@@

A green and orange logo

Description automatically generated

**TẬP ĐOÀN VIỄN THÔNG QUÂN ĐỘI**

**TÀI LIỆU MÔ TẢ**

**Nghiên cứu đề xuất phương pháp khuyến nghị phim theo phiên**

Hà Nội, 2023



# Mục lục

[Mục lục 2](#_Toc151505521)

[Danh mục hình ảnh 4](#_Toc151505522)

[Danh mục bảng biểu 6](#_Toc151505523)

[Phần 1. Đánh giá và tìm hiểu bài toán 7](#_Toc151505524)

[1.1. Tổng quan bài toán 7](#_Toc151505525)

[1.2. Đề xuất luồng giải pháp xây dựng mô hình 14](#_Toc151505526)

[1.3. Các vấn đề cần giải quyết 16](#_Toc151505527)

[Phần 2. Phân tích và xử lý dữ liệu 22](#_Toc151505528)

[2.1. Xử lý dữ liệu 22](#_Toc151505529)

[2.1.1. Tổng hợp các đặc trưng nhóm mô hình 22](#_Toc151505530)

[2.1.2. Tổng hợp các đặc trưng xử lý nhóm dữ liệu categorical 23](#_Toc151505531)

[2.1.3. Kết hợp dữ liệu và xử lý dữ liệu nhiễu 24](#_Toc151505532)

[2.1.4. Thực hiện chuẩn hóa scale 25](#_Toc151505533)

[2.2. Phân tích dữ liệu 26](#_Toc151505534)

[Phần 3. Các mô hình nghiên cứu đề xuất 29](#_Toc151505535)

[3.1. Mô hình khuyến nghị với kiến trúc mạng Gated Recurrent Unit theo phiên tương tác của khách hàng 30](#_Toc151505536)

[3.1.1. Xây dựng thuật toán và kiến trúc mô hình 30](#_Toc151505537)

[3.1.2. Thực nghiệm và phân tích 33](#_Toc151505538)

[3.2. Mô hình khuyến nghị áp dụng mạng học sâu tự chú ý theo phiên tương tác của khách hàng 43](#_Toc151505539)

[3.2.1. Xây dựng thuật toán và kiến trúc mô hình 43](#_Toc151505540)

[3.2.2. Thực nghiệm và phân tích 46](#_Toc151505541)

[3.3. Mô hình khuyến nghị mở rộng quy mô áp dụng kiến trúc Transformer theo phiên tương tác của khách hàng 55](#_Toc151505542)

[3.3.1. Xây dựng thuật toán và kiến trúc mô hình 55](#_Toc151505543)

[3.3.2. Thực nghiệm và phân tích 58](#_Toc151505544)

[3.4. Tổng hợp và phân tích hiệu suất các nhóm mô hình khuyến nghị theo phiên 66](#_Toc151505545)

# Danh mục hình ảnh

[Hình 1.1. Hình mô tả ý tưởng lọc cộng tác truyền thống trong các hệ khuyến nghị phim và chương trình trên các nền tảng số 8](#_Toc151505546)

[Hình 1.2. Quy trình lọc cộng tác phổ biến trong các hệ khuyến nghị phim và chương trình trên các nền tảng số 8](#_Toc151505547)

[Hình 1.3. Một số vấn đề thiên khiến nổi bật trong các hệ khuyến nghị 9](#_Toc151505548)

[Hình 1.4. Một ví dụ về kỹ thuật ensemble các mô hình học máy trong các hệ khuyến nghị 10](#_Toc151505549)

[Hình 2.1. Đặc trưng về thông tin phim 22](#_Toc151505550)

[Hình 2.2. Đặc trưng về đánh giá của người dùng cho phim 23](#_Toc151505551)

[Hình 2.3. Một số đặc trưng về thông tin hồ sơ của người dùng 23](#_Toc151505552)

[Hình 2.4. Tổng hợp các đặc trưng mô tả thể loại của bộ phim 24](#_Toc151505553)

[Hình 2.5. Kết hợp dữ liệu tên và thể loại tạo trường thông tin mô tả phim 24](#_Toc151505554)

[Hình 2.6. Xử lý dữ liệu NaN trong ma trận tiện ích của người dùng và phim 25](#_Toc151505555)

[Hình 2.7. Kết quả trực quan hóa dữ liệu theo phân bố giới tính của người dùng 26](#_Toc151505556)

[Hình 2.8. Kết quả trực quan hoá dữ liệu theo phân bố độ tuổi và nghề nghiệp của người dùng 27](#_Toc151505557)

[Hình 2.9. Trực quan hoá dữ liệu theo số lượt được tương tác của mỗi loại phim 27](#_Toc151505558)

[Hình 2.10. Trực quan hoá dữ liệu theo điểm số được yêu thích của mỗi loại phim 28](#_Toc151505559)

[Hình 2.11. Trực quan hoá dữ liệu mô tả phân bố số lần tương tác của các người dùng với các bộ phim trong hệ thống 28](#_Toc151505560)

[Hình 3.1. Biểu đồ minh họa hiệu suất của mô hình GRU4Rec trong quá trình các round trên bộ dữ liệu mẫu từ TV360 41](#_Toc151505561)

[Hình 3.2. Mô tả kiến trúc SASRec trong quá trình huấn luyện. Ở mỗi bước thời gian, mô hình sẽ xem xét tất cả các thao tác trước đó và chú ý đến các đối tượng cần 'tập trung vào' để xác định hành động tiếp theo 44](#_Toc151505562)

[Hình 3.3. Biểu đồ minh họa hiệu suất của mô hình SASRec trong quá trình các round trên bộ dữ liệu mẫu từ TV360 53](#_Toc151505563)

[Hình 3.4. Mô tả quá trình lấy mẫu âm 56](#_Toc151505564)

[Hình 3.5. Mô tả quá trình lấy mẫu âm theo từng phần tử 56](#_Toc151505565)

[Hình 3.6. Mô tả quá trình lấy mẫu âm theo phiên 57](#_Toc151505566)

[Hình 3.7. Mô tả quá trình lấy mẫu âm theo lô 57](#_Toc151505567)

[Hình 3.8. Biểu đồ minh họa hiệu suất của mô hình TRON trong quá trình các round trên bộ dữ liệu mẫu từ TV360 65](#_Toc151505568)

[Hình 3.9. Biểu đồ minh họa hiệu suất của các mô hình khuyến nghị theo phiên trong quá trình thực nghiệm các round trên bộ dữ liệu mẫu từ TV360 66](#_Toc151505569)

# Danh mục bảng biểu

[Bảng 1.1. Tóm tắt các vấn đề cần giải quyết 17](#_Toc151505570)

[Bảng 3.1. Tổng hợp hiệu suất model GRU4Rec qua các Round trên bộ dữ liệu thực tế trên nền tảng TV360 40](#_Toc151505571)

[Bảng 3.2. Tổng hợp hiệu suất model SASRec qua các Round trên bộ dữ liệu thực tế trên nền tảng TV360 52](#_Toc151505572)

[Bảng 3.3. Tổng hợp hiệu suất model TRON qua các Round trên bộ dữ liệu thực tế trên nền tảng TV360 65](#_Toc151505573)

# Tìm hiểu và nhận định bài toán

## Tổng quan bài toán

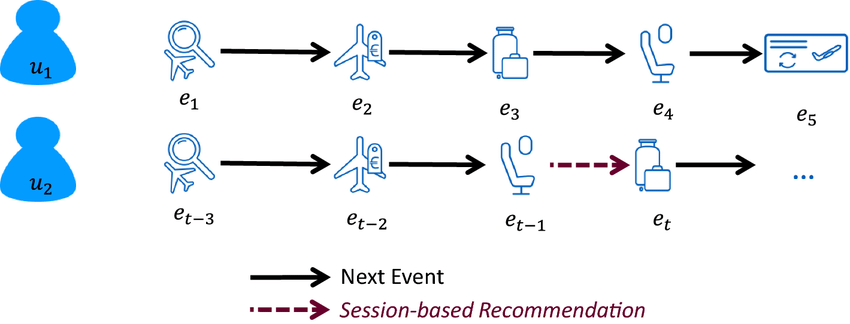
**Mô tả bài toán**:

Nếu người dùng chỉ mới ghé thăm nền tảng xem phim trực tuyến, hệ thống hướng đến mục tiêu có thể giúp người dùng thấy một hạng mục như "Đoán Xem Bạn Sẽ Thích" hiển thị danh sách các bộ phim thú vị mà có thể phù hợp với sở thích và mong muốn xem phim của họ theo phiên đăng nhập và tương tác của người dùng đó. Đây chính là mục tiêu chính của hệ khuyến nghị nói chung và bài toán gợi ý phim theo phiên nói riêng. Hệ thống khuyến nghị phim là một loại hệ thống lọc hướng đến việc cung cấp thông tin cá nhân hóa cho người dùng, từ đó cải thiện trải nghiệm của người dùng và thúc đẩy lợi nhuận kinh doanh.

Trong các tài liệu nghiên cứu về các vấn đề tồn tại hệ thống gợi ý phim (RS) của Viettel, hầu hết các tài liệu đều tập trung vào việc phát minh ra các mô hình khuyến nghị dựa trên lịch sử tương tác dài hạn của người dùng, mà chưa thể giải quyết vấn đề khuyến nghị theo thời gian thực, ngay trên phiên đăng nhập mà người dùng sử dụng tại thời điểm nhất định. Hệ thống đề xuất dựa trên phiên SBRS tập trung vào việc cung cấp các đề xuất được cá nhân hóa dựa trên phiên hiện tại của người dùng, đây là một chuỗi tương tác giữa người dùng và hệ thống. Không giống như các hệ thống đề xuất truyền thống dựa vào lịch sử người dùng lâu dài, SBRS được thiết kế để đáp ứng các tình huống trong đó sở thích của người dùng thay đổi nhanh chóng trong một khung thời gian ngắn. Thách thức chính nằm ở việc nắm bắt được động lực tạm thời và sở thích ngắn hạn của người dùng. SBRS thường được sử dụng trong các tình huống như thương mại điện tử, nơi người dùng thể hiện các sở thích đa dạng và theo ngữ cảnh cụ thể trong các phiên khác nhau.

Đặc điểm chính cần quan tâm:

* Mô hình hóa tuần tự: SBRS mô hình hóa trình tự tương tác của người dùng, xem xét thứ tự các mục trong một phiên.
* Trọng tâm ngắn hạn: Các đề xuất được điều chỉnh cho phù hợp với phiên hiện tại, phản ánh mối quan tâm trước mắt của người dùng.
* Tùy chọn người dùng động: Những thay đổi nhanh chóng trong tùy chọn người dùng được điều chỉnh, nắm bắt những thay đổi ngắn hạn.



Một ví dụ về kỹ thuật khuyến nghị sản phẩm theo phiên các hệ thống thời gian thực

**Các phương pháp liên quan:**

* Mạng RNN và Bộ nhớ ngắn hạn dài (LSTM): LSTM, một loại RNN, đã được sử dụng rộng rãi để lập mô hình tuần tự trong SBRS. Phương pháp rất hữu hiệu trong việc nắm bắt các phần phụ thuộc trong chuỗi tương tác của người dùng, cung cấp thông tin chi tiết về sở thích ngày càng phát triển của người dùng.
* Cơ chế tự chú ý: các mô hình dựa trên Transformer ban đầu được thiết kế để xử lý ngôn ngữ tự nhiên, Transformer đã được áp dụng thành công cho SBRS. Cơ chế tự chú ý cho phép các mô hình tập trung động vào các phần khác nhau của phiên, nắm bắt các phần phụ thuộc phức tạp một cách hiệu quả.
* SASRec (Khuyến nghị tuần tự tự chú ý):
  + Kiến trúc mô hình: SASRec được thiết kế rõ ràng cho SBRS. Cơ chế tự chú ý của nó vượt trội trong việc nắm bắt các phần phụ thuộc tầm xa trong một phiên, mang lại sự hiểu biết sâu sắc về các mẫu tuần tự.
  + Hiệu suất tiên tiến: SASRec đã chứng minh hiệu suất tiên tiến trong các kịch bản dựa trên phiên, khiến nó trở thành chuẩn mực để so sánh trong nghiên cứu liên quan.
* GRU4Rec:
  + Lập mô hình tuần tự với GRU: GRU4Rec sử dụng Đơn vị lặp lại có cổng (GRU) để lập mô hình tuần tự. Nó giải quyết các vấn đề phụ thuộc ngắn hạn và tính chất ngày càng tăng của sở thích người dùng trong một phiên.
  + Xử lý thông tin ngắn hạn: GRU4Rec tập trung vào việc nắm bắt biến đổi ngắn hạn trong hành vi của người dùng, phù hợp với tính chất tạm thời của phiên.
* TRON (Trình đề xuất biến áp dựa trên phiên có thể mở rộng):
  + Khả năng mở rộng và mô hình hóa hiệu quả: TRON giới thiệu khả năng mở rộng cho đề xuất dựa trên phiên bằng cách tối ưu hóa việc lấy mẫu âm. Tận dụng kiến trúc Transformer, TRON mô hình hóa hiệu quả các tương tác tuần tự trong các tình huống quy mô lớn.
* Phương pháp tiếp cận lai:
  + Kỹ thuật kết hợp: Một số nghiên cứu khám phá các phương pháp kết hợp kết hợp điểm mạnh của mô hình tuần tự với các phương pháp lọc cộng tác hoặc dựa trên nội dung, nhằm mục đích hiểu biết toàn diện về sở thích của người dùng.
* Đề xuất dựa trên phiên thời gian thực:
  + Thích ứng động: Nghiên cứu mới nổi tập trung vào SBRS thời gian thực, nhấn mạnh sự cần thiết của các mô hình để thích ứng linh hoạt với sở thích tức thời của người dùng trong một phiên.

