@@bbbbbbbbbbbbbbb

A green and orange logo

Description automatically generated

**TẬP ĐOÀN VIỄN THÔNG QUÂN ĐỘI**

**TÀI LIỆU MÔ TẢ**

**TRIỂN KHAI GỢI Ý CÓ THỂ GIẢI THÍCH ĐƯỢC VỚI ĐỒ THỊ TRI THỨC**

Hà Nội, 2023



# Mục lục

[Mục lục 2](#_Toc154579050)

[Danh mục hình ảnh 4](#_Toc154579051)

[Danh mục bảng biểu 5](#_Toc154579052)

[Phần 1. Đánh giá tìm hiểu bài toán 6](#_Toc154579053)

[1.1. Mô tả bài toán 6](#_Toc154579054)

[1.2. Đề xuất giải pháp 7](#_Toc154579055)

[1.3. Các vấn đề cần giải quyết 8](#_Toc154579056)

[1.4. Xây dựng mô hình đồ thị 10](#_Toc154579057)

[1.4.1. Tổng hợp các đặc trưng nhóm mô hình 10](#_Toc154579058)

[1.4.2. Tạo đồ thị tri thức 11](#_Toc154579059)

[1.5. Trực quan hoá dữ liệu 13](#_Toc154579060)

[1.6. Chia tập dữ liệu huấn luyện và thử nghiệm 15](#_Toc154579061)

[Phần 2. TRIỂN KHAI MÔ HÌNH PGPR 17](#_Toc154579062)

[2.1. Tiền xử lý dữ liệu 17](#_Toc154579063)

[2.2. Nhúng dữ liệu đầu vào 18](#_Toc154579064)

[2.3. Thực nghiệm mô hình PGPR 19](#_Toc154579065)

[2.3.1. Cơ sở lý thuyết 19](#_Toc154579066)

[2.3.2. Kết quả mô hình 25](#_Toc154579067)

[2.4. Đánh giá mô hình 26](#_Toc154579068)

[Phần 3. Triển khai mô hình CAFE 27](#_Toc154579069)

[3.1. Tiền xử lý dữ liệu 27](#_Toc154579070)

[3.2. Nhúng dữ liệu đầu vào 28](#_Toc154579071)

[3.3. Thực nghiệm mô hình CAFE 29](#_Toc154579072)

[3.3.1. Cơ sở lý thuyết 29](#_Toc154579073)

[3.3.2. Kết quả mô hình 32](#_Toc154579074)

[3.4. Đánh giá mô hình 33](#_Toc154579075)

[Phần 4. TỔNG HỢP VÀ ĐÁNH GIÁ KẾT QUẢ 34](#_Toc154579076)

[4.1. Tổng hợp kết quả hai mô hình PGPR và CAFE 34](#_Toc154579077)

[4.2. Đề xuất tích hợp và hướng nâng cấp 34](#_Toc154579078)

# Danh mục hình ảnh

[Hình 1.1. Mô tả đồ thị tri thức 12](#_Toc154579042)

[Hình 1.2. Phân phối chiếc đuôi dài với dữ liệu TV360-1M 14](#_Toc154579043)

[Hình 1.3. Phân phối chiếc đuôi dài với dữ liệu TV360-1M và chia bộ dữ liệu 14](#_Toc154579044)

[Hình 1.4. Phân phối mức yêu thích đối với phim 15](#_Toc154579045)

[Hình 2.1. Quy trình đưa ra đề xuất về phương pháp giải thích theo lộ trình được hướng dẫn bởi chính sách 19](#_Toc154579046)

[Hình 2.2. Mô tả trạng thái của mô hình PGPR 21](#_Toc154579047)

[Hình 2.3. Thuật toán 1: Thuật toán Policy-Guided Path Reasoning 24](#_Toc154579048)

[Hình 3.1. Kiến trúc mô hình CAFE 30](#_Toc154579049)

# Danh mục bảng biểu

[Bảng 1.1. Tóm tắt các vấn đề cần giải quyết 8](#_Toc154579079)

[Bảng 1.2. Bảng mô tả dữ liệu thông tin phim 10](#_Toc154579080)

# Đánh giá tìm hiểu bài toán

## Mô tả bài toán

**Bài toán**: Bài toán gợi ý kèm giải thích là một thách thức quan trọng trong lĩnh vực trí tuệ nhân tạo và học sâu, đặc biệt là trong việc tạo ra các ứng dụng cá nhân hóa. Nhiệm vụ chính của bài toán là xây dựng một mô hình có khả năng dự đoán sở thích của người dùng dựa trên thông tin về hành vi trước đây và các thuộc tính cá nhân.

Để giải quyết bài toán này, cần phải xử lý một lượng lớn dữ liệu đa dạng, bao gồm thông tin các hành vi trước đó của người dùng và các thông tin đặc tính của sản phẩm trên hệ thống.

Mục tiêu cuối cùng của bài toán là xây dựng một mô hình có khả năng hiểu và kết hợp các yếu tố này để đưa ra dự đoán kèm sở thích. Hiệu suất của mô hình được đánh giá thông qua các độ đo như độ chính xác và độ phủ để đảm bảo tính khả thi và độ chính xác của các gợi ý kèm sở thích. Mô hình này sau đó có thể được tích hợp vào ứng dụng để cung cấp trải nghiệm cá nhân hóa, tăng cường tương tác và hài lòng của người dùng trên nền tảng.

**Dữ liệu đầu vào**: Hệ thống gợi ý cần sử dụng dữ liệu gì? Dữ liệu đầu vào có thể bao gồm:

* Lịch sử tương tác của người dùng
* Thông tin về các mục thông tin của sản phẩm

**Các kỹ thuật gợi ý**: Các mô hình học sâu dựa trên đồ thị tri thức được áp dụng cho phương pháp này bao gồm: CoArse-to-FinE neural symbolic reasoning (CAFE) và  Policy-Guided Path Reasoning (PGPR).

**Các chỉ số đánh giá:** Chỉ số đánh giá được ứng dụng trong phương pháp này bao gồm:

* **Độ hồi tưởng xếp hạng k**
* **NDCG xếp hạng k**

## Đề xuất giải pháp

Đối với bài toán gợi ý kèm theo giải thích. Chúng tôi đề xuất hai mô hình học sâu là Policy-Guided Path Reasoning và Coarse-to-Fine Neural Symbolic Reasoning. Các bước thực hiện như sau:

**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.

**Tạo đồ thị tri thức từ các đặc tính của phim và người dùng:** Đồ thị tri thức biểu diễn mối quan hệ giữa phim và người dùng dựa trên tương tác và mối quan hệ tương đồng giữa các phim khác nhau.

**Chia tập huấn luyện và thử nghiệm**: Chia tập dữ liệu huấn luyện và thử nghiệm dựa trên tỷ lệ để phù hợp cho mô hình học và gợi ý kèm giải thích đối với người dùng.

**Lựa chọn mô hình:** Chọn một mô hình học sâu phù hợp để sinh gợi ý có kèm giải thích. Tập các mô hình giải pháp đa dạng được đề xuất, dựa trên các ý tưởng khác nhau, thống nhất sử dụng các mô hình bao gồm Policy-Guided Path Reasoning và Coarse-to-Fine Neural Symbolic Reasoning

**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. Thực hiện đánh giá các đặc trưng quan trọng để xác định những yếu tố nào ảnh hưởng nhiều nhất đến kết quả dự đoán.

**Đánh giá mô hình:** Đánh giá hiệu suất của mô hình bằng cách sử dụng các số liệu đánh giá thích hợp như độ chính xác, độ chính xác, khả năng thu hồi, điểm F1 hoặc diện tích dưới đường cong đặc tính hoạt động của máy thu (AUC-ROC)

## Các vấn đề cần giải quyết

Trong giai đoạn đầu tiên, các vấn đề và công việc cần giải quyết được tóm tắt trong Bảng 1.1 như sau:

Tóm tắt các vấn đề cần giải quyết

| **STT** | **Công việc** |
| --- | --- |
| 1 | **Thử nghiệm Policy Guided Path Reasoning (PGPR)** |
| 2 | Triển khai tiền xử lý dữ liệu cho mô hình cây đồ thị tri thức |
| 3 | Triển khai tỉa cây đồ thị tri thức |
| 4 | Triển khai mô hình PGPR cho toàn tập dữ liệu |
| 5 | Huấn luyện mô hình nhúng TransE |
| 6 | Tạo dữ liệu từ mô hình nhúng TransE |
| 7 | Thực hiện tinh chỉnh tham số mô hình PGPR round 1 |
| 8 | Thực hiện tinh chỉnh tham số mô hình PGPR round 2 |
| 9 | Đánh giá kết quả mô hình PGPR |
| 10 | Phân tích kết quả mô hình PGPR |
| 11 | Thực hiện tạo gợi ý kèm giải thích từ mô hình PGPR |
| 12 | Triển khai tạo gợi ý kèm giải thích lên cơ sở dữ liệu |
| 13 | Thực hiện triển khai thực nghiệm trên tập backtest |
| 14 | Thu thập dữ liệu kết quả trên tập backtest |
| 15 | Phân tích kết quả trên tập backtest |
| 16 | Đánh giá kết quả trên tập backtest |
| 17 | Đưa hướng triển khai tối ưu |
| 18 | Triển khai tối ưu mô hình PGPR |
| 19 | Thực hiện tinh chỉnh tham số mô hình PGPR round 1 sau khi tối ưu |
| 20 | Thực hiện tinh chỉnh tham số mô hình PGPR round 2 sau khi tối ưu |
| 21 | Đánh giá kết quả mô hình PGPR sau khi tối ưu |
| 22 | Phân tích kết quả mô hình PGPR sau khi tối ưu |
| 23 | Thực hiện tạo gợi ý kèm giải thích từ mô hình PGPR sau khi tối ưu |
| 24 | Thực hiện triển khai đánh giá mô hình sau khi tối ưu |
| 25 | Thu thập dữ liệu kết quả trên bộ dữ liệu backtest sau khi tối ưu |
| 26 | Phân tích kết quả trên bộ dữ liệu backtest sau khi tối ưu |
| 27 | Đánh giá kết quả trên bộ dữ liệu backtest sau khi tối ưu |
| 28 | Tiền xử lý và tạo bộ dữ liệu nhúng từ mô hình TranSE cho mô hình CAFE |
| 29 | Triển khai mô hình CAFE cho tập dữ liệu backtest |
| 30 | Thực hiện tinh chỉnh tham số mô hình CAFE round 1 |
| 31 | Thực hiện tinh chỉnh tham số mô hình CAFE round 2 |
| 32 | Đánh giá kết quả mô hình CAFE trên tập dữ liệu backtest |
| 33 | Phân tích kết quả mô hình CAFE trên tập dữ liệu backtest |
| 34 | Thực hiện tạo gợi ý kèm giải thích từ mô hình CAFE |
| 35 | Triển khai luồng tạo gợi ý kèm giải thích lên cơ sở dữ liệu |
| 36 | Thực hiện triển khai thực nghiệm |
| 37 | Thu thập dữ liệu kết quả trên bộ dữ liệu backtest |
| 38 | Phân tích kết quả trên bộ dữ liệu backtest |
| 39 | Đánh giá kết quả trên bộ dữ liệu backtest |
| 40 | Đưa hướng triển khai tối ưu |
| 41 | Triển khai tối ưu mô hình trên bộ dữ liệu backtest |
| 42 | Thực hiện tinh chỉnh tham số mô hình CAFE round 1 sau khi tối ưu |
| 43 | Thực hiện tinh chỉnh tham số mô hình CAFE round 2 sau khi tối ưu |
| 44 | Đánh giá kết quả mô hình CAFE sau khi tối ưu |
| 45 | Phân tích kết quả mô hình CAFE sau khi tối ưu |
| 46 | Thực hiện tạo gợi ý kèm giải thích từ mô hình CAFE sau khi tối ưu |
| 47 | Thực hiện triển khai thực nghiệm sau khi tối ưu |
| 48 | Thu thập dữ liệu kết quả thực tế sau khi tối ưu |
| 49 | Phân tích kết quả của mô hình CAFE trên bộ dữ liệu backtest sau khi tối ưu |
| 50 | Đánh giá kết quả của mô hình CAFÉ trên bộ dữ liệu backtest sau khi tối ưu |

