Computer Science

Heterogeneous Collaborative Filtering for Recommender System Case Study at Business

Vũ Hồng Quân

TABLE OF CONTENTS

01

INTRODUCTION

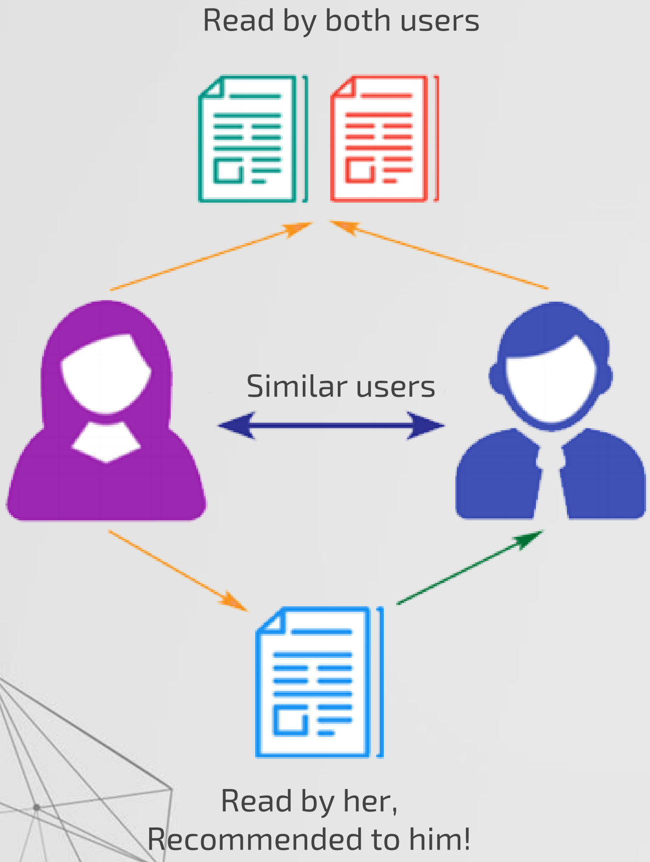
- Definition
- Taxonomy

WHAT IS RECOMMENDER SYSTEM ?????

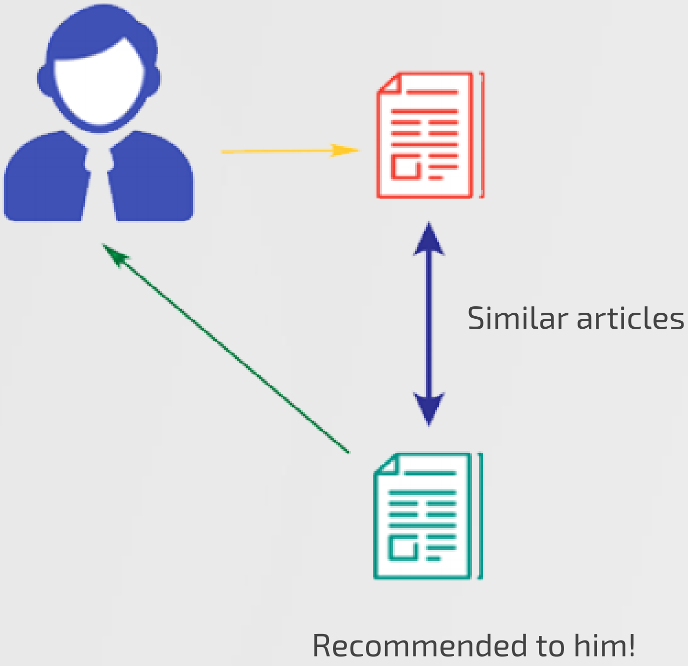
Taxonomy of recommender system

1. Domain
2. Purpose
3. Context
4. Personalization level
5. Whose opinions
6. Privacy and trustworthinesse
7. Interface
8. Algorithms

COLLABORATIVE FILTERING



CONTENT-BASED FILTERING



02

OBJECTIVES

Literature review
Contribution

Literature Review

Kind 1

Only utilize the rating information between users and items but are not concerned about other useful aspects.

Kind 2

Employs ineffective methods to compute the similarities underneath aspects.

Kind 3

Jointly factorizes multiple behavior matrices based on extending the matrix factorization (MF) method.

Kind 4

Approaches the problem from the perspective of learning.

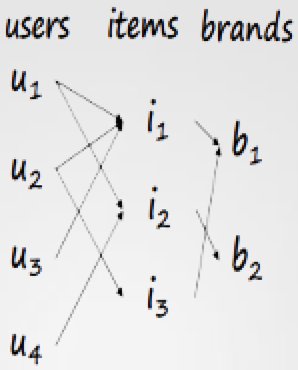
The importance of different aspects

	I_1	I_2	I_3
U_1	✓	✓	
U_2	✓		✓
U_3		✓	
U_4	✓	?	?

