



ĐỒ ÁN 01: BUILD A RECOM- MENDER SYSTEM WITH SPARK

Môn: Ứng dụng Dữ liệu lớn

Lớp: CQ2018/21

Giáo viên: Nguyễn Ngọc Thảo
Phạm Trọng Nghĩa



Thực hiện: 18120167 - Nguyễn Viết Dũng
18120544 – Trương Minh Tài
18120579 – Đặng Minh Thọ
18120584 – Phạm Đình Thực

Graph Convolutional Matrix Completion

1. Tác giả

- Rianne van den Berg (University of Amsterdam)
- Thomas N. Kipf (University of Amsterdam)
- Max Welling (University of Amsterdam, CIFAR)

2. Tóm lược

- o Tác giả xem xét việc sử dụng matrix completion cho các hệ thống khuyến nghị theo quan điểm của dự đoán liên kết trên đồ thị.
- o Dữ liệu tương tác chẳng hạn như xếp hạng phim có thể được biểu thị bằng biểu đồ mục người dùng hai bên với các cạnh được gắn nhãn biểu thị xếp hạng quan sát được
- o Dựa trên tiến bộ gần đây trong học tập sâu về dữ liệu có cấu trúc đồ thị, tác giả đã đề xuất một khuôn khổ bộ mã hóa tự động đồ thị dựa trên thông điệp có thể phân biệt được truyền trên biểu đồ tương tác lưỡng phân.
- o Mô hình của tác giả cho thấy hiệu suất cạnh tranh trên các điểm chuẩn lọc cộng tác tiêu chuẩn. Trong các cài đặt có sẵn thông tin tính năng miễn phí hoặc dữ liệu có cấu trúc như mạng xã hội, khung của tác giả hoàn toàn hoạt động tốt hơn các phương pháp hiện đại gần đây.

3. Giới thiệu

- Với sự phát triển bùng nổ của thương mại điện tử và các nền tảng truyền thông xã hội, các thuật toán đề xuất (recommendation algorithms) đã trở thành công cụ không thể thiếu của nhiều doanh nghiệp.
- Hai nhánh chính của thuật toán giới thiệu thường được phân biệt: hệ thống giới thiệu dựa trên nội dung (content-based recommender systems) và mô hình lọc cộng tác (collaborative filtering models).
- Hệ thống giới thiệu dựa trên nội dung (Content-base recommender systems) sử dụng thông tin nội dung của người dùng và mặt hàng, chẳng hạn như nghề nghiệp và thể loại tương ứng của họ, để dự đoán lần mua tiếp theo của người dùng hoặc xếp hạng của một mặt hàng.
- Các mô hình lọc cộng tác (Collaborative filtering models) giải quyết nhiệm vụ matrix completion bằng cách tính đến dữ liệu tương tác tập chung (collective interaction data) để dự đoán xếp hạng hoặc mua hàng trong tương lai.
- Tác giả xem việc xử lý matrix completion như một vấn đề dự đoán liên kết trên đồ thị: dữ liệu tương tác trong lọc cộng tác (interaction data in collaborative filtering) có thể được biểu thị bằng biểu đồ hai phía (bipartite graph) giữa người dùng và mặt hàng, với xếp hạng / mua hàng được quan sát biểu thị bằng liên kết. Thông tin nội dung có thể được đưa vào khung này một cách tự nhiên dưới dạng các node tính năng. Dự đoán xếp hạng sau đó giảm xuống dự đoán các liên kết được gắn nhãn trong biểu đồ mục người dùng hai bên.
- Tác giả đề xuất sử dụng graph convolutional matrix completion (GC-MC): một framework mã hóa tự động dựa trên đồ thị để matrix completion, được xây dựng dựa trên tiến bộ gần đây trong việc học sâu trên đồ thị. Bộ mã hóa tự động tạo ra các tính

năng tiềm ẩn của người dùng và node item thông qua một dạng thông báo chuyển trên biểu đồ tương tác hai bên. Các đại diện người dùng và mặt hàng tiềm ẩn này được sử dụng để tạo lại các liên kết xếp hạng thông qua bộ giải mã song tuyến (bilinear decoder).