## Xây dựng mô hình đồ thị

### 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 sâu. Việc xử lý các đặc trưng này sẽ giúp mô hình học sâu sử dụng đồ thị trức hiểu rõ các mối quan hệ giữa phim và phim.

Đặc trưng nhóm mô hình có thể kể đến: Các đặc trưng nhóm dữ liệu thông tin phim và các đặc trưng nhóm đánh giá của người dùng qua phim. Dữ liệu sau khi tổng hợp và loại bỏ bớt sẽ như thông tin mô tả dưới đây:”

Bảng mô tả dữ liệu thông tin phim

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tên bảng | Ý nghĩa bảng | Tên trường | Kiểu dữ liệu | Mô tả trường |
| f\_tv360\_film\_category | Thông tin thể loại phim | episode\_id | string | id tập phim |
| episode\_name | string | tên tập phim |
| episode\_description | string | mô tả tập phim |
| series\_id | string | id vỏ phim |
| series\_name | string | tên vỏ phim |
| release\_year | int | năm phát hành |
| country | string | quốc gia phát hành |
| raw\_category\_id | string | id thể loại (dữ liệu gốc) |
| raw\_category\_name | string | tên thể loại (dữ liệu gốc) |
| director\_id | string | id đạo diễn |
| director\_name | string | tên đạo diễn |
| actor\_id | string | id diễn viên |
| actor\_name | string | tên diễn viên |

Tiếp theo đó, chúng tôi nhận thấy rằng đối với đặc trưng raw\_category\_id là một danh sách các mã thể loại phim. Đối với một phim có nhiều thể loại phim thì chúng tôi sẽ tiến hành xử lý danh sách này để biểu diễn dưới dạng đồ thị để biểu diễn các mối quan hệ giữa các đồ thị với nhau.

Các bước xử lý như sau:

Bước 1: Trích xuất đặc trưng từ trường dữ liệu tập phim với trường ‘raw\_category\_id’. Ví dụ, một phim sẽ có các giá trị: "Hành động," "Kinh dị," "Tình cảm,"... ta trích xuất và lấy dữ liệu của từng phim.

Bước 2: Trích xuất và tổng hợp thành dãy giá trị. Ví dụ phim có giá trị dữ liệu là “Hành động, kinh dị, tình cảm, v.v..” sẽ được trích xuất thành [‘Hành động’, ‘kinh dị’, ‘tình cảm’].

Bước 3: Tổng hợp các giá trị của trường dữ liệu thành một dãy các giá trị độc nhất để thực hiện.

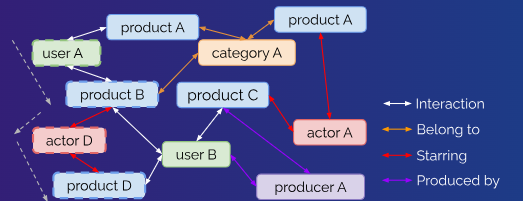
Chuẩn hoá các dữ liệu thành mã dãy thông tin được đánh dấu. Các bước thực hiện như sau:

Bước 1: Thực hiện chuyển giữa mã id người dùng thành một danh sách mã hoá mới để phù hợp với mô hình. Việc ghi lại bộ từ điển mã chuyển đổi được thực hiện bằng cách trích xuất toàn mã id của người dùng. Sau đó, gán với mã id người dùng với một mã id mới. Cuối cùng, lưu bộ từ điển chuyển đổi vào một file để thống nhất sử dụng.

Bước 2: Thực hiện chuyển giữa mã id sản phẩm thành một danh sách mã hoá mới để phù hợp với mô hình. Việc ghi lại bộ từ điển mã chuyển đổi được thực hiện bằng cách trích xuất toàn mã id của sản phẩm. Sau đó, gán với mã id sản phẩm với một mã id mới. Cuối cùng, lưu bộ từ điển chuyển đổi vào một file để thống nhất sử dụng

### Tạo đồ thị tri thức

Việc xây dựng đồ thị tri thức bao gồm các mối quan hệ giữa người dùng và phim, giữa phim và phim được biểu diễn minh hoạ ở hình 1.1.



Mô tả đồ thị tri thức

Theo hình 1.1, các bộ phim có quan hệ với nhau thông qua các nút actor hoặc các nút category với mối quan hệ được đóng chính bởi hoặc mối quan hệ thuộc về tương ứng.

Mục đích của việc xây dựng đồ thị tri thức cũng sẽ biểu diễn các mối quan hệ giữa các bộ phim để tìm ra đường dẫn tốt nhất để gợi ý đối với người dùng. Các bước để xây dựng mô hình đồ thị tri thức như sau:

Bước 1: Khởi tạo bộ hằng số để định nghĩa các mối quan hệ giữa các đối tượng bao gồm:

relation\_id2plain\_name = {

    '0': 'category',

    '1': 'starred\_by',

    '2': 'belong\_to',

    '3': 'directed\_by',

    '4': 'related\_to'

}

relation\_name2entity\_name = {

    'director\_id': 'directed\_by',

    'country\_id': 'belong\_to',

    'category\_id': 'category',

    'actor\_id': 'starred\_by',

    'release\_year': 'related\_to'

}

relation\_name2relation\_id = {

    'director\_id': '3',

    'country': '2',

    'raw\_category\_id': '0',

    'actor\_id': '1',

    'release\_year': '4'

}

Bước 2: Dựa trên dữ liệu về thuộc tính phim, thực hiện tạo bộ ba mối quan hệ giữa thực thể sản phẩm, thực thể thuộc tính, mối quan hệ giữa thực thể sản phẩm và thực thể thuộc tính. Ví dụ như sau:

|  |  |  |  |
| --- | --- | --- | --- |
| Series\_id | Actor\_id | Raw\_category\_id | Director\_id |
| 888 | Diễn viên A, diễn viên B | Category\_1, category\_2 | Director\_1 |

Chuyển đổi thành bộ ba mối quan hệ, ta được dãy các bộ ba như sau:

[[888, diễn viên A, starring], [888, diễn viên A, starring], [888, category\_1, belong\_to], [ 888, category\_2, belong\_to], [888, director\_1, director\_by]].

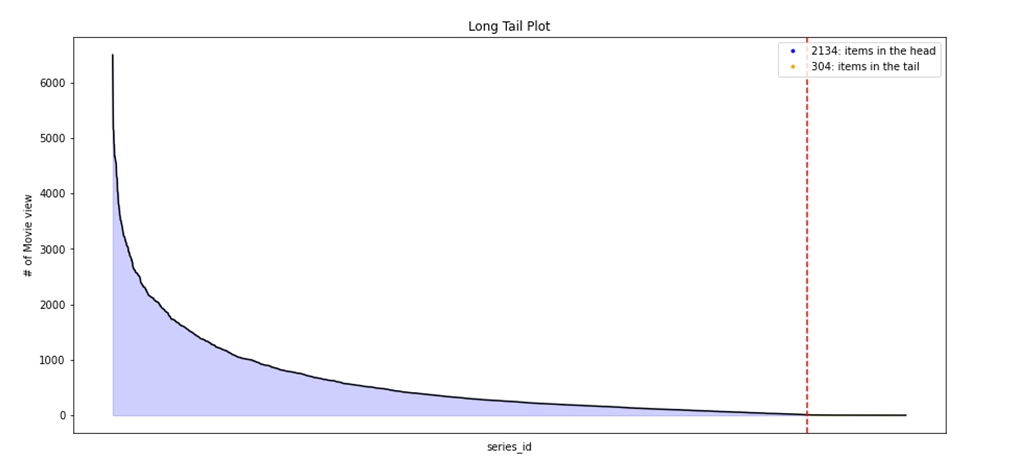
Bước 3: Tạo mã định danh cho thực thể thuộc tính. Đối với thuộc tính cho đồ thị tri thức, có hàng chục thuộc tính khác nhau. Trong mỗi thuộc tính chúng ta lại có hàng chục, hàng trăm giá trị khác nhau. Vậy nên cần tạo mã định danh cho thực thể thuộc tính. Việc tạo mã định danh cho tổ hợp này để sử dụng cho việc tạo gợi ý kèm giải thích.

Bước 4: Lưu đồ thị tri thức được tạo bởi bộ ba dưới dạng mã hoá của bộ mã hoá thực thể sản phẩm và bộ mã hoá thuộc tính.

## Trực quan hoá dữ liệu

Trực quan hóa dữ liệu là quá trình sử dụng các phương tiện hình ảnh, biểu đồ, đồ thị và các phương tiện khác để hiển thị thông tin dữ liệu một cách trực quan và dễ hiểu. Vậy nên, chúng tôi trực quan hoá dữ liệu để xây dựng các kịch bản huấn luyện và chuẩn bị dữ liệu.

Dựa trên biểu đồ 1.2. Biểu đồ thể hiện phân phối chiếc đuôi dài của dữ liệu TV360. Dữ liệu có chênh lệch khá lớn giữa những bộ phim ở top đầu và những bộ phim ở top cuối. Chúng tôi đề xuất kịch bản loại bỏ xếp hạng top 50 bộ phim phổ biến nhất và loại bỏ những bộ phim có mức tương tác nhỏ hơn 10.