Hiểu được bản chất năng động của hành vi người dùng trong các phiên ngắn hạn và phát triển các mô hình có thể nắm bắt và thích ứng một cách hiệu quả với tính linh động lực trong tương tác theo phiên là những thách thức chính và điểm mấu chốt cần quan tâm trong cách giải quyết bài toán khuyến nghị theo phiên trên hệ thống TV360. Bên cạnh đó, tính mở rộng của mô hình cũng cần làm rõ trên bối cảnh dữ liệu lớn.

**Đầu vào**: Dữ liệu đầu vào được trích xuất và thống kê từ hệ quản trị cơ sở dữ liệu của hệ thống, bao gồm các dữ liệu sau:

* Định danh hồ sơ của người dùng
* Hành vi tương tác với phim của người dùng:
  + Thời gian tương tác
  + Thời điểm tương tác
  + Định danh phim được người dùng sử dụng
  + Mức độ và tần suất tương tác
* Thông tin của phim bao gồm:
  + Thông tin về phim hoặc tập phim
  + Thông tin về mô tả phim
  + Thông tin về thể loại phim
  + Thông tin về thời điểm phát hành phim

**Đầu ra**: Đối với dữ liệu TV360, chúng tôi tạo bộ dữ liệu mẫu cung cấp một tập hợp chuỗi các sự kiện tương tác(theo phiên của người dùng trong ngày) dưới dạng tập huấn luyện. Các phiên sẽ có lượng tương tác khác nhau. Với một chuỗi phim được người dùng tương tác mới, mục tiêu là dự đoán những phim mà người dùng sẽ xem, nếu có. Những thông tin như vậy rất có giá trị đối với các doanh nghiệp điện tử, vì nó có thể chỉ ra không chỉ những loại phim/ sản phẩm nào cần gợi ý cho người dùng mà còn cả cách khuyến khích người dùng tương tác ngay để thúc đẩy doanh thu. Ví dụ: nếu hệ thống gợi ý có thể xác định rằng người dùng không có ý định tương tác với phim trong phiên hiện tại, thì nó có thể cố gắng thay đổi hành vi này bằng cách mở rộng cung cấp cho người dùng các khuyến mại, quảng cáo dành riêng cho người dùng theo lịch sử tương tác của phiên đó.

**Các mô hình nghiên cứu được khảo sát thử nghiệm và lựa chọn đề xuất**:

* Xây dựng mô hình khuyến nghị phim theo phiên dựa trên mạng học sâu GRU
* Xây dựng mô hình khuyến nghị phim theo phiên dựa trên mạng học sâu tự chú ý
* Xây dựng mô hình khuyến nghị mở rộng quy mô áp dụng kiến trúc Transformer theo phiên tương tác của khách hàng

**Danh sách phương pháp xử lý các đặc trưng bao gồm**:

* xử lý dữ liệu nhiễu,
* xử lý các đặc trưng null,
* rời rạc hóa,
* chuẩn hóa,
* kết hợp đặc trưng, lựa chọn đặc trưng,
* tạo phiên tương tác,
* lọc dữ liệu tương tác theo phiên,
* xử lý thời gian tương tác theo phiên phù hợp cho nhóm mô hình.

**Các phương pháp đánh giá chính**:

* Recall@K
* MRR@K
* *(với K=30 và K=50).*

## Đề xuất luồng giải pháp xây dựng mô hình

Nhóm nghiên cứu tiến hành đánh giá tìm hiểu bài toán và phân tích các tập đặc trưng cần thiết cho mô hình phù hợp với dữ liệu đầu vào. Tập dữ liệu đầu vào có nhiều đặc trưng khác nhau bao gồm thông tin định danh người dùng, thông tin đặc trưng phim, hành vi tương tác, thời gian tương tác, yếu tố về độ thưa của dữ liệu. Các hành vi được xử lý, trực quan hóa và kiểm tra chất lượng theo từng phương pháp cụ thể trước khi đưa vào mô hình. Qua đó, triển khai các mô hình và phân tích kết quả thực nghiệm để lựa chọn giải pháp phù hợp với bối cảnh hiện tại và phát triển cho các giai đoạn tiếp theo của dự án.

**Phân tích và trực quan hóa dữ liệu:** Bước này cần tiến hành thống kê và khai thác các khía cạnh như số lượng người dùng, số lượng sản phẩm (phim), phân bố người dùng theo tần suất, các phân bố tương tác giữa người dùng và phim, v.v. Phân tích và trực quan hóa dữ liệu để có góc nhìn sâu rộng về hiện trạng dữ liệu, kết hợp với bước Tiền xử lý dữ liệu để đưa đầu vào phù hợp cho các mô hình.

**Tiền xử lý dữ liệu:** Làm sạch và xử lý trước dữ liệu để loại bỏ các giá trị bị thiếu, xử lý các giá trị ngoại lai và chuyển đổi dữ liệu thành định dạng có thể sử dụng được. Bước này là cần thiết để đảm bảo chất lượng của mô hình dự đoán. Cụ thể, do các đặc trưng thực tế có tính chất phức tạp, đa dạng, nên cần phân nhóm các dạng đặc trưng để tiền xử lý bao gồm các bước làm sạch và chuẩn hóa, xử lý giá trị NULL, rời rạc hóa, tạo ra các đặc trưng mới từ dữ liệu gốc để phù hợp với đầu vào mô hình.

**Xác định biến mục tiêu:** Xác định biến mục tiêu cho mô hình. Trong trường hợp này, để dự đoán danh sách các bộ phim có khả năng tương tác với người dùng cao nhất, ta có thể quy về bài toán dự đoán liên kết trong đồ thị với mỗi node trong đồ thị biểu hiện cho người dùng hoặc sản phẩm và học biểu diễn mỗi node dựa trên các kỹ thuật tính toán đồ thị phổ.

**Lựa chọn mô hình:** Chọn một mô hình khai thác các kỹ thuật dựa trên đồ thị phổ phù hợp để dự đoán khả năng tương tác giữa người dùng và phim. Qua quá trình khảo sát và nghiên cứu (phần 1.2), tập các mô hình giải pháp đa dạng được đề xuất dựa trên các kỹ thuật học biểu diễn và nâng cấp khác nhau, bao gồm:

* Xây dựng mô hình khuyến nghị phim theo phiên dựa trên mạng học sâu GRU (hay GRU-based)
* Xây dựng mô hình khuyến nghị phim theo phiên dựa trên mạng học sâu tự chú ý
* Xây dựng mô hình khuyến nghị mở rộng quy mô áp dụng kiến trúc Transformer theo phiên tương tác của khách hàng (Transformer-based)

**Huấn luyện mô hình:** Điều chỉnh mô hình đã chọn trên dữ liệu huấn luyện bằng cách sử dụng các tính năng và biến mục tiêu. Sau khi xây dựng các model based của các mô hình, thực hiện fine-tunning để điều chỉnh mô hình đã được huấn luyện ban đầu trở nên phù hợp hơn với tập dữ liệu đầu vào. Fine-tunning giúp cải thiện hiệu suất dự đoán và tăng độ tin cậy của mô hình. Thực hiện phân tích kết quả dự đoán để hiểu rõ hơn về hiệu suất và hành vi của mô hình. Xem xét các dự đoán sai và xác định nguyên nhân gây ra sự không chính xác, từ đó cải thiện mô hình.

**Đánh giá mô hình:** Đánh giá hiệu suất của mô hình bằng cách sử dụng các số liệu đánh giá thích hợp như Recall và MRR với top-K các sản phẩm được khuyến nghị.

Điều quan trọng cần lưu ý là việc xây dựng một mô hình dự đoán chính xác đòi hỏi phải có kiến thức về miền, kiến thức chuyên môn về phân tích dữ liệu và xem xét cẩn thận các yếu tố khác nhau có thể ảnh hưởng đến các hoạt động của mô hình.

## Các vấn đề cần giải quyết

Trong giai đoạn đầu tiên, các công việc cần giải quyết được tóm tắt trong Bảng 1.1 dựa theo các vấn đề chính về dữ liệu, mục tiêu bài toán và mô hình đề xuất như sau:

* Xử lý dữ liệu
* Phân tích các đặc trưng dữ liệu
* Trực quan hóa dữ liệu
* Biến đổi dữ liệu phù hợp với đầu vào và đầu ra của bài toán
* Xác định các mô hình/ giải pháp phù hợp với bài toán
* Triển khai cài đặt các mô hình/ giải pháp đề xuất:
  + Xây dựng mô hình khuyến nghị phim theo phiên dựa trên mạng học sâu GRU (hay GRU-based)
  + Xây dựng mô hình khuyến nghị phim theo phiên dựa trên mạng học sâu tự chú ý
  + Xây dựng mô hình khuyến nghị mở rộng quy mô áp dụng kiến trúc Transformer theo phiên tương tác của khách hàng (Transformer-based)
* Phân tích kết quả thực nghiệm của các mô hình/ giải pháp đề xuất

