

VIETNAM NATIONAL UNIVERSITY HO CHI MINH CITY
UNIVERSITY OF SCIENCE
FACULTY OF INFORMATION TECHNOLOGY



REPORT

Topic: RECOMMENDER SYSTEM

Course: Computational Thinking

Students:

Nguyễn Thanh Trúc (24127137)

Đào Minh Khoa (24127422)

Trần Lưu Gia Bảo (24127018)

Nguyễn Tiến Cường (24127337)

Nguyễn Khánh Toàn (24127252)

Phạm Trần Anh Quân (24127226)

Supervisors:

PhD. Nguyễn Tiến Huy

PhD. Lê Thanh Tùng

Mr. Trần Hoàng Quân

Ngày 9 tháng 11 năm 2025

Mục lục

1	Abstract	1
2	Introduction	1
2.1	About recommendation system	1
3	Traditional Recommendation System Problems	1
4	Analysis of Personalized Recommender Systems	1
4.1	Collaborative Filtering (CF)	1
4.1.1	Memory-Based CF	2
4.1.2	Model-Based CF	2
5	Deep Learning-Based Recommendation Models	3
6	Other Recommender Systems	4
6.1	Explainable recommender system	4
6.2	Context-Aware-Based CF	4
6.3	Knowledge-Based Systems	4
6.4	Demographic recommender system	5
6.5	Reciprocal recommender systems	5
6.6	Group recommender systems	5
7	Challenges in Recommender System Design	5
7.1	System Robustness	5
7.2	Scalability	5
7.3	Data Bias	5
8	Similarity Measures	5
9	Conclusion	5

1 Abstract

This article presents an overview of recommendation systems, focusing on their components, classification, main techniques, applications, and a comparison of different approaches. Recommendation systems are essential tools in various domains, including e-commerce, entertainment, and social media, as they help users discover relevant content based on their preferences and behaviors.

2 Introduction

2.1 About recommendation system

What is it?

Recommendation systems address the issue of information overload. The systems personalize user experiences by suggesting items such as products, services, or content that align with individual preferences. Therefore, enhance user experiences and increase engagement and satisfaction. Recommendation systems are divided into three types: Content-based filtering, Collaborative filtering, and Hybrid methods. Recommendation systems are now have the knowledge-based

3 Traditional Recommendation System Problems

This section gives the definition for some common problems in traditional recommendation systems that we will discuss later, including:

- Cold Start Problem: Difficulty in making accurate recommendations for new users or items due to lack of historical data.
- Data Sparsity: The challenge of making recommendations when user-item interaction data is sparse.
- Scalability: The ability to efficiently handle large-scale datasets

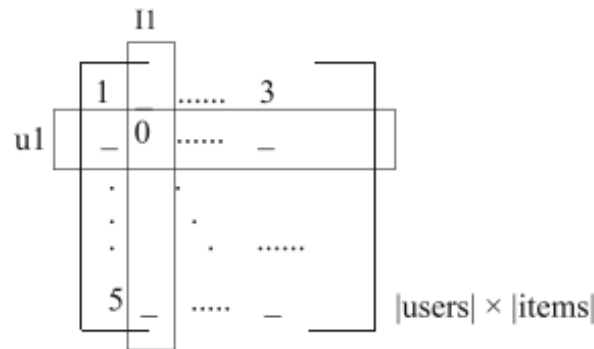
4 Analysis of Personalized Recommender Systems

4.1 Collaborative Filtering (CF)

Driven by user profiles or historical interactions

4.1.1 Memory-Based CF

This method get input is a matrix of user-item interactions (e.g., ratings, clicks, purchases). using cosine similarity or Pearson correlation to find similar users or items. Figure 1 illustrates the user-



Hình 1: Memory-Based Collaborative Filtering

item interaction matrix, where rows represent users and columns represent items. The values in the matrix indicate the interactions (e.g., ratings) between users and items. Memory-based collaborative filtering utilizes this matrix to find similar users or items based on their interactions.

The method is simple and popular but it faces challenges such as sparsity in the interaction matrix and computational complexity in extensive user or item domains.

A general solution to sparsity and high-dimensionality involves transforming high-dimensional sparse vectors into low-dimensional dense ones, a process known as feature embedding or representation learning.

