Hệ gợi ý phim

Cài đặt thử nghiệm các thuật toán

Đặng Quang Trung, Trần Bá Thiết

Đại Học Bách Khoa Hà Nội, Viện công nghệ thông tin và truyền thông, Việt Nam

Liên hệ: 20134145@student.hust.edu.vn

Giới thiệu

Hằng ngày chúng ta có các ý kiến về những thứ chúng ta thích hoặc không thích và thậm chí là không quan tâm đến nó. Ví dụ như bạn xem một chương trình truyền hình trên TV, bạn cảm thấy chương trình rất hay và hài hước hoặc thấy nó nhàm chán hay bạn không tìm thấy chương trình đó ở tất cả mọi kênh. Hoặc chương trình đó diễn ra mà chúng ta không để ý.

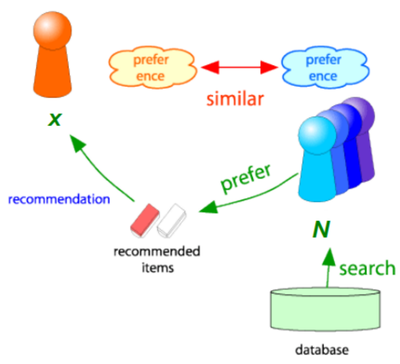
Sở thích của mỗi người là khác nhau, nhưng chúng ta tạo ra các dạng mẫu người dùng. Mọi người có xu hướng thích những thứ tương tự như những thứ mà họ thích.

Vì thế chúng ta cần có các chiến lược để gợi ý sản phẩm cho người dùng. Ở trình hai phương pháp:

- Collaborative Filtering
- Latent Factor Model

Collaborative Filtering

Ý tưởng basic:



- quan sát một người dùng x.
- Tìm tập N các người dùng khác cũng rating giống như các ratings của người dùng x.
- Ước lượng các ratings của người dùng x dựa trên các ratings của những người trong tập N.

Tìm tập người tương đồng:

- Độ đo tương đồng Cosine
  - $sim(x,y) = \cos(\vec{r}_x, \vec{r}_y) = \frac{\vec{r}_x.\vec{r}_y}{\|\vec{r}_x\|.\|\vec{r}_y\|}$
  - Vấn đề: khắc phục được nhược điểm của jaccard nhưng lại bỏ qua các ratings không tốt(người dùng đánh giá thấp bộ phim đó)
- Sử dụng hệ số tương phản cá nhân
  - S_{xy} là tập các bộ phim được rating bởi x và y.
  - $sim(x,,y) = \sum s \in S \downarrow xy \uparrow (r \downarrow xs - r \downarrow x)r \downarrow ys - r \downarrow y)/\sqrt{\sum s \in S \downarrow xy \uparrow (r \downarrow xs - r \downarrow x) \uparrow 2}$

Dự đoán rating người dùng:

$$\hat{r_{xi}} = b_{xi} + \frac{\sum_{j \in N(i;x)} s.(r_{xj} - b_{xj})}{\sum_{j \in N(i;x)} s}$$

Với $b_{xi} = \mu + b_x + b_i$ là độ lệch cơ sở của người dùng x khi đánh giá bộ phim i. Trong đó:

- μ là rating trung bình của tất cả người dùng.
- b_x là độ lệch của người dùng x.

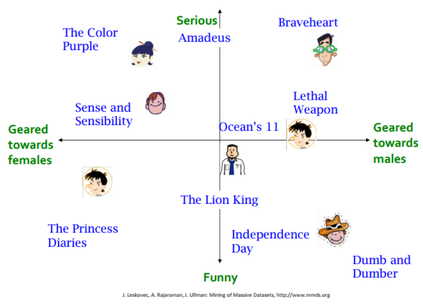
  $b_x = avg(x) - \mu$.
- b_i là độ lệch của bộ phim i.

  $b_i = avg(i) - \mu$.

Latent Factor Models

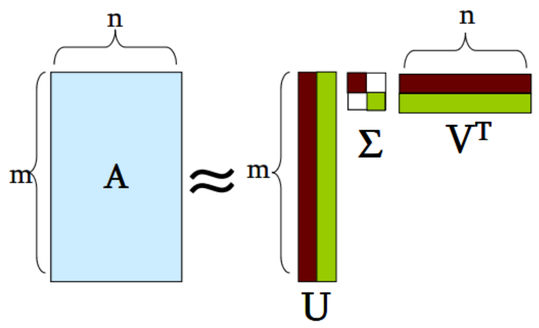
Ý tưởng:

Trong thức tế mỗi người dùng sẽ có những sở thích riêng của mình ví dụ như có người thích phim tình cảm, lãng mạn có người thích phim hành động, khoa học viễn tưởng, ... Những sở thích đó được coi là một nhân tố ẩn của mỗi người dùng.