Tóm tắt các vấn đề cần giải quyết

| **STT** | **Công việc** |
| --- | --- |
| 1 | Tìm hiểu hiện trạng vấn đề/ bài toán |
| 2 | Khảo sát các giải pháp liên quan |
| 3 | Phân tích và đề xuất các hướng giải pháp xây dựng cụ thể cho bài toán |
| 4 | Tổng hợp dữ liệu trên môi trường dev |
| 5 | Tổng hợp features nhóm dữ liệu categorical trên môi trường dev |
| 6 | Thực hiện feature engineering xử lý nhóm dữ liệu categorical trên môi trường dev |
| 7 | Tổng hợp features nhóm dữ liệu không hợp lệ trên môi trường dev |
| 8 | Thực hiện feature engineering nhóm dữ liệu không hợp lệ trên môi trường dev |
| 9 | Thực hiện feature engineering chuẩn hóa dữ liệu trên môi trường dev |
| 10 | Tổng hợp dữ liệu trên môi trường test |
| 11 | Tổng hợp features nhóm dữ liệu categorical trên môi trường test |
| 12 | Thực hiện feature engineering xử lý nhóm dữ liệu categorical trên môi trường test |
| 13 | Tổng hợp features nhóm dữ liệu không hợp lệ trên môi trường test |
| 14 | Thực hiện feature engineering nhóm dữ liệu không hợp lệ trên môi trường test |
| 15 | Thực hiện feature engineering chuẩn hóa dữ liệu trên môi trường test |
| 16 | Trực quan hóa và phân tích đặc trưng tương tác của các phim bộ trong hệ thống |
| 17 | Trực quan hóa và phân tích đặc trưng tương tác của các nhóm thể loại phim được quan tâm trong hệ thống |
| 18 | Trực quan hóa và phân tích đặc trưng hành vi sử dụng của nhóm khách hàng trong hệ thống |
| 19 | Trực quan hóa và phân tích đặc trưng hành vi sử dụng của nhóm khách hàng trong hệ thống |
| 20 | Tổng hợp và phân tích các nhóm phim thiên khiến trong hệ thống |
| 20 | Tổng hợp và phân tích các nhóm phim thiên khiến trong hệ thống |
| 21 | Tổng hợp và xử lý các nhóm phim hiếm tương tác trong hệ thống |
| 22 | Tổng hợp và xử lý các nhóm khách hàng hiếm tương tác trong hệ thống |
| 23 | Xử lý feature tương thích với các mô hình giải pháp |
| 24 | Cấu tạo các dữ liệu huấn luyện và thử nghiệm phù hợp cho các mô hình gợi ý phim bộ theo phiên |
| 25 | Khuyến nghị và đề xuất |
| 26 | Xây dựng kiến trúc mô hình GRU-based cho khuyến nghị phim bộ theo phiên |
| 27 | Xây dựng kịch bản thực nghiệm tinh chỉnh mô hình GRU-based trên bộ dữ liệu mẫu (round 1) trên môi trường dev |
| 28 | Xây dựng kịch bản thực nghiệm tinh chỉnh mô hình GRU-based trên bộ dữ liệu mẫu (round 2) trên môi trường dev |
| 29 | Xây dựng kịch bản thực nghiệm tinh chỉnh mô hình GRU-based trên bộ dữ liệu mẫu (round 3) trên môi trường dev |
| 30 | Xây dựng kịch bản thực nghiệm tinh chỉnh mô hình GRU-based trên bộ dữ liệu mẫu (round 4) trên môi trường dev |
| 31 | Xây dựng kịch bản thực nghiệm tinh chỉnh mô hình GRU-based trên bộ dữ liệu mẫu (round 5) trên môi trường dev |
| 32 | Đánh giá kết quả khuyến nghị của mô hình GRU-based trên môi trường dev |
| 33 | Nhận xét hiệu suất mô hình GRU-based trên môi trường dev |
| 34 | Xây dựng kịch bản thực nghiệm tinh chỉnh mô hình GRU-based trên bộ dữ liệu mẫu (round 1) trên môi trường test |
| 35 | Xây dựng kịch bản thực nghiệm tinh chỉnh mô hình GRU-based trên bộ dữ liệu mẫu (round 2) trên môi trường test |
| 36 | Xây dựng kịch bản thực nghiệm tinh chỉnh mô hình GRU-based trên bộ dữ liệu mẫu (round 3) trên môi trường test |
| 37 | Xây dựng kịch bản thực nghiệm tinh chỉnh mô hình GRU-based trên bộ dữ liệu mẫu (round 4) trên môi trường test |
| 38 | Xây dựng kịch bản thực nghiệm tinh chỉnh mô hình GRU-based trên bộ dữ liệu mẫu (round 5) trên môi trường test |
| 39 | Đánh giá kết quả khuyến nghị của mô hình GRU-based trên môi trường test |
| 40 | Nhận xét hiệu suất mô hình GRU-based trên môi trường test |
| 41 | Khuyến nghị và đề xuất mô hình GRU-based |
| 42 | Xây dựng kiến trúc mô hình Mạng nơ-ron chú ý cho khuyến nghị phim bộ theo phiên |
| 43 | Xây dựng kịch bản thực nghiệm tinh chỉnh mô hình Mạng nơ-ron chú ý trên bộ dữ liệu mẫu (round 1) trên môi trường dev |
| 44 | Xây dựng kịch bản thực nghiệm tinh chỉnh mô hình Mạng nơ-ron chú ý trên bộ dữ liệu mẫu (round 2) trên môi trường dev |
| 45 | Xây dựng kịch bản thực nghiệm tinh chỉnh mô hình Mạng nơ-ron chú ý trên bộ dữ liệu mẫu (round 3) trên môi trường dev |
| 46 | Xây dựng kịch bản thực nghiệm tinh chỉnh mô hình Mạng nơ-ron chú ý trên bộ dữ liệu mẫu (round 4) trên môi trường dev |
| 47 | Xây dựng kịch bản thực nghiệm tinh chỉnh mô hình Mạng nơ-ron chú ý trên bộ dữ liệu mẫu (round 5) trên môi trường dev |
| 48 | Đánh giá kết quả khuyến nghị của mô hình Mạng nơ-ron chú ý trên môi trường dev |
| 49 | Nhận xét hiệu suất mô hình Mạng nơ-ron chú ý trên khuyến nghị phiên trên môi trường dev |
| 50 | Xây dựng kịch bản thực nghiệm tinh chỉnh mô hình Mạng nơ-ron chú ý trên bộ dữ liệu mẫu (round 1) trên môi trường test |
| 51 | Xây dựng kịch bản thực nghiệm tinh chỉnh mô hình Mạng nơ-ron chú ý trên bộ dữ liệu mẫu (round 2) trên môi trường test |
| 52 | Xây dựng kịch bản thực nghiệm tinh chỉnh mô hình Mạng nơ-ron chú ý trên bộ dữ liệu mẫu (round 3) trên môi trường test |
| 53 | Xây dựng kịch bản thực nghiệm tinh chỉnh mô hình Mạng nơ-ron chú ý trên bộ dữ liệu mẫu (round 4) trên môi trường test |
| 54 | Xây dựng kịch bản thực nghiệm tinh chỉnh mô hình Mạng nơ-ron chú ý trên bộ dữ liệu mẫu (round 5) trên môi trường test |
| 55 | Đánh giá kết quả khuyến nghị của mô hình Mạng nơ-ron chú ý trên môi trường test |
| 56 | Nhận xét hiệu suất mô hình Mạng nơ-ron chú ý trên khuyến nghị phiên trên môi trường test |
| 57 | Khuyến nghị và đề xuất mô hình Mạng nơ-ron chú ý trên khuyến nghị phiên |
| 58 | Xây dựng kiến trúc mô hình Transformer-based cho khuyến nghị phim bộ theo phiên |
| 59 | Xây dựng kịch bản thực nghiệm tinh chỉnh mô hình Transformer-based trên bộ dữ liệu mẫu (round 1) trên môi trường dev |
| 60 | Xây dựng kịch bản thực nghiệm tinh chỉnh mô hình Transformer-based trên bộ dữ liệu mẫu (round 2) trên môi trường dev |
| 61 | Xây dựng kịch bản thực nghiệm tinh chỉnh mô hình Transformer-based trên bộ dữ liệu mẫu (round 3) trên môi trường dev |
| 62 | Xây dựng kịch bản thực nghiệm tinh chỉnh mô hình Transformer-based trên bộ dữ liệu mẫu (round 4) trên môi trường dev |
| 63 | Xây dựng kịch bản thực nghiệm tinh chỉnh mô hình Transformer-based trên bộ dữ liệu mẫu (round 5) trên môi trường dev |
| 64 | Đánh giá kết quả khuyến nghị của mô hình Transformer-based trên môi trường dev |
| 65 | Nhận xét hiệu suất mô hình Transformer-based trên khuyến nghị phiên trên môi trường dev |
| 66 | Xây dựng kịch bản thực nghiệm tinh chỉnh mô hình Transformer-based trên bộ dữ liệu mẫu (round 1) trên môi trường test |
| 67 | Xây dựng kịch bản thực nghiệm tinh chỉnh mô hình Transformer-based trên bộ dữ liệu mẫu (round 2) trên môi trường test |
| 68 | Xây dựng kịch bản thực nghiệm tinh chỉnh mô hình Transformer-based trên bộ dữ liệu mẫu (round 3) trên môi trường test |
| 69 | Xây dựng kịch bản thực nghiệm tinh chỉnh mô hình Transformer-based trên bộ dữ liệu mẫu (round 4) trên môi trường test |
| 70 | Xây dựng kịch bản thực nghiệm tinh chỉnh mô hình Transformer-based trên bộ dữ liệu mẫu (round 5) trên môi trường test |
| 71 | Đánh giá kết quả khuyến nghị của mô hình Transformer-based trên môi trường test |
| 72 | Nhận xét hiệu suất mô hình Transformer-based trên khuyến nghị phiên trên môi trường test |
| 73 | Khuyến nghị và đề xuất mô hình Transformer-based |
| 74 | Tổng hợp và trực quan hóa hiệu suất thực nghiệm tốt nhất của các models đề xuất |

|  |  |
| --- | --- |
| 75 | Phân tích và so sánh hiệu quả của models Transformer-based và Mạng nơ-ron chú ý |
| 76 | Phân tích và so sánh hiệu quả của các models GRU-based và Transformer-based |
| 77 | Phân tích và so sánh hiệu quả của các models GRU-based và Mạng nơ-ron chú ý |
| 78 | Tổng kết và khuyến nghị phương án đề xuất và kết quả |

## 

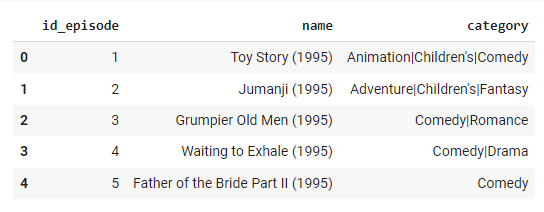
# Phân tích và xử lý dữ liệu

## Xử lý dữ liệu

Với nguồn dữ liệu đầu vào lớn, cần đánh giá tìm hiểu bài toán và phân tích các tập đặc trưng cần thiết cho mô hình. Tập dữ liệu đầu vào có nhiều đặc trưng khác nhau bao gồm hành vi sử dụng dịch vụ phim/chương trình của người dùng; các đặc trưng thông tin của phim (tên, thể loại, ngày sản xuất, ...); thông tin thời điểm và thời lượng tương tác của người dùng với phim. Các hành vi được xử lí, trực quan hóa và kiểm tra chất lượng theo từng phương pháp cụ thể trước khi đưa vào mô hình để lựa chọn những đặc trưng phù hợp nhất tương ứng với mỗi mô hình đề xuất.

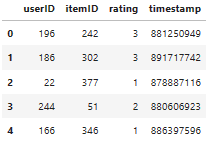
### Tổng hợp các đặc trưng nhóm mô hình

Các đặc trưng nhóm dữ liệu mô hình là tập hợp các thuộc tính hoặc thông tin mô tả về dữ liệu có tính chất nhóm hoặc tương tác trong mô hình học máy và học sâu. Việc xử lý các đặc trưng này cùng một lúc sẽ giúp cải thiện hiệu suất dự đoán của mô hình. Đặc trưng nhóm mô hình có thể kể đến: Các đặc trưng nhóm dữ liệu thông tin phim, hồ sơ người dùng và các đặc trưng nhóm đánh giá của người dùng qua phim.



Đặc trưng về thông tin phim

Đối với bộ dữ liệu mẫu để phục vụ nghiên cứu này, chúng ta quy ước *userID* và *itemID* tương ứng với mã định danh cho hồ sơ người dùng và phim (sản phẩm). Tương tác giữa người dùng và các bộ phim được xác định bằng đánh giá của người dùng với phim đó.



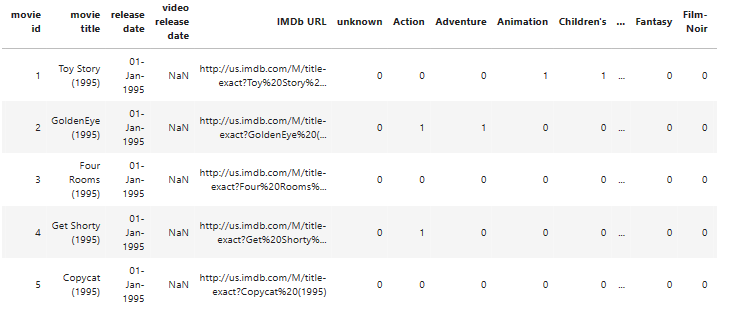
Đặc trưng về đánh giá của người dùng cho phim



Một số đặc trưng về thông tin hồ sơ của người dùng

### Tổng hợp các đặc trưng xử lý nhóm dữ liệu categorical

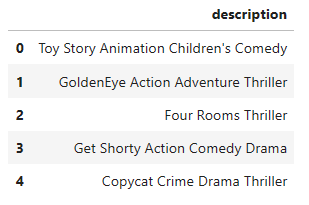
Trong học máy, dữ liệu categorical (biến phân loại) là loại dữ liệu mà các giá trị của nó thuộc vào một tập hợp hữu hạn các nhóm hoặc danh mục. Dữ liệu categorical không thể thực hiện các phép toán số học như các dữ liệu dạng số (numerical). Thay vào đó, chúng được mô tả bằng các nhãn hoặc tên đại diện cho từng danh mục hoặc nhóm riêng biệt. Đặc trưng (feature) của dữ liệu categorical là các biến đại diện cho các nhóm dữ liệu này. Việc xử lý nhóm dữ liệu categorical nhằm tạo biểu diễn đầu vào cho các mô hình thích hợp.



Tổng hợp các đặc trưng mô tả thể loại của bộ phim

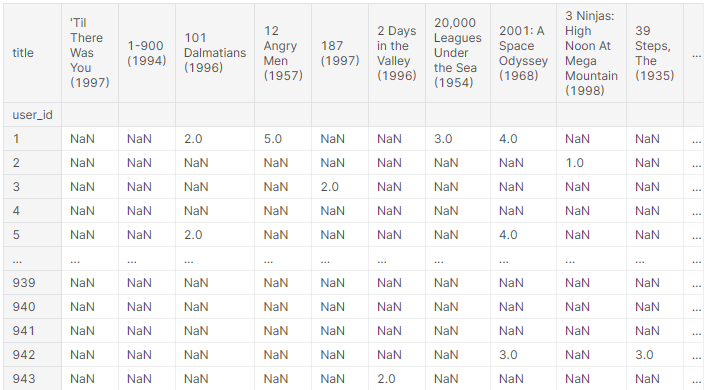
### Kết hợp dữ liệu và xử lý dữ liệu nhiễu

Ensemble data là một khái niệm được sử dụng khi ta kết hợp (ensemble) các tập dữ liệu từ nhiều nguồn khác nhau để tạo ra một tập dữ liệu lớn hơn hoặc cải thiện tính đại diện của dữ liệu. Ý tưởng chính là kết hợp dữ liệu từ nhiều nguồn để có được cái nhìn tổng quát hơn và đa dạng hơn về tập dữ liệu mục tiêu. Kết quả của việc kết hợp dữ liệu người dùng và phim như sau:



Kết hợp dữ liệu tên và thể loại tạo trường thông tin mô tả phim

Làm sạch và chuẩn hóa dữ liệu tiêu dùng là quá trình loại bỏ các giá trị lỗi, không chính xác hoặc thiếu sót trong dữ liệu và chuyển đổi nó thành định dạng chuẩn để phân tích và sử dụng. Cụ thể trong bài toán sẽ tìm các dòng dữ liệu NaN và xử lý các giá trị NaN đó.