Phân phối chiếc đuôi dài với dữ liệu TV360-1M

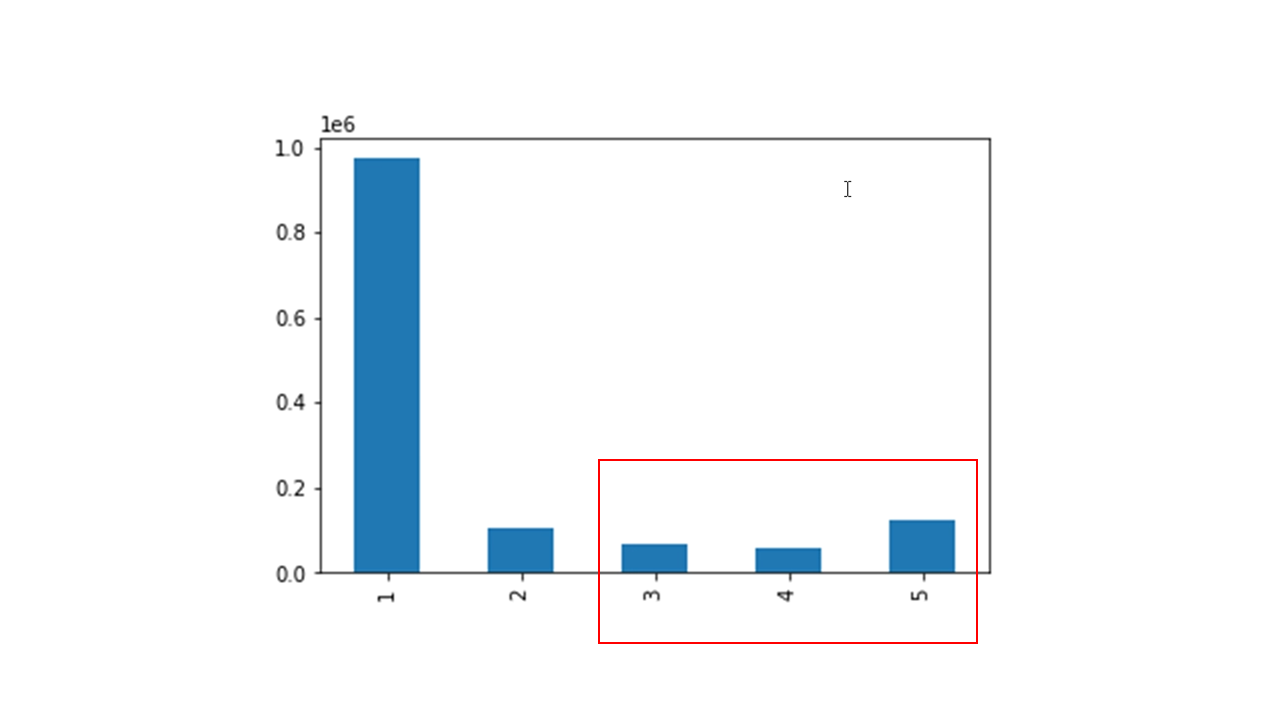
Mặt khác, một cách để thoả thuận với dữ liệu có phân phối chiếc đuôi dài là việc loại bỏ những item quá phổ biến hoặc những item quá ít tương tác. Những item nằm ở giữa được coi là những item phù hợp với cá nhân hoá nhiều hơn. Việc chia cắt hay lấy những item được thể hiện ở hình 1.3.

A graph of a curve

Description automatically generated

Phân phối chiếc đuôi dài với dữ liệu TV360-1M và chia bộ dữ liệu

Tuy nhiên, đối với dữ liệu dựa trên đồ thị tri thức, chúng tôi sẽ áp dụng kịch bản loại bỏ những item phổ biến và những item có tương tác nhỏ hơn 10.



Phân phối mức yêu thích đối với phim

Đối với các mức yêu thích của phim đã được định nghĩa từ những báo cáo trước. Chúng tôi trực quan hoá với hành vi của phim dựa trên tương tác người dùng. Như hình 1.4, chúng tôi thấy rằng lượng phim có mức độ yêu thích là 1 (thời gian xem phim trên thời gian của bộ phim thấp) chiếm phần lớn, nhưng những bộ phim được yêu thích nhất mới là những bộ phim chúng tôi xây dựng kịch bản để đưa vào dữ liệu huấn luyện.

## Chia tập dữ liệu huấn luyện và thử nghiệm

Trong quá trình phát triển mô hình máy học, quá trình chia tập huấn luyện và thử nghiệm đóng vai trò quan trọng để đảm bảo độ chính xác và hiệu suất của mô hình. Tập huấn luyện là giai đoạn mà mô hình học từ dữ liệu được cung cấp, điều chỉnh trọng số và xây dựng khả năng dự đoán. Để đảm bảo tính tổng quát, tập dữ liệu lớn thường được chia thành tập huấn luyện và tập kiểm tra. Tập huấn luyện được sử dụng để "dạy" mô hình, trong khi tập kiểm tra được sử dụng để đánh giá khả năng dự đoán trên dữ liệu mà mô hình chưa từng thấy. Với tỷ lệ bộ dữ liệu được chia là 80% cho bộ dữ liệu huấn luyện và 20% cho bộ dữ liệu kiểm thử.

Dữ liệu được thử nghiệm sẽ là dữ liệu tháng 10. Từ thống kê, chúng tôi lựa chọn bộ dữ liệu người dùng có hành vi tương tác từ 5 – 10 tương tác trên một tháng để cho mô hình học. Bộ dữ liệu được mô tả như sau:

Dữ liệu của mỗi người sẽ được sắp xếp theo thời gian và được chia theo khoảng 80% cho tập huấn luyện và 20% cho tập kiểm thử. Các bước chia tập train-test được mô tả qua các bước sau:

Bước 1: Gom nhóm người dùng theo mã người dùng và sắp xếp theo thời gian. Ví dụ:

|  |  |  |  |
| --- | --- | --- | --- |
| Uid | pid | Watch\_duration | Partition |
| 1 | 3 | 1 | 20231001 |
| 2 | 2 | 2 | 20231005 |
| 2 | 1 | 3 | 20231004 |
| 1 | 5 | 4 | 20231002 |

Đây là bảng mô tả bộ dữ liệu gốc trước khi xử lý. Sau khi xử lý dữ liệu sẽ có dạng như sau:

{

1: [[3, 1, 20231001], [5, 4, 20231002]],

2: [[1, 3, 20231004], [2, 2, 20231005]],

}

Như vậy, bộ dữ liệu đã được gom nhóm theo khoá là mã id của người dùng và giá trị của khoá là mảng lưu trữ các hành vi người dùng được sắp xếp theo thời gian từ trái qua phải.

Bước 2: Chia bộ dữ liệu theo tỷ lệ đã xác định. Chúng tôi ở đây sẽ chia bộ dữ liệu theo tỷ lệ 80% cho bộ huấn luyện và 20% cho bộ dữ liệu kiểm thử. Thuật toán sẽ chia dựa trên giá trị là mảng lưu trữ các hành vi của người dùng được sắp xếp theo thời gian được mô tả như trên.

# TRIỂN KHAI MÔ HÌNH PGPR CHO BỘ DỮ LIỆU BACKTEST

## Tiền xử lý dữ liệu

Tiền xử lý trước khi đưa vào mô hình đồ thị được biểu diễn dưới dạng nạp dữ liệu theo khung đối tượng được biểu diễn dưới dạng đồ thị. Thực hiện việc nạp dữ liệu vào đối tượng đồ thị tri thức và xử lý dữ liệu được thông qua các bước sau:

Bước 1: Nạp dữ liệu với các hằng số được định nghĩa sẵn (các hằng số được định nghĩa ở file pgpr\_utils.py)

Bước 2: Tạo đồ thị từ bộ dữ liệu được xử lý dữ liệu (mô tả tại phần 1.4, 1.5)

Bước 3: Tạo nhãn train/test cho bộ dữ liệu train/text để sử dụng sau này.

Để chạy PGPR, chúng ta cần có các tệp ở định dạng mà mô hình có thể đọc được. Đặc biệt, PGPR cần phải nhóm tất cả các thực thể (ví dụ: tác nhân, danh mục) và các mối quan hệ (ví dụ: đóng vai chính, thuộc về) theo loại của chúng.

Để làm điều đó, hãy sử dụng hàm map\_to\_PGPR trong mô-đun ánh xạ để thực hiện việc chuyển đổi dữ liệu này. Hàm map\_to\_PGPR chỉ lấy tập dữ liệu\_name làm đầu vào. Trong nội bộ, nó sẽ trích xuất các thực thể từ định dạng chuẩn KG của chúng tôi và nhóm tất cả các thực thể và quan hệ theo loại của chúng. Ví dụ, trong bộ ba (phim A, thuộc Danh mục A), mối quan hệ thuộc về cho chúng ta biết rằng đuôi thực thể sẽ là một danh mục. Khai thác các quy tắc ngữ nghĩa này, nó sẽ tạo ra các tệp number\_of\_entity\_types, trong đó mọi người trong số chúng đều lưu trữ các id cục bộ của thực thể gia tăng cho loại thực thể đó. Một cái gì đó tương tự được thực hiện cho các mối quan hệ.

Do đó, chúng tôi có thể mong đợi hai loại tệp chính:

* Thực thể: là các tệp chứa id thực thể cho thực thể tương ứng. Chúng được ánh xạ lại từ 0 đến số thực thể thuộc loại tương ứng.
* Quan hệ: là các tập tin chứa các mối quan hệ. Mỗi tệp có số hàng n\_products và mỗi hàng xác định mối quan hệ giữa sản phẩm đó và danh sách các thực thể.

## Tiền xử lý dữ liệu tỉa cây đồ thị tri thức

Việc xử lý dữ liệu để làm sạch dữ liệu đầu vào giúp tăng kết quả mô hình và giúp mô hình học được tri thức đúng hơn. Việc tỉa cây tri thức ở đây bao gồm việc gom nhóm những thể loại được phân biệt riêng lẻ thành một nhóm. Giúp cây tri thức trở nên gọn gàng. Việc cây tri thức có ít nút hơn sẽ giúp cải thiện được hiệu năng của mô hình.

Các bước thực hiện việc gom nhóm thể loại để tỉa cây tri thức như sau:

* Bước 1: Tạo một bộ từ điển với khoá chính là thể loại của phim, và giá trị của khoá là thể loại được gom nhóm
* Bước 2: Gom nhóm và phân tách các thể loại
* Bước 3: Thực hiện xử lý gom nhóm dữ liệu trên bộ dữ liệu phim
* Bước 4: Ứng dụng bộ từ điển lên cây tri thức

## Nhúng dữ liệu đầu vào

TransE là một phương pháp nhúng biểu đồ tri thức tịnh tiến (KGE). Các phương pháp KGE nhằm mục đích tìm hiểu cách biểu diễn chiều thấp của các thực thể và quan hệ của KG trong khi vẫn giữ được ý nghĩa ngữ nghĩa của chúng. Vì lý do này, nó đã được sử dụng rộng rãi để thể hiện mối quan hệ giữa các thực thể trong KG. Việc nhúng transE là một điều quan trọng trong PGPR. Trên thực tế, nó đã sử dụng tác nhân trong quá trình đào tạo cho hai chức năng tính điểm khác nhau (được trình bày kỹ hơn ở phần 2.3):

* Chức năng chấm điểm Multi-Hop: Hướng dẫn khám phá tác nhân (chọn đường dẫn), ưu tiên các đường dẫn có giá trị cao nhất. Nó được tính bằng tích số chấm giữa nút hiện tại và nút ứng cử viên.
* Chức năng khen thưởng: Gán phần thưởng cho đường đi. Nó được tính bằng cách thực hiện tích số chấm giữa nút đầu tiên của đường dẫn và nút cuối.

