KIPS Trans. Comp. and Comm. Sys.
Vol.12, No.1 pp.25~40
ISSN: 2287-5891 (Print), ISSN 2734-049X (Online)

https://doi.org/10.3745/KTCCS.2023.12.1.25

A Prospective Extension Through an Analysis of the Existing Movie Recommendation Systems and Their Challenges

Cho Nwe Zin Latt† · Muhammad Firdaus†† · Mariz Aguilar††† · Kyung-Hyune Rhee††††

ABSTRACT

Recommendation systems are frequently used by users to generate intelligent automatic decisions. In the study of movie recommendation system, the existing approach uses largely collaboration and content-based filtering techniques. Collaborative filtering considers user similarity, while content-based filtering focuses on the activity of a single user. Also, mixed filtering approaches that combine collaborative filtering and content-based filtering are being used to compensate for each other's limitations. Recently, several AI-based similarity techniques have been used to find similarities between users to provide better recommendation services. This paper aims to provide the prospective expansion by deriving possible solutions through the analysis of various existing movie recommendation systems and their challenges.

Keywords : Recommendation, Collaborative Filtering, Content-based Filtering, Artificial Intelligence, Neural Network

기존 영화 추천시스템의 문헌 고찰을 통한 유용한 확장 방안

Cho Nwe Zin Latt† · Muhammad Firdaus†† · Mariz Aguilar††† · 이 경 현††††

요 약

추천 시스템은 지능적인 자동 결정을 생성하기 위해 사용자가 자주 사용한다. 영화 추천 시스템의 연구에서, 기존 접근 방식은 협업 및 콘텐츠 기반 필터링 기술을 사용한다. 협업 필터링은 사용자 유사성을 고려하는 반면, 콘텐츠 기반 필터링은 단일 사용자의 활동에 중점을 두고 있다. 또한 협업 필터링과 콘텐츠 기반 필터링을 결합한 혼합 필터링 접근법은 서로의 한계를 보완하기 위해 사용되고 있다. 최근엔 더 나은 추천 서비스를 제공하기 위해 사용자 간의 유사성을 찾는데 몇 가지 AI 기반 유사성 기법을 사용하고 있다. 본 논문은 기존의 다양한 영화 추천 시스템과 문제점 분석을 통해 가능한 해결책을 도출하여 유용한 확장 방안을 제공하는 것을 목표로 한다.

키워드 : 추천, 협업 필터링, 콘텐츠 기반 필터링, 인공지능, 신경망

1. Introduction

Recommender system (RS), also known as a recommendation system or a recommender engine, is a type of information system that tries to predict the "rating" or "preferred" that a user will assign to a specific item. For video and audio services like Netflix, YouTube, and Spotify, they are best known as playlist generators; alternatively, product recommenders for sites like Amazon, or content recommender systems, are utilized in a number of applications. Furthermore, these systems can work with a single input, such as music, or with multiple inputs from one platform to another, such as news, literature, and search queries. Popular recommender systems for niche subjects such as restaurants and online dating are also available. Accordingly, recommender systems are generally algorithms that try to provide users with relevant items like movies to watch, texts to read, products to buy, or anything else, depending on the industry. The effectiveness of recommender systems can be assessed using various metrics and offline tests. There are various groups which metrics for recommender systems can be placed in. [1] and [2].

※ This work was supported by a Research Grant of Pukyong National University (2021).
※ 이 논문은 2022년 한국정보처리학회 ASK 2022의 우수논문으로 "Survey for Movie Recommendation System: Challenge and Problem Solution"의 제목으로 발표된 논문을 확장한 것임.
† 준 회 원 : 부경대학교 정보보호학과 박사과정
†† 준 회 원 : 부경대학교 인공지능융합학과 박사과정
††† 비 회 원 : 부경대학교 정보보호학과 석사과정
†††† 종신회원 : 부경대학교 컴퓨터공학부 교수
Manuscript Received : August 2, 2022
Accepted : September 12, 2022
* Corresponding Author : Kyung-Hyune Rhee(khrhee@pknu.ac.kr)

Meanwhile, conventional item-based collaborative filtering (ICF) algorithms have relied on user activities such as ratings and sales to determine item similarity. However, there are many instances in which relationships between items exist in the real world, such as when a same director is behind two films or when two goods work well together [2]. These relations, however, provide fine-grained information about items from many perspectives of meta-data, functionality, etc., in contrast to collaborative similarity, which suggests co-interaction patterns from the user's point of view [2,12]. Although, recommendation research has not been as thorough in accounting for multiple item linkages. Therefore, we propose a system that will equip recommendations. This paper focuses on the movie recommendation system, so our main purpose is to give the most relevant movies which is suitable for the user. Here, we use matrix factorization and Relation Collaborative Filtering (RCF) to exploit multiple item relations in movie recommender systems.

This paper is organized as follows: Section 2 discusses the Taxonomy of recommendation system. In Section 3, we explained the challenges of recommendation system. In Section 4, we review and analysis of the challenge and section 5 concludes this paper.

2. Taxonomy of Techniques in RS

The term recommendation system covers a broad spectrum of topics that are applied in various fields. It is employed in a wide range of real-world applications, including entertainment, e-commerce, services, and social media. In the entertainment industry, it is extensively utilized when watching movies or listening to music. The following related works are some of the other applications that use a recommendation system.

- Movie Recommendation: Netflix uses algorithms to recommend movies based on your interests.
- Music Recommendation: Pandora uses the properties of a song of the artists.
- News Recommendation: Google News and Apple News are only two examples of the many programs that offer news recommendations.

The basic concept of a movie recommendation system is straightforward. Every recommender system has two main components: users and items. The system predicts movies for users, and so the actual movies are its products [2,4,7]. The primary goal of movie recommendation systems is to filter and predict only the movies that a matching user is most likely want to see.

The Machine Learning (ML) algorithms for these recommendation systems use user information from the system's database. This data is used to forecast the user's future behavior based on information from the past. And since the data is so critical to ML projects, it should only be handled by experts [2,7].

Users can search for the most relevant movies applying a range of filtration methods and algorithms used by movie recommendation systems. The most popular ML algorithms used for movie recommendations fall under the categories of content-based filtering and collaborative filtering systems.

Fig. 1 shows the taxonomy of techniques in recommendation system. The detailed description will be explained as follows.

2.1 Content-based Filtering

A method such as content-based filtering makes an effort to predict what a user might enjoy based on the user’s activity. It bases its recommendations on factors such as genre, director, description, actors, etc. This information obtained from a single user is critical in utilizing a technique that employs an ML algorithm to recommend movies which are similar to the user's past choices [14,15,17]. Whereupon, prior movie choices, and preferences of a single person will be used to generate similarity in content-based filtering. So, the recommendation engine first examines the user's prior viewing habits before using this information to find comparable movies. Following that, database access to this data will be provided (e.g., lead actors, director, genre, etc.). The system then suggests movies to the user. The use of only one user's data to generate predictions is a key component of content-based filtering [3,15,21,29].

