-00592-5 INTERNET 6%
2	https://www.mdpi.com/2076-3417/10/21/7748 INTERNET 5%
3	https://link.springer.com/article/10.1007/s42979-023-02207-z INTERNET 3%
4	https://www.itconvergence.com/blog/challenges-and-solutions-for-building-effective-recommendation-systems/ INTERNET 1%
5	https://www.d2l.ai/chapter_recommender-systems/neumf.html INTERNET 1%
6	https://link.springer.com/article/10.1007/s10639-021-10643-8 INTERNET 1%
7	https://towardsdatascience.com/movie-recommender-system-a5dbfdb2585d INTERNET 1%
8	https://www.analyticsvidhya.com/blog/2021/07/recommendation-system-understanding-the-basic-concepts/ INTERNET 1%
9	https://towardsai.net/p/machine-learning/the-multilayer-perceptron-built-and-implemented-from-scratch INTERNET <1%
10	https://moraref.kemenag.go.id/documents/article/98984515036263275 INTERNET <1%
11	https://arxiv.org/pdf/2107.13472 INTERNET <1%
12	https://www.justinmind.com/blog/user-onboarding/ INTERNET <1%
13	https://journalofbigdata.springeropen.com/articles/10.1186/s40537-021-00425-x INTERNET <1%
14	https://medium.com/@nasdag/ray-vs-spark-the-future-of-distributed-computing-b10b9caa5b82 INTERNET <1%

EXCLUDE CUSTOM MATCHES OFF

EXCLUDE QUOTES OFF

EXCLUDE BIBLIOGRAPHY OFF