Chúng ta có thể huấn luyện cách biểu diễn TransE của KG bằng cách thực thi train\_transe\_model.py.

Danh sách siêu tham số transE được báo cáo như sau:

* --epochs: số lượng kỷ nguyên cần huấn luyện.
* --batch\_size: kích thước lô.
* --lr: tốc độ học tập
* --weight\_decay: giảm cân của adam.
* --l2\_lambda: chuẩn hóa l2
* --max\_grad\_norm: cắt bớt độ dốc
* --embed\_size: kích thước nhúng kiến thức.
* --num\_neg\_samples: số lượng mẫu âm tính.

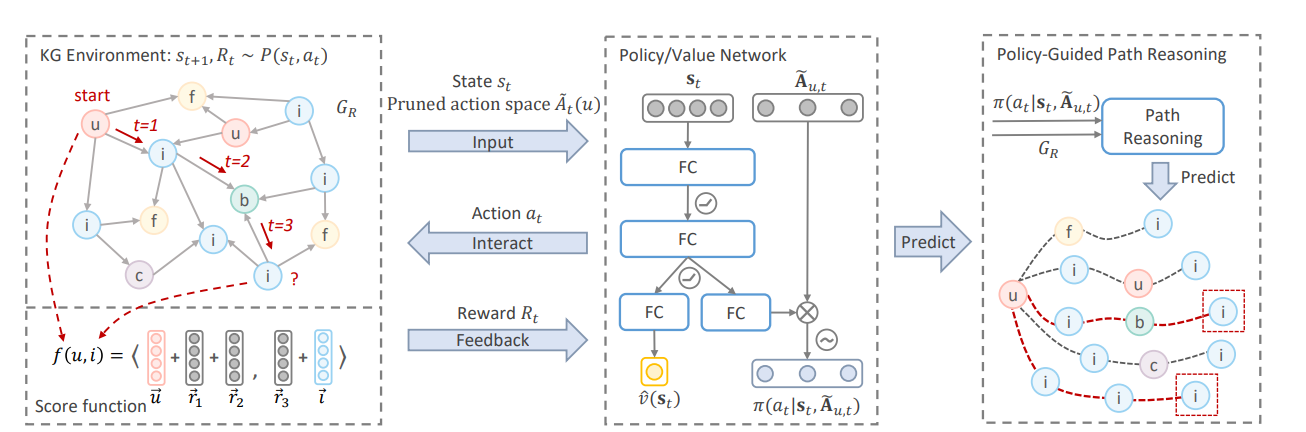
Để đơn giản, chúng tôi đã đặt các giá trị để có thể chạy tập lệnh chỉ ra tập dữ liệu\_name.

## Thực nghiệm mô hình PGPR

### Cơ sở lý thuyết

Phương pháp được đề xuất, Policy-Guided Path Reasoning (PGPR), là một phương pháp học tăng cường (RL) kết hợp đề xuất và khả năng diễn giải bằng cách tận dụng các biểu đồ tri thức để ra quyết định. Mô hình này có chiến lược phần thưởng mềm cải tiến, cắt tỉa hành động có điều kiện của người dùng và chức năng tính điểm nhiều bước. Nó sử dụng thuật toán tìm kiếm biểu đồ hướng dẫn có chính sách để lấy mẫu các đường dẫn lý luận cho đề xuất một cách hiệu quả và hiệu quả. Mô hình này kết hợp một thuật ngữ chính quy hóa nhằm tối đa hóa entropy của chính sách nhằm khuyến khích tác nhân khám phá các đường dẫn đa dạng hơn. Các biểu diễn tiềm ẩn của các thực thể và quan hệ được huấn luyện dựa trên chức năng tính điểm 1 bước nhảy và kích thước nhúng được đặt thành 100. Mô hình sử dụng mạng giá trị chính sách với ma trận trọng số cụ thể cho các thành phần khác nhau. Mô hình được đào tạo bằng cách sử dụng tối ưu hóa Adam và kích thước lô là 64 hoặc 32, tùy thuộc vào tập dữ liệu.

Nhìn chung, kiến trúc của mô hình kết hợp học tăng cường, lý luận đồ thị và khả năng diễn giải để tạo ra các đề xuất có thể giải thích được hỗ trợ bởi quy trình suy luận nhân quả. Nó kết hợp nhiều kỹ thuật khác nhau như chiến lược phần thưởng mềm, cắt tỉa hành động có điều kiện của người dùng và tìm kiếm biểu đồ theo hướng dẫn chính sách để cải thiện hiệu suất đề xuất.



Quy trình đưa ra đề xuất về phương pháp giải thích theo lộ trình được hướng dẫn bởi chính sách

Tổng quan, một đồ thị tri thức G với bộ thực thể E và bộ quan hệ R được định nghĩa như sau: . Trong đó, với mỗi bộ ba biểu diễn mối quan hệ trên thực tế của thực thể và thực thể . Xây dựng một đồ thị tri thức đặc biệt adnhf cho khuyến nghị có thể giải thích được là . Giả định rằng, chúng ta có một bộ thực thể người dùng ký hiệu là và một bộ thực thể sản phẩm ký hiệu là . Hai thực thể kết nối với nhau thông qua mối quan hệ . Có 2 vấn đề cần làm được định nghĩa để giải quyết bài toán như sau:

Định nghĩa 1: (đường dẫn k-hop): Các tác giả giới thiệu khái niệm "đường dẫn k-hop" trong bối cảnh biểu đồ tri thức. Đường dẫn k-hop là một chuỗi các thực thể được kết nối bằng các mối quan hệ trong biểu đồ tri thức. Đường dẫn bắt đầu từ thực thể e0 và kết thúc tại thực thể khác ek. Độ dài của đường dẫn được ký hiệu là k và nó bao gồm k + 1 thực thể và k quan hệ. Mỗi thực thể trong đường dẫn được kết nối với thực thể tiếp theo bằng một mối quan hệ, được biểu diễn dưới dạng ri. Ký hiệu pk (, ) được sử dụng để biểu thị đường dẫn k-hop từ thực thể đến thực thể . Trong biểu đồ tri thức, các thực thể biểu thị các đối tượng hoặc khái niệm trong thế giới thực, trong khi các mối quan hệ biểu thị các kết nối hoặc liên kết giữa các thực thể này. Ví dụ: trong biểu đồ tri thức về phim, các thực thể có thể là tựa phim, diễn viên, đạo diễn và các mối quan hệ có thể được "đóng vai", "đạo diễn bởi", v.v. Biểu đồ tri thức nắm bắt thông tin có cấu trúc về các thực thể và mối quan hệ này, cho phép cho việc lập luận và suy luận. Trong định nghĩa, các tác giả sử dụng một số ký hiệu và ký hiệu để biểu diễn các phần tử khác nhau. Thực thể ở vị trí i trong đường dẫn được ký hiệu là , trong đó i nằm trong khoảng từ 0 đến k. Mối quan hệ kết nối thực thể ei-1 và ei được biểu diễn dưới dạng ri. Ký hiệu (, , ) được dùng để chỉ ra rằng có một mối quan hệ ri kết nối các thực thể và trong biểu đồ tri thức. Ký hiệu (, , ) được sử dụng để biểu thị mối quan hệ hai chiều giữa các thực thể và . Ví dụ: Khái niệm về đường dẫn k-hop cho phép chúng ta duyệt qua biểu đồ tri thức và tìm kết nối giữa các thực thể. Ví dụ: nếu chúng ta có biểu đồ tri thức về sách, chúng ta có thể tìm đường đi 2 bước từ cuốn sách cụ thể này sang cuốn sách khác bằng cách tuân theo hai mối quan hệ. Giả sử chúng ta muốn tìm đường dẫn 2 bước từ cuốn A đến cuốn C và các mối quan hệ trong biểu đồ tri thức là "được viết bởi" và "tương tự". Chúng ta có thể tìm thấy một đường dẫn như thế này: A -(được viết bởi)-> Author1 -(similar to)-> B -(writing by)-> Author2 -(similar to)-> C. Trong ví dụ này, A được kết nối với B thông qua quan hệ "được viết bởi" và B được kết nối với C thông qua quan hệ "tương tự", tạo thành đường dẫn 2 bước từ A đến C.

Định nghĩa 2: Định nghĩa 3.2. (Vấn đề KGRE-Rec): Vấn đề Lý do Sơ đồ tri thức cho Khuyến nghị có thể giải thích (KGRE-Rec). Bài toán KGRE-Rec liên quan đến biểu đồ tri thức (GR), người dùng (u) thuộc một tập hợp người dùng (U) và hai số nguyên (K và N). Mục tiêu của bài toán KGRE-Rec là tìm một tập hợp các khuyến nghị cho người dùng (u) từ một tập hợp các mục (I). Bộ đề xuất phải bao gồm N mục ({in }n ∈[N ] ⊆ I) và mỗi mục trong bộ phải được liên kết với một đường dẫn lý luận (pk) giữa người dùng (u) và mục (in). Đường dẫn lý luận (pk) biểu thị các kết nối hoặc mối quan hệ logic giữa người dùng (u) và mục được đề xuất (in) trong biểu đồ tri thức (GR). Các đường suy luận được đánh số từ 2 đến K, trong đó K là số nguyên lớn hơn hoặc bằng 2. Số lượng đề xuất (N) xác định kích thước của bộ đề xuất. Tóm lại, bài toán KGRE-Rec nhằm mục đích tìm ra một tập hợp các đề xuất cho người dùng bằng cách xem xét biểu đồ tri thức và các kết nối logic giữa người dùng và các mục được đề xuất. Các khuyến nghị cần được hỗ trợ bởi các đường dẫn lý luận trong biểu đồ tri thức.

A diagram of a computer program

Description automatically generated

Mô tả trạng thái của mô hình PGPR

Để xác định được đường dẫn tối ưu cho việc gợi ý người dùng, cần định nghĩa các mối quan hệ của đồ thị với 2 dạng là

* Biểu diễn mối quan hệ nghịch đảo dưới dạng
* Biểu diễn cho đường dẫn của mô hình mong muốn dưới dạng

Đầu tiên, xét trạng thái đầu tiên của mô hình là trạng thái State. Tại **trạng thái**  tại bước t được định nghĩa bằng một bộ ba , trong đó, u đại diện cho thực thể người dùng, là thực thể đại diện cho thuật toán thực hiện tại bước t, là trạng thái cũ dẫn đến bước t. Dựa trên định nghĩa đường dẫn k-hop, việc định nghĩa k-bước lịch sử là sự kết hợp của tất cả các thực thể và mối quan hệ thực thể xảy ra trong k bước. Ví dụ .

