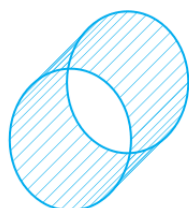


ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN
KHOA TOÁN-TIN HỌC



BÁO CÁO ĐỒ ÁN

HỆ THỐNG TƯ VẤN

Đề tài:

XÂY DỰNG HỆ THỐNG ĐỀ XUẤT ẢNH BẰNG MÔ
HÌNH NEURAL COLABORATIVE FILTERING DỰA
TRÊN BỘ DỮ LIỆU PINTEREST

NHÓM UK - 20KDL

GIẢNG VIÊN
Huỳnh Thanh Sơn

Ngày 9 tháng 1 năm 2024

Danh sách các thành viên nhóm UK

STT	Tên	MSSV
1	Ngô Trương Minh Đạt	20280013
2	Nguyễn Minh Phúc	20280074
3	Nguyễn Thị Hoa	20280033

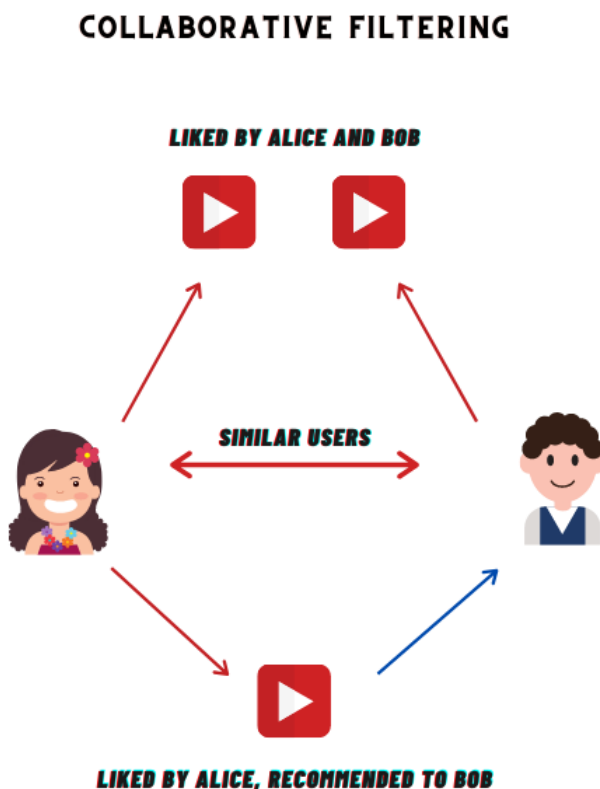
Mục lục

1	Giới thiệu	3
2	Giới thiệu các mô hình Neural Colaborative Filtering	4
2.1	General framework	4
2.2	Mô hình Generalized Matrix Factorization (GMF)	5
2.3	Multi-Layer Perceptron (MLP)	6
2.4	Sự kết hợp của GMF và MLP (Mô hình Neural Matrix Factorization)	7
3	Experiments	9
3.1	Experimental Setup	9
3.1.1	Dataset	9
3.1.2	Evaluator metric - NDCG	9
3.1.3	Training details	10
3.2	Experimental Result	10
4	Kết luận	12

1 Giới thiệu

Hệ thống đề xuất, được thiết kế riêng để phù hợp với sở thích cá nhân và ngày càng trở nên phổ biến trong nhiều dịch vụ trực tuyến khác nhau ví dụ như thương mại điện tử, quảng cáo, và các mạng xã hội. Những hệ thống này chủ yếu tập trung vào việc dự đoán xác suất mà một người dùng sẽ chọn một sản phẩm cụ thể, dựa vào lịch sử tương tác trước đây như việc mua hàng và lượt nhấp chuột.

Phương pháp Lọc Cộng tác (Collaborative Filtering - CF) tiếp cận vấn đề này bằng cách đặt giả thuyết rằng những người dùng có mô hình hành vi tương tự nhau sẽ có xu hướng thích những sản phẩm giống nhau. Việc xác định độ “giống nhau” giữa các users có thể dựa vào mức độ quan tâm (rating) của các users này với các items khác mà hệ thống đã biết trong quá khứ.