Xử lý dữ liệu NaN trong ma trận tiện ích của người dùng và phim

Với dữ liệu văn bản mô tả phim, chúng ta cần đưa qua các mô hình ngôn ngữ để trích chọn các đặc trưng quan trọng thể hiện được ngữ nghĩa của văn bản mô tả ứng với mỗi phim. Trong quá trình thực nghiệm nghiên cứu này, nhóm nghiên cứu đề xuất sử dụng mô hình "Sentence-BERT: Nhúng câu bằng cách sử dụng mạng Siamese BERT." Những biểu diễn nhúng qua mô hình của mỗi thông tin mô tả phim có thể được so sánh độ tương đồng về mặt ngữ nghĩa, ví dụ như bằng cách sử dụng độ tương đồng cosine.

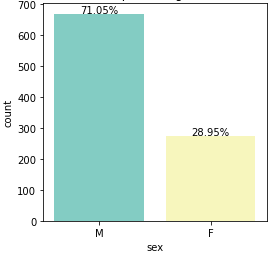
### Thực hiện chuẩn hóa scale

Mục tiêu của việc chuẩn hóa scale là chuyển đổi dữ liệu ban đầu thành dạng có trung bình bằng 0 và độ lệch chuẩn bằng 1, đảm bảo rằng các đặc trưng có phạm vi giá trị tương đương và không ảnh hưởng mạnh đến mô hình.

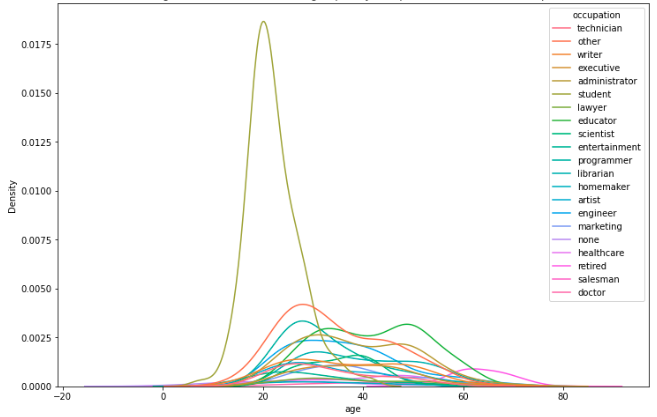
Chúng tôi sử dụng StandardScaler cho việc chuẩn hóa dữ liệu. Nó trừ đi trung bình của từng đặc trưng và chia kết quả cho độ lệch chuẩn của từng đặc trưng. Kết quả cuối cùng là dữ liệu có trung bình bằng 0 và độ lệch chuẩn bằng 1.

## Phân tích dữ liệu

Trong bài toán gợi ý phim sử dụng các mô hình gợi ý sản phẩm có tính cá nhân hóa với thông tin người dùng, nhóm nghiên cứu tiến hành một số bước trực quan hoá dữ liệu theo các thông tin hồ sơ và hành vi sử dụng sản phẩm của người dùng trong hệ thống.

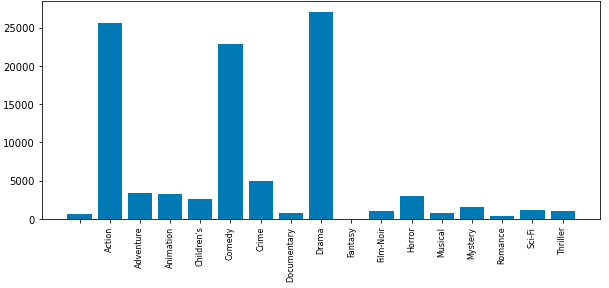


Kết quả trực quan hóa dữ liệu theo phân bố giới tính của người dùng

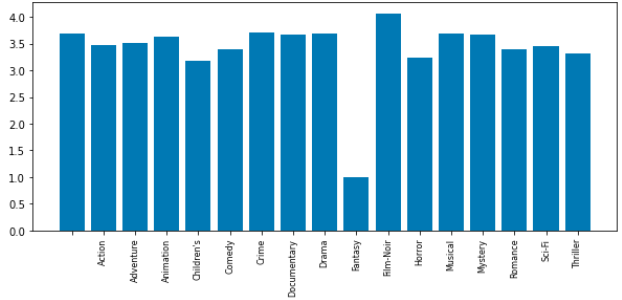


Kết quả trực quan hoá dữ liệu theo phân bố độ tuổi và nghề nghiệp của người dùng

Nhóm nghiên cứu tiến hành phân tích và xác định thể loại phim nào nhận được nhiều tương tác nhất (hình 2.9) và thể loại phim nào nhận được nhiều đánh giá tích cực nhất từ người dùng (hình 2.10).

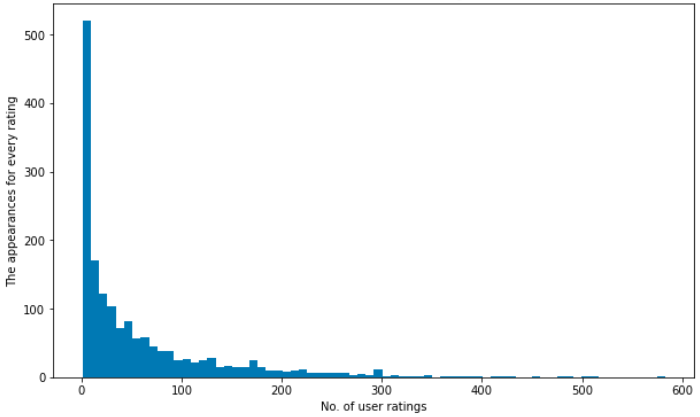


Trực quan hoá dữ liệu theo số lượt được tương tác của mỗi loại phim



Trực quan hoá dữ liệu theo điểm số được yêu thích của mỗi loại phim

Hình 2.11 mô tả phân bố số lần tương tác của các người dùng với các bộ phim trong hệ thống. Phần lớn các người dùng trong hệ thống chỉ tương tác dưới 10 bộ phim, và tồn tại một lượng không nhỏ người dùng đã tương tác hơn 100 bộ phim.



Trực quan hoá dữ liệu mô tả phân bố số lần tương tác của các người dùng với các bộ phim trong hệ thống

# Các mô hình nghiên cứu đề xuất

Khuyến nghị theo thời gian thực được cá nhân hóa là một tính năng quan trọng được áp dụng rộng rãi đối với các nền tảng trực tuyến như NetFlix, VieOn hay TV360. Trong khi những tiến bộ gần đây trong mô hình học sâu đã mang lại kết quả đầy hứa hẹn trong các đề xuất theo phiên, các hệ thống được thiết lập như GRU4Rec dựa trên mạng nơ ron hồi quy, và SASRec dựa trên mạng học sâu tự chú ý đã mang lại kết quả nghiên cứu đầy hứa hẹn nhưng thường gặp khó khăn trong việc duy trì độ chính xác và khả năng mở rộng khi xử lý với khối lượng sản phẩm lớn. Bên cạnh các mô hình phổ biến mạnh mẽ dựa trên mạng học sâu như GRU hay cơ chế tự chú ý, chúng tôi đề xuất cải tiến mô hình khuyến nghị mở rộng quy mô áp dụng kiến trúc Transformer theo phiên tương tác của khách hàng (*Recommender with Scalable Transformer-based User’s Session – TRON*) bằng cách sử dụng các hàm mất mát và lấy mẫu âm được tối ưu hóa tích hợp các chức năng lấy mẫu âm tính top-K và hàm mất mát theo danh sách phim để nâng cao độ chính xác của khuyến nghị. Các mô hình đề xuất cho bài toán khuyến nghị cho khách hàng theo phiên được mô tả chi tiết dưới đây.

## Mô hình khuyến nghị với kiến trúc mạng Gated Recurrent Unit theo phiên tương tác của khách hàng

Hệ thống khuyến nghị phim bộ trên nền tảng TV360 theo phiên dự đoán sản phẩm (phim) tiếp theo mà khách hàng sẽ tương tác dựa trên các hoạt động trước đó của họ trong phiên. Phiên là một chuỗi các tương tác giữa người dùng và sản phẩm theo thời gian thực, được biểu thị dưới dạng trong đó là độ dài phiên.

GRU4Rec là một mô hình được đề xuất cho bài toán hệ thống khuyến nghị theo phiên (Session-based Recommendation System) trên nền tảng số TV360. GRU4Rec là viết tắt của "Gated Recurrent Unit for Recommender Systems." Đây là một mô hình sử dụng mạng GRU, một dạng cơ bản của mạng nơ-ron hồi quy (RNN), để mô hình hóa hành vi người dùng trong hệ thống khuyến nghị.

Cụ thể, GRU4Rec được thiết kế để mô hình dữ liệu chuỗi, chẳng hạn như lịch sử tương tác của người dùng với các sản phẩm. Mô hình này sử dụng cổng (gate) để kiểm soát việc truyền thông tin trong quá trình học. Điều này giúp nó nhớ thông tin quan trọng và loại bỏ thông tin không quan trọng, điều này thường xuyên quan trọng trong các ứng dụng khuyến nghị.

GRU4Rec có thể được sử dụng trong bối cảnh khuyến nghị các gói sản phẩm dịch vụ, gói phim bộ, hoặc các sản phẩm tương thích với người dùng dựa trên lịch sử tiếp xúc của họ với hệ thống theo thời gian thực.

### Xây dựng thuật toán và kiến trúc mô hình

GRU4Rec ban đầu dành cho các khuyến nghị dựa trên phiên, trong đó các phiên thường ngắn được coi là độc lập. Mỗi khi người dùng truy cập nền tảng số TV360, họ có thể được coi là không xác định, tức là không có gì trong lịch sử của họ được sử dụng, ngay cả khi nó được biết đến. Thiết lập này rất phù hợp với nhiều ứng dụng trên hệ thống. Điều này có nghĩa là khi mô hình được đánh giá, trạng thái ẩn bắt đầu từ 0 cho mỗi phiên kiểm tra.

Tuy nhiên, các mô hình dựa trên RNN cũng rất phù hợp cho việc thiết lập khuyến nghị cá nhân hóa nhận biết chuỗi trình tự thực tế ít quan trọng hơn (tức là toàn bộ lịch sử người dùng được sử dụng làm trình tự để dự đoán các mục trong tương lai trong trình tự). Có hai điểm khác biệt chính:

(1) Các trình tự dài hơn đáng kể trong các đề xuất nhận biết trình tự. Điều này cũng có nghĩa là BPTT (lan truyền ngược theo thời gian) rất hữu ích trong trường hợp này. Đối với các đề xuất dựa trên phiên, các thiết kế BPTT không cải thiện mô hình.

(2) Việc đánh giá trong thiết lập nhận biết chuỗi trình tự phải được bắt đầu từ giá trị cuối cùng của trạng thái ẩn (tức là giá trị được tính trên phần huấn luyện của lịch sử người dùng).

Qua nhận định và nghiên cứu, chương này mô tả luồng áp dụng kiến trúc mô hình GRU4Rec (Gated Recurrent Unit for Recommendation) được thiết kế đặc biệt cho hệ thống khuyến nghị, đặc biệt là trong ngữ cảnh của việc khuyến nghị sản phẩm trong thời gian thực trong các phiên tương tác của người dùng. Dưới đây là mô tả thuật toán và kiến trúc của GRU4Rec cho bài toán:

1. Đầu vào:

* Sequential Interaction Data: Dữ liệu đầu vào là chuỗi các sự kiện tương tác của người dùng trong phiên hiện tại. Các sự tương tác này có thể bao gồm xem, mua sắm, đánh giá sản phẩm, và các hành vi khác.
* Embedding Layer: Sản phẩm được biểu diễn dưới dạng vector thông qua một lớp nhúng (embedding layer). Pha này giúp giảm chiều dữ liệu và tạo ra biểu diễn số chiều thấp hơn cho mỗi sản phẩm.

2. Mạng GRU (Gated Recurrent Unit):

* Temporal Learning: Các vector nhúng của sản phẩm được đưa vào mạng GRU theo thứ tự thời gian trong phiên tương tác hiện tại. Mỗi bước thời gian đại diện cho một sự tương tác cụ thể.
* Hidden State: GRU duy trì một trạng thái ẩn (hidden state) để lưu giữ thông tin về lịch sử tương tác của người dùng trong phiên hiện tại.

3. Dự đoán:

* Cổng đầu ra: Cổng đầu ra của GRU được sử dụng để dự đoán sản phẩm tiếp theo cho người dùng trong phiên hiện tại. Cổng này thường được kết nối với một lớp softmax để tạo ra phân phối xác suất cho tất cả các sản phẩm có thể.