Không gian hành động , không gian hành động hoàn chỉnh của trạng thái được định nghĩa là tất cả các cạnh đầu ra có thể có của thực thể , ngoại trừ các thực thể đã được truy cập đến. Ký hiệu: . Điều này dẫn đến hiện tượng, long-tail distribution (phân phối đuôi dài), nghĩa là một số nút sẽ có nhiều đầu ra so với phần còn lại. Vậy nên việc duy trì kích thước của không gian hành động dựa trên mức độ lớn nhất là khá kém hiệu quả về mặt không gian. Để giải quyết điều này, hàm tính điểm ánh xạ bất kỳ cạnh (r, e) (∀r ∈ R, ∀e ∈ E) nào tới điểm có giá trị thực tùy thuộc vào người dùng u. Chi tiết nằm trong bài báo, tôi sẽ tập trung vào những thuật toán nổi bật để giải bài toán tổng quát.

Với bất kỳ người dùng nào, không có nhãn đầu ra nào được biết trước trong bài toán KGRE-Rec, do đó, việc xem xét thuật toán thưởng nhị phân cho biết liệu tác nhân có đạt được mục tiêu hay không là không khả thi. Thay vào đó, tác nhân được khuyến khích khám phá càng nhiều con đường “tốt” càng tốt. Một cách trực quan, trong bối cảnh khuyến nghị, đường dẫn “tốt” là đường dẫn đến một mục mà người dùng sẽ tương tác với xác suất cao. Vì mục đích này, mô hình chỉ trao phần thưởng đối với trạng thái cuối dựa trên cách tính điểm khác hàm .

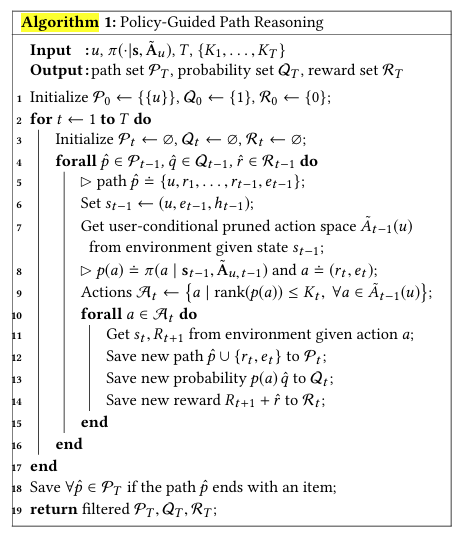
Với nhiều đầu ra, việc chuyển tiếp trạng thái sang được định nghĩa như sau:

Cho trạng thái và một hành động

Một ngoại lệ là trạng thái ban đầu là ngẫu nhiên, được xác định bởi thực thể người dùng ban đầu. Để đơn giản, chúng tôi giả sử việc phân phối trước của người dùng tuân theo phân phối thống nhất sao cho mỗi người dùng đều được lấy mẫu như nhau ngay từ đầu.

A multi-hop scoring là một phương pháp được sử dụng trong bối cảnh lý luận về biểu đồ tri thức để đưa ra đề xuất. Nó được thiết kế để đo lường tính chính xác hoặc mức độ liên quan của một mục đối với người dùng dựa trên thông tin không đồng nhất phong phú có trong biểu đồ tri thức. Nó tận dụng nhiều bước nhảy hoặc kết nối trong biểu đồ để tính điểm cho một mục, có tính đến các mối quan hệ và kết nối giữa các thực thể khác nhau trong biểu đồ. Hàm tính điểm nhiều bước được sử dụng như một phần của hàm mục tiêu trong việc huấn luyện biểu diễn biểu đồ tri thức và có thể cải thiện hiệu suất đề xuất của phương pháp.

Trong biểu đồ tri thức, có một thuộc tính là khi biết loại thực thể đầu và mối quan hệ hợp lệ, chúng ta có thể xác định loại thực thể đuôi. Điều này có nghĩa là mối quan hệ giữa thực thể đứng đầu và thực thể đuôi được xác định bởi loại của thực thể đứng đầu và mối quan hệ giữa chúng. Chúng ta có thể mở rộng thuộc tính này bằng cách tạo quy tắc chuỗi về các loại thực thể và kiểu quan hệ. Quy tắc chuỗi này bao gồm một chuỗi các thực thể và quan hệ: {e0, r1, e1, r2, ..., rk, ek}. Nếu chúng ta biết kiểu của thực thể e0 và tất cả các quan hệ r1, r2, ..., rk, thì chúng ta có thể xác định duy nhất kiểu của tất cả các thực thể khác e1, e2, ..., ek trong chuỗi. Điều này có nghĩa là các loại thực thể trong chuỗi phụ thuộc vào nhau và có thể được suy ra dựa trên các loại thực thể chính đã biết và các mối quan hệ. Dựa trên quy tắc dây chuyền của các kiểu thực thể và kiểu quan hệ, chúng tôi giới thiệu khái niệm về mẫu. Mẫu là một chuỗi cụ thể của các thực thể và quan hệ tuân theo quy tắc dây chuyền. Bằng cách biết các loại thực thể chính, các mối quan hệ và các loại thực thể trong mẫu, chúng ta có thể xác định loại của tất cả các thực thể khác trong mẫu. Các mẫu cung cấp một cách để biểu diễn và suy luận về mối quan hệ giữa các thực thể trong biểu đồ tri thức dựa trên loại của chúng và mối quan hệ giữa các thực thể.



Thuật toán 1: Thuật toán Policy-Guided Path Reasoning

Lý luận về đường dẫn có hướng dẫn của chính sách (PGPR) là một phương pháp đưa ra đề xuất có thể giải thích được trên biểu đồ tri thức. Nó kết hợp học tập tăng cường với biểu đồ tri thức để tạo ra các đề xuất và đưa ra lời giải thích. PGPR đào tạo tác nhân RL cách điều hướng trong môi trường sơ đồ tri thức và tìm các mục "tốt" cho mỗi người dùng. Đường dẫn điều hướng của tác nhân đóng vai trò giải thích cho các mục được đề xuất. Mô tả bước cuối cùng để giải quyết vấn đề đề xuất trên biểu đồ tri thức bằng cách sử dụng mạng chính sách đã được huấn luyện. Một cách tiếp cận là lấy mẫu đường dẫn cho từng người dùng dựa trên mạng chính sách, nhưng điều này có thể dẫn đến thiếu tính đa dạng của đường dẫn do tác nhân có xu hướng tìm kiếm nhiều lần trên cùng một đường dẫn với phần thưởng tích lũy cao nhất. Để giải quyết vấn đề này, phương pháp được đề xuất sử dụng tìm kiếm chùm tia được hướng dẫn bởi xác suất hành động và phần thưởng để khám phá các đường dẫn ứng viên và các mục được đề xuất cho mỗi người dùng. Thuật toán 1 phác thảo quy trình, lấy người dùng, mạng chính sách, đường chân trời và kích thước lấy mẫu được xác định trước làm đầu vào và cung cấp một tập hợp các đường dẫn T-hop cho người dùng với xác suất và phần thưởng tạo đường dẫn tương ứng. Mỗi đường dẫn trong đầu ra kết thúc bằng một thực thể vật phẩm được liên kết với xác suất tạo đường dẫn và phần thưởng đường dẫn.

## Kết quả mô hình baseline

Danh sách siêu tham số train\_agent được báo cáo như sau:

* --epochs: Số lượng epoch tối đa.
* --batch\_size: Kích thước lô.
* --lr: Tốc độ học tập.
* --max\_acts: Số lượng hành động tối đa.
* --max\_path\_len: Độ dài đường dẫn tối đa.
* --gamma: hệ số chiết khấu phần thưởng.
* --ent\_weight: hệ số trọng lượng để giảm entropy.
* --act\_dropout: tỷ lệ bỏ qua hành động.
* --state\_history: độ dài lịch sử trạng thái.
* --hidden: Số lượng mẫu.

Kịch bản thực nghiệm:

* Kịch bản 1: Thực nghiệm với toàn bộ dữ liệu
* Kịch bản 2: Thực nghiệm với bộ dữ liệu loại bỏ item phổ biến
* Kịch bản 3: Thực nghiệm với bộ dữ liệu loại bỏ item phổ biến kết hợp tinh chỉnh parameter
* Kịch bản 4: Thực nghiệm với bộ dữ liệu lấy 1000 người dùng thường xuyên tương tác
* Kịch bản 5: Thực nghiệm với bộ dữ liệu lấy rating >= 3

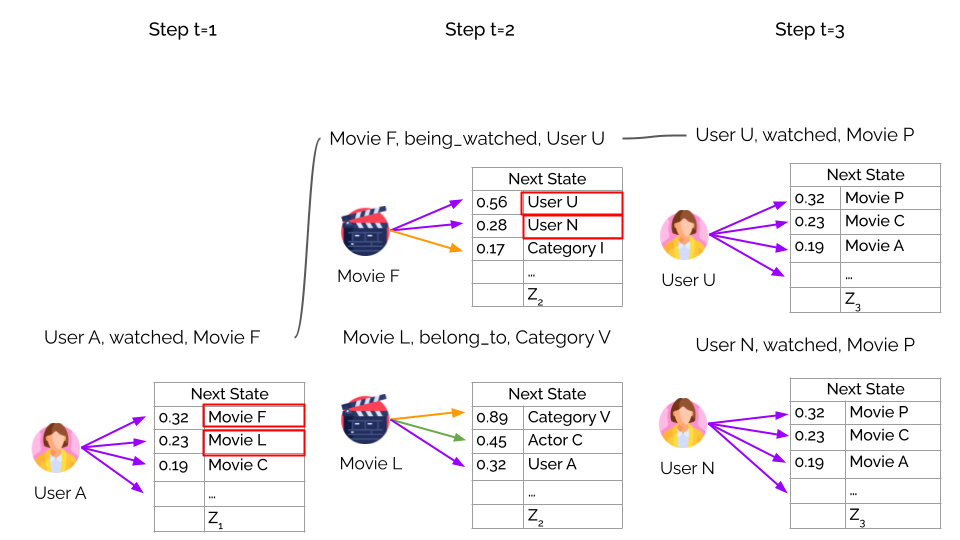
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Recall@10** | **Recall@30** | **Recall@50** | **NDCG@50** |
| Round 1 | 0.0280 | 0.0322 | 0.0383 | 0.327 |
| Round 2 | 0.0251 | 0.0289 | 0.0321 | 0.351 |

### Tạo gợi ý kèm giải thích và kiểm tra lỗi

Mô hình tạo gợi ý dựa trên đường dẫn giữa người dùng và người dùng trên tương tác giữa các phim với nhau. Bênh cạnh đó mô hình tạo gợi ý dựa trên các thể loại người dùng hay xem.