- **User-based:** Measures similarity between the target user and all the other users.
- **Item-based:** Measures the similarity between the item that a target user is interested in and all the other items.

In both case, if similarity above a threshold, it then categorized as neighbor. The output of memory-based CF is a prediction, $\hat{r}_{u,i}$, for an unrated item i by a target user u , typically derived as a weighted average of neighbor ratings.

4.1.2 Model-Based CF

Uses machine learning, deep learning, or data mining algorithms to build a predictive model for recommendations. Able to capture complex relationships.

- **Clustering-Based Models:** Use unsupervised learning like K-Nearest Neighbor (KNN) or Locality-Sensitive Hashing (LSH) to group similar users or items, making the system more scalable.
- **Matrix Factorization (MF)-Based Models:** Decompose the user-item utility matrix into lower-dimensional matrices representing latent factors (e.g., SVD, PMF, NMF) to uncover dependencies and user-item structures.
- **Deep Learning-Based Models:** Utilize neural networks (e.g., Autoencoders, CNNs, RNNs) to learn complex patterns in user-item interactions, capturing non-linear relationships for improved recommendations.

5 Deep Learning-Based Recommendation Models

Since learning-based models outperform neighborhood-based models, Deep Neural Networks (DNNs) have drastically increased the research on recommender systems. It can capture complex user-item relationships.

For the input, feature vectors are created for users and items, which may include demographic information, item attributes, and historical interactions.

The network outputs a prediction score or vector that represents the probability of a user interacting with or preferring an item.

Deep learning-based recommendation models include:

- **Multilayer perceptron-based recommendation:**
- **Autoencoders:** Neural networks that learn efficient data representations by encoding and decoding input data. In recommender systems, they can reconstruct user-item interaction matrices to predict missing values.
- **Convolutional Neural Networks (CNNs):** Primarily used for image data, CNNs can also be applied to recommender systems by extracting features from item images or user profiles to enhance recommendations.
- **Recurrent Neural Networks (RNNs):** Designed for sequential data, RNNs can model user behavior over time,

- **Restricted boltzmann machines-based recommendations:** RBM is a generative stochastic neural network that can learn a probability distribution over its set of inputs. In recommender systems, RBMs can model user-item interactions to predict user preferences.
- **Deep reinforcement learning-based recommendations:** This approach uses reinforcement learning techniques to optimize recommendation strategies based on user feedback and interactions over time.
- **Adversarial network-based recommendation:** This method employs generative adversarial networks (GANs) to improve recommendation quality by generating realistic user-item interactions and enhancing the model's ability to capture complex patterns.
- **Graph neural network-based recommender systems:** GNNs leverage graph structures to model relationships between users and items, capturing complex interactions and improving recommendation accuracy.

6 Other Recommender Systems

6.1 Explainable recommender system

6.2 Context-Aware-Based CF

enhances recommendation quality by incorporating contextual information. provide more relevant and timely suggestions by using dynamic information like time, location, weather.

6.3 Knowledge-Based Systems

Knowledge-Based recommender systems are uniquely suited for high-consideration, infrequent purchase domains (e.g., automobiles, real estate) where traditional methods falter. This category employs two main methodologies:

- **Knowledge Graph Embedding (KGE):** This method encodes external knowledge from sources like knowledge graphs into higher-level, dense vector representations.
- **Path-Based RS:** This method leverages connectivity patterns, known as meta-paths, within a knowledge graph. By analyzing these paths, the system can evaluate the semantic similarity between entities

6.4 Demographic recommender system

6.5 Reciprocal recommender systems

6.6 Group recommender systems

7 Challenges in Recommender System Design

7.1 System Robustness

7.2 Scalability

7.3 Data Bias

8 Similarity Measures

Similarity measures are fundamental to certain recommendation approaches, particularly memory-based collaborative filtering, where they are used to identify neighboring users or items. The most fundamental of these are the Pearson Correlation Coefficient and Cosine Similarity. Pearson Correlation Coefficient (PCC): This measures the linear correlation between the ratings of two users (u and v) on their co-rated items ($I_{u,v}$), defined as: Cosine Similarity (COS): This measures the cosine of the angle between two user rating vectors in the item space, defined

9 Conclusion