User-Item purchase relation

	B_1	B_2
I_1	✓	
I_2		✓
I_3	✓	

Item-Brand relation



Heterogeneous network

Capturing visitor behaviors

- Consumer browses
- Consumer becomes interested in one or more products
- Consumer adds product to basket or a list with the intent to buy
- Consumer buys products
- Consumer consumes product
- Consumer rates the product
- Consumer resells product or otherwise disposes of it

1
2
3
4
5
6
7
Rate this!
We buy used...
Used

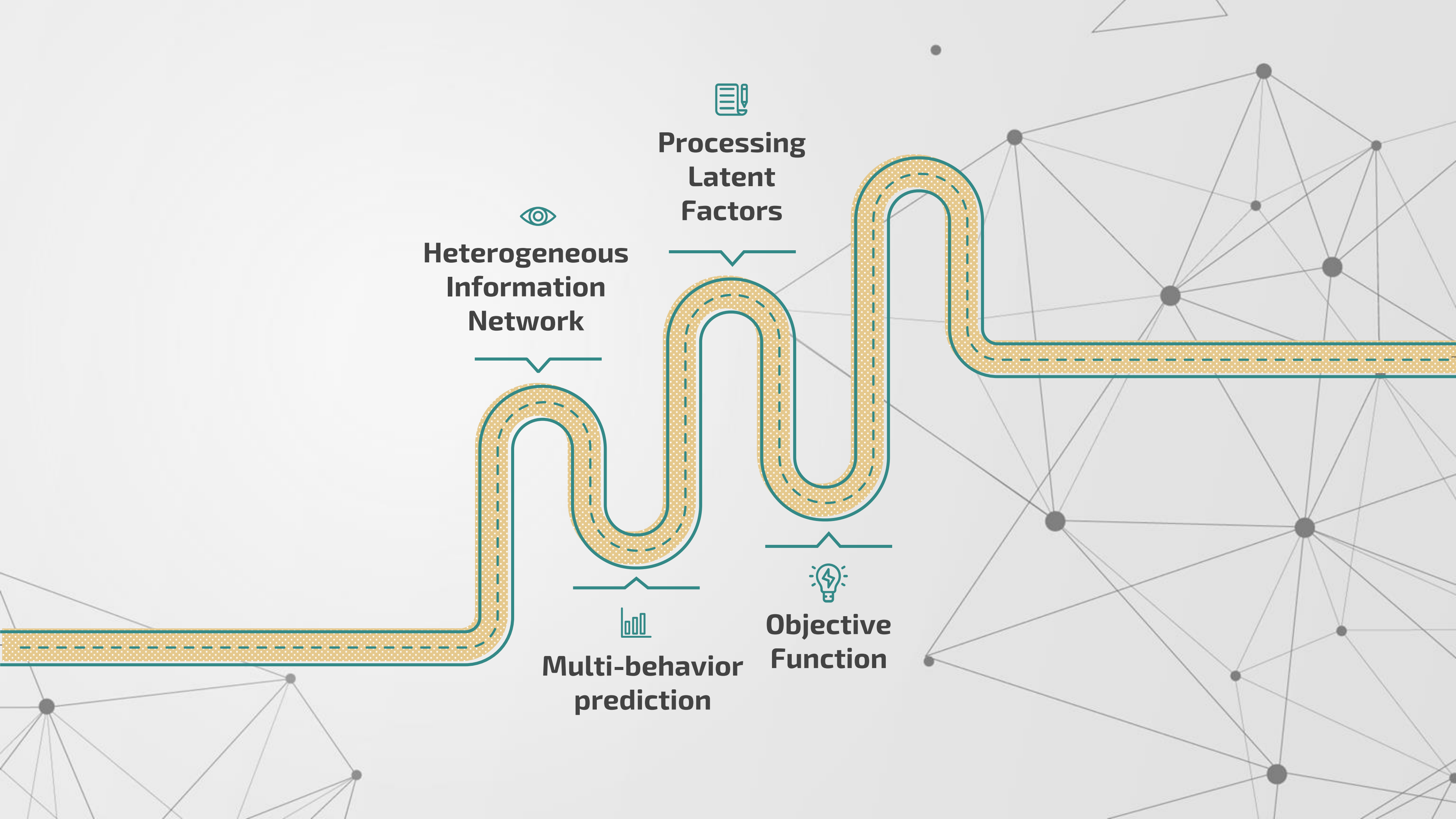
Contribution

This work proposes a new model named Heterogeneous Neural Collaborative Filtering (HNCF) which combines the two important factors: multi-aspect information and the translational relationships between different behaviors.

- From a defined Heterogeneous Information Network, the model first determines the meta paths based on the structure of the dataset
- Use PathSim that captures the subtlety of peer similarity to compute the similarity matrices between users with other users and items with other items.
- Aspect-level latent factors are learned through a standard Multi-Layer Perceptron then fused by the attention mechanism.
- This model focuses on aggregating these result factors with multi-behaviors prediction layers to improve the performance.