Tạo gợi ý của mô hình PGPR

Trích xuất các đường dẫn từ chính sách đã học trước đó π. Điều này được thực hiện bằng cách thực hiện tìm kiếm chùm bằng cách sử dụng xác suất π(at|st,Au,t) làm điểm cho mỗi bước.



Minh hoạ cho ví dụ trích xuất đường dẫn có giải thích của mô hình đồ thị

Kiểm tra lỗi:

* Mô hình tạo gợi ý kèm giải thích mang tính mới mẻ và đồ thị tri thức chưa được xây dựng hoàn chỉnh.
* Chưa đa dạng được thực thể liên kết nên các đường dẫn gợi ý vẫn chưa được đa dạng và phần lớn là gợi ý với đường dẫn người dùng.

### Đánh giá mô hình

* Kết quả mô hình đạt khoảng từ 2%-5% tại recall at 50. Kết quả kém do đầu vào mô hình tri thức còn sơ sài – chúng tôi dựa trên dữ liệu hiện có của bộ dữ liệu bao gồm đạo diễn và diễn viên của bộ phim. Bộ phim đã có thêm các nút từ những thực thể của bộ phim hoặc diễn viên, đạo diễn. Tuy nhiên, mô hình lại luôn gợi ý kết quả với dữ liệu tương đồng với người dùng.
* Kết quả từ các kịch bản thực nghiệm mang kết quả tăng dần, đối với bộ dữ liệu loại bỏ những item phổ biến giúp mô hình tăng 1-2 % tại độ hồi tưởng xếp hạng 50.
* Kết quả mô hình đạt kết quả tốt nhất tại kịch bản lấy những tương tác với rating > 3.

## Kết quả mô hình tiến hành tối ưu

Danh sách siêu tham số train\_agent được báo cáo như sau:

* --epochs: Số lượng epoch tối đa.
* --batch\_size: Kích thước lô.
* --lr: Tốc độ học tập.
* --max\_acts: Số lượng hành động tối đa.
* --max\_path\_len: Độ dài đường dẫn tối đa.
* --gamma: hệ số chiết khấu phần thưởng.
* --ent\_weight: hệ số trọng lượng để giảm entropy.
* --act\_dropout: tỷ lệ bỏ qua hành động.
* --state\_history: độ dài lịch sử trạng thái.
* --hidden: Số lượng mẫu.

Kịch bản thực nghiệm:

* Kịch bản 1: Thực nghiệm với toàn bộ dữ liệu
* Kịch bản 2: Thực nghiệm với bộ dữ liệu loại bỏ item phổ biến

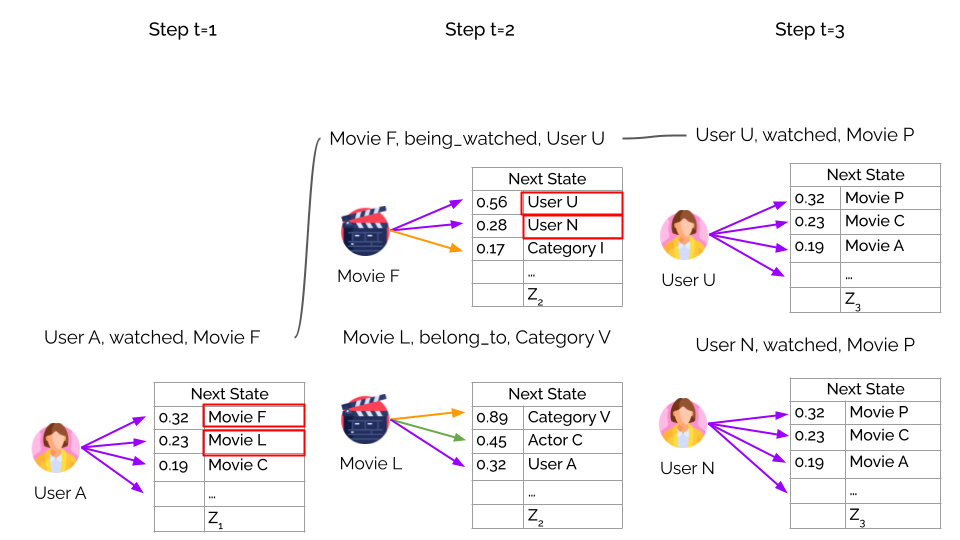
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Recall@10** | **Recall@30** | **Recall@50** | **NDCG@50** |
| Round 1 | 0.0280 | 0.0322 | 0.0383 | 0.327 |
| Round 2 | 0.0251 | 0.0289 | 0.0321 | 0.351 |

### Tạo gợi ý kèm giải thích và kiểm tra lỗi

Mô hình tạo gợi ý dựa trên đường dẫn giữa người dùng và người dùng trên tương tác giữa các phim với nhau. Bênh cạnh đó mô hình tạo gợi ý dựa trên các thể loại người dùng hay xem.

Tạo gợi ý của mô hình PGPR

Trích xuất các đường dẫn từ chính sách đã học trước đó π. Điều này được thực hiện bằng cách thực hiện tìm kiếm chùm bằng cách sử dụng xác suất π(at|st,Au,t) làm điểm cho mỗi bước.



Minh hoạ cho ví dụ trích xuất đường dẫn có giải thích của mô hình đồ thị

Kiểm tra lỗi:

* Mô hình tạo gợi ý kèm giải thích mang tính mới mẻ và đồ thị tri thức chưa được xây dựng hoàn chỉnh.
* Chưa đa dạng được thực thể liên kết nên các đường dẫn gợi ý vẫn chưa được đa dạng và phần lớn là gợi ý với đường dẫn người dùng.

### Đánh giá mô hình

* Kết quả mô hình đạt khoảng từ 2%-5% tại recall at 50. Kết quả kém do đầu vào mô hình tri thức còn sơ sài – chúng tôi dựa trên dữ liệu hiện có của bộ dữ liệu bao gồm đạo diễn và diễn viên của bộ phim. Bộ phim đã có thêm các nút từ những thực thể của bộ phim hoặc diễn viên, đạo diễn. Tuy nhiên, mô hình lại luôn gợi ý kết quả với dữ liệu tương đồng với người dùng.
* Kết quả từ các kịch bản thực nghiệm mang kết quả tăng dần, đối với bộ dữ liệu loại bỏ những item phổ biến giúp mô hình tăng 1-2 % tại độ hồi tưởng xếp hạng 50.
* Kết quả mô hình đạt kết quả tốt nhất tại kịch bản lấy những tương tác với rating > 3.

# Triển khai mô hình CAFE

## Tiền xử lý dữ liệu

Tiền xử lý trước khi đưa vào mô hình đồ thị được biểu diễn dưới dạng nạp dữ liệu theo khung đối tượng được biểu diễn dưới dạng đồ thị. Thực hiện việc nạp dữ liệu vào đối tượng đồ thị tri thức và xử lý dữ liệu được thông qua các bước sau:

Bước 1: Nạp dữ liệu với các hằng số được định nghĩa sẵn (các hằng số được định nghĩa ở file cafe\_utils.py)

Bước 2: Tạo đồ thị từ bộ dữ liệu được xử lý dữ liệu (mô tả tại phần 1.4, 1.5)

Bước 3: Tạo nhãn train/test cho bộ dữ liệu train/text để sử dụng sau này.

Để chạy CAFE, chúng tôi cần các tập dữ liệu và tệp KG ở định dạng mà mô hình có thể đọc được. Để làm điều đó, chúng tôi sẽ sử dụng hàm map\_to\_CAFE từ mô-đun trình ánh xạ của chúng tôi.

Hàm map\_to\_CAFE lấy đầu vào là tên tập dữ liệu, kích thước mong muốn của tập tàu train\_size và tập xác thực kích thước valid\_size được mặc định là 0 như trước đây. Hàm này thực hiện cả việc nhóm quan hệ/thực thể và phân chia kiểm thử tàu do sự phụ thuộc nội bộ của mô hình.

Trong nội bộ, nó sẽ trích xuất các thực thể từ định dạng stardard KG của chúng tôi và tạo các tệp sau:

* kg\_entities.txt.gz: Tập hợp tất cả các thực thể. Nó được biểu thị bằng bộ ba (entity\_global\_id, thực thể\_local\_id, tên\_thực thể). Tên thực thể sẽ được người dùng sử dụng sau này để tạo giải thích bằng văn bản.
* kg\_relations.txt.gz: Tập hợp các quan hệ. Nó ngầm bao gồm mối quan hệ ngược lại (ví dụ: đã xem: đang được\_xem, đóng vai chính: được gắn dấu sao).
* kg\_rules.txt.gz: Xác định các quy tắc metapath.
* kg\_triples.txt.gz: Tập hợp các bộ ba KG. Khác với PGPR, ví dụ như các bộ ba ngược (phim, do\_người đóng vai chính, diễn viên) sẽ có (diễn viên, diễn viên chính, phim). Nó cũng bao gồm bộ ba tương tác của người dùng như (người dùng, đã xem, phim).
* train.txt.gz: Train set, trong đó mỗi hàng được tạo bởi user\_id và danh sách các sản phẩm được người dùng đó tương tác.
* test.txt.gz: Bộ kiểm tra, trong đó mỗi hàng được tạo bởi user\_id và danh sách các sản phẩm được người dùng đó tương tác.

## Nhúng dữ liệu đầu vào

TransE là một phương pháp nhúng biểu đồ tri thức tịnh tiến (KGE). Các phương pháp KGE nhằm mục đích tìm hiểu cách biểu diễn chiều thấp của các thực thể và quan hệ của KG trong khi vẫn giữ được ý nghĩa ngữ nghĩa của chúng. Vì lý do này, nó đã được sử dụng rộng rãi để thể hiện mối quan hệ giữa các thực thể trong KG. Việc nhúng transE là một điều quan trọng trong PGPR. Trên thực tế, nó đã sử dụng tác nhân trong quá trình đào tạo cho hai chức năng tính điểm khác nhau (được trình bày kỹ hơn ở phần 2.3):

* Chức năng chấm điểm Multi-Hop: Hướng dẫn khám phá tác nhân (chọn đường dẫn), ưu tiên các đường dẫn có giá trị cao nhất. Nó được tính bằng tích số chấm giữa nút hiện tại và nút ứng cử viên.
* Chức năng khen thưởng: Gán phần thưởng cho đường đi. Nó được tính bằng cách thực hiện tích số chấm giữa nút đầu tiên của đường dẫn và nút cuối.

Chúng ta có thể huấn luyện cách biểu diễn TransE của KG bằng cách thực thi train\_transe\_model.py.