4. Huấn luyện

* Online Learning: Mô hình GRU4Rec có khả năng thích ứng với dữ liệu mới ngay khi nó được thêm vào mà không cần huấn luyện lại toàn bộ mô hình. Điều này làm cho nó phù hợp cho việc triển khai trong môi trường thời gian thực.

GRU4Rec xử lý dữ liệu theo phiên, giúp nó nhận diện các mẫu tương tác đặc biệt trong mỗi phiên của người dùng. Qua đó, mô hình tập trung vào việc dự đoán sản phẩm tiếp theo trong phiên hiện tại của người dùng.

### Thực nghiệm và phân tích

Nhóm nghiên cứu tiến hành xây dựng và thực nghiệm mô hình GRU4Rec cho khuyến nghị phim bộ theo phiên trên bộ dữ liệu mẫu từ TV360 với chuỗi lịch sử tương tác giữa khách hàng và các bộ phim. Mô hình được cài đặt thực nghiệm sử dụng các gói hỗ trợ sau:

* python 3.8
* matplotlib == 3.3.3
* pytorch-lightning==2.1.0
* torch==1.13.1
* tensorboardx
* onnx
* pandas
* gdown
* mlflow
* tqdm
* pytest
* ipykernel
* autopep8
* yapf

Trong quá trình thực nghiệm nghiên cứu mô hình cho bài toán, thực nghiệm được tinh chỉnh với 5 kịch bản khác nhau trên bộ dữ liệu thực nghiệm nghiên cứu Yoochoose – dữ liệu thể hiện hoạt động của người dùng trong sáu tháng với khoảng 33 triệu bản ghi bao gồm các trường: mã định danh của phiên, mốc thời gian tương tác, mã định danh của sản phẩm và thể loại sản phẩm.

Mô hình thực nghiệm nghiên cứu được yêu cầu chạy thuật toán trên dữ liệu đào tạo để dự đoán sản phẩm nào sẽ được tương tác trong mỗi phiên trong dữ liệu thử nghiệm. Bộ dữ liệu thực nghiệm nghiên cứu có kích thước lớn giúp tính toán và đo đạc để xác định mức độ triển khai khả thi vào thực tế với dữ liệu của nền tảng TV360.

Các kịch bản thực nghiệm qua các Round được mô tả và tổng hợp chi tiết dưới đây, cụ thể:

**Round** **1**: Khởi tạo các tham số mô hình

* "model": "gru4rec",
* "dataset": "yoochoose",
* "hidden\_size": 100,
* "num\_layers": 1,
* "dropout": 0.0,
* "num\_batch\_negatives": null,
* "num\_uniform\_negatives": 2048,
* "reject\_uniform\_session\_items": false,
* "reject\_in\_batch\_items": true,
* "sampling\_style": "batchwise",
* "loss": "bpr-max",
* "bpr\_penalty": 0.5,
* "max\_epochs": 1,
* "batch\_size": 32,
* "max\_session\_length": 200,
* "lr": 0.2,
* "limit\_val\_batches": 1.0,
* "accelerator": "gpu",
* "overfit\_batches": 0,
* "output\_bias": true,
* "share\_embeddings": true,
* "original\_gru": false,
* "shuffling\_style": "no\_shuffling",
* "final\_activation": true,
* "optimizer": "adagrad"

**Round** **2**:

* "model": "gru4rec",
* "dataset": "yoochoose",
* "hidden\_size": 100,
* "num\_layers": 1,
* "dropout": 0.0,
* "num\_batch\_negatives": null,
* "num\_uniform\_negatives": 2048,
* "reject\_uniform\_session\_items": false,
* "reject\_in\_batch\_items": true,
* "sampling\_style": "batchwise",
* "loss": "ssm",
* "max\_epochs": 3,
* "batch\_size": 32,
* "max\_session\_length": 200,
* "lr": 0.2,
* "limit\_val\_batches": 1.0,
* "accelerator": "gpu",
* "overfit\_batches": 0,
* "output\_bias": true,
* "share\_embeddings": true,
* "original\_gru": false,
* "shuffling\_style": "no\_shuffling",
* "final\_activation": true,
* "optimizer": "adagrad"

**Round** **3**:

* "model": "gru4rec",
* "dataset": "yoochoose",
* "hidden\_size": 200,
* "num\_layers": 2,
* "dropout": 0.0,
* "num\_batch\_negatives": null,
* "num\_uniform\_negatives": 2048,
* "reject\_uniform\_session\_items": false,
* "reject\_in\_batch\_items": true,
* "sampling\_style": "batchwise",
* "loss": "bce",
* "max\_epochs": 3,
* "batch\_size": 32,
* "max\_session\_length": 200,
* "lr": 0.2,
* "limit\_val\_batches": 1.0,
* "accelerator": "gpu",
* "overfit\_batches": 0,
* "output\_bias": true,
* "share\_embeddings": true,
* "original\_gru": false,
* "shuffling\_style": "no\_shuffling",
* "final\_activation": true,
* "optimizer": "adagrad"

**Round 4:**

* "model": "gru4rec",
* "dataset": "yoochoose",
* "hidden\_size": 200,
* "num\_layers": 1,
* "dropout": 0.0,
* "num\_batch\_negatives": null,
* "num\_uniform\_negatives": 2048,
* "reject\_uniform\_session\_items": false,
* "reject\_in\_batch\_items": true,
* "sampling\_style": "eventwise",
* "loss": "bpr-max",
* "bpr\_penalty": 0.5,
* "max\_epochs": 3,
* "batch\_size": 32,
* "max\_session\_length": 200,
* "lr": 0.2,
* "limit\_val\_batches": 1.0,
* "accelerator": "gpu",
* "overfit\_batches": 0,
* "output\_bias": true,
* "share\_embeddings": true,
* "original\_gru": false,
* "shuffling\_style": "no\_shuffling",
* "final\_activation": true,
* "optimizer": "adagrad"

**Round 5:**

* "model": "gru4rec",
* "dataset": "yoochoose",
* "hidden\_size": 100,
* "num\_layers": 1,
* "dropout": 0.05,
* "num\_batch\_negatives": null,
* "num\_uniform\_negatives": 2048,
* "reject\_uniform\_session\_items": false,
* "reject\_in\_batch\_items": true,
* "sampling\_style": "eventwise",
* "loss": "bpr-max",
* "bpr\_penalty": 0.5,
* "max\_epochs": 3,
* "batch\_size": 32,
* "max\_session\_length": 200,
* "lr": 0.2,
* "limit\_val\_batches": 1.0,
* "accelerator": "gpu",
* "overfit\_batches": 0,
* "output\_bias": true,
* "share\_embeddings": true,
* "original\_gru": false,
* "shuffling\_style": "no\_shuffling",
* "final\_activation": true,
* "optimizer": "adagrad"

Các cấu hình trên trình bày chi tiết các tham số và tùy chọn của mô hình.

* model: Mô hình cơ sở sẽ được sử dụng (ví dụ: "sasrec", "gru4rec").
* dataset: Tập dữ liệu được sử dụng để đào tạo
* Hidden\_size: Kích thước của các lớp ẩn và phần nhúng sản phẩm.
* num\_layers: Số lớp trong mô hình.
* dropout: Tỷ lệ loại bỏ áp dụng cho các lớp của mô hình.
* num\_batch\_Negatives: Số lượng mẫu âm tính trong lô. Bị giới hạn bởi batch\_size - 1.
* num\_uniform\_ Negatives: Số lượng mẫu âm được lấy mẫu thống nhất.
* reject\_uniform\_session\_items: Nếu đúng, các sản phẩmtrong cùng một phiên sẽ không được sử dụng làm mẫu âm thống nhất. Trở nên chậm nếu num\_uniform\_ Negatives lớn.
* reject\_in\_batch\_items: Nếu đúng, các sản phẩm trong cùng một phiên sẽ không được sử dụng làm mẫu âm theo lô.
* sampling\_style: Kiểu lấy mẫu âm tính sẽ sử dụng (ví dụ: "theo sự kiện", "theo phiên", ). Có tác động đáng kể đến tốc độ huấn luyện.
* loss: Hàm mất mát sẽ sử dụng (ví dụ: "bce", "bpr-max", "ssm").
* bpr\_penalty: Hệ số phạt cho BPR-Max. Bỏ qua nếu không sử dụng BPR-Max loss.
* max\_epochs: Số lượng epochs đào tạo tối đa.
* batch\_size: Kích thước lô được sử dụng để đào tạo và xác nhận.
* max\_session\_length: Độ dài tối đa của một phiên. Các phiên dài hơn sẽ bị cắt ngắn.
* lr: Tốc độ học của trình tối ưu hóa.
* limit\_val\_batches: Tỉ lệ dữ liệu xác thực sẽ sử dụng cho bước xác thực.
* accelerator: Loại thiết bị được sử dụng để đào tạo (ví dụ: "gpu", "cpu").
* overfit\_batches: Đặt thành 0 để không overfitting. Xem tài liệu PyTorch Lightning để biết thêm chi tiết.
* share\_embeddings: Nếu đúng, trọng số nhúng được chia sẻ thông tin giữa lớp đầu vào và đầu ra.
* output\_bias: Nếu đúng, nó sẽ bao gồm độ thiên khiến trong lớp đầu ra.
* shuffling\_style: Kiểu xáo trộn để sử dụng cho tập dữ liệu huấn luyện (ví dụ:"no\_shuffling","shuffling\_with\_replacement","shuffling\_without\_replacement").
* optimizer: bộ tối ưu hóa được sử dụng cho việc đào tạo (ví dụ: "adam", "adagrad")
* original\_gru: lựa chọn sử dụng mô hình GRU-based gốc (chưa sửa đổi phù hợp với bài toán khuyến nghị theo phiên)

**Đối với kịch bản đánh giá hiệu suất mô hình GRU4Rec trên bộ dữ liệu thực tế hệ thống TV360 được chia theo phiên tương tác sản phảm của khách hàng:**

Kết quả thực nghiệm được tổng hợp với 5 kịch bản tương tự trên bộ dữ liệu thực tế TV360 – dữ liệu thể hiện hoạt động của người dùng trong ba tháng với các bản ghi bao gồm các trường: mã định danh của phiên, mốc thời gian tương tác, mã định danh của phim và thể loại phim.

Mô hình thu được kết quả cuối cùng đánh giá trên Recall@K và MRR@K (đánh giá chính trên độ đo với K={30,50} ).

Độ đo Recall@K trong hệ thống khuyến nghị là một phép đo đánh giá hiệu suất của mô hình dựa trên khả năng nắm bắt và đề xuất những sản phẩm thực sự được người dùng quan tâm trong một danh sách đề xuất có kích thước là K. Các sản phẩm theo phiên được khuyến nghị đánh giá theo công thức sau:

Độ đo “Mean Reciprocal Rank at K (MRR@K)” là một phép đo đánh giá hiệu suất của hệ thống khuyến nghị theo phiên. Độ đo này đo lường khả năng của mô hình khuyến nghị trong việc đặt các sản phẩm được xếp hạng cao về thứ tự ưu tiên trong một danh sách đề xuất có kích thước là K. Độ đo theo MRR@K cung cấp một đánh giá tổng quan về khả năng của hệ thống khuyến nghịđưa ra sản phẩm được xếp hạng cao nhất. Nếu MRR@K cao, điều này chỉ ra rằng mô hình có xu hướng đặt những sản phẩm quan trọng lên đầu danh sách đề xuất.

Tổng hợp hiệu suất model GRU4Rec qua các Round trên bộ dữ liệu thực tế trên nền tảng TV360

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Recall@30** | **Recall@50** | **MRR@30** | **MRR@50** |
| **Round 1** | 0.462 | 0.554 | 0.647 | 0.713 |
| **Round 2** | 0.367 | 0.446 | 0.527 | 0.582 |

Biểu đồ minh họa hiệu suất của mô hình GRU4Rec trong quá trình các round trên bộ dữ liệu mẫu từ TV360

**Nhận xét đánh giá**:

Qua kết quả tổng hợp ở bảng 3.1 và các hình minh họa quá trình huấn luyện mô hình, ta có thể nhận xét về khả năng triển khai của mô hình GRU4Rec như sau:

* Mô hình chuỗi tương tác: GRU4Rec xuất sắc trong việc nắm bắt sự phụ thuộc tuần tự trong các phiên người dùng. Kiến trúc Gated Recurrent Unit (GRU) của nó giúp nó mô hình hóa mặt thời gian của các tương tác người dùng một cách hiệu quả, đảm bảo sự hiểu biết tinh tế về sở thích của nhóm khách hàng theo thời gian. Khả năng của mô hình nắm bắt sự ưa thích ngắn hạn và dài hạn làm cho nó phù hợp cho các tình huống nơi động lực phiên đóng một vai trò quyết định.
* Khả năng học trực tuyến của GRU4Rec giúp nó thích ứng thời gian thực với hành vi người dùng. Điều này đặc biệt quan trọng trong môi trường động nơi sở thích của người dùng thay đổi nhanh chóng.
* Hiệu suất của mô hình phụ thuộc lớn vào sự có sẵn của dữ liệu phiên gần đây để đưa ra các khuyến nghị thời gian thực chính xác.
* Tính mở rộng và hiệu suất huấn luyện: GRU4Rec có khả năng mở rộng và xử lý các bộ dữ liệu lớn một cách hiệu quả. Quá trình huấn luyện của nó được tối ưu hóa cho dữ liệu chuỗi, cho phép học hiệu quả ngay cả với lịch sử phiên rộng lớn.
* Xem xét: Tài nguyên tính toán cần thiết cho việc huấn luyện mô hình có thể thay đổi đáng kể dựa trên kích thước và phức tạp của bộ dữ liệu.
* GRU4Rec là một mô hình linh hoạt và hiệu quả cho các hệ thống đề xuất sản phẩm thời gian thực, đặc biệt là khi người dùng có các phiên tương tác ngắn và muốn nhận được đề xuất nhanh chóng và chính xác.