The goal of content-based filtering, as illustrated in Fig. 2, is to suggest the content of an item and the user profile. A user might, for instance, get a movie recommendation based on the synopsis of other movies. As

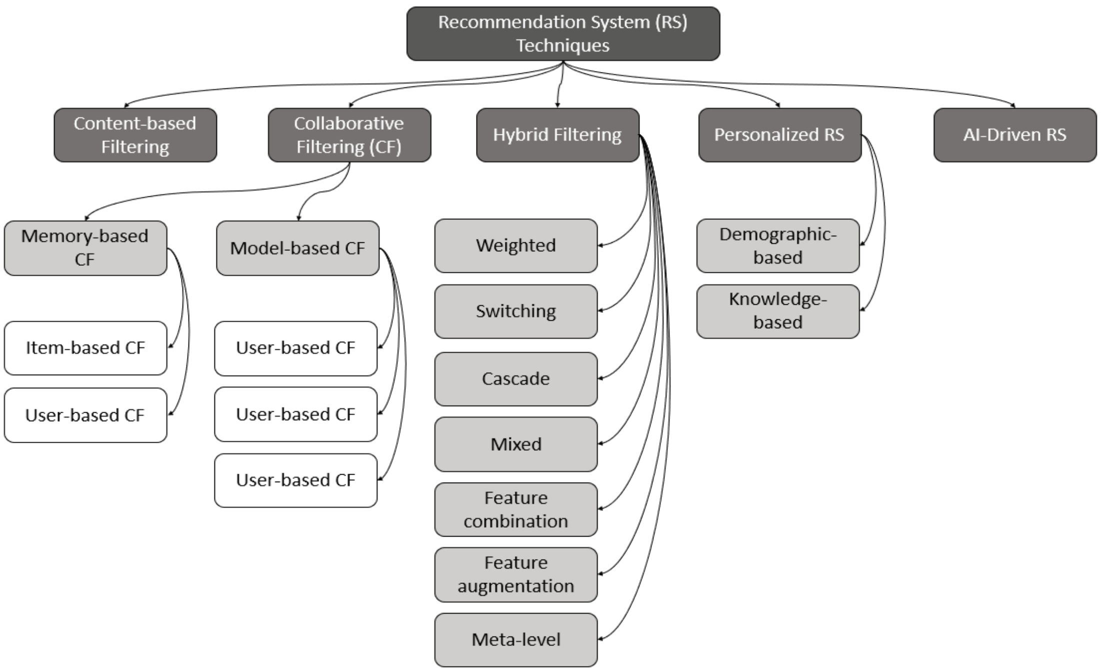

Fig. 1. Taxonomy of Techniques in Recommendation System

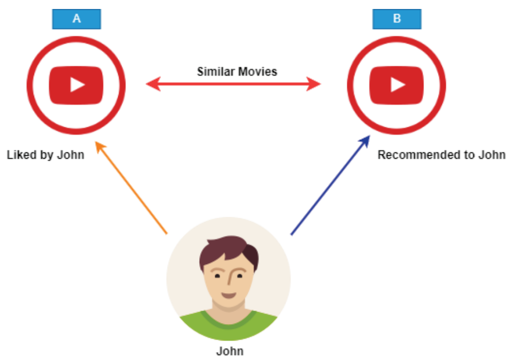

Fig. 2. Content-based Filtering

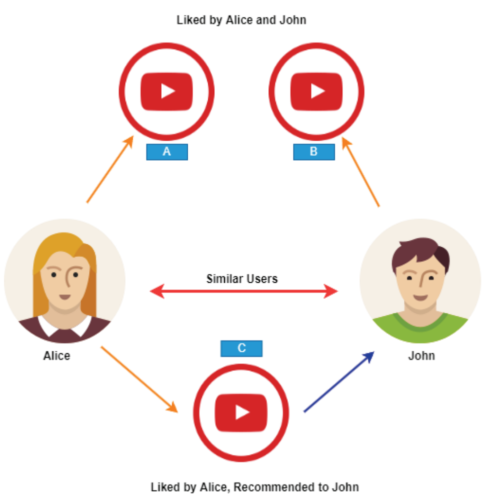

Fig. 3. Collaborative Filtering

a result, a content-based strategy makes it simpler to suggest new products.

Additionally, it offers a feature in the content that helps justify the recommendation. Finding a certain feature like photographs or films of any type might be difficult. This is referred to as an over-specialization issue. Since the users have never recommended anything outside of their user profiles, there is not enough information to make recommendations [3].

2.2 Collaborative Filtering

As the name implies, this filtering technique relies on interactions between the data subject and other users to produce the best results, with the system comparing their behaviors. So, what is the mechanism for this? The history of each user in the database serves as the foundation for this movie recommendation system and the machine learning algorithm on which it is based. It combines film preferences with the usage habits of various users. In simple words, collaborative filtering is based on system users interacting with the objects (movies) [1,6]. This means that the results of this ML-based recommendation system are influenced by all users, in contrast to content-based filtering, which only models input from a single user.

There are two types of collaborative filtering algorithms:

Table 1. Comparison of Recommendation Techniques

Techniques	Advantages	Disadvantages
Content-based Filtering	• The system didn't make product recommendations using the user's data. • The system can suggest new things to users depending on how closely their specifications matches other items.	• In order to produce a recommendation list, we must analyze and find every item feature. • Since the system was independent of how the customer rated this item, a quality assessment of the product was not included.
Collaborative Filtering	• The system didn't use demographic data to suggest products. • The system pair people with things that are comparable. • The system is able to suggest to the user stuff outside of their tastes that they might enjoy.	• The highly performing item list determines the system's quality. • A problem with cold starts exists when recommendation products to new users.
Personalized Recommendation	• It provides recommendations before any item is reviewed by the user and is not based on user-item ratings. • Helping users make knowledgeable decisions. • Generating new information by analyzing data and knowledge already stored. • Wisely and effectively managing large quantities of structured and unstructured data. • Technique of making decisions more productive and consistent. • Ensuring effective documentation of crucial information that consumers can access readily. • Easy data storage for future use.	• The collection of demographic information raises privacy concerns. • Problem of Stability vs. Plasticity • Large-scale data and information gathering, management, and manipulation. • Potential system problems including redundant rules and circular dependencies are being observed. • Addressing the constraints imposed by scientific and cognitive methods. • Navigating knowledge's largely abstract nature. • Delivering a system that is only as good as the data and information contained in it. • Requiring substantial and accurate data to function properly.
Hybrid Recommendation	• It combines the benefits of collaborative filtering and content-based filtering. • It is based on the evaluation of user and the content description. • Avoid overspecializing • Enhance the rate of client satisfaction.	• Undergo cold start issues. • An issue with early product ratings. • The sparsity issue.
AI-driven Approaches	• It can improve accuracy and accurateness and reducing errors. • Decreasing errors and considerably increasing the level of precision. • It has the enormous benefit of being unbiased, allowing for more precise decision-making.	• It can be very expensive and takes a lot time and resources. • The vast majority of production difficulties can be quickly resolved with the use of algorithms, but machines will only carry out the duties for which they have been built and created.

