The Comparison of Recommendation Approaches in MovieLens

Lichao Han
Ho Wai Lok

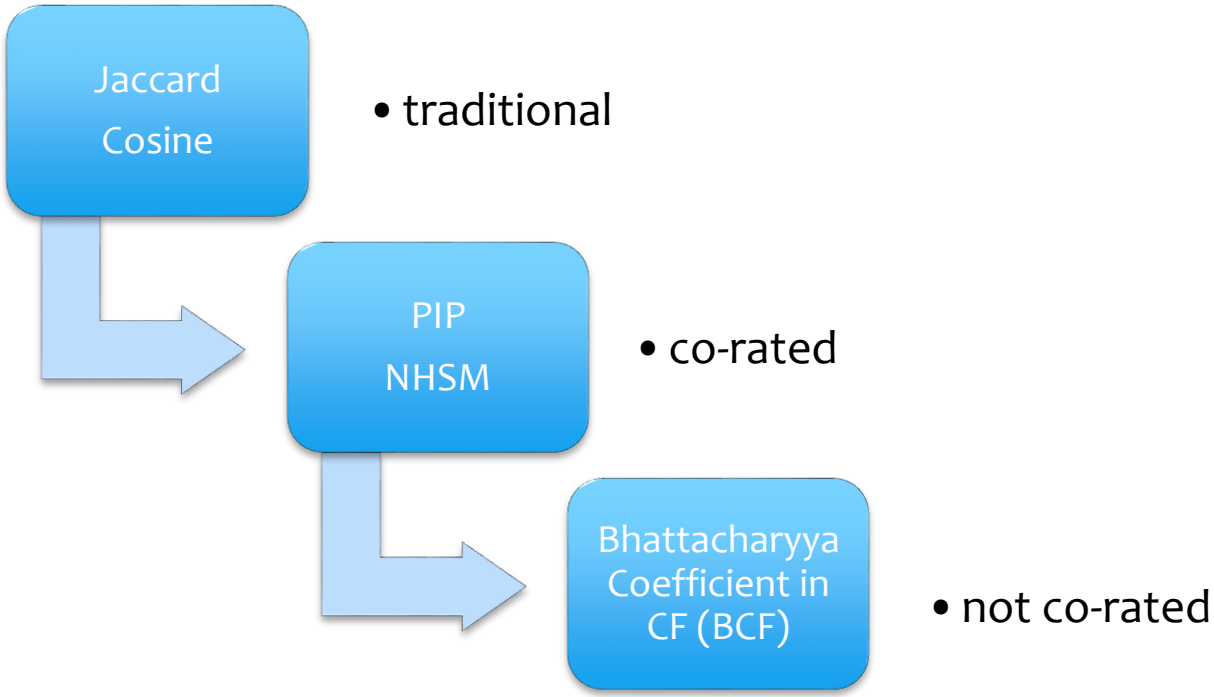
Outline

- Introduction
- Similarity Measures
- Implementation
- Results
- Reference

Introduction

- Recommender System (RS)
- Provides personalized suggestions for products or items to individual users
- Content-based VS Collaborating filtering (CF)
- Neighborhood based CF

Similarity Measures



Jaccard

$$s(U,V)^{Jacc} = \frac{|I_U \text{ Ç } I_V|}{|I_U \text{ È } I_V|}$$

- I_U is the set of items rated by user *U*

Cosine

$$s(U,V)^{Cos} = \frac{\sum_{I \in I'} (r_{UI})(r_{VI})}{\sqrt{\sum_{I \in I'} r_{UI}^2 \sum_{I \in I'} r_{VI}^2}}$$

- r_{UI} is the rating made by user *U* on item *I* and *I'* is the set of co-rated items

Jaccard & Cosine

Similarity Measures	Advantages	Disadvantages
Jaccard	Easy to compute	Suffers from few co-rated item problem; Does not take absolute value (rating) into account; Uses local information of the ratings only
Cosine	Easy to compute	Suffers from few co-rated item problem; Outputs high similarity in spite of significant difference in ratings; Uses local information of the ratings only