Hình 1: Colaborative filtering

Ví dụ: Hai người dùng, A và B, có sở thích chung là xem phim hình sự, đặc biệt là những bộ phim họ đánh giá từ 4 đến 5 sao. Qua việc phân tích lịch sử xem phim của B, ta nhận thấy rằng B rất thích bộ phim "Người phán xử". Từ thông tin này, hệ thống có thể suy luận rằng A cũng có khả

năng sẽ hứng thú với bộ phim này. Do đó, hệ thống sẽ gợi ý "Người phán xử" cho A.

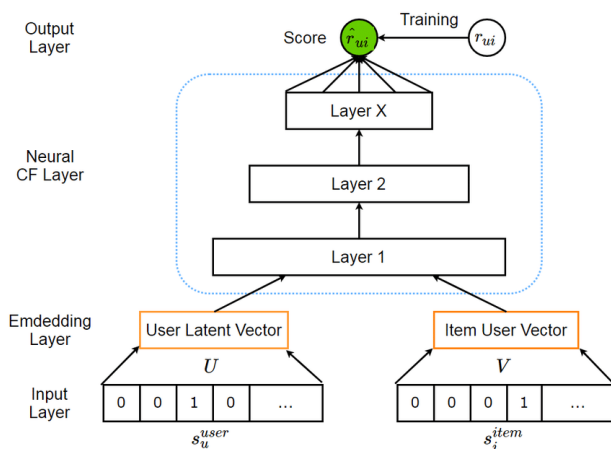
Trong số các phương pháp Collaborative Filtering khác nhau, kỹ thuật Phân rã Ma trận (Matrix Factorization - MF) là một trong những phương pháp được ưa chuộng nhất. Kỹ thuật này biểu diễn user và item trong một latent space chung. Để làm điều này, MF sử dụng một vector đặc trưng tiềm ẩn để mô tả mỗi user hoặc mỗi item. Điều này có nghĩa là cả người dùng và mặt hàng được biểu diễn dưới dạng các vector trong latent space này. Sau đó, sự tương tác của người dùng với một mặt hàng cụ thể được mô hình hóa bằng cách tính toán tích vô hướng của hai vector tiềm ẩn này - một đại diện cho người dùng và một đại diện cho mặt hàng. Tích vô hướng này cung cấp một giá trị số mô tả mức độ tương tác hoặc sự quan tâm dự đoán của user đối với item đó.

Dù Phân rã Ma trận (MF) được đánh giá cao trong lĩnh vực lọc cộng tác, nhưng có ý kiến cho rằng khả năng của nó bị giới hạn do việc sử dụng hàm tương tác đơn giản như tích vô hướng. Cụ thể, khi dự đoán xếp hạng dựa trên phản hồi cụ thể, hiệu suất của mô hình MF có thể tăng lên nếu như tích hợp thêm các yếu tố phân biệt người dùng và sản phẩm vào trong hàm tương tác. Dù chỉ là sự điều chỉnh nhỏ trong toán tử tích vô hướng, nhưng nó lại mở ra khả năng cải thiện đáng kể bằng cách phát triển một hàm tương tác riêng biệt, tinh vi hơn, để mô hình hóa các tương tác giữa đặc trưng tiềm ẩn của người dùng và sản phẩm. Việc sử dụng tích vô hướng, vốn chỉ là sự kết hợp tuyến tính của các đặc trưng tiềm ẩn, có thể chưa đủ để thể hiện đầy đủ độ phức tạp trong dữ liệu tương tác của người dùng.

Trong project này, chúng em sử dụng Deep Neural Network (DNNs) để học hàm tương tác trong hệ thống đề xuất, nhằm xây dựng một hệ thống đề xuất hình ảnh cho người dùng Pinterest.

2 Giới thiệu các mô hình Neural Collaborative Filtering

2.1 General framework



Hình 2: Neural collaborative filtering framework

Sử dụng cách biểu diễn nhiều lớp để mô hình hóa tương tác của user-item y_{ui} , trong đó output của một layer là input của layer tiếp theo. Input dưới cùng (theo Hình 2) là 2 vector đặc trưng \mathbf{v}_u^U và \mathbf{v}_i^I biểu diễn user u và item i . Trong project này, do chú trọng vào việc lọc cộng tác một cách nguyên bản, chúng tôi chỉ sử dụng thông tin cơ bản về người dùng và các mục được chọn làm đặc trưng đầu vào, chuyển đổi chúng thành vector thưa dạng nhị phân thông qua kỹ thuật one-hot encoding.