- Lợi ích của việc đưa vào công thức matrix completion như một nhiệm vụ nhằm dự đoán liên kết trên biểu đồ hai phía (bipartite graph) trở nên đặc biệt rõ ràng khi đồ thị đề xuất được đi kèm với thông tin bên ngoài có cấu trúc như mạng xã hội.
- Tác giả chứng minh rằng mô hình bộ mã hóa tự động biểu đồ (graph auto-encoder model) của họ kết hợp hiệu quả dữ liệu tương tác với thông tin bên mà không cần sử dụng đến các khuôn khổ lặp lại như trong (Federico Monti, Michael M. Bronstein, and Xavier Bresson. Geometric matrix completion with recurrent multi-graph neural networks. NIPS, 2017.)

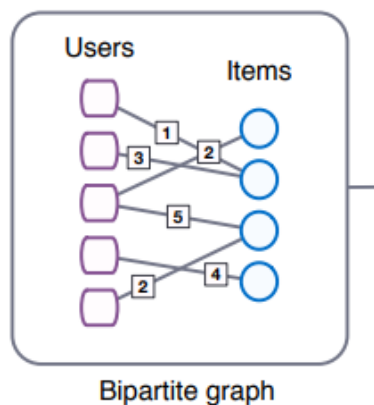
4. Matrix completion dưới dạng dự đoán liên kết trong biểu đồ lưỡng phân

a. KHỞI TẠO

Input: Ma trận dữ liệu được tạo ra với 2 chiều bao gồm đánh giá của user tương ứng với item đánh giá. Rating của item sẽ có giá trị từ 1-5. Những điểm 0 là những dữ liệu chưa được đánh giá mà ta sẽ tìm kiếm khi hoàn thành thuật toán.

	Items			
user	0	1	0	0
	0	3	0	0
	2	0	5	0
	0	0	0	4
	0	0	2	0

(5 users, 4 items)



Output: đồ thị vô hướng như trên.

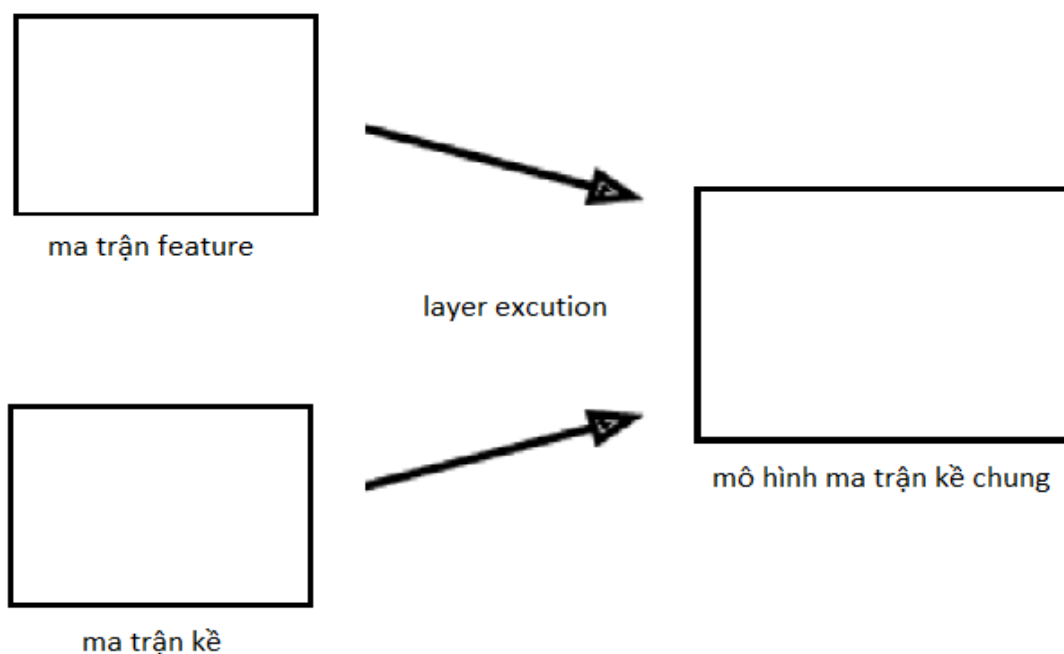
Xây dựng model đồ thị vô hướng sẽ cho chúng ta có nhiều phương pháp tiếp cận bài toán hơn.

Các phương pháp vượt trội đã được kiểm thử qua bài báo là phương pháp end-to-end và graph auto-encoders. Nhóm nghiên cứu lựa chọn graph auto-encoders để cải thiện và thực hiện bài toán trên.

b. GRAPH AUTO-ENCODER

Đồ thị auto-encoders cũng được xây dựng tương tự như end-to-end graph.