Danh sách siêu tham số transE được báo cáo như sau:

* --epochs: số lượng kỷ nguyên cần huấn luyện.
* --batch\_size: kích thước lô.
* --lr: tốc độ học tập
* --weight\_decay: giảm cân của adam.
* --l2\_lambda: chuẩn hóa l2
* --max\_grad\_norm: cắt bớt độ dốc
* --embed\_size: kích thước nhúng kiến thức.
* --num\_neg\_samples: số lượng mẫu âm tính.

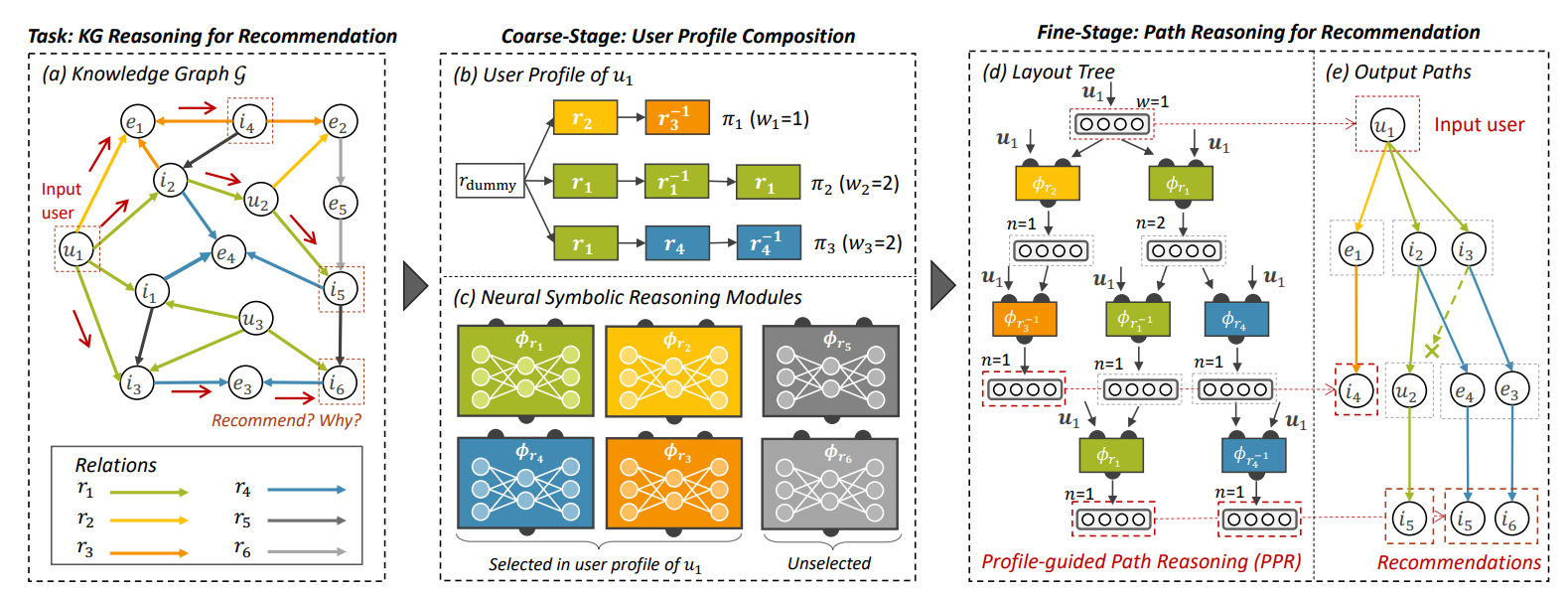
Để đơn giản, chúng tôi đã đặt các giá trị để có thể chạy tập lệnh chỉ ra tập dữ liệu\_name.

## Thực nghiệm mô hình CAFE

### Cơ sở lý thuyết

Giai đoạn thô trong mô hình CAFE:

Trong phương pháp CAFE, giai đoạn thô đề cập đến bước đầu tiên trong đó hồ sơ người dùng được tạo để nắm bắt các hành vi nổi bật của người dùng. Hồ sơ người dùng này đóng vai trò như một bản phác thảo thô về lịch sử hành vi của người dùng và cung cấp các tín hiệu có giá trị về mẫu đường dẫn nào có nhiều khả năng dẫn đến các mục tiềm năng mà người dùng quan tâm. Hồ sơ người dùng được tạo bằng cách tổng hợp các phương trình trên tất cả người dùng trong biểu đồ tri thức (KG). Nó nắm bắt các mô hình lấy người dùng làm trung tâm và giúp hướng dẫn quá trình tìm đường ở giai đoạn hoàn thiện. Mục tiêu của giai đoạn thô là rút ra các đường dẫn lý luận cho các đề xuất dựa trên hồ sơ người dùng. Những đường dẫn lý luận này, thu được thông qua đường dẫn lý luận dưới sự hướng dẫn của hồ sơ người dùng, có thể được coi là lời giải thích cho các đề xuất. Phương pháp CAFE tận dụng thuật toán tìm đường cải tiến được gọi là Lý do đường dẫn hướng dẫn hồ sơ (PPR) để tìm một cách hiệu quả một loạt đường dẫn trên KG quy mô lớn, sử dụng hồ sơ người dùng làm hướng dẫn. **Thành phần hồ sơ người dùng** ở giai đoạn thô là **một** **bước quan trọng** trong phương pháp CAFE, vì nó giúp đạt được chất lượng đề xuất thỏa đáng bằng cách nắm bắt hành vi của người dùng và cung cấp hướng dẫn cho quy trình lập luận đường dẫn.



Kiến trúc mô hình CAFE

Thành phần hồ sơ người dùng trong Cafe bao gồm việc chọn một tập hợp con các mẫu lấy người dùng làm trung tâm từ tập hợp các mẫu ứng cử viên, P và gán trọng số cho các mẫu này để phản ánh hành vi nổi bật của mỗi người dùng. Để tạo tập hợp ứng cử viên gồm các mẫu lấy người dùng làm trung tâm, thuật toán dựa trên bước đi ngẫu nhiên có sẵn được sử dụng để trích xuất các mẫu từ các cặp mục người dùng được tương tác trong biểu đồ tri thức (KG). Thay vì chỉ định trọng số dựa trên tần suất của các mẫu được truy xuất, Cafe nhằm mục đích tạo hồ sơ người dùng được cá nhân hóa để mô tả rõ ràng hành vi của người dùng. Trọng số được xác định dựa trên mức độ nổi bật của từng mẫu, được đo bằng khả năng các đường dẫn "chính xác" được tạo bởi mẫu đó. Thành phần hồ sơ người dùng trong CAFE rất quan trọng để nắm bắt hành vi của người dùng và cung cấp hướng dẫn cho quá trình suy luận đường dẫn ở giai đoạn tinh tế. Nó giúp xác định các mẫu đường dẫn có nhiều khả năng dẫn đến các mục tiềm năng mà mỗi người dùng quan tâm, từ đó cải thiện chất lượng đề xuất. Hàm F Profile được sử dụng trong phương pháp Cafe để chọn các mẫu nổi bật và xác định trọng số của chúng trong thành phần hồ sơ người dùng. Nó nhằm mục đích tối ưu hóa một hàm mục tiêu giúp cân bằng chất lượng của các đường dẫn lý luận do các mẫu tạo ra và hiệu suất xếp hạng của các đề xuất. Bằng cách gán trọng số cho các mẫu ứng cử viên, chức năng giúp tạo hồ sơ người dùng được cá nhân hóa nhằm nắm bắt các hành vi nổi bật của người dùng và hướng dẫn quy trình lý luận đường dẫn để tạo ra các đề xuất chi tiết. Bài toán tối ưu hóa được giải quyết bằng hàm đảm bảo chỉ những mẫu có trọng số dương mới được giữ trong hồ sơ người dùng, nâng cao hiệu lực và hiệu quả của quá trình đề xuất. Hàm là một phần của phương pháp heuristic được sử dụng trong CAFE để chấm điểm thành phần hồ sơ người dùng và cải thiện chất lượng đề xuất. Mục đích của hàm trong phương pháp CAFE là xác định trọng số của các mẫu ứng cử viên trong thành phần hồ sơ người dùng. Nó nhằm mục đích chọn các mẫu nổi bật đóng góp cho hồ sơ người dùng và loại bỏ các mẫu có trọng số âm. Hàm là một phần của phương pháp heuristic được sử dụng trong Cafe để chấm điểm thành phần hồ sơ người dùng. Nó giúp tối ưu hóa hàm mục tiêu giúp cân bằng giữa chất lượng của các đường dẫn lý luận do các mẫu tạo ra và hiệu suất xếp hạng của các đề xuất. Bằng cách gán trọng số cho các mẫu ứng cử viên, chức năng Hồ sơ F cho phép tạo hồ sơ người dùng được cá nhân hóa để nắm bắt các hành vi nổi bật của người dùng và hướng dẫn quy trình lý luận đường dẫn để tạo ra các đề xuất chi tiết. Bài toán tối ưu được giải quyết bằng hàm F Profile đảm bảo chỉ những mẫu có trọng số dương mới được giữ trong hồ sơ người dùng, nâng cao hiệu lực và hiệu quả của quá trình đề xuất.

Giai đoạn Lý do đưa ra khuyến nghị. Văn bản được đánh dấu đề cập đến một giai đoạn cụ thể trong phương pháp đề xuất được gọi là "Giai đoạn tinh tế" tập trung vào lý luận về lộ trình đưa ra đề xuất. Trong bối cảnh của nghiên cứu này, suy luận đường dẫn đề cập đến quá trình xác định trình tự các bước hoặc đường dẫn từ nút người dùng đến nút mục trong biểu đồ tri thức (KG). Mục đích của Giai đoạn tinh chỉnh này là tạo ra các dự đoán hoặc đề xuất chi tiết bằng cách tìm đường dẫn phù hợp nhất cho từng người dùng dựa trên hồ sơ người dùng của họ. Hồ sơ người dùng được tạo dưới dạng bản phác thảo thô về hành vi của người dùng, ghi lại các hành vi nổi bật của người dùng từ lịch sử của họ. Những hồ sơ người dùng này cung cấp các tín hiệu có giá trị về mẫu đường dẫn nào có nhiều khả năng dẫn đến các mục tiềm năng mà người dùng quan tâm. Fine-Stage sử dụng thuật toán tìm đường dẫn cải tiến được gọi là Lý do đường dẫn hướng dẫn hồ sơ (PPR) để tìm một loạt đường dẫn trên KG quy mô lớn một cách hiệu quả và hiệu quả. PPR tận dụng kho mô-đun lý luận biểu tượng thần kinh, là các thành phần hỗ trợ lý luận và ra quyết định, để hướng dẫn quá trình tìm đường. Mục tiêu của Giai đoạn tinh chỉnh là rút ra các đường dẫn lý luận có thể được sử dụng để đưa ra khuyến nghị cho người dùng. Bằng cách kết hợp lý luận đường dẫn vào quy trình đề xuất, phương pháp đề xuất nhằm đạt được hiệu suất đề xuất tốt hơn so với các phương pháp hiện có. Hiệu quả của Giai đoạn tinh tế và cách tiếp cận tổng thể được đánh giá thông qua các thử nghiệm sâu rộng trên bốn điểm chuẩn trong thế giới thực và đã quan sát thấy mức tăng đáng kể về hiệu suất đề xuất.