Phía trên lớp input là lớp embedding, là một fully connected layer, để biến đổi 1 vector thưa về dạng vector dày đặc. User (item) embedding thu được có thể coi là vector ẩn cho user (item) trong bối cảnh của mô hình yếu tố ẩn. User embedding và item embedding được đưa vào một kiến trúc thần kinh nhiều lớp, được gọi là lớp lọc cộng tác thần kinh, để ánh xạ các vector ẩn thành điểm dự đoán. Mỗi lớp trong các lớp Neural Collaborative Filtering có thể được tùy chỉnh để khám phá các cấu trúc ẩn nhất định của tương tác người user-item. Kích thước của lớp ẩn cuối cùng X quyết định khả năng của mô hình. Lớp đầu ra cuối cùng là điểm dự đoán \hat{y}_{ui} , và việc training được thực hiện bằng cách giảm thiểu điểm chênh lệch giữa \hat{y}_{ui} và giá trị mục tiêu của nó y_{ui} .

Mô hình dự đoán của NCF được xây dựng như sau:

$$\hat{y}_{ui} = f(\mathbf{P}^T \mathbf{v}_u^U, \mathbf{Q}^T \mathbf{v}_i^I | \mathbf{P}, \mathbf{Q}, \theta_f)$$

Trong đó $\mathbf{P} \in \mathbb{R}^{M \times K}$ and $\mathbf{Q} \in \mathbb{R}^{N \times K}$, là latent factor matrix của users và items, và θ_f là tham số mô hình của hàm tương tác f . Hàm f được định nghĩa là mạng neural nhiều lớp, nó thể được xây dựng như sau:

$$f(\mathbf{P}^T \mathbf{v}_u^U, \mathbf{Q}^T \mathbf{v}_i^I) = \phi_{out}(\phi_X \dots (\phi_2(\phi_1(\mathbf{P}^T \mathbf{v}_u^U, \mathbf{Q}^T \mathbf{v}_i^I)) \dots))$$

Trong đó ϕ_{out} biểu thị hàm ánh xạ cho lớp đầu ra và ϕ_X là lớp NCF thứ X , và có tổng cộng X lớp NCF.

2.2 Mô hình Generalized Matrix Factorization (GMF)

Do việc one-hot encoding của user (item) ID của lớp đầu vào, vector embedding thu được có thể được xem là vector ẩn của user (item). latent vector của người dùng \mathbf{p}_u là $\mathbf{P}^T \mathbf{v}_u^U$ và latent vector của item \mathbf{q}_i là $\mathbf{Q}^T \mathbf{v}_i^I$. Hàm ánh xạ của lớp Neural CF đầu tiên được định nghĩa như sau:

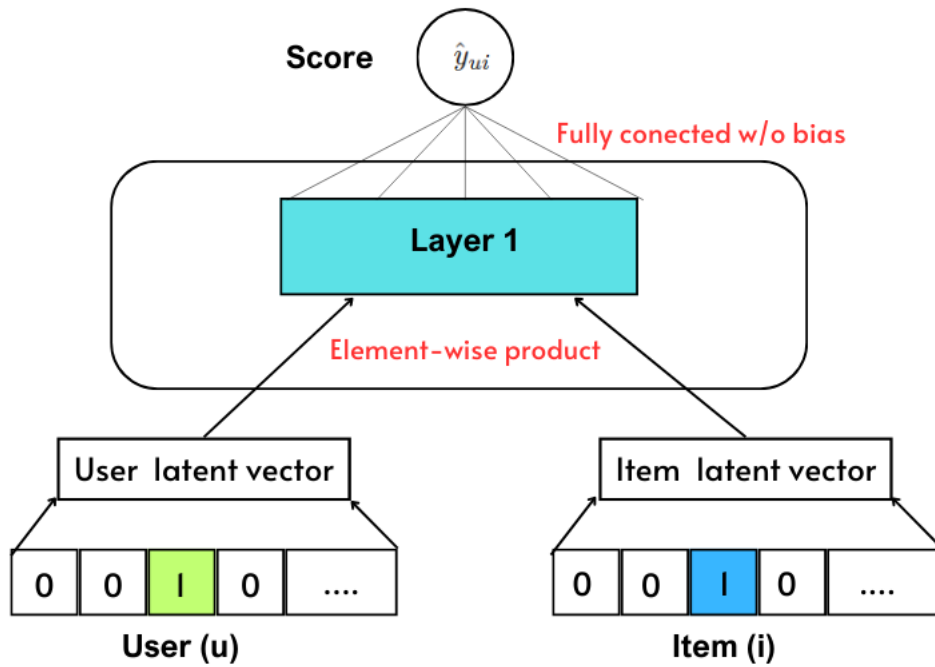
$$\phi(\mathbf{p}_u, \mathbf{q}_i) = \mathbf{p}_u \odot \mathbf{q}_i$$