Đồ thị này bao gồm:

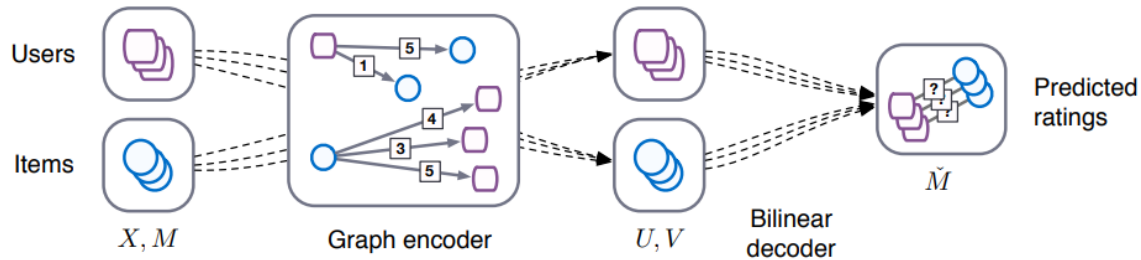


- 1) Graph encoder được xây dựng từ ma trận đặc trưng và ma trận kề của ma trận ban đầu.
 - ⇒ Xây dựng ma trận nhúng Z .
 - ⇒ Bộ decoder được cấu tạo từ embedding matrix Z để dự đoán các phần tử thiếu của ma trận kề nó.
 - ⇒ Với bipartite graph, ta có thể xây dựng lại từ đồ thị encoder với 2 cực là user và item. Phân hóa theo rating sẽ chỉ còn 1 và 0. Tương ứng với đã đánh giá và chưa đánh giá.
 - ⇒ Tương tự như vậy, ta có thể xây dựng lại từ đồ thị decoder với 2 cực user và item. Xây dựng lại ma trận mới đã được encoder (graph auto-encoder) bằng cách giảm tối đa tỉ lệ sai giữa graph này với ma trận cơ bản. các ví dụ cho tỉ lệ sai này bao gồm:

root means square error hay the cross entropy khi xét các mức rating là các class khác nhau (1,2,3,4,5).

c. GRAPH CONVOLUTIONAL ENCODER

Quá trình gửi thông tin qua Graph convolutional – Matrix completion được thực hiện như sau



Weight sharing được sử dụng thông suốt quá trình forward-pass này bởi vì GC chỉ sử dụng các node lân cận nó. Đồng thời weight sharing được sử dụng khá nhiều và phù hợp với GC. Các vector-valued được gửi qua và thay đổi phù hợp với rating mà nó có.

Thay vì gửi trực tiếp dữ liệu qua bước tiếp theo trong mạng neural network, có thể xây dựng 1 hệ thống tập trung giúp cho giá trị của dữ liệu đang có được đánh giá cao phụ thuộc vào model.

d. BILINEAR DECODER

Để xây dựng lại liên kết trong bipartite graph, ta sử dụng bilinear decoder và tách riêng 5 lớp khác nhau cho 5 rating. Với việc chỉ định lại rating của user tới item và ngược lại, bộ decoder tạo ra một phân phối xác suất trên các mức rating thông qua 1 bilinear operation. Từ đó tìm ra được predicted ratings cho user lên item.

e. MODEL TRAINING

Sử dụng thêm 1 ma trận omega với mục đích lưu trữ các rating không có tác động lên giá trị của predicted rating, hay là các rating không được xét tới. nhờ đó, training chỉ tối ưu hóa các rating được đánh giá.

Node dropout để mô hình có thể tổng quát hóa cho các xếp hạng không được quan sát, nó được đào tạo trong 1 thiết lập để giảm giá trị các node đó bằng cách loại bỏ ngẫu nhiên.

Mini-batching được sử dụng để giảm thiểu lượng dữ liệu khổng lồ ban đầu. loại bỏ trực tiếp 1 lượng user và item không quan trọng. cải thiện means of regularization và giảm lượng bộ nhớ cần thiết để train model.