- User-based collaborative filtering: The goal is to discover patterns in the movie preferences of the target user and other database users.
- Item-based collaborative filtering: The basic concept is to identify comparable products (movies) that target users rate or interact with.

According to the current method of movie recommendation systems, combining both tactics will produce the most gradual and explicit results [11,12]. Users with similar interests to another user are discovered using collaborative filtering. Then it suggests an item or product, assuming that the other user will take into account the recommendation based on comparisons with other users who have similar preferences. It implies that this technique connects people who have similar interests and makes recommendations based on their preferences [1,4,6].

The idea behind collaborative filtering is to work cooperatively with the user or movie id. In Fig. 2, for instance, Alice and John are two users. John prefers the movies B and C whereas Alice prefers A, B, and C. Movie C will be suggested to John since John and Alice both have A and B [6,62].

2.3 Hybrid Recommendation

In a hybrid approach, we combine the two advised methods of content-based and collaborative filtering to maximize benefits, improve outcomes, and lessen problems and challenges associated with these applications. Multi-methods are used in this technique.

Many challenges exist in the recommendation system. This section will cover the challenges with the current recommendation system.

- Weighted: We numerically added each recommended component, and the system assigned each one a unique score.
- Switching: The system presents the user with a number of recommendations and asks them to choose one of them.
- Mixed: The system provides the user with recommendations for a variety of goods all at once.
- Feature Combination: Several information sources are combined to create features for the recommendation system.
- Feature Augmentation: It is used to compute a set of features for recommender systems, which is one of the key components of the following technique.
- Cascade: In the recommendation list, items with a high weighted priority score appear first, followed by items with a low weighted score, in descending order.
- Meta-level: It is one of the input methods utilized to create a model for the algorithms step following the recommendation system.

By combining these several approaches, high performance is achieved while reducing issues and challenges that arise when using only content-based or collaborative filtering.

2.4 Personalized Recommendation

1) Demographic-based RS

Regardless of genre or other considerations, demographic recommendations show popular and highly rated movies to viewers with similar demographic backgrounds. As a result, it gives a straightforward yet easily implementable outcome because it does not take into account each person's unique taste. This recommendation system technique used user profile data such as age, gender, interests, and opinions about rating products to find common users with similar interests and ratings of divided users by age group and demographic area [2,7].

2) Knowledge-based RS

Knowledge-based system is a vital subfield of artificial intelligence. It makes a decision based on the knowledge they already have and their ability to comprehend the content of the material being processed. A knowledge-based method is one in which a recommender system bases its recommendations on the user's specific inquiries rather than the user's rating history. The user may be asked for a set of guidelines, an interdiction for how the results should be shown, and a sample of an item. After that, the system performs a search in its item database and displays matching results [10,12].

Knowledge-based recommender systems can be quite helpful even though they are a tool that addresses a particular issue, especially when used in conjunction with other types of recommender systems. They can serve as a temporary fix for the cold start issue before switching to collaborative filtering or a content-based system after ratings have been gathered [10,12,15].

2.5 AI-driven Approaches

A set of machine learning algorithms called artificial intelligence recommendation systems, sometimes known as recommender systems, are employed by developers to forecast consumer preferences and provide pertinent recommendations to users. AI recommendation systems filter and suggest the most appropriate products to a specific user using data science and the data of the users. The content recommendation system is compared to an experienced shop assistant who can offer more enticing products while also raising conversion rates since they are familiar with the user's wants, preferences, and requirements [7,13,29]. Managing requests for proposals (REPs) is one of the most repetitive yet critical processes in a sales organization, so a recommendation system based on AI can be a solution. Evalueserve team developed an AI platform to help teams break down REPs into smaller portions, analyze sections, and then recommend a response for each one [13]. The AI platform assists the business in producing a zero draft, thereby expediting the start of a much bigger project. A machine can now do what a human used to do in processing a full document, increasing business efficiency and improving response team experience [18,24,69].

The teams can use the recommendation system to increase accuracy and support their response rather than having to get every component correct. These kinds of AI-driven solutions and platforms enable businesses to improve decision-making and produce lasting, useful results. As this type of AI engine develops, everything that was previously only feasible by humans becomes faster, more efficient, and more precise. This is merely one illustration of the benefits AI can provide [2,7,13].

3. Challenges

We discuss the numerous issues with the current recommendation system in this section. Meanwhile, in this section, we will explore the various problems that the current recommendation system presents.

3.1 Cold Start

When new users or items are added to the system, this problem develops. It is difficult to predict the preferences or interests of users because a new product can only be recommended to users with ratings or reviews, which results in recommendations that are less accurate [1]. For instance, a user can only be given a recommendation for a recently released movie after it has received some ratings. It is challenging to solve a problem brought on by the addition of a new user or item because it is only possible to locate another person with the same interests or preferences by first learning about the person's previous actions [14,15,37,40].

3.2 Sparsity

It frequently occurs that most users don't rate or review the products they buy, making the rating model highly sparse and potentially causing data sparsity issues. This makes it harder to identify groups of users that share ratings or interests [37,34].

3.3 Privacy

A person usually needs to provide personal information to the recommendation system (and have experience with hyper-personalization) to receive more helpful services. However, doing so raises concerns about data privacy and security, and many users are hesitant to do so. In order to offer individualized services, the recommendation system is obligated to have the personal information to be used to the fullest extent possible. The recommendation systems must ensure user confidence to address this problem [43,44].

3.4 Scalability

One of the most challenging aspects of the recommendation system is the scalability of algorithms using real-world dataset. Scalability is a major concern for these dataset because user-item interactions like ratings and reviews generate a lot of changing data. Large dataset results are inefficiently interpreted by recommendation systems; advanced large-scaled approaches are required to address this issue [55-57].

4. Analysis and Discussion

4.1 Current Techniques and Solutions

A list of CF techniques is presented by Su and Khoshgoftaar [66]. The authors provide an overview of CF theory and briefly address the major issues, such as sparsity, scalability, synonymy, gray sheep, shilling attacks, privacy, etc. Additionally, they display a table of CF technique overviews. In their review of 210 publications on RS, Park et al. [67] categorized the papers based on their data mining methods, application fields, and the year and journal in which they were published. The papers were further divided into eight application fields (films, music, etc.).

On the other hand, content-based RS takes into account contextual information about the products to match consumers with goods that are comparable to ones they have previously enjoyed. By using this technique, the description of the thing is composed of a list of its textual characteristics. In user profiles that explain the products, a user likes, user histories including ratings and descriptions of prior purchases are employed. Pandora and News Weeder both use such recommender algorithms. The item description must be clear for this method to work properly. The cold start users and limited data descriptions, however, are issues with these systems. Additionally, Patel, Krupa, et al. [14] indicated in the research that they experienced cold-start, sparseness, and shilling attacks issues.

Cho Yoon Ho et al. [63]'s solution to the sparsity and scalability issues in the collaborative filtering approach makes use of the decision tree induction method, association rule mining methods, and data warehousing technologies. Because of it, a novel hybrid technique that makes use of web usage mining has increased the effectiveness of the collaborative filtering approach. The Apriori technique was utilized by authors to detect the common patterns using weblogs as a database. And the author uses the decision tree induction method to categorize the customers.