Trong đó, \odot là tích vô hướng của các vector, và sau đó chiếu vector này lên output:

$$\hat{y}_{ui} = a_{out}(\mathbf{h}^T(\mathbf{p}_u \odot \mathbf{q}_i))$$

a_{out} là activation function, và \mathbf{h} là trọng số cạnh của lớp đầu ra.

Trong project này, chúng em sử dụng activation function cho mô hình GMF là hàm sigmoid, $\sigma = \frac{1}{1+e^{-x}}$



Hình 3: Generalized Matrix Factorization

2.3 Multi-Layer Perceptron (MLP)

NCF sử dụng hai phương thức khác nhau để phân tích người dùng và sản phẩm, nên việc ghép nối các đặc điểm từ cả hai phương thức là một lựa chọn hợp lý. Tuy nhiên, chỉ ghép nối vector một cách đơn giản không đủ để mô tả tương tác giữa các thuộc tính ẩn của người dùng và mục, điều này không thể hiện đầy đủ hiệu quả của lọc cộng tác. Để khắc phục điều này, ta thêm vào các lớp ẩn trên vector đã ghép, sử dụng MLP tiêu chuẩn để phân tích sự tương tác giữa thuộc tính ẩn của người dùng và sản phẩm. Cách tiếp cận này giúp mô hình có khả năng linh hoạt và phi tuyến đáng kể trong việc học các tương tác giữa \mathbf{p}_u và \mathbf{q}_i , không giống như cách tiếp cận của GMF chỉ tích vô hướng. Mô hình MLP trong NCF được xác định như sau:

Layer 1:

$$\mathbf{z}_1 = \phi_1(\mathbf{p}_u, \mathbf{q}_i) = \begin{bmatrix} \mathbf{p}_u \\ \mathbf{q}_i \end{bmatrix}$$

Các layer còn lại:

$$\phi_2(\mathbf{z}_1) = a_2(\mathbf{W}_2^T \mathbf{z}_1 + b_2)$$

...

$$\phi_L(\mathbf{z}_{L-1}) = a_L(\mathbf{W}_L^T \mathbf{z}_{L-1} + b_L)$$

$$\hat{y}_{ui} = \sigma(\mathbf{h}^T \phi_L(\mathbf{z}_{L-1}))$$

Trong đó \mathbf{W}_x , \mathbf{b}_x , và \mathbf{a}_x lần lượt biểu thị ma trận trọng số, vector bias, và hàm activation function cho lớp perceptron thứ x .

Trong việc chọn hàm activation function cho các lớp MLP, có nhiều lựa chọn như sigmoid, tanh, và ReLU (Rectifier).