03

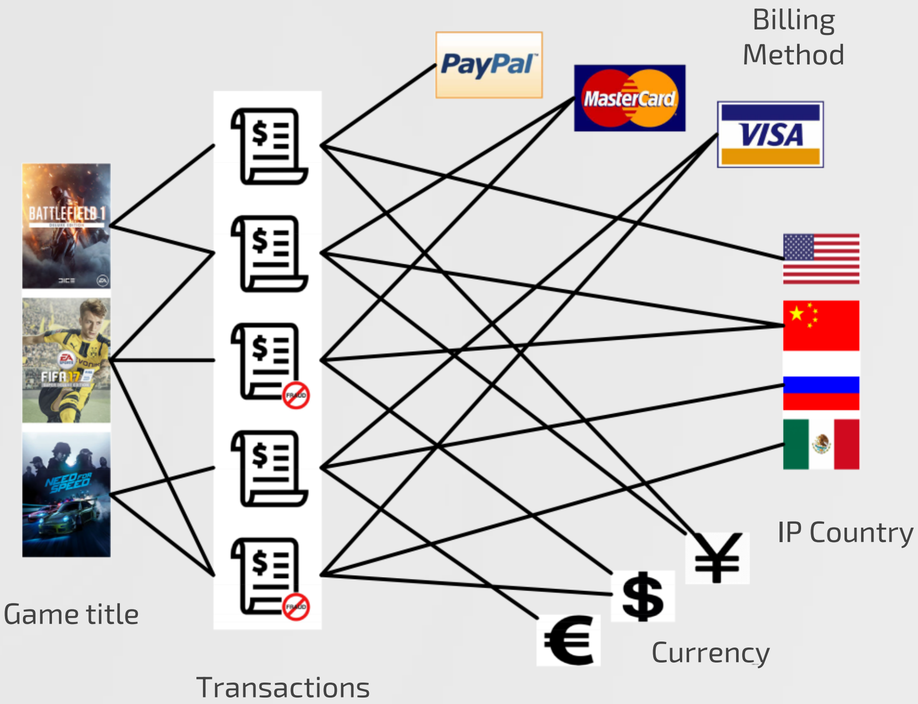
METHODOLOGY


Heterogeneous Information Network
Processing Latent Factors
Multi-behavior prediction
Objective Function

Heterogeneous Information Network

Firstly, the model compute the similarity between users and user, items and items through Heterogeneous Information Network (HIN).

The recommendation domain comprises of various object types (e.g., billing method, country, currency, and title in E-commerce recommendation) with rich relations among them, which establish a natural HIN.



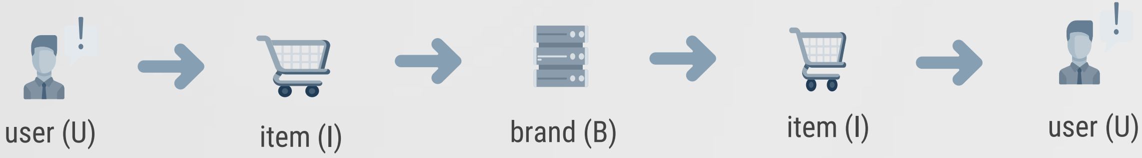
Definition

Definition 1: Information Network. An information network is defined as a directed graph G = (V, E) with an object type mapping function $\varphi : V \rightarrow A$ and a link type mapping function $\psi : E \rightarrow R$, where each object $v \in V$ belongs to one particular object type $\varphi(v) \in A$, and each link $e \in E$ belongs to a particular relation $\psi(e) \in R$

Definition 1.2: Heterogeneous/Homogeneous information network. The information network is called a heterogeneous information network if the types of objects $|A| > 1$ or the types of relations $|R| > 1$; otherwise, it is a homogeneous information network.

Definition 2: Meta path. A meta path P is a path defined on the graph of network schema TG = (A, R), and is denoted in the form of $A_1 \rightarrow A_2 \rightarrow \ldots \rightarrow A_l$ which defines a respectively composite relation $R = R_1 \circ R_2 \circ \ldots \circ R_l$ between type A_1 and A_l, where $\circ$ denotes the composition operator on relations.

(UCU) is a meta path



(UIBIU) is a meta path

PathSim

Given a symmetric meta path P, PathSim between two objects of the same type x and y is:

$$s(x,y)=\frac{2\times|p(x,y)|}{|p(x,x)|+|p(y,y)|}$$

where p(x,y) is a path instance between x and y, p(x,x) is that between x and x, and p(y,y) is that between y and y.

Let define M as the commuting matrix for a meta path $P=(A_1,A_2,\dots,A_l)$
$M=W_{A_1A_2}W_{A_2A_3}\dots W_{A_{l-1}A_l}$ where $W_{A_iA_{i+1}}$ is the adjacency between two types A_i and A_{i+1}. From that, $s(x_i,x_j)=\frac{2M_{ij}}{M_{ii}+M_{jj}}$ is PathSim between two objects x_i, x_j

Adjacency matrix W_{UC}

	Electronics	Furnitures	Appliances	Accessories
Quang	2	1	0	0
Chung	50	20	0	0
Mai	2	0	1	0
Nghia	2	1	0	0
Quan	0	0	1	1

We have $W_{CU} = (W_{UC})^T \Rightarrow M = W_{UC} W_{CU}$ (UC - CU)

$$M = \begin{bmatrix} 5 & 120 & 4 & 5 & 0 \\ 120 & 2900 & 100 & 120 & 0 \\ 4 & 100 & 5 & 4 & 1 \\ 5 & 120 & 4 & 5 & 0 \\ 0 & 0 & 1 & 0 & 2 \end{bmatrix}$$