A clustering method has been put out by Magdalini Eirinaki et al. [64] in order to provide the customer with better and quicker recommendations. They employed semantically coherent clusters for the clustering process. Additionally, they employ Domain ontology, which is based on terms taken from online content, for recommendations. For better suggestions, Feng Hsu Wanga et al [65] applied clustering and association rule mining utilizing online usage mining. They employed Hierarchical Bisecting Mediods for clustering.

We can see how the suggested system makes use of the Candidate Generation functionality, Ranking, and Relational Collaborative Filtering functions. Therefore, we constructed a matrix factorization type based on a neural network to create a candidate generation model. Input data is crucial when carrying out the primary purpose of this matrix factorization type to model which function to anticipate.

In order to address the challenges in Section 3, in this paper, we summarize some of the current solutions for the movie recommendation system problem as follows.

1) Candidate Generation

For example, when the user choose and display the first video that user, John will see when he accesses YouTube? Out of all of these millions of videos, a user is first shown an episode of Candidate Generation. So, to create a candidate generation model, and employed neural network-based matrix factorization. [15,16]. A matrix factorization's primary goal is to create a model that predicts how much each user will enjoy a given item (scoring). In order to function properly, these estimate functions require input data. According to reports, demographic data, search queries, and video IDs make up the bulk of YouTube's input data. The score of the items for the appropriate user and the learned user and item embedding are significant results of matrix factorization. These learned embeddings are also helpful in the Re-ranking portion, which will be covered later, and other transfer learning activities [16,18].

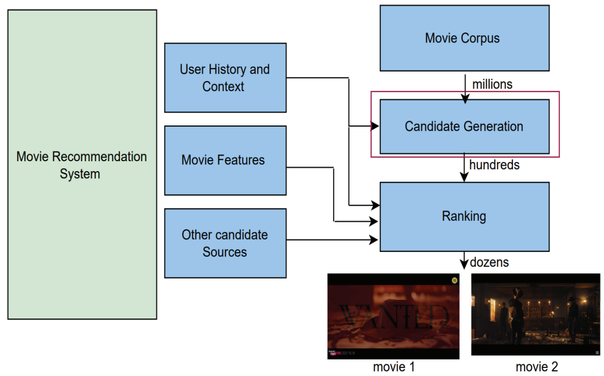

Fig 4. Movie Recommendation Architecture (Candidate Generation)

Fig. 4 depicts the overall structure of our recommendation system. The system employs two neural networks, one of which is utilized for candidate generation and the other for ranking. The candidate generation network extracts a small subset (hundreds of movies) from a large corpus using events from the user's viewing history. These candidates are intended to be extremely accurate and broadly applicable to the user. The candidate generation network can only give substantial customization when using collaborative filtering. The movie ID, search term, and demographic data are the primary input data used for movies. When only a few “best” recommendations are shown in a list, it is necessary to use a fine-level representation to determine relevance among candidates with high memory.

The ranking network achieves this by giving each video a score based on an objective function that has been predetermined as well as a wide range of parameters that describe both the video and the user. The highest-scoring films are displayed to the user in order of their score. The two-step recommendation method enables us to choose recommendations from a large pool of movies (millions) while assuming that the user will find the few movies that do appear on the device to be customized and enjoyable [15,16,18,19].

2) Ranking

The most appropriate video for the relevant user in the Ranking section must be chosen from among the hundreds of videos generated by Candidate Generation. Because there are fewer films inputted in this step than in Candidate Generation, we can include as many features as we need in the Ranking Model. The computation time is significantly reduced when compared to the Candidate Generation section, as it has been reduced from millions to hundreds. The architecture of the Ranking Model is similar to that of the Candidate Generation. The distinction is found in the substitution of Weighted Linear Regression for Softmax Classification. To fine-tune the ranking, the objective function that predicts how long a relevant user will watch the relevant video will be used. The user will be recommended based on the estimated video [15,17,18].

The most interesting characteristic of the Ranking model is the component that determines the Ranking score. The model learns the odds of the expected watch time (based on how long the user will watch the video) and the Ranking score using weighted logistic regression. The score is determined based on how long the viewer watched the movie after clicking the link, and this score is used to determine the weight of input to the logistic regression. The text does not go into great detail regarding the scoring calculating mechanism. But in the research, there are other ways to calculate, and the process is not particularly difficult. Equal weight is given to the negative impression (non-click videos). Example 1: The weight of the positive impression must be higher than 1, E[T] (1+P) equals stated. Additionally, when serving the Ranking score it will be determined by e**x, which is a unique feature [17,18].

The method is utilized to determine the classification score or regression value during training and serving must be the same if the results are normal. The Ranking model on YouTube, however, employs a different set of formulas. It is intriguing to observe the relationships between odds and expected watch time, weighted logistic regression, and $e^{**}(w_x + b)$.

Thus, we may determine the serving formula by determining the Odd value.

$$Odds = e^{**}\ W_x + b + b$$

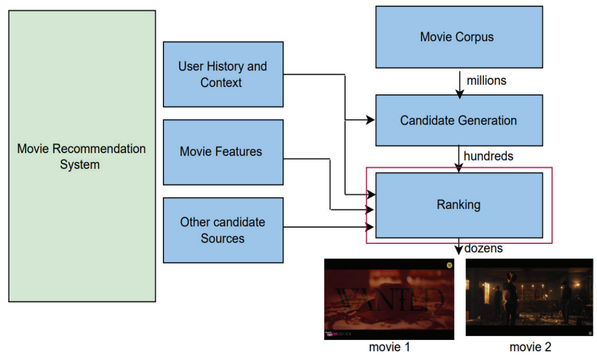

Fig. 5. Movie Recommendation Architecture (Ranking)

In other words, the serving purpose is to determine Odds (Expected Watch Time).

The probability ration of an event or result is known as the Odds.

The positive weight (w) for that positive video can lead to greater positive weight, according to the Weighted Logistic Regression method (w). In other words, we will place more positive weight than negative weight because we want more pleasant impressions than bad impressions. Thus, the Odds formula is modified as shown below.

$$Odds\,(i) = \frac{w_i p}{1 - w_i p}$$

According to the study, there is a very minimal chance that a person will really click the video link.

As a result, if we continue to change the Odds formula, we will get the following.

$$Odds\,(i) = \frac{w_i p}{1 - w_i p} \approx w_i p = T_i p = E(T_i)$$

We can substitute T_i for w_i in this case since it calculates weight (w_i) based on the viewing time.

The result returned is the Expected watching time, or E (T_i), of how long the video is watched after multiplying the Probability of clicking on the movie by the viewing time (T_i).

The Expected watching time is calculated as $e^{**}(w_x + b)$. found in Serving is E(T) [13-16,18,22].

3) Matrix Factorization

Matrix factorization can generate latent characteristics by multiplying two distinct types of entities. Collaborative filtering uses matrix factorization to ascertain the relationship between the entities of items and users. We would like to forecast user ratings of store items using the input of user ratings so that users can receive recommendations based on the prediction. Assuming that the rankings are integers between 1 and 5, and that there are five users and five films in the ranking table, the matrix is shown in the table below [15,19,20].