- Các ratings chịu ảnh hưởng sâu sắc bởi một bộ các nhân tố(factor) rất cụ thể cho miền.
- Một số các factors là rất khó quan sát được và khó ước lượng sự tác động của chúng đối với ratings của người dùng.
- Mục tiêu là suy luận những nhân tố tiềm ẩn từ dữ liệu đánh giá bằng cách sử dụng các kỹ thuật toán học.

SVD:

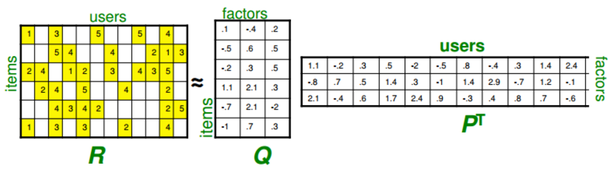


Hình 1: Hình ảnh minh họa svd

Trong đó:

- U là ma trận với các cột là các vector trái.
- S là ma trận cùng kích thước như A giá trị đơn là đường chéo.
- V^T có các hàng là cá vector đơn phải

Dự đoán rating người dùng:



Hình 2: Ma trận sau khi giảm chiều sử dụng svd

Trong đó:

- A = R là ma trận người dùng ban đầu.
- Q là ma trận nhân tố của phim.
- P^T là ma trận nhân tố của người dùng.

  $P^T = \sum V^T$

Để dự đoán rating của người dùng x cho một bộ phim ra chỉ cần tính:

$$r_{xi} = \underbrace{\mu}_{overallmeanrating} + \underbrace{b_x}_{biasforuserx} + \underbrace{b_i}_{biasformoivei} + q_x.p_i$$

Trong đó:

- q_i là hàng i của ma trận Q.
- p_x là cột x của ma trận P.
- f là số factors(hay độ dài vector của hàng i và cột x).

Vấn đề tối thiểu hóa lỗi và overfitting:

$$\min_{Q,P} \sum_{(x,i) \in R} (r_{xi} - (\mu + b_x + b_i + q_i.p_x))^2$$

$$+(\lambda_1 \sum_i \|q_i\|^2 + \lambda_2 \sum_x \|p_x\|^2 + \lambda_3 \sum_x \|b_x\|^2 + \lambda_4 \sum_i \|b_i\|^2)$$

Trong đó:

- $\lambda_1, \lambda_2, \lambda_3, \lambda_4$ là các hệ số phạt.
  + Để đơn giản ở đây tất cả dùng chung λ
- b_i, b_x là độ lệch của moive và người dùng.
- μ là raitng trung bình của ma trận người dùng.
- q_i, p_x là các vector hàng và cột của ma trận Q,P.

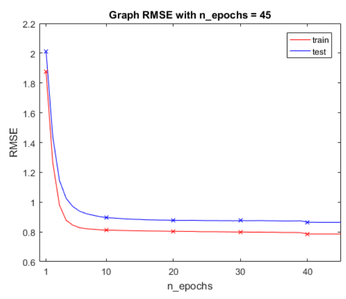
Kết quả thực nghiệm

Collaborative:

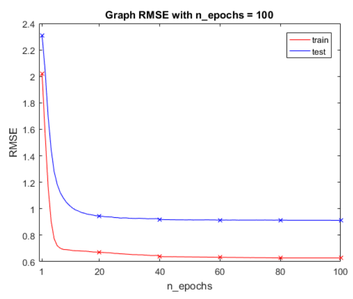
Bộ dư liệu	k(hàng xóm)	RMSE
ml-100k	3	0.9928
ml-100k	10	0.9236
ml-100k	15	0.921
ml-100k	5	0.9537
ml-1m	5	0.8835

Latent factor model:

Bộ dữ liệu	k(factor)	n_eopchs	RMSE
ml-100k	40	100	0.9161
ml-1m	40	45	0.8626



Hình 3: Hình ảnh RMSE với tập dữ liệu ml-100k



Hình 4: Hình ảnh RMSE với tập dữ liệu ml-1m

Kết luận:

- Cả hai phương pháp dễ tiếp cận và cài đặt trong thực tiễn.
- Qua thực nghiệm thấy rằng Latent cho kết quả tốt hơn Collaborative Filltering basic

Nội dung của poster này về hê gợi ý môn học khai phá web, mã học phần IT4868