- Hàm sigmoid giới hạn đầu ra của neuron trong phạm vi (0,1), có thể làm hạn chế hiệu suất của mô hình. Nó còn gặp phải tình trạng bão hòa, tức là neuron không còn học hỏi khi đầu ra gần 0 hoặc 1.
- Tuy tanh được coi là lựa chọn ưu việt hơn và đã được sử dụng rộng rãi, nhưng nó chỉ giải quyết một phần vấn đề của sigmoid, vì nó cơ bản là phiên bản điều chỉnh của sigmoid ($\tanh(\frac{x}{2}) = 2\sigma(x) - 1$).
- Vì vậy, ưu tiên sử dụng ReLU, vì nó phản ánh thực tế tốt hơn và đã được chứng minh không gặp phải tình trạng bão hòa. Ngoài ra, ReLU còn thúc đẩy kích hoạt thưa, phù hợp với dữ liệu thưa và giảm khả năng mô hình bị quá khớp. Kết quả thực nghiệm cho thấy ReLU mang lại hiệu suất cao hơn một chút so với tanh, trong khi tanh lại hiệu quả hơn nhiều so với sigmoid.

2.4 Sự kết hợp của GMF và MLP (Mô hình Neural Matrix Factorization)

Từ 2 biến thể của Neural Collaborative Filtering là Generalized Matrix Factorization - sử dụng một kernel tuyến tính để mô hình hóa sự tương tác của các đặc trưng ẩn; và Multi-Layer Perceptron - sử dụng một kernel phi tuyến để học hàm tương tác từ dữ liệu. Chúng ta có thể kết hợp GMF và MLP dưới hình thức của NCF để chúng có thể hỗ trợ lẫn nhau để mô hình hóa tốt hơn các tương tác phức tạp giữa người dùng và sản phẩm.

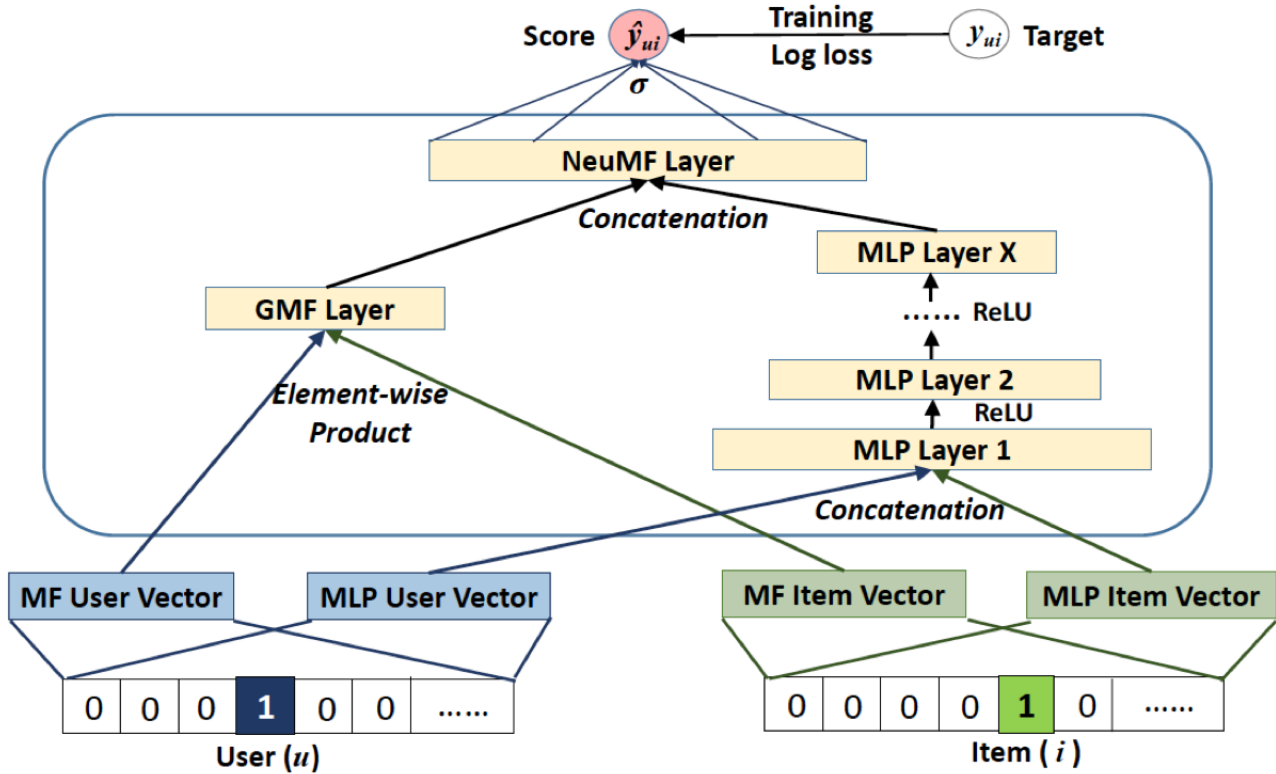
Một giải pháp để kết hợp đơn giản nhất là cho GMF và MLP chia sẻ cùng một lớp embedding, sau đó kết hợp đầu ra của các hàm tương tác của chúng. Cách tiếp cận này tương tự với Neural Tensor Network. Cụ thể, mô hình kết hợp GMF với MLP một lớp có thể được xây dựng như sau:

- GMF model: $\hat{y}_{ui} = \mathbf{h}^T a_{out}(\mathbf{p}_u \odot \mathbf{q}_i)$
- MLP model (1 linear layer): $\hat{y}_{ui} = \mathbf{h}^T a_{out}(\mathbf{W} \begin{bmatrix} \mathbf{p}_u \\ \mathbf{q}_i \end{bmatrix} + \mathbf{b})$

Kết hợp của GMF và MLP:

$$\hat{y}_{ui} = \mathbf{h}^T a_{out}(\mathbf{p}_u \odot \mathbf{q}_i + \mathbf{W} \begin{bmatrix} \mathbf{p}_u \\ \mathbf{q}_i \end{bmatrix} + \mathbf{b})$$

Tuy nhiên, việc chia sẻ các nhúng của GMF và MLP có thể hạn chế hiệu suất của mô hình kết hợp. Ví dụ, điều này có nghĩa là GMF và MLP phải sử dụng cùng một kích thước của nhúng; đối với các tập dữ liệu mà kích thước nhúng tối ưu của hai mô hình có sự chênh lệch lớn, giải pháp này có thể không đạt được kết hợp tối ưu.



Hình 4: Neural Matrix Factorization

Để mang lại sự linh hoạt hơn cho mô hình kết hợp, ta cho phép GMF và MLP học các embedding riêng biệt, và kết hợp hai mô hình bằng cách nối lớp ẩn cuối cùng của chúng (Hình 4). Được xây dựng như sau:

$$\begin{aligned}\phi^{GMF} &= \mathbf{p}_u^G \odot \mathbf{q}_i^G \\ \phi^{MLP} &= a_L(\mathbf{W}_L^T(a_{L-1}(\dots a_2(\mathbf{W}_2^T \begin{bmatrix} \mathbf{p}_u^M \\ \mathbf{q}_i^M \end{bmatrix} + \mathbf{b}_2) \dots)) + \mathbf{b}_L) \\ \hat{y}_{ui} &= \sigma(\mathbf{h}^T \begin{bmatrix} \phi^{GMF} \\ \phi^{MLP} \end{bmatrix})\end{aligned}$$

trong đó \mathbf{p}_u^G và \mathbf{p}_u^M là user embedding của GMF và MLP, \mathbf{q}_i^G và \mathbf{q}_i^M là item embedding của GMF và MLP.

Mô hình này kết hợp tính tuyến tính của MF (Matrix Factorization) và tính phi tuyến của DNNs (Deep Neural Networks) để mô hình hóa các cấu trúc ẩn của người dùng–sản phẩm.

3 Experiments

3.1 Experimental Setup

3.1.1 Dataset

Trong project này, chúng em sử dụng bộ dataset từ paper Learning Image and User Features for Recommendation in Social Networks, được crawl trực tiếp từ mạng xã hội Pinterest.

Tuy nhiên, chúng em chỉ sử dụng một tập con của bộ dữ liệu gốc, bao gồm khoảng 60,000 người dùng và hơn 1 triệu đánh giá của người dùng pinterest cho ảnh. Với 1 là người dùng đó có ghim ảnh đó và 0 là không xác định.

Chúng em chia dữ liệu với 70% là tập training và 30% là tập testing.