The similarity between two users *Quang* and *Chung* underneath meta path (UCU):

$$s(Quang, Chung) = s(x_1, x_2) = \frac{2M_{12}}{M_{11} + M_{22}}$$

$$= \frac{2 \times 120}{5 + 2900} = 0.08261618$$

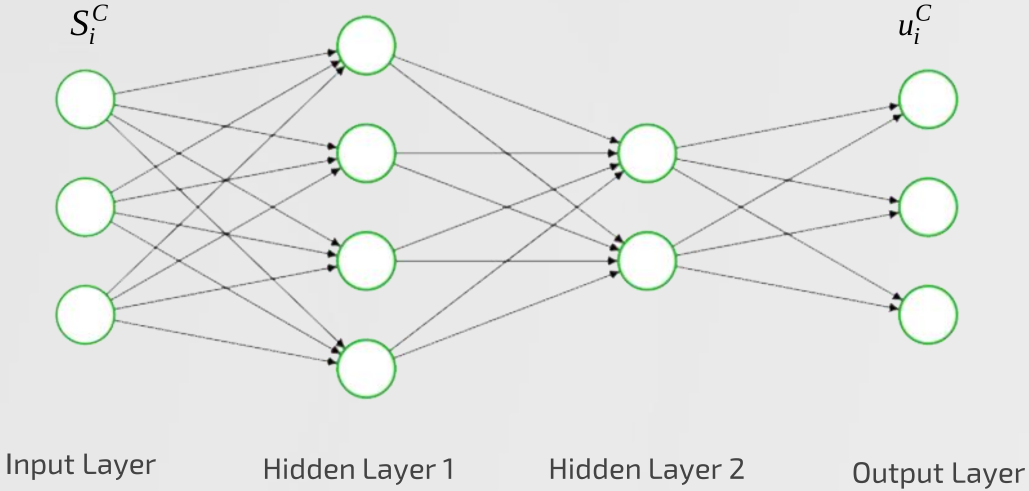
Processing latent factors

Multi-layer Perceptron (MLP)

To have a network that is able to learn different aspect-level latent factors separately, this model employs a Multi-layer Perceptron (MLP) which gives it a large level of flexibility and non-linearity

ReLu is used as the activation function.

Let define S_i^C as the vector that represents the similarity between U_i and all the other users.



Processing latent factors

Multi-layer Perceptron (MLP)

The initial similarity vector between user U_i and other users is projected to a low-dimensional aspect-level latent factor:

$$a_0 = S_i^C$$

$$a_1 = RELU(\mathrm{W}_1^T \times a_0 + b_1)$$

$$a_2 = RELU(\mathrm{W}_2^T \times a_1 + b_2)$$

...

$$u_i^C = RELU(\mathrm{W}_n^T \times a_{n-1} + b_n)$$

Processing latent factors

Attention Mechanism

We need to incorporate aspect-level latent factors together to acquire upholstery factors. The attention mechanism is chosen due to its proven effectiveness in various domains

There are two kinds of attention mechanisms:

- Hard attention (reinforcement learning to learn where to pay attention)
- Soft attention (weighted learning using the backpropagation algorithm)

Given the user's category-aspect latent factor u_i^C we use a two-layer network to compute the attention score s_i^C by the following equation:

$$s_i^C = \mathrm{W}_2^T \times f(\mathrm{W}_1^T \times u_i^C + b_i) + b_2$$

Processing latent factors

Attention Mechanism

Finally, we can apply the Softmax function and a normal ratio to obtain the attention value for the aspect-level latent factors

$$w_i^C = \frac{\exp(s_i^C)}{\sum_{k=1}^{A} \exp(s_i^k)}$$

With each k-aspect latent factor of user u_i , we have attention weights w_i^k respectively, the aggregated latent factor u_i can be calculated by the following equation:

$$u_i = \sum_{k=1}^{A} \mathrm{w}_i^k \times u_i^k$$

Multi-behavior prediction

After aggregating the results, the factors processed together with the multi-behavior prediction layers to give a probability of the interaction between the user U_i and item I_j is the highlight of the model to enhance the performance of the recommender system.

The transfer scheme of two relational behaviors (from h_t to h_k) is formulated as:

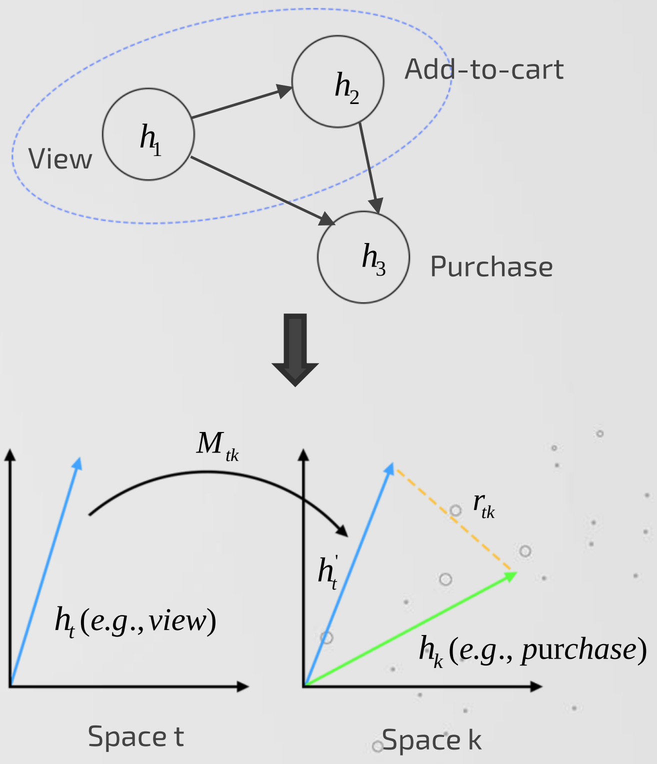
$$f_{h_t \to h_h} = h_t \times M_{tk} + r_{tk}$$