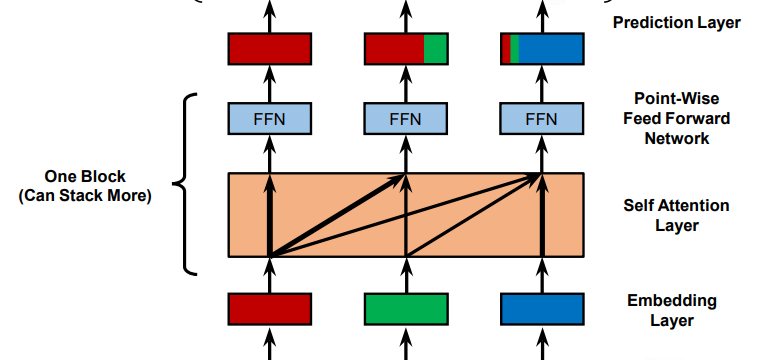
## Mô hình khuyến nghị áp dụng mạng học sâu tự chú ý theo phiên tương tác của khách hàng

Hệ thống khuyến nghị phim bộ trên nền tảng TV360 theo phiên dự đoán sản phẩm (phim) tiếp theo mà khách hàng sẽ tương tác dựa trên các hoạt động trước đó của họ trong phiên. Gần đây, các mô hình dựa theo Transfomer và cơ chế chú ý mới đã đạt được hiệu suất vượt trội. Không giống như các mô hình xử lý tín hiệu chuỗi hiện có sử dụng mô-đun tích chập hoặc hồi quy, phương pháp được đề xuất ở chương này có tên là 'tự chú ý', có hiệu quả cao và có khả năng khám phá các mối quan hệ mang tính ngữ cảnh giữa các thành phần trong tín hiệu chuỗi. Lấy cảm hứng từ phương pháp này, chúng tôi tìm cách áp dụng cơ chế tự chú ý vào các vấn đề khuyến nghị trên phiên theo thời gian thực.

### Xây dựng thuật toán và kiến trúc mô hình

Phương pháp xây dựng mô hình khuyến nghị cho người sử dụng hệ thống theo phiên dựa trên ý tưởng tự chú ý có thể giải quyết cả hai vấn đề: trích rút ra bối cảnh từ tất cả các hành động trong quá khứ (như RNN), và mặt khác, có thể đưa ra các dự đoán chỉ trong một phạm vi nhỏ tính trên số lượng thao tác (như chuỗi Markov). Cụ thể, chúng tôi xây dựng mô hình đề xuất tuần tự dựa trên sự chú ý (SASRec), mô hình này gán trọng số một cách thích ứng cho các mục trước đó ở mỗi giai đoạn thời gian trong một phiên tương tác của người dùng.

Mô hình được đề xuất vượt trội đáng kể so với các phương pháp đề xuất tuần tự dựa trên chuỗi Markov, CNN hay RNN hiện đại trên một số bộ dữ liệu điểm chuẩn. Do cơ chế tự chú ý, SASRec có xu hướng xem xét sự phụ thuộc lâu dài vào các tập dữ liệu dày đặc, đồng thời tập trung vào các hoạt động của người dùng gần đây hơn trên các tập dữ liệu thưa thớt. Điều này chứng tỏ rất quan trọng để xử lý thích ứng các bối cảnh dữ liệu với mật độ khác nhau trên hệ thống TV360. Hơn nữa, thành phần cốt lõi (tức là khối tự chú ý) của SASRec phù hợp để tăng tốc độ tính toán song song, dẫn đến một mô hình nhanh hơn nhiều so với các lựa chọn thay thế dựa trên CNN/RNN. Ngoài ra, chúng tôi đánh giá cao về khả năng mở rộng của SASRec.



Mô tả kiến trúc SASRec trong quá trình huấn luyện. Ở mỗi bước thời gian, mô hình sẽ xem xét tất cả các thao tác trước đó và chú ý đến các đối tượng cần 'tập trung vào' để xác định hành động tiếp theo

Tổng quan về kiến trúc của mô hình SASRec cho hệ thống khuyến nghị dựa trên phiên:

1. Mã hóa đầu vào:

- Các tương tác giữa người dùng và sản phẩm phim tuần tự được biểu diễn dưới dạng một chuỗi tương tác.

- Mỗi vị trí được đánh chỉ mục chuyển đổi thành một vectơ biểu diễn bằng cách sử dụng lớp nhúng.

2. Mã hóa vị trí:

- Để thu nhận được thông tin thời gian, mã hóa vị trí được thêm vào phần nhúng sản phẩm.

- Điều này giúp mô hình phân biệt giữa các sản phẩm dựa trên vị trí của chúng trong chuỗi.

3. Cơ chế tự chú ý:

- Cốt lõi của SASRec là cơ chế tự chú ý, cho phép mô hình tập trung vào các phần khác nhau của chuỗi khi đưa ra dự đoán.

- Khả năng tự chú ý cho phép mô hình cân nhắc tầm quan trọng của các mục khác nhau trong chuỗi dựa trên mức độ liên quan của chúng với mục hiện tại được dự đoán.

- Điểm chú ý được tính bằng cơ chế chú ý theo tỷ lệ chuẩn hóa tích vô hướng.

4. Chú ý đa ngữ cảnh:

- SASRec thường sử dụng nhiều góc độ (head) chú ý để nắm bắt các khía cạnh đa dạng của các mối phụ thuộc tuần tự.

- Sự chú ý đa ngữ cảnh cho phép mô hình tìm hiểu các phép chiếu tuyến tính khác nhau của chuỗi đầu vào, cung cấp các biểu diễn phong phú hơn.

5. Mạng chuyển tiếp theo vị trí:

- Sau các lớp tự chú ý, các mạng chuyển tiếp theo vị trí được áp dụng vào mô hình.

- Các mạng này giới thiệu tính phi tuyến tính và giúp mô hình nắm bắt được các mối quan hệ phức tạp giữa các sản phẩm.

6. Lớp đầu ra:

- Lớp đầu ra tạo ra phân bố xác suất trên tất cả các sản phẩm trong tập dữ liệu cho dự đoán tương tác với sản phẩm nào tiếp theo trong chuỗi.

- Mô hình được huấn luyện để tối đa hóa khả năng xác thực các sản phẩm tiếp theo trong chuỗi huấn luyện.

7. Hàm mục tiêu: Sử dụng hàm Entropy chéo nhị phân để tính toán sự khác biệt giữa phân phối sản phẩm được dự đoán và thao tác thực tế.

### Thực nghiệm và phân tích

Nhóm nghiên cứu tiến hành xây dựng và thực nghiệm mô hình SASRec cho khuyến nghị phim bộ theo phiên trên bộ dữ liệu mẫu từ TV360 với chuỗi lịch sử tương tác giữa khách hàng và các bộ phim. Mô hình được cài đặt thực nghiệm sử dụng các gói hỗ trợ sau:

* python 3.8
* matplotlib == 3.3.3
* pytorch-lightning==2.1.0
* torch==1.13.1
* tensorboardx
* onnx
* pandas
* gdown
* mlflow
* tqdm
* pytest
* ipykernel

Trong quá trình thực nghiệm nghiên cứu mô hình cho bài toán, thực nghiệm được tinh chỉnh với 5 kịch bản khác nhau trên bộ dữ liệu thực nghiệm nghiên cứu Yoochoose – dữ liệu thể hiện hoạt động của người dùng trong sáu tháng với khoảng 33 triệu bản ghi bao gồm các trường: mã định danh của phiên, mốc thời gian tương tác, mã định danh của sản phẩm và thể loại sản phẩm.

Mô hình thực nghiệm nghiên cứu được yêu cầu chạy thuật toán trên dữ liệu đào tạo để dự đoán sản phẩm nào sẽ được tương tác trong mỗi phiên trong dữ liệu thử nghiệm. Bộ dữ liệu thực nghiệm nghiên cứu có kích thước lớn giúp tính toán và đo đạc để xác định mức độ triển khai khả thi vào thực tế với dữ liệu của nền tảng TV360.

Các kịch bản thực nghiệm qua các Round được mô tả và tổng hợp chi tiết dưới đây, cụ thể:

**Round** **1**: Khởi tạo các tham số mô hình

* "model": "sasrec",
* "dataset": "yoochoose",
* "hidden\_size": 200,
* "num\_layers": 2,
* "dropout": 0.05,
* "num\_batch\_negatives": 0,
* "num\_uniform\_negatives": 1,
* "reject\_uniform\_session\_items": true,
* "reject\_in\_batch\_items": true,
* "sampling\_style": "eventwise",
* "loss": "bce",
* "max\_epochs": 1,
* "batch\_size": 128,
* "max\_session\_length": 50,
* "lr": 0.001,
* "limit\_val\_batches": 1.0,
* "accelerator": "gpu",
* "overfit\_batches": 0,
* "share\_embeddings": true,
* "output\_bias": false,
* "shuffling\_style": "no\_shuffling",
* "optimizer": "adam"

**Round** **2**:

* "model": "sasrec",
* "dataset": "yoochoose",
* "hidden\_size": 200,
* "num\_layers": 2,
* "dropout": 0.05,
* "num\_batch\_negatives": 0,
* "num\_uniform\_negatives": 1,
* "reject\_uniform\_session\_items": true,
* "reject\_in\_batch\_items": true,
* "sampling\_style": "batchwise",
* "loss": "ssm",
* "max\_epochs": 1,
* "batch\_size": 128,
* "max\_session\_length": 50,
* "lr": 0.001,
* "limit\_val\_batches": 1.0,
* "accelerator": "gpu",
* "overfit\_batches": 0,
* "share\_embeddings": true,
* "output\_bias": false,
* "shuffling\_style": "no\_shuffling",
* "optimizer": "adam"

**Round** **3**:

* " model": "sasrec",
* "dataset": "yoochoose",
* "hidden\_size": 200,
* "num\_layers": 2,
* "dropout": 0.05,
* "num\_batch\_negatives": 0,
* "num\_uniform\_negatives": 1,
* "reject\_uniform\_session\_items": true,
* "reject\_in\_batch\_items": true,
* "sampling\_style": "eventwise",
* "loss": "bpr-max",
* "bpr\_penalty": 0.5,
* "max\_epochs": 1,
* "batch\_size": 128,
* "max\_session\_length": 50,
* "lr": 0.001,
* "limit\_val\_batches": 1.0,
* "accelerator": "gpu",
* "overfit\_batches": 0,
* "share\_embeddings": true,
* "output\_bias": false,
* "shuffling\_style": "no\_shuffling",
* "optimizer": "adam"

**Round 4:**

* **"**model": "sasrec",
* "dataset": "yoochoose",
* "hidden\_size": 200,
* "num\_layers": 4,
* "dropout": 0.05,
* "num\_batch\_negatives": 0,
* "num\_uniform\_negatives": 1,
* "reject\_uniform\_session\_items": true,
* "reject\_in\_batch\_items": true,
* "sampling\_style": "batchwise",
* "loss": "bce",
* "max\_epochs": 1,
* "batch\_size": 128,
* "max\_session\_length": 50,
* "lr": 0.001,
* "limit\_val\_batches": 1.0,
* "accelerator": "gpu",
* "overfit\_batches": 0,
* "share\_embeddings": true,
* "output\_bias": false,
* "shuffling\_style": "no\_shuffling",
* "optimizer": "adam"

**Round 5:**

* "model": "sasrec",
* "dataset": "yoochoose",
* "hidden\_size": 200,
* "num\_layers": 4,
* "dropout": 0.05,
* "num\_batch\_negatives": 0,
* "num\_uniform\_negatives": 1,
* "reject\_uniform\_session\_items": true,
* "reject\_in\_batch\_items": true,
* "sampling\_style": "eventwise",
* "loss": "ssm",
* "max\_epochs": 1,
* "batch\_size": 128,
* "max\_session\_length": 50,
* "lr": 0.001,
* "limit\_val\_batches": 1.0,
* "accelerator": "gpu",
* "overfit\_batches": 0,
* "share\_embeddings": true,
* "output\_bias": false,
* "shuffling\_style": "no\_shuffling",
* "optimizer": "adam"

Các cấu hình trên trình bày chi tiết các tham số và tùy chọn của mô hình.