3.1.2 Evaluator metric - NDCG

Normalized Discounted Cumulative Gain (NDCG) là metric được dùng phổ biến nhất trong các bài toán learning to rank. NDCG xem xét thứ tự và sự liên quan quan trọng của các tài liệu, đồng thời chú trọng việc đưa ra các tài liệu có liên quan cao và danh sách được đề xuất. Trước khi tìm hiểu NDCG, ta cần biết 2 metric liên quan là Cumulative Gain (CG) và Discounted Cumulative Gain (DCG):

- Cumulative Gain (CG) của một tập các tài liệu được truy xuất là tổng các điểm liên quan (relevance score) của chúng đối với câu truy vấn, được định nghĩa như sau:

$$CG_p = \sum_{i=1}^p rel_i$$

- Discounted Cumulative Gain (DCG) là phiên bản có trọng số của CG, sử dụng logarit để giảm relevance score tương ứng với vị trí của các kết quả. Điều này hữu ích với việc muốn ưu tiên cao hơn cho một vài mục tiêu đầu tiên sau khi phân tích hiệu suất của một hệ thống:

$$DCK_p = \sum_{i=1}^p \frac{rel_i}{\log_2(i+1)}$$

DCG dựa trên giả định sau:

- Các tài liệu có liên quan cao sẽ hữu ích hơn nếu xuất hiện sớm hơn trong kết quả tìm kiếm

- Các tài liệu có liên quan cao sẽ hữu ích hơn các tài liệu có liên quan bên lề tốt hơn các tài liệu không liên quan

Normalized Discounted Cumulative Gain (NDCG) cố gắng nâng cao DCG để phù hợp hơn với các ứng dụng thực tế. Bởi tập hợp các mục được truy xuất có thể khác nhau về kích thước giữa các truy vấn hay hệ thống, NDCG cố gắng so sánh hiệu suất bằng các sử dụng phiên bản chuẩn hóa của DCG. Nói cách khác, nó sắp xếp các tài liệu của 1 danh sách kết quả theo mức độ liên quan, tìm vị trí p có DCG cao nhất, và sử dụng để chuẩn hóa DCG như sau:

$$NDCG_p = \frac{DCG_p}{IDCG_p}$$

Trong đó, IDCG (Ideal Discounted Cumulative Gain), được định nghĩa như sau:

$$IDCG_p = \sum_{i=1}^{|REL_p|} \frac{2^{rel_i} - 1}{\log_2(i + 1)}$$

3.1.3 Training details

Nhóm em sử dụng các siêu tham số cho mô hình như sau:

- Batch size: 256
- Learning rate: 0.001
- Số lượng epoch: 20, dừng sau 2 epoch nếu hàm loss không thay đổi.

Và các tham số:

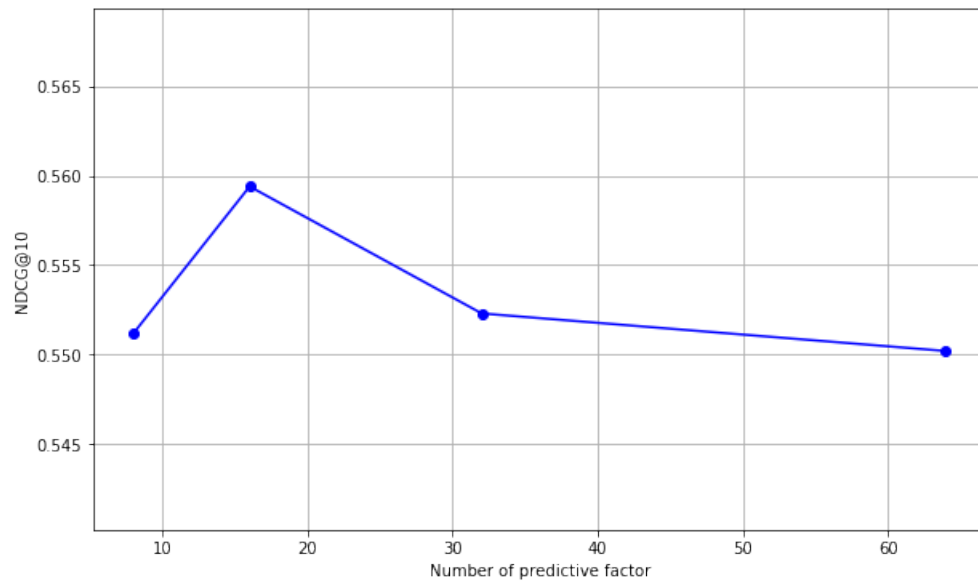
- Số lượng lớp MLP: 3
- Số lượng predictive factor: nhóm em thử trong tập [8,16,32,64], và tốt nhất là 16.
- Loss function: Mean Squared Error (MSE):