When we aggregate all of the behaviors on a sequence of behaviors to some considered behavior, we obtain the equation:

$$h_k = \sum_t (h_t \times M_{tk} + r_{tk})$$

where t is considered as the precedent behaviors of that k_{th} behavior

Transfer Scheme



Objective function

The probability of the interaction between the user u_i and item v_j is formulated according to the equation

$$y_{k-ij} = sigmoid(h_k \times (u_i \odot v_j)) = \frac{1}{1+\exp(-h_k \times (u_i \odot v_j))}$$

Further assuming that the data points are randomly generated independently of each other, we could write:

$$p(\alpha, \alpha^- \mid w) = \prod_{i,j\in\alpha} y_{ij} \prod_{i,k\in\alpha^-} (1 - y_{ik})$$

$$p(\alpha, \alpha^- \mid w) = \prod_{i,j\in\alpha\cup\alpha^-} (y_{ij})^{y_{ij}} \prod_{i,k\in\alpha^-} (1 - y_{ik})^{1-y_{ij}}$$

Objective function

Loss function respectively to k_{th} behavior:

$$L_k = \sum_{i,j \in \alpha \cup \alpha^-} (y_{k-ij} \times \log y_{k-ij}) + (1 - y_{k-ij}) \times \log(1 - y_{k-ij}))$$

Loss function should be fused by a sum weighted to maintain the different level of behaviors:

$$L = \sum_{k=1}^{K} \lambda_k L_k$$

where $\sum_{k=1}^{K} \lambda_k = 1$

04

RESULTS ANALYSIS

- Dataset and Evaluation
- Compare NeuCF and HNCF
- Analysis of Attention Mechanism
- Compare the number of layers

Dataset and Evaluation

Dataset

This model leverages a real-world E-commerce dataset from a large multi-category online store in the Middle East. The data was collected from October to November 2019.

82533	Interactions
10000	Users
6676	Items
99.88%	Sparsity

Dataset and Evaluation

Evaluation

To study how the model works for the recommender system, the model will apply the ***leave-one-out*** evaluation.

- Keep the final interaction of the user for testing and the remaining for training.
- For each user, randomly gets 99 items that are unrelated to the user as negative samples.

We use both the Hit Ratio (HR) and Normalized Discounted Cumulative Gain (NDCG) to evaluate the model performance. They are defined as:

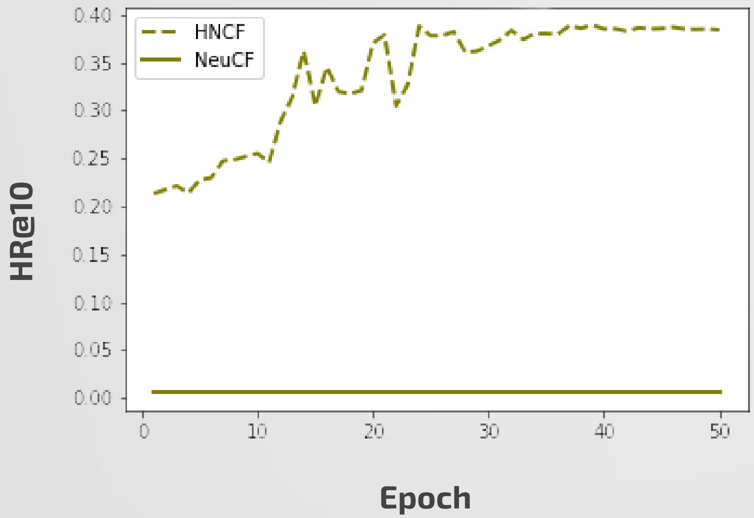
$$HR = \frac{|\mathrm{keys}|}{|\mathrm{users}|}$$

$$NDCG = \frac{1}{|users|} \sum_{i=1}^{|users|} \frac{\log(p_i + 1)}{\log 2}$$

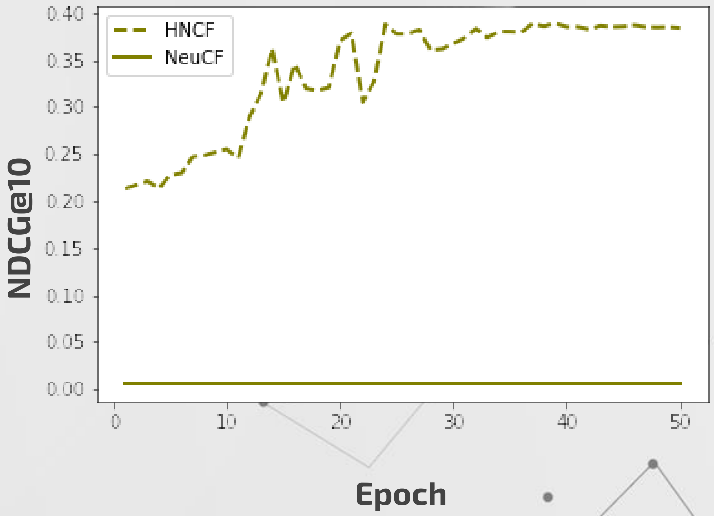
where |keys| is the number of users whose test item appears in the recommended list and p_i is the position of the test item in the list for the i_{th} test user.