PIP

$$s(U,V)^{PIP} = \sum_{p \in I} PIP(r_{U,P}, r_{V,P})$$

where $PIP(r_{U,P}, r_{V,P}) = Proximity(r_{U,P}, r_{V,P}) * Impact(r_{U,P}, r_{V,P}) * Popularity(r_{U,P}, r_{V,P})$

PIP

$$Agreement(r_1, r_2) \begin{cases} \textit{false} \text{ if } (r_1 > R_{med} \text{ and } r_2 < R_{med}) \text{ or} \\ \qquad (r_1 < R_{med} \text{ and } r_2 > R_{med}) \\ \textit{true} \text{ otherwise} \end{cases}$$

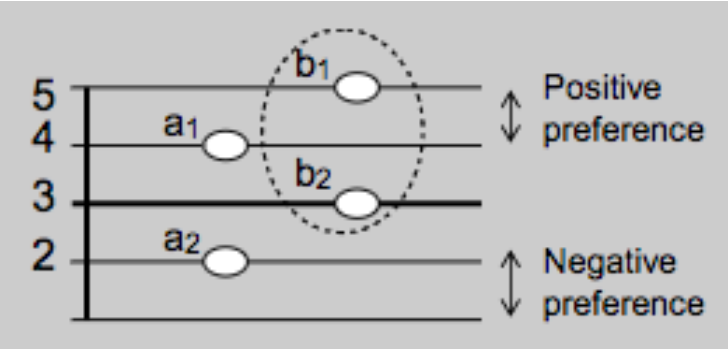
where $R_{med} = \dfrac{R_{\max} + R_{\min}}{2}$

PIP

$$Proximity(r_1, r_2) = \left\{ \left\{ 2 \times \left(R_{\max} - R_{\min} \right) + 1 \right\} - D(r_1, r_2) \right\}^2$$

where $D(r_1, r_2) = |r_1 - r_2|$ if $Agreement(r_1, r_2)$ is true

$= 2|r_1 - r_2|$ if $Agreement(r_1, r_2)$ is false



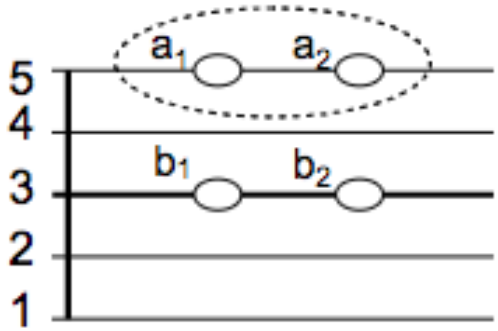
PIP

if $Agreement(r_1, r_2)$ is true

$$Impact(r_1, r_2) = \left(\left|r_1 - R_{med}\right| + 1\right)\left(\left|r_2 - R_{med}\right| + 1\right)$$

if $Agreement(r_1, r_2)$ is false

$$Impact(r_1, r_2) = \frac{1}{\left(\left|r_1 - R_{med}\right| + 1\right)\left(\left|r_2 - R_{med}\right| + 1\right)}$$



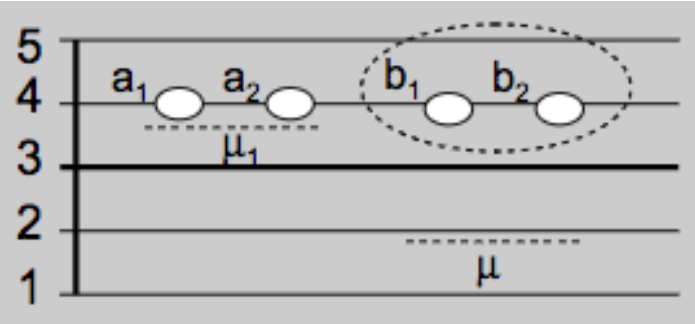
PIP

if ($r_1 > m_k$ and $r_2 > m_k$) or ($r_1 < m_k$ and $r_2 < m_k$)

