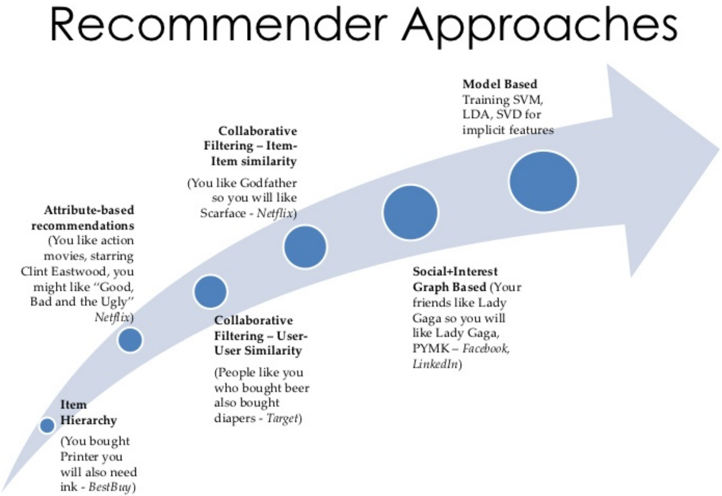
Know your customers

Aris Tritas   Laurent Cetinsoy

13 décembre 2016

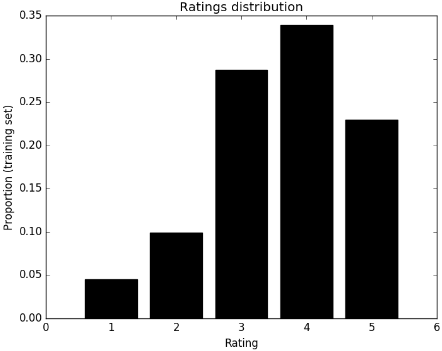
Overview


Recommender Approaches
Item Hierarchy (You bought Printer you will also need ink - BestBuy)
Attribute-based recommendations (You like action movies, starring Clint Eastwood, you might like "Good, Bad and the Ugly" Netflix)
Collaborative Filtering – User-User Similarity (People like you who bought beer also bought diapers - Target)
Collaborative Filtering – Item-Item similarity (You like Godfather so you will like Scarface - Netflix)
Social+Interest Graph Based (Your friends like Lady Gaga so you will like Lady Gaga, PYMK – Facebook, LinkedIn)
Model Based Training SVM, LDA, SVD for implicit features

Customer affinity dataset

Courtesy AlixML

- More than 2 millions of ratings of 3560 items from 93705 users
- Sparsity : $< 1\%$ of matrix filled
- Split in 2.5M train & 1.3M test examples
- #$\{$Ratings/Item$\}$: median=71, mean=712
- #$\{$Ratings/User$\}$: median=13, mean=28



Collaborative filtering

Non negative factorization

$$R = \begin{pmatrix} n_{11} & n_{12} & \cdots & n_{1i} \\ n_{21} & n_{22} & \cdots & n_{2i} \\ \vdots & \vdots & \vdots & \vdots \\ n_{u1} & n_{u2} & \cdots & n_{ui} \end{pmatrix} = \overbrace{\begin{pmatrix} n_{11} & \cdots & n_{1k} \\ n_{21} & \cdots & n_{2k} \\ n_{31} & \cdots & n_{3k} \\ \vdots & \vdots & \vdots \\ n_{u1} & \cdots & n_{uk} \end{pmatrix}}^{U} \cdot \overbrace{\begin{pmatrix} n_{11} & n_{12} & \cdots & n_{1i} \\ n_{21} & n_{22} & \cdots & n_{2i} \\ \vdots & \vdots & \vdots & \vdots \\ n_{k1} & n_{k2} & \cdots & n_{ki} \end{pmatrix}}^{V^{\mathrm{T}}}$$

A new rating is computed as follows :

$$\tilde{r}_{ij} := u_i \cdot v_j^{\mathrm{T}} = \sum_{p=1}^{k} u_{ip} v_{pj}$$

Collaborative filtering

Non Negative Matrix Factorization

$$U^*, V^* = \arg\min_{U,V} \frac{1}{2} ||R - UV^{\mathrm{T}}||^2 + \lambda \cdot \Omega(U, V)$$

$$R \in \mathbb{R}^{n\times m}, U \in \mathbb{R}^{n\times k}, V \in \mathbb{R}^{m\times k}$$