* model: Mô hình cơ sở sẽ được sử dụng (ví dụ: "sasrec", "gru4rec").
* dataset: Tập dữ liệu được sử dụng để đào tạo
* Hidden\_size: Kích thước của các lớp ẩn và phần nhúng sản phẩm.
* num\_layers: Số lớp trong mô hình.
* dropout: Tỷ lệ loại bỏ áp dụng cho các lớp của mô hình.
* num\_batch\_Negatives: Số lượng mẫu âm tính trong lô. Bị giới hạn bởi batch\_size - 1.
* num\_uniform\_ Negatives: Số lượng mẫu âm được lấy mẫu thống nhất.
* reject\_uniform\_session\_items: Nếu đúng, các sản phẩmtrong cùng một phiên sẽ không được sử dụng làm mẫu âm thống nhất. Trở nên chậm nếu num\_uniform\_ Negatives lớn.
* reject\_in\_batch\_items: Nếu đúng, các sản phẩm trong cùng một phiên sẽ không được sử dụng làm mẫu âm theo lô.
* sampling\_style: Kiểu lấy mẫu âm tính sẽ sử dụng (ví dụ: "theo sự kiện", "theo phiên", ). Có tác động đáng kể đến tốc độ huấn luyện.
* loss: Hàm mất mát sẽ sử dụng (ví dụ: "bce", "bpr-max", "ssm").
* max\_epochs: Số lượng epochs đào tạo tối đa.
* batch\_size: Kích thước lô được sử dụng để đào tạo và xác nhận.
* max\_session\_length: Độ dài tối đa của một phiên. Các phiên dài hơn sẽ bị cắt ngắn.
* lr: Tốc độ học của trình tối ưu hóa.
* limit\_val\_batches: Tỉ lệ dữ liệu xác thực sẽ sử dụng cho bước xác thực.
* accelerator: Loại thiết bị được sử dụng để đào tạo (ví dụ: "gpu", "cpu").
* overfit\_batches: Đặt thành 0 để không overfitting. Xem tài liệu PyTorch Lightning để biết thêm chi tiết.
* share\_embeddings: Nếu đúng, trọng số nhúng được chia sẻ thông tin giữa lớp đầu vào và đầu ra.
* output\_bias: Nếu đúng, nó sẽ bao gồm độ thiên khiến trong lớp đầu ra.
* shuffling\_style: Kiểu xáo trộn để sử dụng cho tập dữ liệu huấn luyện (ví dụ:"no\_shuffling","shuffling\_with\_replacement","shuffling\_without\_replacement").
* optimizer: bộ tối ưu hóa được sử dụng cho việc đào tạo (ví dụ: "adam", "adagrad")

**Đối với kịch bản đánh giá hiệu suất mô hình SASRec trên bộ dữ liệu thực tế hệ thống TV360 được chia theo phiên tương tác sản phảm của khách hàng:**

Kết quả thực nghiệm được tổng hợp với 5 kịch bản tương tự trên bộ dữ liệu thực tế TV360 – dữ liệu thể hiện hoạt động của người dùng trong ba tháng với các bản ghi bao gồm các trường: mã định danh của phiên, mốc thời gian tương tác, mã định danh của phim và thể loại phim.

Mô hình thu được kết quả cuối cùng đánh giá trên Recall@K và MRR@K (đánh giá chính trên độ đo với K={30,50} ).

Độ đo Recall@K trong hệ thống khuyến nghị là một phép đo đánh giá hiệu suất của mô hình dựa trên khả năng nắm bắt và đề xuất những sản phẩm thực sự được người dùng quan tâm trong một danh sách đề xuất có kích thước là K. Các sản phẩm theo phiên được khuyến nghị đánh giá theo công thức sau:

Độ đo “Mean Reciprocal Rank at K (MRR@K)” là một phép đo đánh giá hiệu suất của hệ thống khuyến nghị theo phiên. Độ đo này đo lường khả năng của mô hình khuyến nghị trong việc đặt các sản phẩm được xếp hạng cao về thứ tự ưu tiên trong một danh sách đề xuất có kích thước là K. Độ đo theo MRR@K cung cấp một đánh giá tổng quan về khả năng của hệ thống khuyến nghị đưa ra sản phẩm được xếp hạng cao nhất. Nếu MRR@K cao, điều này chỉ ra rằng mô hình có xu hướng đặt những sản phẩm quan trọng lên đầu danh sách đề xuất.

Tổng hợp hiệu suất model SASRec qua các Round trên bộ dữ liệu thực tế trên nền tảng TV360

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Recall@30** | **Recall@50** | **MRR@30** | **MRR@50** |
| **Round 1** | 0.462 | 0.554 | 0.647 | 0.713 |
| **Round 2** | 0.367 | 0.446 | 0.527 | 0.582 |

Biểu đồ minh họa hiệu suất của mô hình SASRec trong quá trình các round trên bộ dữ liệu mẫu từ TV360

**Nhận xét đánh giá**:

Qua kết quả tổng hợp ở bảng 3.2 và các hình minh họa quá trình huấn luyện mô hình, ta có thể nhận xét về khả năng triển khai của mô hình SAS4Rec như sau:

* Mô hình hóa chuỗi tương tác theo thời gian: SASRec đáp ứng tốt trong việc nắm bắt các mối quan hệ phụ thuộc và các biểu diễn theo thời gian trong các tương tác tuần tự giữa người dùng và sản phẩm.
* Khả năng chia sẻ thông tin, tham số: Sử dụng cùng một bộ tham số cho tất cả các vị trí trong chuỗi, phát huy hiệu quả tham số.
* Hiệu quả: Cơ chế tự chú ý cho phép mô hình xem xét toàn bộ chuỗi một cách hiệu quả, ngay cả khi có chuỗi tương tác theo phiên dài.
* SASRec không dựa vào yếu tố hồi quy như GRU4Rec hay RNN-based, khiến nó có khả năng mô hình hóa thông tin song song và tính toán hiệu quả hơn.
* Kiến trúc SASRec đã chứng tỏ hiệu suất cạnh tranh trong các kịch bản khuyến nghị dựa trên phiên, đặc biệt khi tính chất tuần tự của các tương tác là rất quan trọng để đưa ra dự đoán chính xác.

## Mô hình khuyến nghị mở rộng quy mô áp dụng kiến trúc Transformer theo phiên tương tác của khách hàng

Hệ thống khuyến nghị phim bộ trên nền tảng TV360 theo phiên dự đoán sản phẩm (phim) tiếp theo mà khách hàng sẽ tương tác dựa trên các hoạt động trước đó của họ trong phiên. Phiên là một chuỗi các tương tác giữa người dùng và sản phẩm theo thời gian thực, được biểu thị dưới dạng trong đó là độ dài phiên. Các sản phẩm mà khách hàng đã tương tác trong một phiên được coi là mẫu dương, ký hiệu là Ngược lại, các sản phẩm mà người dùng chưa tương tác được gọi là mẫu âm, được biểu thị dưới dạng và là tổng số các mục có sẵn. Việc đào tạo một mô hình để thực hiện dự đoán sản phẩm tiếp theo trên thường không khả thi do kích thước lớn của trong kịch bản thực tế. Do đó, một cách tiếp cận phổ biến là huấn luyện mô hình để phân biệt giữa mẫu dương và mẫu âm, điều này có thể đạt được thông qua lấy mẫu âm. Một thách thức lớn trong việc lấy mẫu âm là tính hiệu quả. Việc lấy mẫu trực tiếp từ có thể tốn kém về mặt tính toán vì nó yêu cầu loại trừ các sản phẩm có trong . Vấn đề này trở nên nghiêm trọng khi tăng số lượng mẫu âm, dẫn đến thời gian huấn luyện mô hình kéo dài, ảnh hưởng đến hiệu suất triển khai thực tế.

### Xây dựng thuật toán và kiến trúc mô hình

Kiến trúc mô hình đề xuất trong chương này được dựa theo một kỹ thuật khuyến nghị áp dụng kiến trúc Transformer theo phiên có thể mở rộng bằng cách sử dụng lấy mẫu âm được tối ưu hóa. Được thúc đẩy bởi các hạn chế về khả năng mở rộng và hiệu suất của các mô hình phổ biến như SASRec và GRU4Rec, mô hình khuyến nghị mở rộng quy mô áp dụng kiến trúc Transformer theo phiên tương tác của khách hàng (nhóm nghiên cứu Trung tâm Phân tích Dữ liệu có thể đặt tên là *Recommender with Scalable Transformer-based User’s Session – TRON*) bằng cách sử dụng các hàm mất mát và lấy mẫu âm được tối ưu hóa tích hợp các chức năng lấy mẫu âm tính top-K và hàm mất mát theo danh sách phim để nâng cao độ chính xác của khuyến nghị.

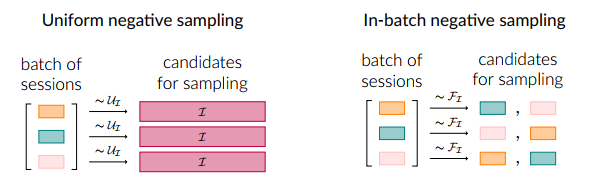
TRON cải tiến dựa trên SASRec, nâng cao khả năng mở rộng và hiệu suất cho các bộ dữ liệu quy mô lớn trên hệ thống TV360. Những cải tiến chính bao gồm:

1. Lấy mẫu âm tính: nhiều mẫu đồng nhất và mẫu trong lô

2. Chỉ lan truyền ngược trên các mẫu âm top-k

3. Hàm mất mát: áp dụng softmax được lấy mẫu

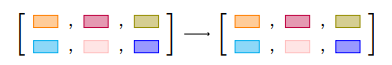
Mô hình sử dụng kết hợp lấy mẫu âm từ phân phối thống nhất và phân phối mật độ theo kinh nghiệm trên bộ sản phẩm . Hãy xem xét một lô bao gồm phiên người dùng. Tại mỗi bước thời gian trong mỗi phiên người dùng , chúng tôi lấy mẫu và Các mẫu này sau đó được ghép nối để tạo thành một vectơ ngẫu nhiên có số chiều với . Các mẫu âm cho toàn bộ lô được biểu diễn dưới dạng .



Mô tả quá trình lấy mẫu âm

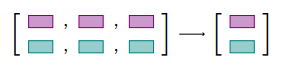
Quá trình lấy mẫu này có thể được thực hiện theo nhiều cách khác nhau, dẫn đến hình dạng âm cực khác nhau có tác động đáng kể đến việc truyền dữ liệu giữa CPU và GPU:

* Lấy mẫu âm theo từng phần tử:



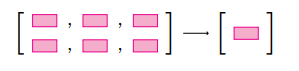
Mô tả quá trình lấy mẫu âm theo từng phần tử

* Lấy mẫu âm theo phiên



Mô tả quá trình lấy mẫu âm theo phiên

* Lấy mẫu âm theo lô



Mô tả quá trình lấy mẫu âm theo lô

Cơ chế chỉ lan truyền ngược trên top-K các mẫu âm: chúng tôi lấy mẫu một tập hợp các mẫu âm và đạt được điểm cho mỗi sản phẩm của phiên tại bước thời gian trong Áp dụng hàm top-k cho các sản phẩm được tính điểm, chúng ta chọn các giá trị âm top-k dựa theo điểm Những sản phẩm top-k này sau đó được sử dụng để cập nhật trong bước lan truyền ngược, trong khi những sản phẩm còn lại sẽ bị loại bỏ.

Hàm mất mát: Áp dụng Softmax được lấy mẫu.

### Thực nghiệm và phân tích

Nhóm nghiên cứu tiến hành xây dựng và thực nghiệm mô hình TRON cho khuyến nghị phim bộ theo phiên trên bộ dữ liệu mẫu từ TV360 với chuỗi lịch sử tương tác giữa khách hàng và các bộ phim. Mô hình được cài đặt thực nghiệm sử dụng các gói hỗ trợ sau:

* python 3.8
* matplotlib == 3.3.3
* pytorch-lightning==2.1.0
* torch==1.13.1
* tensorboardx
* onnx
* pandas
* gdown
* mlflow
* tqdm
* pytest
* ipykernel

Trong quá trình thực nghiệm nghiên cứu mô hình cho bài toán, thực nghiệm được tinh chỉnh với 5 kịch bản khác nhau trên bộ dữ liệu thực nghiệm nghiên cứu Yoochoose – dữ liệu thể hiện hoạt động của người dùng trong sáu tháng với khoảng 33 triệu bản ghi bao gồm các trường: mã định danh của phiên, mốc thời gian tương tác, mã định danh của sản phẩm và thể loại sản phẩm.

Mô hình thực nghiệm nghiên cứu được yêu cầu chạy thuật toán trên dữ liệu đào tạo để dự đoán sản phẩm nào sẽ được tương tác trong mỗi phiên trong dữ liệu thử nghiệm. Bộ dữ liệu thực nghiệm nghiên cứu có kích thước lớn giúp tính toán và đo đạc để xác định mức độ triển khai khả thi vào thực tế với dữ liệu của nền tảng TV360.