Compare NeuCF and HNCF

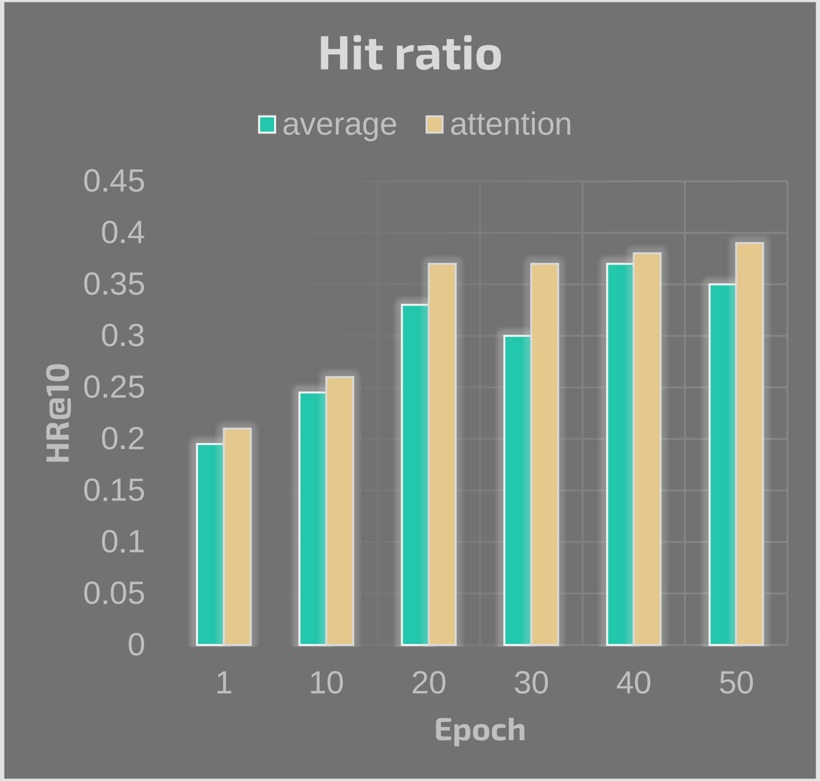
Hit Ratio

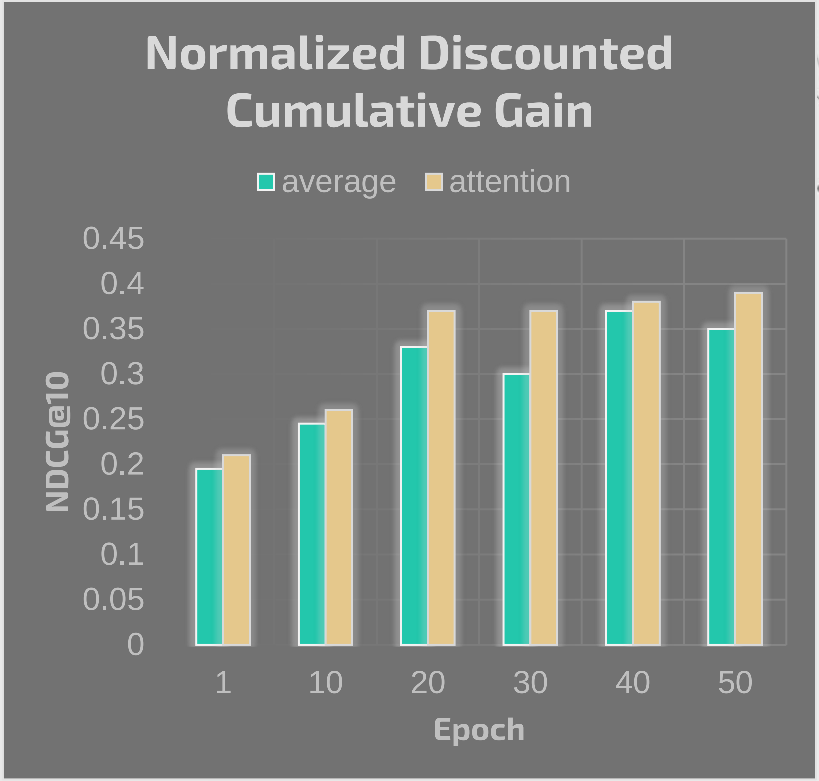


Normalized Discounted Cumulative Gain



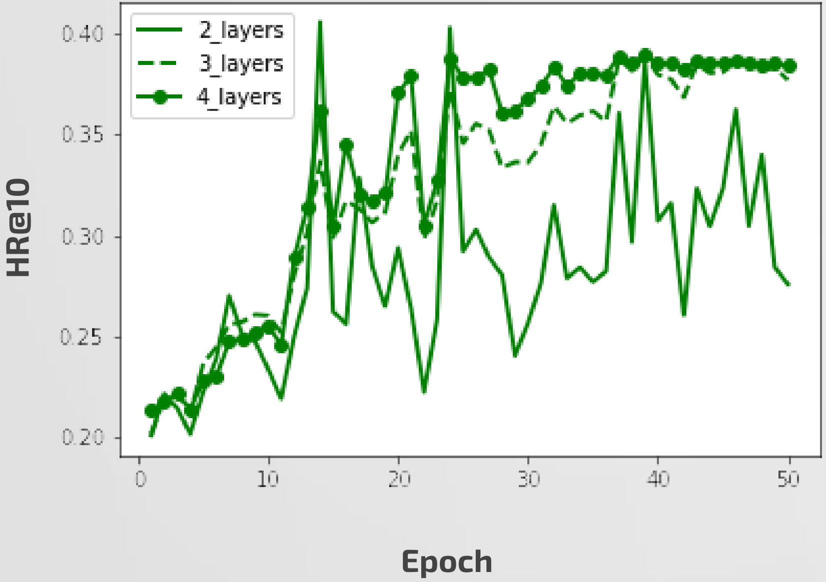
Analysis of Attention Mechanism



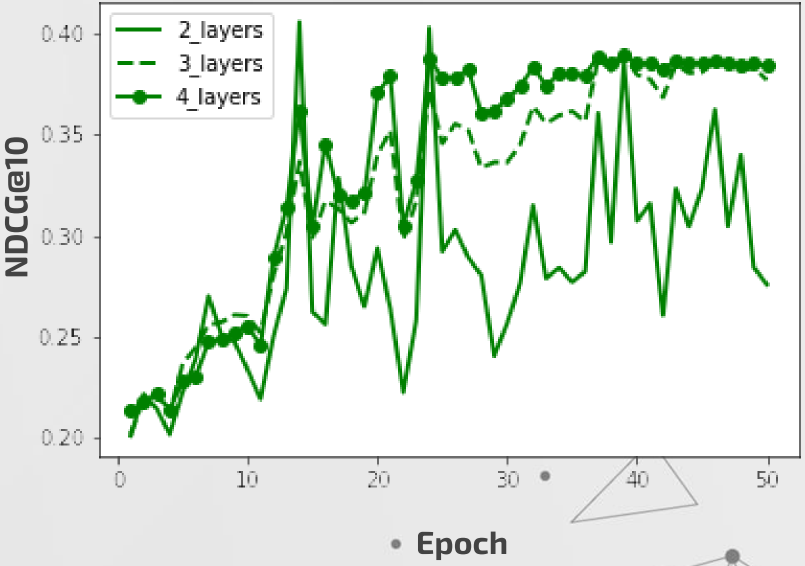


Compare the number of layers

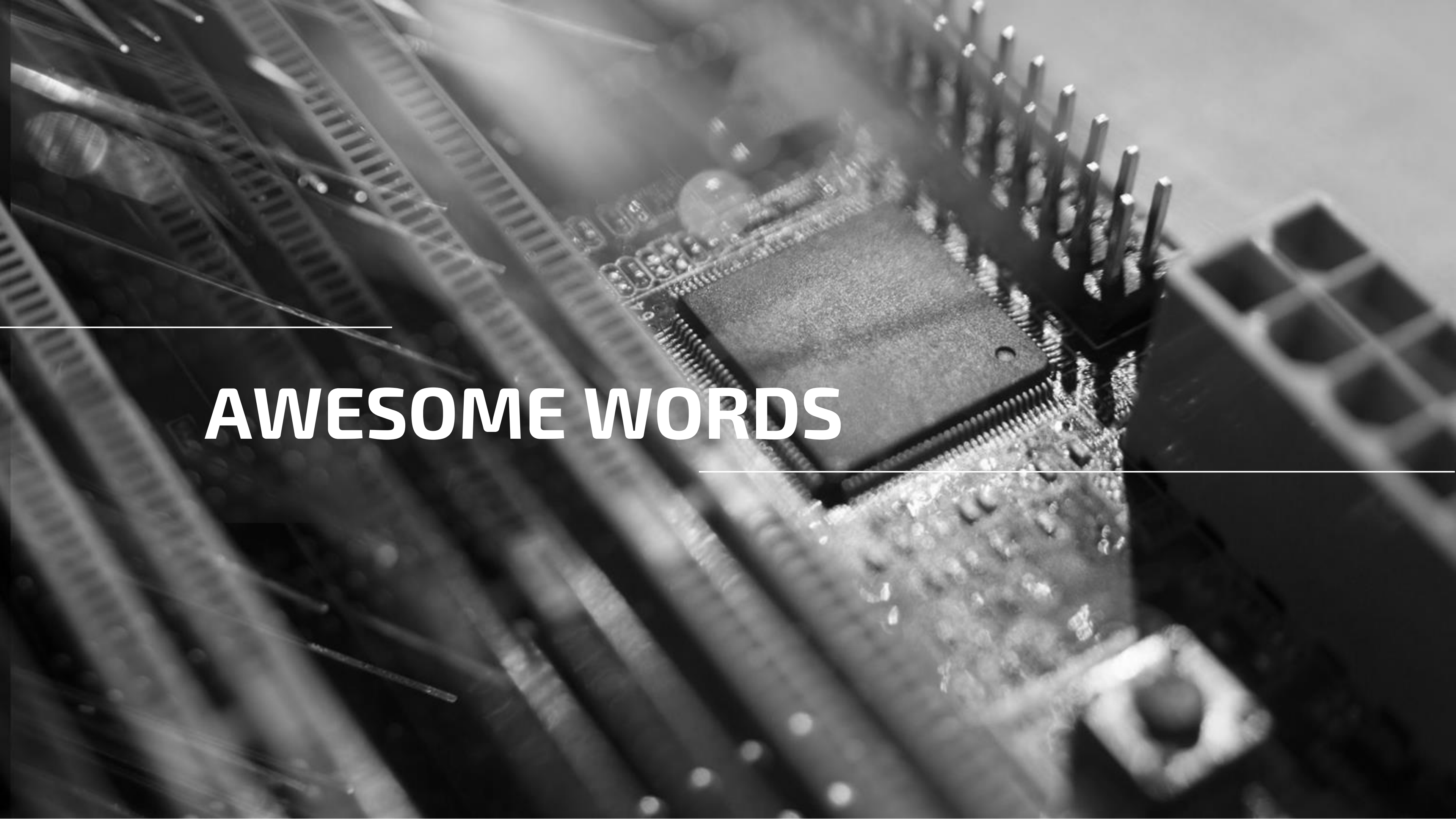
Hit Ratio



Normalized Discounted Cumulative Gain