Since not every user rate all the movies, the matrix has a large number of missing values which makes it sparse. In order to offer full values for the multiplication, the null values that the user did not provide would be filled with 0. For instance, when a given movie stars their favorite actor and actress in the action genre, two people may rate it highly. According to Table 2, which is shown above, Users 1 and 3 both gave high ratings to Movie2 and Movie3.

In another instance, User 4 chooses not to rate Movie4. So, by using ratings provided by users with similar preferences to Movie4, it aims to identify other users who is similar to User 4 and then predict whether or not other users would enjoy the film.

The most crucial results of matrix factorization, as shown in Fig. 6, are learnt user and item embedding as well as the user's item scoring. In a re-ranking part, learn embedding can also be employed for other transfer learning. We discovered that the tactics utilized in the offline evaluation were different from

Table 2: User's Ratings Table on Movie

	Movie1	Movie2	Movie3	Movie4	Movie5
U1		5	4	2	1
U2	1			5	3
U3	1	4	4	1	
U4			2		2
U5	3	1	1		

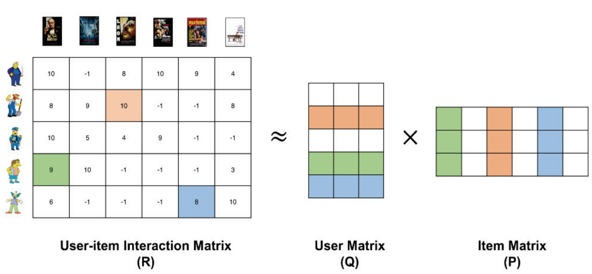

Fig. 6. Matrix Factorization

those used in online serving as a result of our training using Candidate Generation. The offline evaluation, according to the paper, concentrates on model training and precision training. A/B testing is reportedly used to simulate online evaluation. Although there was no real difference between offline and online outcomes [15,17-20].

4) SoftMax Formula

To address the accuracy in prediction problem, we propose the recommendation as extreme multiclass classification, which categorizes a video watch w_t at a given time t among millions of movies i depending ν on a user u and a context C, as indicated below.

$$P(w_t = i | U, C) = \frac{e^{v_i u}}{\sum_{j \in V} e^{v_i u}}$$

Simply mapping sparse entities into a dense vector is what embedding in the equation above entails. Therefore, the main purpose of the deep neural network is to learn user embedding u as a function of the user's history and context, which can then be applied to classify between movies using a SoftMax classifier. Additionally, we train the model based on watched data using the implicit feedback mechanism as opposed to the explicit approach that is currently employed in streaming platforms like YouTube. The reason is that it is readily available and capable of producing recommendations in circumstances when explicit feedback is extremely common.

We employ the SoftMax formula to determine the score. For multi-classification, SoftMax is a formula for determining the probability of the relevant class; the probability of all these classes added together equals 1. As a result, this formula's likelihood can be employed as scoring system. If we can enter the user and content pair into the model, we will obtain the pertinent movie and scoring. The outcome of this Candidate Generation is user, content embedding, and movie embedding. Without flooding the user with information, we can suggest relevant movies to the user [2,4,6,7,18,19,58].

4.2 Discussion and Potential Solution

We discussed the challenges in Section 3: cold start, sparsity, privacy, and scalability. We will discuss about potential solutions for the recommendation system in this section.

1) Solution for Challenge 1: Cold Start

The main challenge with the cold start problem is a lack of information needed to generate recommendations. Procedures for gathering this unavailable data, which can be classified into two types, are included in the proposed solutions. The data can be collected either explicitly by asking questions of the user or implicitly by leveraging pre-existing data [23,24]. Meanwhile, another study expresses a similar idea. These methods, as well as a few solutions that employ them, are discussed in the sections that follow. Based on a review of the current literature, some desired characteristics have been identified, regardless of how missing data are gathered and solutions that should be implemented [25-31].

a) Relevant Recommendation Accuracy

If a recommendation grabs the user's attention, it is relevant. The percentage of appropriate recommendations among recommendations determines how accurate it is [27,28]. This paper introduces a number of metrics for evaluating precision [18]. This parameter can be used to evaluate the efficiency and utility of the system. If less relevant test recommendations are offered, the system's accuracy rate may decrease as it learns the user's profile. Overall correctness must be maintained in solutions. One method is to select the fewest number of query items that are the most informative [27].

b) Minimizing Bias

Ratings are expected to include user and product interactions. However, some evaluations are made without taking interactions into account. Popular products, for example, frequently receive high ratings, and some consumers rank products regardless of their experience with them [25,26]. Such biased ratings make it difficult to personalize suggestions. To avoid this, solutions employ baseline predictors or bias factors, but they require the rating history for proper modeling [26]. Appropriate techniques should be available to detect rating abnormalities immediately, particularly in cold start situations.

c) Adaptability

It is preferable that solutions are adaptive in light of the concept of the recommendation system [30]. A recommendation system can employ a variety of filtering and grading formats. A flexible solution requires less effort to integrate into the system.

d) Diversity

In an e-Market, there are typically several product categories, such as electronics, home goods, apparel, etc [31]. All-encompassing recommendations are necessary for a good RS. For this, it must ascertain the user's preferences across many fields [32]. In light of this designation, solutions should take it into account.

2) Solution for Challenge 2: Sparsity

The effectiveness of hybrid recommenders was compared to that of pure collaborative and content-based filtering techniques in several studies. It was found that hybrid approaches could deliver more accurate recommendations than pure ones. Additionally, the manufacturer advises that these techniques can prevent cold starts and sparse system frequency [34,35].

Netflix is a great illustration of how hybrid recommendation systems can be used. The website compares the opinions and search habits of similar users (i.e. collaborative filtering) and offers movies that have qualities in common with films that a user has highly rated [35,39,40]. If user preferences data is available and a high degree of customization is required, content-based filtering techniques can help with user-specific recommendations [42,36]. When explicit feedback is provided and the data sparsity is not too high, collaborative filtering performs better. When we have item attributes, collaborative filtering combines the advantages of both content and collaborative filtering to improve models by removing data sparsity and lowering errors [37,38,40,42,63].