5. Các công trình liên quan

- Auto-encoders
 - Bộ mã hóa tự động dựa trên người dùng hoặc mặt hàng là một collaborative filtering models hiện đại gần đây có thể được coi là một trường hợp đặc biệt của mô hình mã hóa tự động biểu đồ (graph auto-encoder model), trong đó chỉ có người dùng hoặc mặt hàng mới được xem xét bộ mã hóa
 - AutoRec là mô hình đầu tiên trong đó vectơ xếp hạng được quan sát một phần của người dùng (hoặc của mặt hàng) được tham chiếu vào một không gian ẩn thông qua lớp mã hóa (encoder layer) và được tạo lại bằng cách sử dụng lớp giải mã (decoder layer) với ean squared reconstruction error loss.
 - Thuật toán CF-NADE có thể được coi là một trường hợp đặc biệt của auto-encoder architecture nói trên, dựa trên người dùng, thông báo chỉ được chuyển từ các mục đến người dùng và trong trường hợp dựa trên mục, điều ngược lại được giữ nguyên. CF-NAD là một mô hình mã hóa tự động làm giảm tín hiệu, trong đó một phần của không gian đầu vào bị loại bỏ ngẫu nhiên trong mỗi lần lặp lại.
- Matrix factorization
 - Matrix factorization giảm chiều của graph. Giảm chiều của ma trận tương tác giữa user – item thành rectangular matrix. Kết quả của phương pháp này có thể được cải thiện qua nhiều phương pháp khác nhau. Sử dụng alternating least square. Từ 2 nguồn dữ liệu user và items được xem như factors, từ đó đánh giá được dữ liệu thiếu trong ma trận X cần tìm. Nổi bật trong bài ta có:
 - Probabilistic matrix factorization(PMF): sử dụng bayesian trong ma trận X cần tìm kiếm, trong model này, ma trận X cho rằng mỗi kết quả đều độc lập với nhau, được đánh giá riêng biệt và có phương sai chung.
 - Biased matrix factorization (BMF): sử dụng sự liên kết giữa user và item để xác định giá trị trong ma trận X. ví dụ 1 người thường xuyên đánh giá phim ở mức 2-3 khi gặp bộ phim đánh giá trung bình <3 sao sẽ không thể đánh giá 4-5 cho bộ phim đó. Và ngược lại.
 - Neural network matrix factorization (NNMF): với phương pháp khởi tạo đơn giản như random function model. NNMF mang tính cải thiện hơn so với PMF hay BMF, bằng việc lọc và gửi dữ liệu qua các lớp neural network, NNMF còn giới thiệu phương pháp xây dựng lại matrix mới với low rank data.
- Matrix completion with side information
 - Trong matrix completion mục tiêu là làm gần đúng rating matrix với low-rank rating matrix. Tuy nhiên, giảm thiểu bậc là một vấn đề khó giải quyết giảm bậc bằng một giảm bậc chuẩn hạt nhân (tổng các giá trị kỳ dị của ma trận).
 - Geomatic matrix completion model giới thiệu 1 regularization bằng cách thêm vào dữ liệu phụ trong đoạn graph chứa user và items. Là một phương pháp tối ưu hơn cho alternating least square optimization sử dụng trong graph-regularized matrix completion problem.
 - Gần đây nhất, side information được sử dụng trong matrix completion thông qua sử dụng convolutional neural networks trong graph, tích hợp việc định hướng trong graph và tạo model cho dynamic rating generation process.

6. Thực nghiệm

- Dataset:
 - o MovieLens (100K, 1M, and 10M), Flixster, Douban, and YahooMusic.
 - o Bộ dữ liệu MovieLens bao gồm xếp hạng của người dùng cho các mục (chẳng hạn như phim) và tùy chọn kết hợp thông tin người dùng / mục bổ sung dưới dạng các tính năng. Số lượng: 100K, 1M, and 10M
 - o Đối với Flixster, Douban và YahooMusic, tác giả sử dụng các tập con được xử lý lại của các tập dữ liệu này được cung cấp từ <https://github.com/fmonti/mgcnn>. Các tập dữ liệu này chứa các biểu đồ con của 3000 người dùng và 3000 mặt hàng và các đồ thị tương tác giữa người dùng với người dùng và mặt hàng tương ứng của họ (nếu có).
 - o Thống kê tập dữ liệu được tóm tắt trong Bảng 1.