$$Popularity(r_1, r_2) = 1 + \left(\frac{r_1 + r_2}{2} - m_k \right)^2$$

otherwise

$$Popularity(r_1, r_2) = 1$$



NHSM

$$s(U,V)^{NHSM} = s(U,V)^{JPSS} \times s(U,V)^{URP}$$, where

$$s(U,V)^{JPSS} = s(U,V)^{PSS} \times s(U,V)^{Jacc'}$$

$$s(U,V)^{URP} = 1 - \frac{1}{1 + \exp(-|m_U - m_V| \times |s_U - s_V|)}$$

NHSM

$$s(U,V)^{PSS} = \sum_{p \in I} PSS(r_{U,P}, r_{V,P})$$

, where

$$PSS(r_{U,P}, r_{V,P}) = Proximity(r_{U,P}, r_{V,P}) * Significance(r_{U,P}, r_{V,P}) * Singularity(r_{U,P}, r_{V,P})$$

$$Proximity(r_{U,P}, r_{V,P}) = 1 - \frac{1}{1 + \exp(-|r_{U,P} - r_{V,P}|)}$$

$$Significance(r_{U,P}, r_{V,P}) = \frac{1}{1 + \exp(-|r_{U,P} - r_{med}| \times |r_{V,P} - r_{med}|)}$$

$$Singularity(r_{U,P}, r_{V,P}) = 1 - \frac{1}{1 + \exp - \left(\left| \frac{r_{U,P} + r_{V,P}}{2} \right| - \mu_p \right)}$$

- μ_p is the average rating of item *p*
- $r_{U,P}$ is the rating of item *p* by user *u*

NHSM

$$s(U,V)^{JPSS} = s(U,V)^{PSS} \times s(U,V)^{Jacc'}$$

where $s(U,V)^{Jacc'} = \frac{|I_U \cap I_V|}{|I_U| \cdot |I_V|}$

$$s(U,V)^{PSS} = \sum_{p \in I} PSS(r_{U,P}, r_{V,P})$$

NHSM

$$s(U,V)^{URP} = 1 - \frac{1}{1 + \exp(-|\mu_U - \mu_V| \times |\sigma_U - \sigma_V|)}$$

- μ_U and μ_V is the mean rating of user *U* and *P* respectively
- σ_U and σ_V represents the standard variance of user *U* and *P*

PIP & NHSM

Similarity Measures	Advantages	Disadvantages
PIP	Captures global information of the concerned item	Suffers from few co-rated item problem; Unnecessarily penalize more than once while computing *proximity* and *impact* factors
NHSM	Captures global information of the concerned item	Suffers from few co-rated item problem

Bhattacharyya Coefficient in CF (BCF)

$$s(U,V)^{BC} = Jacc(U,V) + \sum_{i \in I_U} \sum_{j \in I_V} BC(i,j) loc(r_{Ui}, r_{Vj})$$

, where

$$BC(i,j) = BC(\widehat{p_i}, \widehat{p_j}) = \sum_{h=1}^{m} \sqrt{(\widehat{p_{ih}})(\widehat{p_{jh}})}$$

$$loc(r_{Ui}, r_{Vj}) = \frac{(r_{Ui} - \overline{r}_U)(r_{Vj} - \overline{r}_V)}{s_U s_V}$$

Bhattacharyya Coefficient in CF (BCF)

$$BC(i,j) = BC(\widehat{p_i}, \widehat{p_j}) = \sum_{h=1}^{m} \sqrt{(\widehat{p_{ih}})(\widehat{p_{jh}})}$$, where $\widehat{p_{ih}} = \frac{\#h}{\#i}$

- *m* is the number of bins
- *#i* is the number of users rated the item *I*
- *#h* is the number of users item *i* with rating value '*h*'
- $\sum_{h=1}^{m} \widehat{p_{ih}} = \sum_{h=1}^{m} \widehat{p_{jh}} = 1$

Bhattacharyya Coefficient in CF (BCF)

$$BC(i,j) = BC(\widehat{p_i}, \widehat{p_j}) = \sum_{h=1}^{m} \sqrt{(\widehat{p_{ih}})(\widehat{p_{jh}})}$$, where $\widehat{p_{ih}} = \frac{\# h}{\# i}$