3) Solution for Challenge 3: Privacy

The privacy protection methods for recommendation systems can be divided into two categories: those based on statistics and those based on cryptography. The statistical approach is realistic and considers the cost of computation in recommendation systems [43, 44]. It can conceal the user's privacy by removing sensitive information from the data. Although it requires more computational resources, encryption technology is a more secure approach [48,49]. Information can be protected entirely without being lost by utilizing homomorphic encryption technology, ensuring the accuracy of the recommendations made by the recommendation system [46,47].

a) Statistics Methods

An advanced, highly efficient form of privacy protection is the statistical approach. Typically, critical information in the data file containing user activity is hidden and the cost of hostile data collectors and attackers is increased by deleting features, creating confusion, increasing noise, and so forth. Different writers have presented a number of anonymization algorithms with various anonymization techniques. The models with the highest levels of approval and acceptance that produce adequate results in anonymization are k-anonymity, l-diversity, and t-closeness. The two most widely used privacy models, k-anonymity and l-diversity, are used to measure the degree of privacy and protect sensitive data from record linkage and attribute linkage attacks respectively. For a variety of attack and privacy scenarios, supplementary secrecy models like t-closeness [44] and m-invariance [45] are also offered. To optimize the benefits of anonymized data sets, a variety of anonymizing techniques are used, including suppression [46], generalization [47,48], anatomization [49], slicing [49], and disassociation [50].

b) Cryptography Methods

In this paper [28], homomorphic encryption is offered as a notion. Because of the unique properties of homomorphic encryption, we are able to directly execute various operations on ciphertext rather than plaintext operations to get the same results while maintaining the confidentiality of plaintext data. In multiparty secure computing and cloud computing, homomorphic encryption techniques will be crucial. The foundational technology for privacy protection in the information age is multi-party confidential computing, which has emerged in recent years as one of the most active study areas in the world of cryptography. Although there are two or more participants, multi-party secret computing is referred to as a whole in academia [46,48].

Table 3. Survey of Current Works to Address RSs' Challenges

Ref. Paper No.	Cold Start	Sparsity	Privacy	Scalability
[23][24]	○	×	×	×
[25][28]	○	×	○	×
[26][27]	○	○	×	×
[30][31] [32][33]	○	○	○	×
[34][36] [37][40]	○	○	×	○
[35][38] [39][41] [42]	×	○	○	×
[43][49] [51]	×	○	○	×
[44][45] [52][53]	×	×	○	○
[46][47] [48][50]	×	×	○	×
[54][55] [56]	×	×	×	○
[57][63]	×	○	×	○

It is frequently referred to as bilateral confidential computing when there are two participants. Private data can be owned thanks to multi-party confidential computing. In order to optimize the usage of private data without compromising data privacy, several participants can cooperate to use these private data for computation without disclosing the secrecy of their private data. A common multi-party calculation is the recommendation system [44-46].

4) Solution for Challenge 4: Scalability

The birth of singular value decomposition (SVD) was a result of a need to increase scalability [63]. The term "single value decomposition" is abbreviated as SVD. A method of factorizing a matrix that can lower the dimensionality of the data [52,53]. LSI (Latent Semantic Indexing) utilizes SVD to overcome concerns with polysemy and synonymy for information retrieval purposes [54]. Scalability was increased by using SVD extensively. SVD and neural networks have been used in research to enhance prediction [55]. This method turned CF into a classification issue using SVD. Its output was fed into an algorithm for artificial neural networks, which may be trained to produce more accurate predictions. Group Lens also utilized SVD in several RS. To compare the outcomes of predictions made using the SVD and the simple CF, a study has been conducted [50]. There were two main kinds of experiments done. The efficiency of perdition was compared in the first tests between SVD and CF. The efficiency of SVD and CF in generating Top-N predictions were contrasted in the second experiment. Another study on SVD and CF was conducted [57] that sequentially applied both of these methods. In order to minimize the dimensionality of the input data, SVD was first used. After that, reduced dimensionality was subjected to user-based CF. The adoption of SVD increased scalability.

4.3 Summary

The most common challenges in recommendation systems are cold stat, sparsity, privacy and scalability. Based on previous studies, we will survey those challenges in this paper. We described the problems and their solutions in detail above. We will review the articles which potentially resolve the concerns in this part. The summary of the challenges which the research paper can (○) or can not (×) overcome is shown in the following table.

5. Conclusion

Movie recommendation systems have proven to be the most effective means of dealing with the problem of information overload. They make decision-making easier by saving time and energy. This work presented the analysis of the current methods, challenges, and potential solutions for the movie recommendation system.

For future research, we will focus on improving the existing methodologies and algorithms in order to increase the accuracy of prediction and recommendations. We will also work on the security and privacy of recommendation systems based on Blockchain technology.

References

[1] G. Suganeshwari and S. P. Syed Ibrahim, “A survey on collaborative filtering based recommendation system,” in *Proceedings of the 3rd international symposium on big data and cloud computing challenges (ISBCC–16’)*. Springer, Cham, pp.503-518, 2016.

[2] S. Rastogi, D. Agarwal, J. Jain, and K. P. Arjun, “Demographic Filtering for Movie Recommendation System Using Machine Learning,” in *Proceedings of International Conerence on Recent Trends in Computing*. Springer, Singapore, pp.549-557, 2022.

[3] J. Shu, X. Shen, H. Liu, B. Yi, and Z. Zhang, “A content-based recommendation algorithm for learning resources,” *Multimedia Systems*, Vol.24, No.2, pp.163-173, 2018.

[4] M. K. Kharita, A. Kumar, and P. Singh, “Item-based collaborative filtering in movie recommendation in real time,” in *2018 First International Conference on Secure Cyber Computing and Communication (ICSCCC)*. IEEE, pp.340-342, 2018.

[5] S. K. Huwanshi, and R. K. Pateriya, “Collaborative filtering techniques in recommendation systems,” *Data, Engineering and Applications*. Springer, Singaporre, pp.11-21, 2019.

[6] R. Sujithra Alias Kanmani, B. Surendiran, and S. P. Ibrahim, “Recency augmented hybrid collaborative movie recommendation system,” *International Journal of Information Technology*, Vol.13, No.5, pp.1829-1836, 2021.

[7] N. Pereira, and S. L. Varma, “Financial planning recommendation system using content-based collaborative and demographic filtering,” In Smart Innovations in Communication and Computational Sciences, Springer, Singapore, pp.141-151, 2019.

[8] F. Yang, “A hybrid recommendation algorithm–based intelligent business recommendation system,” *Journal of Discrete Mathematical Sciences and Cryptography*, Vol.21, No.6, pp.1317-1322, 2018.

[9] S. Sharma, V. Rana, and M. Malhotra, “Automatic recommendation system based on hybrid filtering algorithm,” *Education and Information Technologies*, Vol.27, No.2, pp.1523-1538, 2022.

[10] Y. Wang et al., “An enhanced multi-modal recommendation based on alternate training with knowledge graph representation,” *IEEE Access*, Vol.8, pp.213012-213026, 2020.

[11] N. F. AL-Bakri and S. H. Hashim, "Collaborative filtering recommendation model based on k-means clustering," *Al-Nahrain Journal of Science*, Vol.22, No.1, pp.74-79, 2019.

[12] S. S. Choudhury, S. N. Mohanty, and A. K. Jagadev, "Multimodal trust based recommender system with machine learning approaches for movie recommendation," *International Journal of Information Technology,* Vol.13, No.2, pp.475-482, 2021.

[13] Q. Zhang, J. Lu, and Y. Jin, "Artificial intelligence in recommender systems," *Complex & Intelligent Systems,* Vol.7, No.1, pp.439-457, 2021.

[14] K. Patel and H. B. Patel, "A state-of-the-art survey on recommendation system and prospective extensions," *Computers and Electronics in Agriculture,* Vol.178, pp.105779, 2020.