05
CONCLUSION



AWESOME WORDS

Future

This study can be applied to most e-commerce sites because it solves the basic problems of online shopping such as Cold-start and sparsity. Multi-behaviors property enables it to be applied in the Movie recommendation domain. Besides, techniques of exploiting lots of aspects can be helpful in social networks to suggest friends and content to users.

TIMELINE

Sep
Literature Review

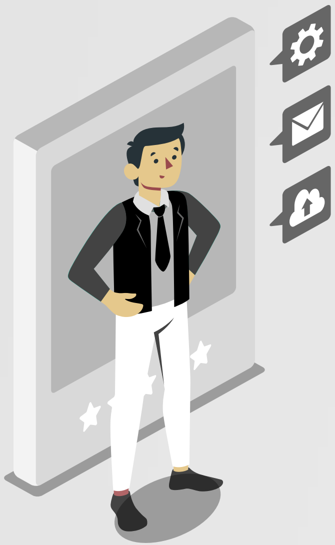
Oct
Propose contributions

Nov
Implemetation and Analysis

Dec
Report and slides

D-Day
Presentation

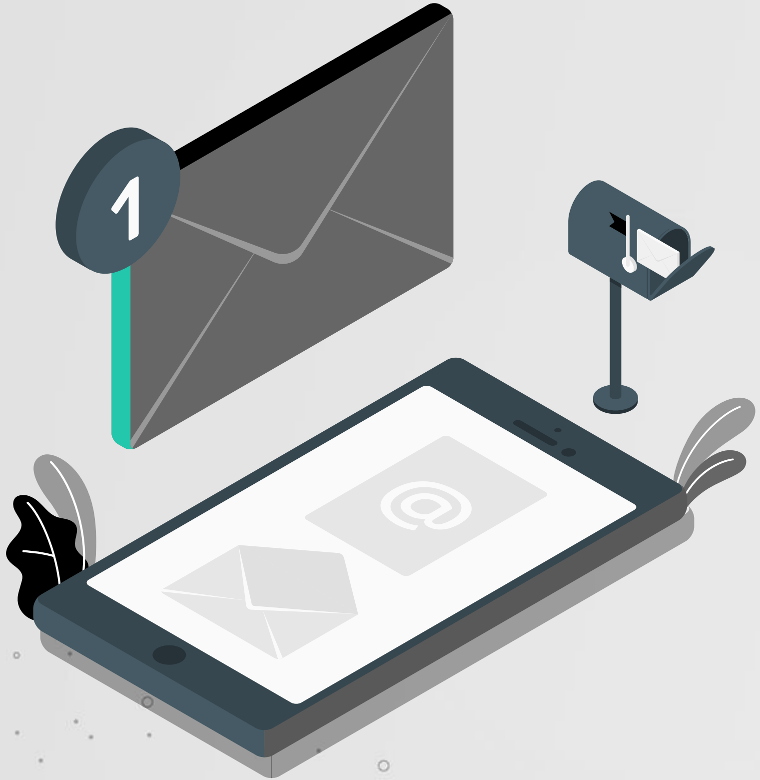
About team



Vũ Hồng Quân
Computer Science

Instructor
Dr. Duy Hùng Phan

Coming soon



THANKS

Does anyone have any questions?

quanvhhe130299@fpt.edu.vn
(+84) 942141869
FPT University