Collaborative filtering

Non Negative Matrix Factorization

$$U^*, V^* = \arg\min_{U,V} \frac{1}{2} ||R - UV^{\mathrm{T}}||^2 + \lambda \cdot \Omega(U, V)$$

$$R \in \mathbb{R}^{n \times m}, U \in \mathbb{R}^{n \times k}, V \in \mathbb{R}^{m \times k}$$

Usual regularization : $\Omega(U, V) = ||U||^2 + ||V||^2$

Collaborative filtering

Non Negative Matrix Factorization

$$U^*, V^* = \arg\min_{U,V} \frac{1}{2} ||R - UV^{\mathrm{T}}||^2 + \lambda \cdot \Omega(U, V)$$

$$R \in \mathbb{R}^{n \times m}, U \in \mathbb{R}^{n \times k}, V \in \mathbb{R}^{m \times k}$$

Usual regularization : $\Omega(U, V) = ||U||^2 + ||V||^2$
Tikhonov regularization : $\Omega(U, V) = \sum_i \mathcal{J}(i)||U_i||^2 + \sum_i \mathcal{I}(j)||V_i||^2$

NNMF

Gradient Descent optimization

Update rule using prediction error $e_{ij} = r_{ij} - \tilde{r}_{ij}$:

- $u_i \leftarrow u_i + \gamma \cdot (e_{ij} \cdot v_j - \lambda \cdot u_i)$
- $v_j \leftarrow v_j + \gamma \cdot (e_{ij} \cdot u_i - \lambda \cdot v_j)$

avec $\gamma > 0$ le taux d'apprentissage, $\lambda > 0$ le coef. de régularisation

NNMF

Algorithm 1 ALS-WR

while not happy **do**
  for $i \in [n]$ **do**
    $\hat{\boldsymbol{U}}_i \leftarrow \arg\min_{\boldsymbol{U}} \sum_{j \in \mathcal{J}(i)} (r_{ij} - \boldsymbol{U}\hat{V}_j^{\mathrm{T}}) + \lambda \cdot \#\mathcal{J}(i)||\boldsymbol{U}||$
  end for
  for $j \in [m]$ **do**
    $\hat{\boldsymbol{V}}_j \leftarrow \arg\min_{\boldsymbol{U}} \sum_{i \in \mathcal{I}(j)} (r_{ij} - \hat{\boldsymbol{U}}_i V^{\mathrm{T}}) + \lambda \cdot \#\mathcal{I}(j)||\boldsymbol{V}||$

  end for
end while

Results

NMF

$k = 15$

Model	ratings	k	lambda	RMSE
GD	500k	15	0.02	0.863
GD	All	15	0.02	0.951
ALS	500k	15	0.03	0.877
ALS	All	15	0.03	0.925
ALS	All	30	0.03	0.923
ALS	All	100	0.1	0.934

Results

Gradient descent

$k = 15, \lambda = 0.02$

Ratings	Left deletion	Righ deletion	RMSE
500k	0	0	0.863
500k	0	0.2	0.879
All	0	0	0.898
All	0	0.2	0.911

NMF - Results

ALS-WR

NMF optimized with ALS-WR

$k = 15$

Rating	Left deletion	right deletion	RMSE
500k	0	0	0.877
500k	0.2	0	0.8867
500k	0.15	0.15	0.895
All	0	0	
All	0	0.2	0.939
All*	0	0	RMSE = 0.934

* $k = 30, \lambda = 0.03$

Collaborative Filtering Networks

[Salakhutdinov et al. 06]

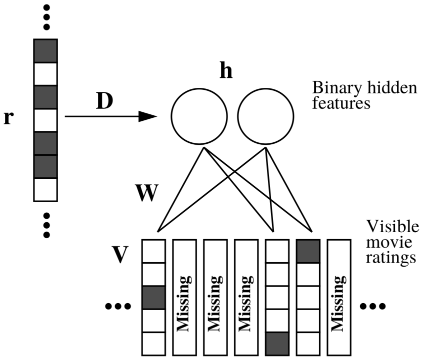


Figure 2. Conditional RBM. The binary vector **r**, indicating rated/unrated movies, affects binary states of the hidden units.

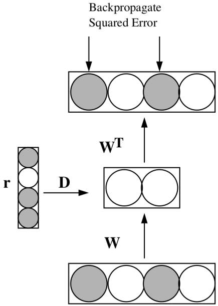


Figure 5. The "unrolled" RBM used to create an autoencoder network which is then fine-tuned using backpropagation of error derivatives.