[15] M. Goyani and N. Chaurasiya, "A review of movie recommendation system: Limitations, survey and challenges," *ELCVIA: Electronic Letters on Computer Vision and Image Analysis,* Vol.19, No.3, pp.18-37, 2020.

[16] W. C. Kang and J. McAuley, "Candidate generation with binary codes for large-scale top-n recommendation," in *Proceedings of the 28th ACM International Conference on Information and Knowledge Management,* pp.1523-1532, 2019.

[17] Y. Deldjoo, M. Schedl, P. Cremonesi, and G. Pasi, "Recommender systems leveraging multimedia content," *ACM Computing Surveys (CSUR),* Vol.53, No.5, pp.1-38, 2020.

[18] P. Covington, J. Adams, and E. Sargin, "Deep neural networks for youtube recommendations," in *Proceedings of the 10th ACM Conference on Recommender Systems,* pp.191-198, 2016.

[19] C. Chen, M. Zhang, Y. Zhang, Y. Liu, and S. Ma, "Efficient neural matrix factorization without sampling for recommendation," *ACM Transactions on Information Systems (TOIS),* Vol.38, No.2, pp.1-28, 2020.

[20] B. Zhu, F. Ortega, J. Bobadilla, and A. Gutiérrez, "Assigning reliability values to recommendations using matrix factorization," *Journal of Computational Science,* Vol.26 pp.165-177, 2018.

[21] J. Bai et al., "Personalized bundle list recommendation," in *The World Wide Web Conference,* pp.60-71, 2019.

[22] X. Xin, X. He, Y. Zhang, Y. Zhang, and J, Jose, "Relational collaborative filtering: Modeling multiple item relations for recommendation," in *Proceedings of the 42nd international ACM SIGIR Conference on Research and Development in Information Retrieval*, pp.125-134, 2019.

[23] J. Feng, Z. Xia., X. Feng, and J. Peng, "RBPR: A hybrid model for the new user cold start problem in recommender systems," *Knowledge-Based Systems,* Vol.214, pp.106732, 2021.

[24] D. Jiang, Z. Liu, L. Zheng, and J. Chen, "Factorization meets neural networks: A scalable and efficient recommender for solving the new user problem," *IEEE Access,* Vol.8, pp.18350-18361, 2020.

[25] V. W. Anelli et al., "Elliot: A comprehensive and rigorous framework for reproducible recommender systems evaluation," in *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp.2405-2414, 2021.

[26] L. Zhang, Q. Wei, L. Zhang, B. Wang, and W. H. Ho, "Diversity balancing for two-stage collaborative filtering in recommender systems," *Applied Sciences*, Vol.10, No.4, pp.1257, 2020.

[27] Y. Cao, X. Chen, L. Yao, X. Wang, and W. E. Zhang, "Adversarial attacks and detection on reinforcement learning-based interactive recommender systems," In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval,* pp.1669-1672, 2020.

[28] F. Rezaimehr and C. Dadkhah, "A survey of attack detection approaches in collaborative filtering recommender systems," *Artificial Intelligence Review,* Vol.54, No.3, pp.2011-2066, 2021.

[29] U. Javed et al., "A review of content-based and context-based recommendation systems," *International Journal of Emerging Technologies in Learning (iJET),* Vol.16, No.3, pp.274- 306, 2021.

[30] D. Jannach, A. Manzoor, W. Cai, and L. Chen, "A survey on conversational recommender systems," *ACM Computing Surveys (CSUR)*, Vol.54, No.5, pp.1-36, 2021.

[31] F. Narducci, P. Basile, M. de Gemmis, P. Lops, and G. Semeraro, "An investigation on the user interaction modes of conversational recommender systems for the music domain," *User Modeling and User-Adapted Interaction*, Vol.30, No.2, pp.251-284, 2020.

[32] I. Fernández-Tobías, M. Braunhofer, M. Elahi, F. Ricci, and I. Cantador, "Alleviating the new user problem in collaborative filtering by exploiting personality information," *User Modeling and User-Adapted Interaction*, Vol.26, No.2, pp.221-255, 2016.

[33] G. Shani and A. Gunawardana, "Evaluating recommendation systems," in *Recommender Systems Handbook1st ed.* USA: Springer, ch. 8, pp.257297, 2011.

[34] G. Guo et al., "Resolving data sparsity by multi-type auxiliary implicit feedback for recommender systems," *Knowledge-Based Systems,* Vol.138, pp.202-207, 2017.

[35] N. F. Al-Bakri and S. H. Hashim, "Reducing data sparsity in recommender systems," *Al-Nahrain Journal of Science,* Vol.2, No.2, pp.138-147, 2018.

[36] S. Ahmadian, M. Afsharchi, and M. Meghdadi, "A novel approach based on multi-view reliability measures to alleviate data sparsity in recommender systems," *Multimedia Tools and Applications*, Vol.78, No.13, pp.17763-17798, 2019.

[37] G. Guo, "Integrating trust and similarity to ameliorate the data sparsity and cold start for recommender systems," in *Proceedings of the 7th ACM Conference on Recommender Systems*, pp.451-454, 2013.

[38] Idrissi, Nouhaila, and Ahmed Zellou, "A systematic literature review of sparsity issues in recommender systems," *Social Network Analysis and Mining,* Vol.10, No.1, pp.1-23, 2020.

[39] J. F. G. da Silva, N. N. de Moura Junior, and L. P. Caloba, "Effects of data sparsity on recommender systems based on collaborative filtering," *International Joint Conference on Neural Networks (IJCNN), IEEE*, pp.1-8, 2018.

[40] G. Guo, "Improving the performance of recommender systems by alleviating the data sparsity and cold start problems," *Twenty-Third International Joint Conference on Artificial Intelligence,* 2013.

[41] X. Dong, L. Yu, Z. Wu, Y. Sun, L. Yuan, and F. Zhang "A hybrid collaborative filtering model with deep structure for recommender systems," *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol.31, No.1, 2017.

[42] D. Anand and K. K. Bharadwaj, "Utilizing various sparsity measures for enhancing accuracy of collaborative recommender systems based on local and global similarities," *Expert Systems with Applications,* Vol.38, No.5, pp.5101-5109, 2011.

[43] L. Sweeney, "k-anonymity: A model for protecting privacy," *International Journal of Uncertainty, Fuzziness and Knowledge-based System,* Vol.10, No.5, pp.557-570, 2002.

[44] B. C. Fung, K. Wang, and S. Y. Philip, "Anonymizing classification data for privacy preservation," *IEEE Transactions on Knowledge and Data Engineering,* Vol.19, No.5, pp.711-725, 2007.

[45] A. Machanavajjhala, D. Kifer, J. Gehrke, and M. Venkitasubramaniam, "l-diversity: Privacy beyond k-anonymity," *ACM Transactions on Knowledge Discovery from Data (TKDD),* Vol.1, No.1, pp.3-es, 2007.

[46] N. Li, T. Li, and S. Venkatasubramanian, "t-closeness: Privacy beyond k-anonymity and l-diversity," *In IEEE 23rd International Conference on Data Engineering, IEEE,* pp.106-115, 2006.