Dataset	Users	Items	Features	Ratings	Density	Rating levels
Flixster	3,000	3,000	Users/Items	26,173	0.0029	0.5, 1, ..., 5
Douban	3,000	3,000	Users	136,891	0.0152	1, 2, ..., 5
YahooMusic	3,000	3,000	Items	5,335	0.0006	1, 2, ..., 100
MovieLens 100K (ML-100K)	943	1,682	Users/Items	100,000	0.0630	1, 2, ..., 5
MovieLens 1M (ML-1M)	6,040	3,706	—	1,000,209	0.0447	1, 2, ..., 5
MovieLens 10M (ML-10M)	69,878	10,677	—	10,000,054	0.0134	0.5, 1, ..., 5

Table 1: Number of users, items and ratings for each of the MovieLens datasets used in our experiments. We further indicate rating density and rating levels.

- MovieLens 100K: Tác giả sử dụng thêm thêm các thông tin bổ sung (tuổi, giới tính và nghề nghiệp) và thể loại phim) vào các vector đặc trưng cho người dùng và phim, đồng thời so sánh hiệu suất của mô hình của tác giả với (GC-MC + Feat) và không bao gồm các tính năng này (GC-MC). Train mô hình với 1,000 full-batch epochs và đánh giá bằng RMSE trung bình trong 5 lần chạy với các lần khởi tạo ngẫu nhiên. Kết quả

thể hiện ở bảng

Model	ML-100K + Feat
MC [3]	0.973
IMC [11, 31]	1.653
GMC [12]	0.996
GRALS [25]	0.945
sRGCNN [22]	0.929
GC-MC (Ours)	0.910
GC-MC+Feat	0.905

Table 2: RMSE scores⁶ for the MovieLens 100K task with side information on a canonical 80/20 training/test set split. Side information is either presented as a nearest-neighbor graph in user/item feature space or as raw feature vectors. Baseline numbers are taken from [22].

- MovieLens 1M và 10M: tác giả so sánh với các thuật toán collaborative filtering tiên tiến nhất hiện nay, chẳng hạn như AutoRec , LLorma và CF-NADE.
 - Đối với ML-1M, tác giả sử dụng accumulate messages through summation, sử dụng tỷ lệ bỏ cuộc (dropout rate) là 0,7 và chuẩn hóa đối xứng(symmetric normalization).
 - Vì ML-10M có gấp đôi số lớp xếp hạng, tác giả sử dụng gấp đôi số ma trận hàm cơ sở trong bộ giải mã. Hơn nữa, tác giả sử dụng stacking accumulation, dropout là 0,3 và symmetric normalization.
 - Train với 3,500 full-batch epochs, and 18,000 mini-batch iterations (20 epochs with batch size 10,000) trên tập ML-1M và ML-10M
 -

Model	ML-1M	ML-10M
PMF [20]	0.883	—
I-RBM [26]	0.854	0.825
BiasMF [16]	0.845	0.803
NNMF [7]	0.843	—
LLORMA-Local [17]	0.833	0.782
I-AUTOREC [27]	0.831	0.782
CF-NADE [32]	0.829	0.771
GC-MC (Ours)	0.832	0.777

Table 4: Comparison of average test RMSE scores on five 90/10 training/test set splits (as in [32]) without the use of side information. Baseline scores are taken from [32]. For CF-NADE, we report the best-performing model variant.

- Flixster, Douban và YahooMusic: Các tập dữ liệu này chứa thông tin về người dùng và mặt hàng dưới dạng đồ thị. Tác giả tích hợp thông tin bên dựa trên biểu đồ này vào framework của mình bằng cách sử dụng vector kề (chuẩn hóa theo mức độ) làm vector đặc trưng cho người dùng / mặt hàng tương ứng.
 - Đối với lớp (single dense feature embedding): thực hiện tích chập đồ thị trên biểu đồ người user-user hoặc item-item
 - Sử dụng dropout rate là 0,7 và 64 hidden units cho dense side information layer (với kích hoạt ReLU) và sử dụng left normalization và các thông báo trong raph convolution layer được tích lũy bằng cách ghép nối (trái ngược với phép tổng hợp). Tất cả các mô hình được train với 200 epochs
 - Đánh giá mô hình tác giả train bộ training gốc và báo cáo hiệu suất ủa bộ kiểm tra
 - Kết quả được tóm tắt trong Bảng 3.
 -

Model	Flixster	Douban	YahooMusic
GRALS	1.313/1.245	0.833	38.0
sRGCNN	1.179/0.926	0.801	22.4
GC-MC	0.941/0.917	0.734	20.5

Table 3: Average RMSE test set scores for 5 runs on Flixster, Douban, and YahooMusic, all of which include side information in the form of user and/or item graphs. We replicate the benchmark setting as in [22]. For Flixster, we show results for both user/item graphs (right number) and user graph only (left number). Baseline numbers are taken from [22].