- Example:

$$I = (1,0,2,0,1,0,2,0,3,0)^T \qquad J = (0,1,0,2,0,1,0,2,0,3)^T$$

$$BC(i,j) = \sum_{h=1}^{3} \sqrt{\widehat{I_h J_h}} = \sqrt{\left(\tfrac{2}{5}\right) * \left(\tfrac{2}{5}\right)} + \sqrt{\left(\tfrac{2}{5}\right) * \left(\tfrac{2}{5}\right)} + \sqrt{\left(\tfrac{1}{5}\right) * \left(\tfrac{1}{5}\right)} = 1$$

Bhattacharyya Coefficient in CF (BCF)

Similarity Measures	Advantages	Disadvantages
Bhattacharyya Coefficient in CF (BCF)	Does not depend on co-rated items; Use both local and global information; Comprehensively use numerical values of the ratings	Relatively difficult to compute

Implementation

- Dataset
  - MovieLens 100K
  - 943 users
  - 1682 movie items
  - 100000 ratings (scale: 1,2,3,4,5)
  - 80% training data; 20% testing data
- Programming Language
  - Python

Implementation

- Similarity Measures
  - Jaccard
  - Cosine
  - PIP
  - NHSM
  - Bhattacharyya

Implementation

- Rating Prediction

$$\widehat{r_{ui}} = \bar{r}_u + \frac{\sum_{k=1}^{K} s(U_u, U_k) * (r_{ki} - \bar{r}_{k_u})}{\sum_{k=1}^{K} |s(U_u, U_k)|}$$

- $\bar{r}_u$ is the average of the ratings made the user U_u
- $s(U_u, U_k)$ denotes the similarity value between user U_u and its *k*th neighbor
- $\bar{r}_{k_u}$ is the average of the ratings made by *k*th neighbor of the user U_u
- r_{ki} is the rating made by *k* neighbor on *i*th item

Implementation

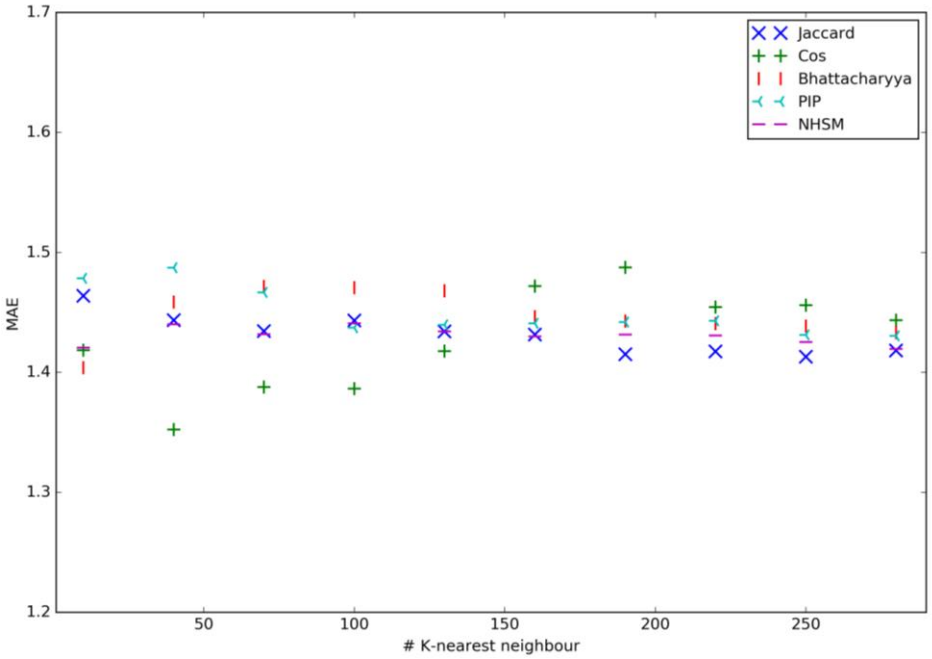
- Mean Absolute Error

$$MAE = \frac{\sum_{i=1}^{MAX} \left| r_i - \hat{r}_i \right|}{MAX}$$

r_i: actual rating
$\hat{r}_i$: predicted rating by a CF algorithm
MAX: number of times the predictions are performed by a CF algorithm