$$\mathcal{L} = \sum_{(u,i) \in \mathcal{Y} \cup \mathcal{Y}^-} (y_{ui} - \hat{y}_{ui})^2$$

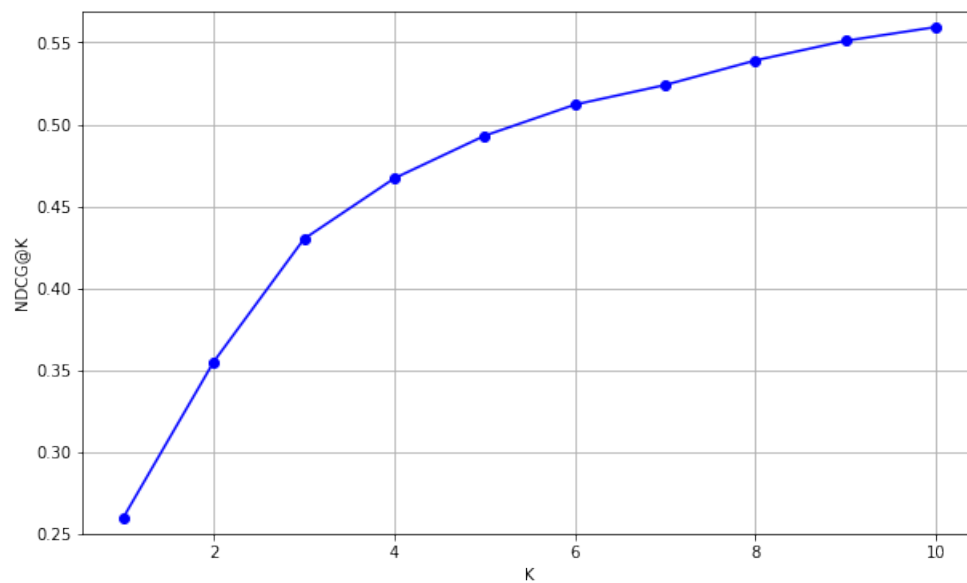
3.2 Experimental Result

Hình 5 là $NDCG_{10}$ dựa trên số lượng predictive factor, có thể thấy chênh lệch không nhiều về performance với số lượng predictive factor khác nhau, và số lượng 16 cho ra performance tốt nhất.

Với hình 6, với số lượng predictive factor là 16, tính NDCG của top N với $N \in 1, 2, 3, 4, 5, 6, 7, 8, 9, 10$. Ta thấy NDCG tỷ lệ thuận với N , nhưng càng về sau thì performance tăng không đáng kể.



Hình 5: Performance of NDCG@10 *w.r.t.* the number of predictive factors



Hình 6: Evaluation of Top-K item recommendation with 16 predictive factors

4 Kết luận

Trong project này, chúng em đã các kiến trúc mạng Neural Collaborative Filtering - GMF, MLP và NeuMF để mô hình hóa các tương tác giữa người dùng và sản phẩm theo những cách khác nhau.

Các mô hình đơn giản nhưng đạt được hiệu suất tương đối tốt.

Link Github: <https://github.com/ngotruongminhdat/Final-Project-RS>

Tài liệu

- [1] Learning Image and User Features for Recommendation in Social Networks
https://openaccess.thecvf.com/content_iccv_2015/papers/Geng_Learning_Image_and_ICCV_2015_paper.pdf
- [2] Neural Collaborative Filtering
<https://arxiv.org/pdf/1708.05031.pdf>
- [3] Supercharging collaborative filtering with neural networks
<https://towardsdatascience.com/neural-collaborative-filtering-96cef1009401>
- [4] Discounted cumulative gain
https://en.wikipedia.org/wiki/Discounted_cumulative_gain