Book Recommendation System

IT562 : Recommendation Engine & Applications

Prof. Sourish Dasgupta

Team : Angry Nerds

Team Members:

- Tasneem Pulavwala (201711007)
- Jainam Shah (201711010)
- Vandan Thaker (201711013)
- Dharit Parikh (201711022)

Dataset

- Book-Crossing Dataset compiled by Cai-Nicolas Ziegler
- Total ratings : 1.1 million
- Ratings of 270,000 books by 90,000 users
- Rating scale : 1-10

Our Approach

ESVD(Density-oriented Approach)

- Top 1000 top rated book ratings : 162,767
- Top 1500 active users ratings : 620,489
- Intersection of most rated book ratings and most active user ratings : 73,101
- Apply SVD and fill missing values
- Combine this with original rating matrix : 487,018 (previously : 433,671) i.e. 12.3% more denser

Our Approach (contd...)

- Apply SVD on the entire (denser) user-books rating matrix
- Predict ratings of all unrated books for a given user, store it as list [P]
- Find user's favourable authors (and genres) of top 10 rated books, set {K}
- Find 5000 books that have highest similarity with user's favourable authors (and genres) {K}, store it as list [L]
- Sort and select top 200 books on the basis of their estimated ratings from SVD and recommend 8 books out of them
- The remaining 2 books are the top 2 books from [P] - [L]
- Repeat last 2 steps for 5 iterations to get total of 50 recommendations

Our Approach (contd..)

Reinforcement learning:

- State Space - last N books user has browsed
- Action Space - books recommended to the user
- Reward
  - Click - 5
  - Like - 10
  - Rate - rating given
  - Ignored - 0
- Memory

Reinforcement Learning (contd..)

Memory:

- For a state space $s = \{s_0, s_1, s_2, \ldots, s_n\}$, action space $a = \{a_0, a_1, a_2, \ldots, a_{10}\}$, reward r for each item in action space
- Store triple $((s, a) \longrightarrow r)$ in memory set $\{M\}$
- Each time a user interacts with items in the recommended list, we add these items to the end of state space and remove the same number of items in the top of the state space.
- For example, recommending a list of five items $\{a_1, \ldots, a_5\}$ to a user, if the user clicks a_1 and rates a_5, then update $s = \{s_3, \ldots, s_n, a_1, a_5\}$.

Reinforcement Learning (contd..)

Similarity:

- We calculated the similarity of the current state-action pair, say $p_t(s_t, a_t)$, to each existing historical state-action pair in the memory.

$$Cosine(p_t, m_i) = \alpha \frac{s_t s_i^{\top}}{||s_t|| ||s_i||} + (1 - \alpha) \frac{a_t a_i^{\top}}{||a_t|| ||a_i||},$$

- With the increase of similarity between p_t and m_i , there is a higher chance of p_t mapping to the reward r_i.

Cold-Start Problem

- New User
  - Ask user to provide favourite books or rate for popular books
  - Otherwise show top rated books
- New Book
  - Content based recommendation using author (and genre) similarity

Top rated books by user

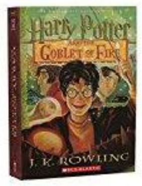
- Goodbye to the Buttermilk Sky
- The Witchfinder (Amos Walker Mystery Series)
- More Cunning Than Man: A Social History of Rats and Man
- Clara Callan
- Harry Potter and the Chamber of Secrets
- Where You'll Find Me: And Other Stories

Top recommended books

Top reccomended books for this user:

['The Return of the King (The Lord of the Rings, Part 3)']

['Harry Potter and the Goblet of Fire (Book 4)']

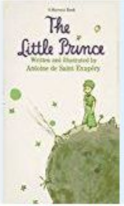


['Harry Potter and the Prisoner of Azkaban (Book 3)']

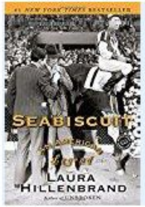
['The Two Towers (The Lord of the Rings, Part 2)']

['Gone with the Wind']

['The Little Prince']



['Seabiscuit: An American Legend']



['Griffin & Sabine: An Extraordinary Correspondence']

Thank you