### Kết quả mô hình

Việc huấn luyện mô hình bao gồm hai phần chính:

* Việc tìm hiểu hồ sơ của người dùng sau này sẽ được sử dụng để hướng dẫn tìm kiếm.
* Việc học các Mô-đun suy luận biểu tượng thần kinh.

Chúng ta có thể tìm hiểu chính sách bằng cách thực thi train\_neural\_symbolic.py. Danh sách siêu tham số train\_agent được báo cáo như sau:

* --epochs: Số lượng epoch tối đa.
* --batch\_size: Kích thước lô.
* --lr: Tốc độ học tập.
* --deep\_module: Có sử dụng mô-đun sâu hay không.
* --embed\_size: Kích thước nhúng KG.
* use\_dropout: có sử dụng dropout hay không.
* Rank\_weight: hệ số trọng số cho việc giảm thứ hạng.
* topk\_candidates: hệ số trọng số cho việc mất thứ hạng.

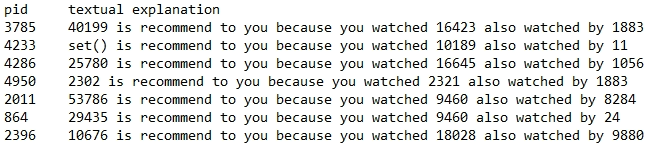
Kịch bản thực nghiệm:

* Kịch bản 1: Thực nghiệm với toàn bộ dữ liệu
* Kịch bản 2: Thực nghiệm với bộ dữ liệu loại bỏ item phổ biến
* Kịch bản 3: Thực nghiệm với bộ dữ liệu loại bỏ item phổ biến kết hợp tinh chỉnh parameter
* Kịch bản 4: Thực nghiệm với bộ dữ liệu lấy 1000 người dùng thường xuyên tương tác
* Kịch bản 5: Thực nghiệm với bộ dữ liệu lấy rating >= 3

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Recall@10** | **Recall@30** | **Recall@50** | **NDCG@50** |
| Round 1 | 0.0112 | 0.0126 | 0.0133 | 0.0301 |
| Round 2 | 0.0121 | 0.0146 | 0.0157 | 0.0252 |
| Round 3 | 0.0122 | 0.0148 | 0.0159 | 0.0257 |
| Round 4 | 0.0091 | 0.0100 | 0.013 | 0.0299 |
| Round 5 | 0.0132 | 0.0154 | 0.0164 | 0.0230 |

## Tạo gợi ý kèm giải thích và kiểm tra lỗi

Mô hình tạo gợi ý dựa trên đường dẫn giữa người dùng và người dùng trên tương tác giữa các phim với nhau. Bênh cạnh đó mô hình tạo gợi ý dựa trên các thể loại người dùng hay xem.



Tạo gợi ý của mô hình CAFE

Mô hình CAFE sinh gợi ý nhưng dựa hầu hết trên việc lịch sử tương tác của người dùng. Chưa sử dụng các thực thể như diễn viên và thể loại trong đồ thị.

Vì đường dẫn lý luận là nguồn sẽ được sử dụng để tạo ra lời giải thích bằng văn bản, nên việc tăng chất lượng đường dẫn lý luận (về một số thuộc tính) do đó sẽ nâng cao chất lượng của các lời giải thích bằng văn bản. Nên cần đầu ra của văn bản khuyến nghị cần đa dạng dựa trên các thực thể trong đồ thị.

## Đánh giá mô hình

* Kết quả mô hình đạt khoảng từ 1% tại recall at 50. Kết quả kém do đầu vào mô hình tri thức còn sơ sài – chúng tôi dựa trên dữ liệu hiện có của bộ dữ liệu bao gồm đạo diễn và diễn viên của bộ phim. Bộ phim đã có thêm các nút từ những thực thể của bộ phim hoặc diễn viên, đạo diễn. Tuy nhiên, mô hình lại luôn gợi ý kết quả với dữ liệu tương đồng với người dùng.
* Kết quả từ các kịch bản thực nghiệm mang kết quả tăng dần, đối với bộ dữ liệu loại bỏ những item phổ biến giúp mô hình tăng nhưng không đáng kể.
* Kết quả mô hình đạt kết quả tốt nhất tại kịch bản lấy những tương tác với rating > 3.
* Kết quả kém hơn do có thể tạo hồ sơ người dùng dựa trên tương tác trên các item kém. Có thể người dùng tương tác thấp và người dùng tương tác cao chênh lệch nhau quá nhiều. Các nút liên kết còn ít nên dẫn đến việc đồ thị tri thức còn nghèo nàn.

# TỔNG HỢP VÀ ĐÁNH GIÁ KẾT QUẢ

## Tổng hợp kết quả hai mô hình PGPR và CAFE

Mô hình CAFE là phương pháp suy luận biểu tượng thần kinh CoArse-to-FinE để đưa ra đề xuất có thể giải thích được, trong khi mô hình PGPR là phương pháp lý luận đường dẫn cũng không thể lập mô hình hồ sơ người dùng. Mô hình CAFE kết hợp biểu đồ tri thức (KG) có thể vào hệ thống đề xuất và tạo hồ sơ người dùng dưới dạng bản phác thảo thô về hành vi của người dùng để hướng dẫn quá trình tìm đường cho các đề xuất

Mặt khác, mô hình PGPR không xem xét hồ sơ người dùng và mẫu mẫu từ biểu đồ tri thức mà không có thông tin được cá nhân hóa. Kết quả thử nghiệm trên tập MovieLen1M cho thấy mô hình CAFE với hồ sơ người dùng tổng hợp thể hiện hiệu suất đề xuất tốt hơn so với mô hình PGPR và các đường cơ sở khác. Tuy nhiên, trên tập dữ liệu TV360-1M. Mô hình PGPR lại đạt hiệu quả tốt hơn mô hình CAFE.

Mô hình CAFE tận dụng hồ sơ người dùng để tìm ra các đường dẫn lấy người dùng làm trung tâm với chất lượng cao hơn, có nhiều khả năng dẫn đến các mục mà người dùng quan tâm hơn. Nhìn chung, mô hình CAFE vượt trội hơn mô hình PGPR về hiệu suất đề xuất trên các điểm chuẩn khác nhau

## Đề xuất tích hợp và hướng nâng cấp

Mô hình có thể được tích hợp vào hệ thống đề xuất để đưa ra các đề xuất có thể giải thích được. Bước đầu tiên là kết hợp các biểu đồ tri thức (KG) vào hệ thống, điều này có thể đạt được bằng cách sử dụng lý luận KG rõ ràng.

Đối với mô hình CAFE tạo hồ sơ người dùng dưới dạng bản phác thảo thô về hành vi của người dùng, ghi lại các hành vi nổi bật của người dùng từ lịch sử và cung cấp các tín hiệu có giá trị về các mẫu đường dẫn có nhiều khả năng dẫn đến các mục tiềm năng mà người dùng quan tâm. Để khai thác tốt hơn hồ sơ người dùng, một thuật toán tìm đường cải tiến có tên Lý do đường dẫn hướng dẫn hồ sơ (PPR) đã được phát triển, thuật toán này tận dụng các mô-đun suy luận biểu tượng thần kinh để tìm ra một loạt đường dẫn trên KG quy mô lớn một cách hiệu quả. Bằng cách tích hợp mô hình CAFE vào hệ thống, người dùng có thể nhận được các đề xuất không chỉ chính xác mà còn đi kèm với những lời giải thích dựa trên các đường dẫn lý luận xuất phát từ hồ sơ người dùng và lý luận KG. Tuy nhiên, với kết quả mô hình còn thấp chưa có thể đáp ứng được kỳ vọng.

Đối với mô hình PGPR, Mô hình có thể được tích hợp vào hệ thống bằng cách thực hiện theo quy trình từng bước. Đầu tiên, mạng lưới chính sách đã được đào tạo cần được đưa vào hệ thống khuyến nghị. Hệ thống sẽ lấy người dùng làm đầu vào và sử dụng mạng chính sách để tạo ra một tập hợp các mục ứng viên và đường dẫn suy luận tương ứng cho người dùng. Để đảm bảo tính đa dạng của đường dẫn, tìm kiếm chùm tia có thể được sử dụng, được hướng dẫn bởi xác suất hành động và phần thưởng, để khám phá các đường dẫn ứng viên và các mục được đề xuất cho mỗi người dùng. Thuật toán cho quy trình này được mô tả chi tiết trong Thuật toán 1, lấy người dùng, mạng chính sách, đường chân trời và kích thước lấy mẫu được xác định trước làm đầu vào và cung cấp một tập hợp các đường dẫn T-hop cho người dùng với xác suất và phần thưởng tạo đường dẫn tương ứng. Đầu ra của quá trình tích hợp sẽ là tập hợp các mục được đề xuất cùng với các đường dẫn suy luận liên quan của chúng, cung cấp bằng chứng có thể diễn giải được cho các đề xuất đưa ra cho người dùng.

Các lời giải thích có thể được để cung cấp khi người dùng chỉ vào item để đưa ra lời giải thích gây sự tò mò cho người dùng.

Tuy kết quả của mô hình đồ thị tri thức còn thấp nhưng tiềm năng của đồ thị vô cùng lớn. Các mô hình đồ thị tri thức sẽ có thể mở rộng và khai thác được những mối quan hệ giữa phim và phim, người dùng và người dùng, người dùng và phim. Việc bổ sung mở rộng thêm dữ liệu dễ dàng hơn so với các mô hình khác. Một số hướng nâng cấp các mô hình đồ thị tri thức như sau:

* Xây dựng đồ thị tri thức bằng việc bổ sung thêm những nút liên quan đến đặc trưng của phim
* Loại bỏ những cạnh được coi là độ yêu thích của người dùng đến phim chưa cao. Như vậy giúp thuật toán tính toán đường dẫn tốt hơn. Các thuật toán chưa sử dụng được trọng số của cạnh nên tính toán sẽ nhập nhằng giữa các cạnh có trọng số thấp và trọng số cao.

Xây dựng hàm hậu xử lý để tăng cường các ánh xạ cho các khía cạnh bổ sung đã bị suy giảm bằng quá trình xử lý hậu xử lý, ví dụ: đa dạng tương tác được liên kết (LID), đa dạng thực thể được chia sẻ (SED), nồng độ loại đường dẫn (PTC)