[47] N. Li, T. Li, and S. Venkatasubramanian, "Closeness: A new privacy measure for data publishing," *IEEE Transactions on Knowledge and Data Engineering,* Vol.22, No.7, pp.943-956, 2009.

[48] V. W. Anelli et al., "Pursuing privacy in recommender systems: The view of users and researchers from regulations to applications," *In Fifteenth ACM Conference on Recommender Systems,* pp.838-841, 2021.

[49] R. Bosri, M. S. Rahman, M. Z. A. Bhuiyan, and A. Al Omar, "Integrating blockchain with artificial intelligence for privacy-preserving recommender systems," *IEEE Transactions on Network Science and Engineering,* Vol.8, No.2, pp.1009-1018, 2020.

[50] L. Belli et al., "Privacy-aware recommender systems challenge on twitter's home timeline," a*rXiv preprint arXiv,* pp.13715, 2020.

[51] I. Mazeh and E. Shmueli, "A personal data store approach for recommender systems: Enhancing privacy without sacrificing accuracy," *Expert Systems with Applications,* Vol.139, pp.112858, 2020.

[52] X. Chi, C. Yan, H. Wang, W. Rafique, and L. Qi, "Amplified locality-sensitive hashing-based recommender systems with privacy protection," *Concurrency and Computation: Practice and Experience,* Vol.34, No.14, pp.e5681, 2022.

[53] T. B. Ogunseyi, T. Bo, and C. Yang, "A privacy-preserving framework for cross-domain recommender systems," *Computers & Electrical Engineering,* Vol.93, pp.107213, 2021.

[54] D. Pramod, "Privacy-preserving techniques in recommender systems: State-of-the-art review and future research agenda," *Data Technologies and Applications ahead-of-print,* 2022.

[55] M. Srifi, A. Oussous, A. Ait Lahcen, and S. Mouline, "Recommender systems based on collaborative filtering using review texts—a survey," *Information*, Vol.11, No.6, pp.317, 2020.

[56] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, "Application of dimensionality reduction in recommender system-a case study," Minnesota Univ Minneapolis Dept of Computer Science, 2000.

[57] B. M. Sarwar, “Sparsity, scalability, and distribution in recommender systems,” Ph.D. disertation, University of Minnesota, 2001.

[58] N. Bhalse and R. Thakur, “Algorithm for movie recommendation system using collaborative filtering,” *Materials Today: Proceedings*, 2021.

[59] S. Sharma, V. Rana, and M. Malhotra, “Automatic recommendation system based on hybrid filtering algorithm,” *Education and Information Technologies*, Vol.27, No.2, pp.1523-1538, 2022.

[60] B. Yi et al., “Deep matrix factorization with implicit feedback embedding for recommendation system,” *IEEE Transactions on Industrial Informatics,* Vol.15, No.8, pp.4591-4601, 2019.

[61] F. Yang, “A hybrid recommendation algorithm–based intelligent business recommendation system,” *Journal of Discrete Mathematical Sciences and Cryptography*, Vol.21, No.6, pp.1317-1322, 2018.

[62] L. Jiang, Y. Cheng, L. Yang, J. Li, H. Yan, and X. Wang, “A trust-based collaborative filtering algorithm for E-commerce recommendation system,” *Journal of Ambient Intelligence and Humanized Computing*, Vol.10, No.8, pp.3023-3034, 2019.

[63] S. Amara and R. R. Subramanian, “Collaborating personalized recommender system and content-based recommender system using TextCorpus,” *6th International Conference on Advanced Computing and Communication Systems (ICACCS). IEEE*, pp.105-109, 2020.

[64] C. S. Wang, B. S. Chen, and J. H. Chiang, “TDD-BPR: The topic diversity discovering on Bayesian personalized ranking for personalized recommender system,” *Neurocomputing,* Vol.441, pp.202-213, 2021.

[65] I. Kangas, M. Schwoerer, and L. J. Bernardi, “Recommender systems for personalized user experience: lessons learned at Booking. com,” *In Fifteenth ACM Conference on Recommender Systems*, pp.583-586, 2021.

[66] J. Bobadilla, F. Ortega, A. Gutiérrez, and S. Alonso, “Classification-based deep neural network architecture for collaborative filtering recommender systems,” 2020.

[67] U. Javed, K. Shaukat, I. A. Hameed, F. Iqbal, T. M. Alam, and S. Luo, “A review of content-based and context-based recommendation systems,” *International Journal of Emerging Technologies in Learning (iJET)*, Vol.16, No.3, pp.274-306, 2021.

[68] R. Singla, S. Gupta, A. Gupta, and D. K. Vishwakarma, “FLEX: a content based movie recommender,” *In International Conference for Emerging Technology (INCET), IEEE,* pp.1-4, 2020.

[69] L. David, A. Thakkar, R. Mercado, and O. Engkvist, “Molecular representations in AI-driven drug discovery: A review and practical guide,” *Journal of Cheminformatics*, Vol.12, No.1, pp.1-22, 2020.

Cho Nwe Zin Latt

https://orcid.org/0000-0002-4258-9509
e-mail : chocho1612@pukyong.ac.kr

She received her M.S. degree from Pukyong National University. She is currently a Ph.D. student in the Laboratory of Information Security and Internet Applications (LISIA) at Pukyong National University. Her research interests includes applied cryptography, blockchain, security and privacy protection in recommendation system and supply chain.

Muhammad Firdaus

https://orcid.org/0000-0003-0104-848X
e-mail : mfirdaus@pukyong.ac.kr

He received his M.S. degree in Telecommunication Engineering from Institut Teknologi Bandung (ITB), Indonesia. He is currently a Ph.D. student in the Laboratory of Information Security and Internet Applications (LISIA) at Pukyong National University. His research interests include applied cryptography, blockchain, security and privacy protection in wireless communications and networking.

Mariz Aguilar

https://orcid.org/0000-0002-3710-4749
e-mail : marizaguilar@pukyong.ac.kr

She received a B.S degree in the Department of Information Technology, Quezon City University, Philippines in 2014. She is now taking her Master's Degree in Information Security, Pukyong National University, South Korea. Her research interests include Blockchain, IoT Security, and Supply Chain.

Kyung-Hyune Rhee

https://orcid.org/0000-0003-0466-8254

e-mail : khrhee@pknu.ac.kr

He received his M.S. and Ph.D. degrees from Korea Advanced Institute of Science and Technology (KAIST), Republic of Korea in 1985 and 1992, respectively. He worked as a senior researcher in Electronic and Telecommunications Research Institute (ETRI), Republic of Korea from 1985 to 1993. He also worked as a visiting scholar in the University of Adelaide, the University of Tokyo, and the University of California, Irvine. He has served as a Chairman of Division of Information and Communication Technology, Colombo Plan Staff College for Technician Education in Manila, the Philippines. He is currently a professor in the Department of IT Convergence and Application Engineering, Pukyong National University, Republic of Korea. His research interests center on security and the evaluation of blockchain technology, key management and its applications, and AI-enabled security evaluation of cryptographic algorithms.