Hybrid collaborative Filtering with Autoencoders

[Strub2016]

$$f(\boldsymbol{x}) = \sigma(\boldsymbol{W_1 x} + \boldsymbol{b_1}),\ g(\boldsymbol{y}) = \sigma(\boldsymbol{W_2 y} + \boldsymbol{b_2})\ \ nn(\boldsymbol{x}) = g(f(\boldsymbol{x}))$$

$$L_{2,\alpha,\beta} = \alpha(\sum_{j \in \mathcal{C}(\tilde{x})})[nn(\tilde{x}_j) - x_j]^2 + \beta(\sum_{j \notin \mathcal{C}(\tilde{x})} [nn(\tilde{x}_j) - x_j]^2)$$

avec $\tilde{x} \in \mathbb{R}^N, W_1 \in \mathbb{R}^{K \times n}, W_1 \in \mathbb{R}^{n \times K}, b_1 \in \mathbb{R}^k, b_2 \in \mathbb{R}^N$

Hybrid collaborative Filtering with Autoencoders [Strub2016]

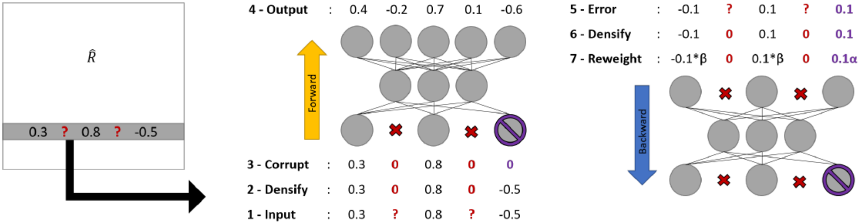
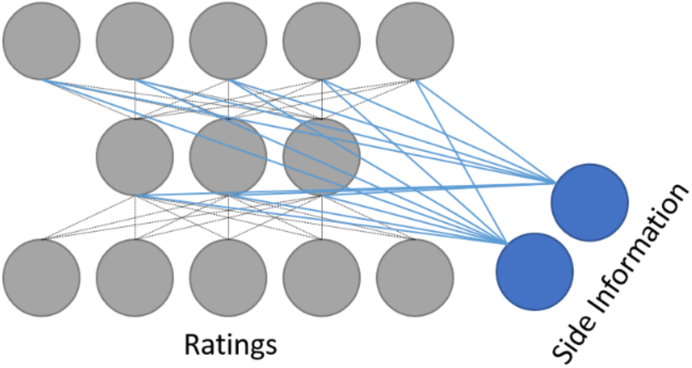


FIGURE – On the forward pass the sparse input vector is densified and then corrupted. Before backpropagation, the error is reweighted and missing values are put to zero.

Injecting side-information



Results

Hybrid collaborative filtering with Autoencoders

Model	Rating	Left removal	right removal	RMSE
AE Item Vector	2M	0	0	0.938
AE User Vector	2M	0	0	0.913
AE User Vector	2M	0	0.2	0.947

Results

Comparison between ALS and autoencoder networks.

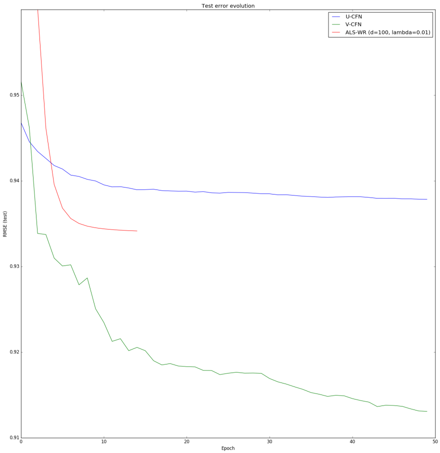


FIGURE – Test error evolution

Conclusion, further work and ideas

- Add terms accounting for bias or temporal dynamics [Koren2009]

Conclusion, further work and ideas

- Add terms accounting for bias or temporal dynamics [Koren2009]
- Further investigate dependency on succession

Conclusion, further work and ideas

- Add terms accounting for bias or temporal dynamics [Koren2009]
- Further investigate dependency on succession
- Hyperparameter tuning, better validation, effect of expert polling ?

- Y. Koren, R. Bell, C. Volinsky - Matrix factorization techniques for recommender systems. 2009 IEEE Computer Society
- F. Strub, J. Mary, R. Gaudel - Hybrid Collaborative Filtering with Auto-Encoders. Arxiv
- F. Strub, J. Mary, R. Gaudel - https://github.com/fstrub95/Autoencoders_cf
- https://github.com/lcetinsoy/collaborative-filtering