- Thảo luận:
 - ML-100K với thông tin phụ hoạt động tốt hơn. Tác giả thử nghiệm với tập dữ liệu không sử dụng thêm thông tin phụ với phương pháp sRGCNN sử dụng tích chập đồ thị trên đồ thị lân cận gần nhất của người dùng và mặt hàng, đồng thời học các biểu diễn theo cách lặp đi lặp lại bằng cách sử dụng RNN. Kết quả chứng minh rằng ước tính trực tiếp ma trận xếp hạng từ các đại diện người dùng/mặt hàng đã học bằng cách sử dụng mô hình decoder đơn giản có thể hiệu quả hơn, đồng thời hiệu quả hơn về mặt tính toán.
 - Kết quả của ML-1M và ML-10M chứng minh rằng có thể mở rộng phương pháp thành các bộ dữ liệu lớn hơn, đưa nó vào vùng lân cận của các phương pháp lọc cộng tác hiện đại nhất gần đây của người dùng hoặc dựa trên mặt hàng về mặt dự đoán màn biểu diễn. Tại thời điểm này, điều quan trọng cần lưu ý là một số kỹ thuật được giới thiệu trong CF-NADE, chẳng hạn như layer-specific learning rates, special ordinal loss function và auto-regressive modeling of ratings
 - Đối với tập dữ liệu Flixster, Douban và YahooMusic, mô hình của chúng tôi đạt được kết quả hiện đại, trong khi sử dụng một cài đặt siêu tham số duy nhất trên cả ba tập dữ liệu

7. Kết luận

- Tác giả giới thiệu về graph convolutional matrix completion (GC-MC): một framework mã hóa tự động đồ thị (graph auto-encoder) cho nhiệm vụ matrix completion trong các hệ thống khuyến nghị. Bộ mã hóa chứa một lớp tích chập đồ thị (graph convolution layer) tạo nhúng người và mặt hàng thông qua biểu đồ tương tác hai phía giữa người dùng - mặt hàng. Được kết hợp với bộ giải mã song tuyến (bilinear decoder), các xếp hạng mới được dự đoán dưới dạng các cạnh được gắn nhãn.

- Framework mã hóa tự động biểu đồ tổng quát hóa một cách tự nhiên để bao gồm thông tin bên cho cả người dùng và mặt hàng. Trong cài đặt, mô hình được đề xuất hoạt động tốt hơn các phương pháp liên quan gần đây nhờ một biên độ lớn, như được chứng minh trên một số bộ dữ liệu điểm chuẩn với thông tin bên dựa trên tính năng và đồ thị. Mô hình của tác giả có thể được đào tạo trên các bộ dữ liệu quy mô lớn hơn thông qua chia nhỏ ngẫu nhiên. Trong cài đặt này, mô hình đạt được kết quả cạnh tranh với tính năng lọc cộng tác hiện đại nhất gần đây.
- Trong tương lai, tác giả muốn mở rộng mô hình này sang dữ liệu đa phương thức ở quy mô lớn (bao gồm văn bản, hình ảnh và thông tin dựa trên biểu đồ khác), chẳng hạn như hiện diện trong nhiều nền tảng đề xuất thực nghiệm. Trong một cài đặt như vậy, mô hình GC-MC có thể được kết hợp với RNN (đối với văn bản) hoặc CNN (đối với hình ảnh). Để giải quyết khả năng mở rộng, cần phải phát triển các chương trình gần đúng hiệu quả, chẳng hạn như lấy mẫu con các vùng lân cận.