Các kịch bản thực nghiệm qua các Round được mô tả và tổng hợp chi tiết dưới đây, cụ thể:

**Round** **1**: Khởi tạo các tham số mô hình

* "hidden\_size": 200,
* "num\_layers": 2,
* "dropout": 0.05,
* "num\_batch\_negatives": 127,
* "num\_uniform\_negatives": 16384,
* "reject\_uniform\_session\_items": false,
* "reject\_in\_batch\_items": true,
* "sampling\_style": "batchwise",
* "topk\_sampling": true,
* "topk\_sampling\_k": 100,
* "loss": "ssm",
* "max\_epochs": 1,
* "batch\_size": 128,
* "max\_session\_length": 50,
* "lr": 0.0005,
* "limit\_val\_batches": 1.0,
* "accelerator": "gpu",
* "overfit\_batches": 0,
* "share\_embeddings": true,
* "output\_bias": false,
* "shuffling\_style": "no\_shuffling",
* "optimizer": "adam"

**Round** **2**:

* "hidden\_size": 200,
* "num\_layers": 2,
* "dropout": 0.05,
* "num\_batch\_negatives": 16,
* "num\_uniform\_negatives": 512,
* "reject\_uniform\_session\_items": false,
* "reject\_in\_batch\_items": true,
* "sampling\_style": "sessionwise",
* "topk\_sampling": true,
* "topk\_sampling\_k": 100,
* "loss": "bce",
* "max\_epochs": 2,
* "batch\_size": 128,
* "max\_session\_length": 50,
* "lr": 0.0005,
* "limit\_val\_batches": 1.0,
* "accelerator": "gpu",
* "overfit\_batches": 0,
* "share\_embeddings": true,
* "output\_bias": false,
* "shuffling\_style": "no\_shuffling",
* "optimizer": "adam"

**Round** **3**:

* "hidden\_size": 200,
* "num\_layers": 2,
* "dropout": 0.05,
* "num\_batch\_negatives": 127,
* "num\_uniform\_negatives": 8192,
* "reject\_uniform\_session\_items": false,
* "reject\_in\_batch\_items": true,
* "sampling\_style": "sessionwise",
* "topk\_sampling": true,
* "topk\_sampling\_k": 100,
* "loss": "bce",
* "max\_epochs": 1,
* "batch\_size": 128,
* "max\_session\_length": 50,
* "lr": 0.0005,
* "limit\_val\_batches": 1.0,
* "accelerator": "gpu",
* "overfit\_batches": 0,
* "share\_embeddings": true,
* "output\_bias": false,
* "shuffling\_style": "no\_shuffling",
* "optimizer": "adam"

**Round 4:**

* "hidden\_size": 200,
* "num\_layers": 2,
* "dropout": 0.05,
* "num\_batch\_negatives": 127,
* "num\_uniform\_negatives": 8192,
* "reject\_uniform\_session\_items": false,
* "reject\_in\_batch\_items": true,
* "sampling\_style": "sessionwise",
* "topk\_sampling": true,
* "topk\_sampling\_k": 100,
* "loss": "bpr-max",
* "bpr\_penalty": 0.125,
* "max\_epochs": 1,
* "batch\_size": 128,
* "max\_session\_length": 50,
* "lr": 0.0005,
* "limit\_val\_batches": 1.0,
* "accelerator": "gpu",
* "overfit\_batches": 0,
* "share\_embeddings": true,
* "output\_bias": false,
* "shuffling\_style": "no\_shuffling",
* "optimizer": "adam"

**Round 5:**

* "hidden\_size": 200,
* "num\_layers": 2,
* "dropout": 0.05,
* "num\_batch\_negatives": 127,
* "num\_uniform\_negatives": 8192,
* "reject\_uniform\_session\_items": false,
* "reject\_in\_batch\_items": true,
* "sampling\_style": "batchwise",
* "topk\_sampling": true,
* "topk\_sampling\_k": 100,
* "loss": "ssm",
* "max\_epochs": 1,
* "batch\_size": 128,
* "max\_session\_length": 50,
* "lr": 0.0005,
* "limit\_val\_batches": 1.0,
* "accelerator": "gpu",
* "overfit\_batches": 0,
* "share\_embeddings": true,
* "output\_bias": false,
* "shuffling\_style": "no\_shuffling",
* "optimizer": "adam"

Các cấu hình trên trình bày chi tiết các tham số và tùy chọn của mô hình.

* model: Mô hình cơ sở sẽ được sử dụng (ví dụ: "sasrec", "gru4rec").
* dataset: Tập dữ liệu được sử dụng để đào tạo
* Hidden\_size: Kích thước của các lớp ẩn và phần nhúng sản phẩm.
* num\_layers: Số lớp trong mô hình.
* dropout: Tỷ lệ loại bỏ áp dụng cho các lớp của mô hình.
* num\_batch\_Negatives: Số lượng mẫu âm tính trong lô. Bị giới hạn bởi batch\_size - 1.
* num\_uniform\_ Negatives: Số lượng mẫu âm được lấy mẫu thống nhất.
* reject\_uniform\_session\_items: Nếu đúng, các sản phẩmtrong cùng một phiên sẽ không được sử dụng làm mẫu âm thống nhất. Trở nên chậm nếu num\_uniform\_ Negatives lớn.
* reject\_in\_batch\_items: Nếu đúng, các sản phẩm trong cùng một phiên sẽ không được sử dụng làm mẫu âm theo lô.
* sampling\_style: Kiểu lấy mẫu âm tính sẽ sử dụng (ví dụ: "theo sự kiện", "theo phiên", ). Có tác động đáng kể đến tốc độ huấn luyện.
* loss: Hàm mất mát sẽ sử dụng (ví dụ: "bce", "bpr-max", "ssm").
* max\_epochs: Số lượng epochs đào tạo tối đa.
* batch\_size: Kích thước lô được sử dụng để đào tạo và xác nhận.
* max\_session\_length: Độ dài tối đa của một phiên. Các phiên dài hơn sẽ bị cắt ngắn.
* lr: Tốc độ học của trình tối ưu hóa.
* limit\_val\_batches: Tỉ lệ dữ liệu xác thực sẽ sử dụng cho bước xác thực.
* accelerator: Loại thiết bị được sử dụng để đào tạo (ví dụ: "gpu", "cpu").
* overfit\_batches: Đặt thành 0 để không overfitting. Xem tài liệu PyTorch Lightning để biết thêm chi tiết.
* share\_embeddings: Nếu đúng, trọng số nhúng được chia sẻ thông tin giữa lớp đầu vào và đầu ra.
* output\_bias: Nếu đúng, nó sẽ bao gồm độ thiên khiến trong lớp đầu ra.
* shuffling\_style: Kiểu xáo trộn để sử dụng cho tập dữ liệu huấn luyện (ví dụ:"no\_shuffling","shuffling\_with\_replacement","shuffling\_without\_replacement").
* optimizer: bộ tối ưu hóa được sử dụng cho việc đào tạo (ví dụ: "adam", "adagrad")
* topk\_sampling: Nếu đúng, việc lấy mẫu âm top-k được kích hoạt.
* topk\_sampling\_k: Nếu kích hoạt topk\_sampling, tham số này chỉ định số lượng mẫu âm top-K sẽ được sử dụng để đào tạo mô hình.

**Đối với kịch bản đánh giá hiệu suất mô hình TRON** **trên bộ dữ liệu thực tế hệ thống TV360 được chia theo phiên tương tác sản phảm của khách hàng:**

Kết quả thực nghiệm được tổng hợp với 5 kịch bản tương tự trên bộ dữ liệu thực tế TV360 – dữ liệu thể hiện hoạt động của người dùng trong ba tháng với các bản ghi bao gồm các trường: mã định danh của phiên, mốc thời gian tương tác, mã định danh của phim và thể loại phim.

Mô hình thu được kết quả cuối cùng đánh giá trên Recall@K và MRR@K (đánh giá chính trên độ đo với K={30,50} ).

Độ đo Recall@K trong hệ thống khuyến nghị là một phép đo đánh giá hiệu suất của mô hình dựa trên khả năng nắm bắt và đề xuất những sản phẩm thực sự được người dùng quan tâm trong một danh sách đề xuất có kích thước là K. Các sản phẩm theo phiên được khuyến nghị đánh giá theo công thức sau:

Độ đo “Mean Reciprocal Rank at K (MRR@K)” là một phép đo đánh giá hiệu suất của hệ thống khuyến nghị theo phiên. Độ đo này đo lường khả năng của mô hình khuyến nghị trong việc đặt các sản phẩm được xếp hạng cao về thứ tự ưu tiên trong một danh sách đề xuất có kích thước là K. Độ đo theo MRR@K cung cấp một đánh giá tổng quan về khả năng của hệ thống khuyến nghị đưa ra sản phẩm được xếp hạng cao nhất. Nếu MRR@K cao, điều này chỉ ra rằng mô hình có xu hướng đặt những sản phẩm quan trọng lên đầu danh sách đề xuất.

Tổng hợp hiệu suất model TRON qua các Round trên bộ dữ liệu thực tế trên nền tảng TV360

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Recall@30** | **Recall@50** | **MRR@30** | **MRR@50** |
| **Round 1** | 0.462 | 0.554 | 0.647 | 0.713 |
| **Round 2** | 0.367 | 0.446 | 0.527 | 0.582 |

Biểu đồ minh họa hiệu suất của mô hình TRON trong quá trình các round trên bộ dữ liệu mẫu từ TV360

**Nhận xét đánh giá**:

Qua kết quả tổng hợp ở bảng 3.3 và các hình minh họa quá trình huấn luyện mô hình, ta có thể nhận xét về khả năng triển khai của mô hình TRON như sau:

* Mô hình TRON được đề xuất của chúng tôi cho thấy độ chính xác vượt trội trên tất cả các tập dữ liệu ngoại trừ MRR@30 trên tập dữ liệu thực nghiệm TV360, đồng thời thể hiện thời gian đào tạo nhanh hơn SASRec SSM do lấy mẫu âm tính theo đợt và top-k.
* TRON chứng tỏ khả năng mở rộng được cải thiện khi tập dữ liệu ngày càng lớn hơn, bằng chứng là tốc độ chậm tương đối giảm so với SASRec.

## Tổng hợp và phân tích hiệu suất các nhóm mô hình khuyến nghị theo phiên

Tổng hợp và đánh giá kết quả của các phương pháp khuyến nghị theo phiên được đề xuất:

Biểu đồ minh họa hiệu suất của các mô hình khuyến nghị theo phiên trong quá trình thực nghiệm các round trên bộ dữ liệu mẫu từ TV360

**Tổng hợp đánh giá và nhận xét**:

Qua kết quả tổng hợp ở bảng 3.4 và các hình minh họa quá trình huấn luyện mô hình, ta có thể nhận xét về khả năng triển khai của các mô hình đề xuất như sau:

Đánh giá ngoại tuyến các thử nghiệm của chúng tôi so với các mô hình đề xuất được trình bày phần trên. Mô hình GRU4Rec ... SASRec trên tất cả các tập dữ liệu.

Mặc dù các nghiên cứu thực nghiệm trước đây về các tập dữ liệu nhỏ hơn chỉ ra tính ưu việt của SASRec so với GRU4Rec, những phát hiện của chúng tôi về các tập dữ liệu lớn hơn và thời gian đào tạo thực tế không tuân theo nhận định trên.

Sự khác biệt này cũng có thể là do phương pháp đánh giá mở rộng của chúng tôi, giúp tránh những điểm yếu liên quan đến đánh giá dựa trên lấy mẫu và không chỉ dựa vào tương tác cuối cùng của phiên.

Mặt khác, SASRec thể hiện thời gian đào tạo chậm hơn trên tất cả các kịch bản thử nghiệm với tập dữ liệu và chỉ cải thiện độ chính xác trên .... Điều này cho thấy rằng việc sử dụng các mẫu âm bổ sung trong hàm mất mát theo điểm như BCE sẽ tác động tiêu cực đến độ chính xác của mô hình.

SAS Rec SSM cho thấy kết quả đầy hứa hẹn, vượt trội so với GRU4Rec+ trên tập dữ liệu Diginetica và thể hiện độ chính xác cạnh tranh cho hai tập dữ liệu còn lại.

Mô hình TRON được đề xuất của chúng tôi cho thấy độ chính xác vượt trội trên tất cả các tập dữ liệu ngoại trừ MRR@30 trên tập dữ liệu thực nghiệm TV360, đồng thời thể hiện thời gian đào tạo nhanh hơn SASRec SSM do lấy mẫu âm tính theo đợt và top-k.

TRON chứng tỏ khả năng mở rộng được cải thiện khi tập dữ liệu ngày càng lớn hơn, bằng chứng là tốc độ chậm tương đối giảm so với SASRec.

Trên tập dữ liệu, TRON cho thấy độ chính xác tăng hơn 6,5% ở cả Recall@20 và MRR@20, cũng như tốc độ đào tạo tăng 1090% so với GRU4Rec+. Mặc dù xử lý nhiều tiêu cực hơn nhưng TRON vẫn duy trì được 92% tốc độ đào tạo của SASRec.