Results

- Mean Absolute Error



Results

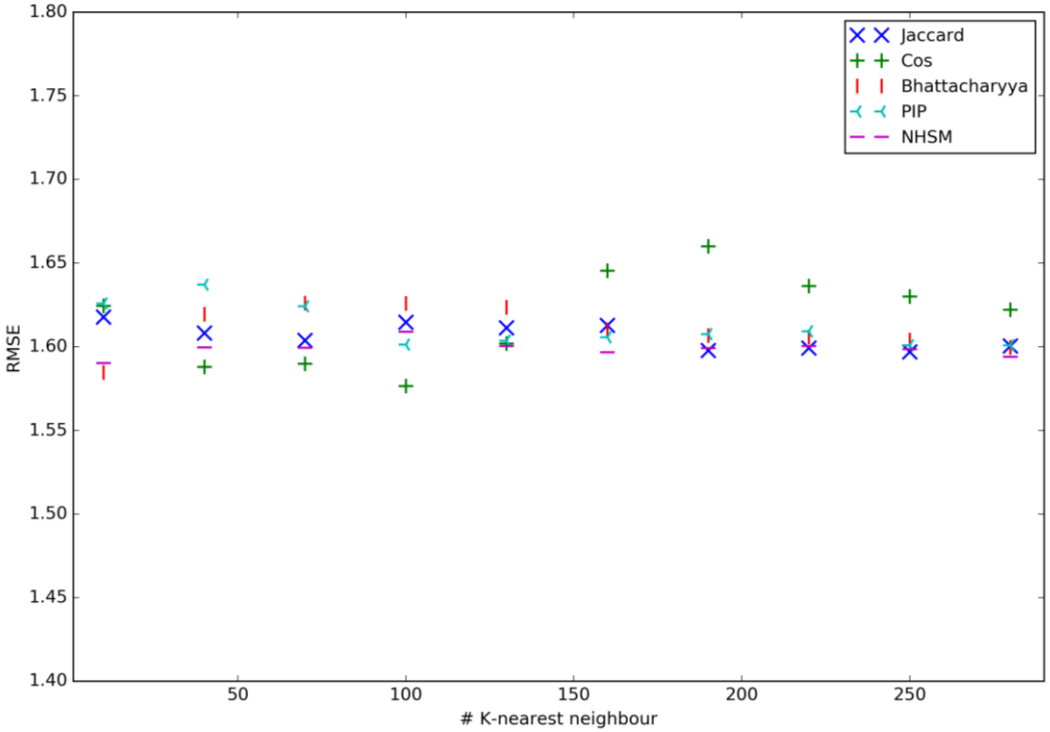
- Root Mean Squared Error

$$RMSE = \sqrt{\frac{1}{MAX} \sum_{i=1}^{MAX} \left(r_i - \hat{r}_i\right)^2}$$

r_i : actual rating
$\hat{r}_i$: predicted rating by a CF algorithm
MAX: number of times the predictions are performed by a CF algorithm

Experiment and Results

- Root Mean Squared Error



References

[1] Bidyut Kr. Patra, Raimo Launonen, Ville Ollikainen, Sukumar Nandi (2015). A New Similarity Measure Using Bhattacharyya Coefficient for Collaborative Filtering In Sparse Data.

[2] Haifeng Liu, Zheng Hu, Ahmad Mian, Hui Tian, Xuzhen Zhu (2014). A New User Similarity Model To Improve The Accuracy Of Collaborative Filtering.

[3] Hyung Jun Ahn (2008). A New Similarity Measure For Collaborative Filtering To Alleviate The New User Cold-Starting Problem.

[4] Keunho Choi, Yongmoo Suh (2013). A New Similarity Function For Selecting Neighbors For Each Target Item In Collaborative Filtering.

References

[5] Sama Jamalzehi, Mohammad Bagher Menhaj (2016). A New Similarity Measure Based On Item Proximity And Closeness For Collaborative Filtering Recommendation

[6] Gediminas Adomavicius, Alexander Tuzhilin (2005). Toward the Next Generation Of Recommender Systems: A Survey Of The State-Of-The-Art And Possible Extensions

[7] Sama Jamalzehi, Mohammad Bagher Manhaj (2016). Scalable User Similarity Estimation Based On Fuzzy Proximity For Enhancing Accuracy Of Collaborative Filtering Recommendation