Các phương pháp đánh giá mô hình

1. RMSE:

Là chỉ số căn bậc 2 của trung bình sai số dự đoán so với giá trị rating trên tập test. Nó thực hiện điều này bằng cách đo sự khác biệt giữa các giá trị dự đoán và giá trị thực tế. RMSE càng nhỏ tức là sai số càng bé thì mức độ ước lượng cho thấy độ tin cậy của mô hình càng cao.

$$RMSE = \sqrt{\sum_{i=1}^n \frac{(\hat{y}_i - y_i)^2}{n}}$$

Với:

- \hat{y}_i là giá trị ước lượng
- y_i là biến độc lập
- $n = (N - k - 1)$
- N : số tổng lượng quan sát
- K : tổng lượng biến

Giá trị sai số này thấp hơn so với giá trị sai số phổ biến của các thuật toán content-based và collaborative-filtering khác khi hầu hết RMSE đều khoảng 0.9x.

2. Hit rate

Mục đích: Xem xét top 10 đề xuất tốt như thế nào?

Để đánh giá top 10, ta sử dụng hit rate, nghĩa là, nếu người dùng xếp hạng một đối tượng nằm trong top 10 đề xuất, ta coi đó là một “hit”. Ta sẽ tính tổng số lần xuất hiện trên tổng đề xuất.

Quá trình thực hiện:

- Tìm tất cả các item trong lịch sử người dùng của tập huấn luyện
- Xóa 1 trong các item (Leave-One-Out cross-validation).
- Dùng các item khác để cung cấp cho recommender và yêu cầu đề xuất top N
- Nếu item đã xóa xuất hiện trong top N đề xuất thì đó là 1 “hit” và ngược lại.

3. **nDCG**: Normalized Discounted Cumulative Gain là metric được dùng phổ biến nhất trong các bài toán learning to rank. nDCG xem xét thứ tự và sự liên quan độ quan trọng của các đối tượng, đồng thời chú trọng việc đưa ra các đối tượng có liên quan cao vào danh sách được đề xuất [1].

- a. **CG** (Cumulative Gain): Tổng gains của K items hàng đầu của recommended (chưa tính đến thứ tự của các items) [2].

$$CG_{@K} = \sum_{i=1}^K G_i$$

- b. **DCG** (Discounted Cumulative Gain): là phiên bản có trọng số của CG, sử dụng logarit để giảm relevance score tương ứng với vị trí của các kết quả [2].

$$DCG_{@K} = \sum_{i=1}^K \frac{G_i}{\log_2(i+1)}$$

- c. **NDCG** (Normalized Discounted Cumulative Gain): phiên bản DCG với hệ số chuẩn hóa ở mẫu số bằng việc sắp xếp các items của 1 danh sách kết quả theo mức độ liên quan [2].

$$NDCG_{@K} = \frac{DCG_{@K}}{IDCG_{@K}}$$

$$IDCG_{@K} = \sum_{i=1}^{K^{ideal}} \frac{G_i^{ideal}}{\log_2(i+1)}$$

Đánh giá mô hình:

Ta tiến hành đánh giá mô hình đã được cài đặt theo 3 độ đo trên các tập dữ liệu:

- MovieLens 1M
 - o MovieLens 1M movie ratings.
 - o size: 6 MB.
 - o Data: 1,000,000 ratings.
 - o Thu thập: Từ 6,000 users cho 4,000 movies.
 - o Released 2/2003.
 - o The ratings dataframe is 95.53% sparse.
- MovieLens 10M
 - o MovieLens 10M movie ratings.
 - o size: 63 MB. (Larger than MovieLens 10M)

- Data: 10M và 100K tag applications.
- Thu thập từ: 72,000 users cho 10,000 movies.
- Released 1/2009.
- The ratings dataframe is 98.66% sparse. (sparser than MovieLens 1M)

Kết quả:

	RMSE	Hit rate@10	nDCG@10
MovieLens 1M	0.857358		0.022819
MovieLens 10M	0.798771		0.005446

Tài liệu tham khảo

- [1] "NDCG Evaluation Metric for Recommender Systems," [Online]. Available: <https://machinelearninginterview.com/topics/machine-learning/ndcg-evaluation-metric-for-recommender-systems/>.
- [2] "Đánh giá các mô hình học máy," [Online]. Available: https://viblo.asia/p/danh-gia-cac-mo-hinh-hoc-may-RnB5pp4D5PG#_dcg--ndcg-16.
- [3] [Online]. Available: <https://viblo.asia/p/introduction-to-recommender-systems-aWj53LQ8K6m> .
- [4] [Online]. Available: <https://viblo.asia/p/matrix-factorization-phuong-phap-goi-y-dua-tren-ky-thuat-phan-ra-ma-tran-p1-Az45bapNlxY>.