The Netflix Prize and Recommender Systems

Andreas Töscher and Michael Jahrer

25. Juni 2008

Andreas Töscher

- Telematik student since 2003
- BA 2006
- Computational Intelligence / Computer Vision

Michael Jahrer

- Telematik student since 2003
- BA 2006
- System on Chip Design / Computer Vision / Computational Intelligence

Netflix as Company

Netflix

- Netflix is a US online movie rental service
- Lend movies over mail
- over 100.000 titles
- 55 million DVDs total
- Productive start at 1997
- Have own recommendation system called "Cinematch"
  - Based on linear neighborhood model with a lot of data conditioning
- Approximately 60% of Netflix members select their movies based on movie recommendations

Netflix Prize

- Grand Prize, 1 Mio. US-Dollar for 10% improvement in prediction accuracy
- Progress Prize, 50.000 US-Dollar, October every year
- Starts Oct, 2 2006
- End Oct, 2 2011
  - or someone reach 10% improvement in RMSE
- www.netflixprize.com

Overview Recommendation Systems

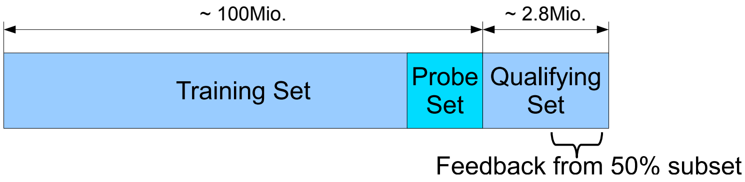
- Extraction of user´s taste
- Top-k recommendations
  - List on online account
  - Recommendations as personal email

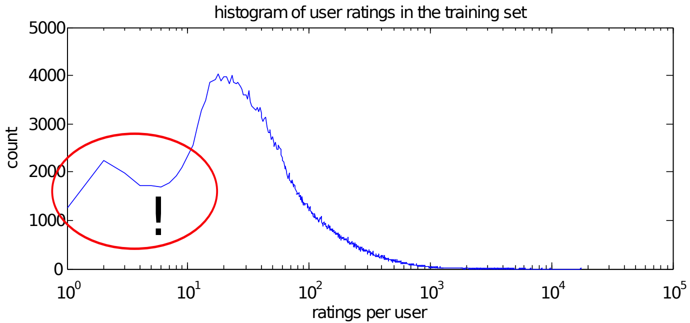
- Examples
  - Netflix
  - Amazon.com
  - last.fm

Netflix Facts

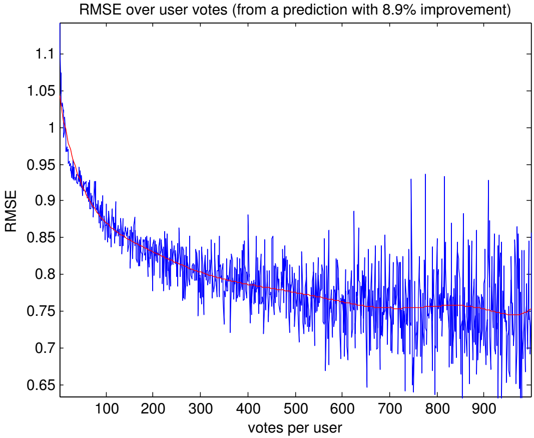
- Dataset consists of 100 Mio. entries
- Quadruples of $< movie, user, rating, date >$
- Integer ratings from 1 to 5
- Error measure: RMSE (root mean square error)
  $RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(r_i - t_i)^2}$ $(r_i = prediction \;\; t_i = target)$
- Over 30.000 contestants from 170 countries

Dataset Details





Average error as function of user votes



- Grand Prize level at $\sim$ 100 votes/user

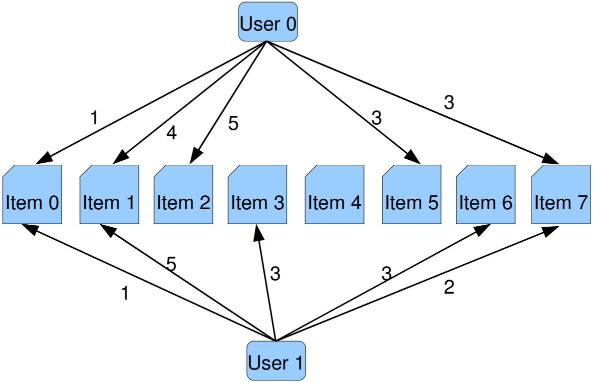
RMSE Scores

- 0.8563 (10.0%) Grand Prize
- 0.8643 (9.15%) Leader
- 0.8667 (8.90%) Our current progress
- 0.8712 (8.43%) Progress Prize Winner 2007
- 0.9514 (0.0%) Netflix Cinematch
- 1.0540 (-10.78%) Movie Average

Overview of Models

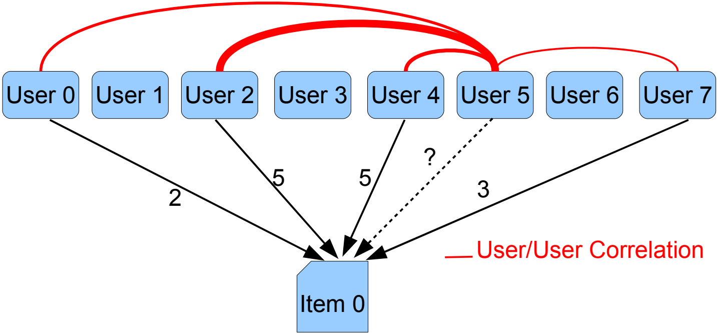
- Similarity based
  - Item based
  - User based
- Latent factor model
- Neural Networks
- Restricted Boltzmann Machines
- Hybrid approaches
- Target: Rating prediction of any user/item combination

Similarity between users



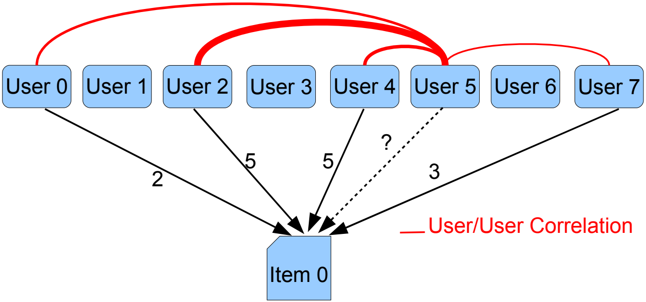
- $U_0 = [1, 4, 3]$
- $U_1 = [1, 5, 2]$

Prediction from user similarity



- Pearson correlation between users
- Rating prediction for user 5 on item 0:
  - select k-best correlating users which voted item 0
  - weight the ratings of the k-best users with their correlation to user 5

Prediction from user similarity



- $N(u,i)$ ... k-best correlating users
- c_{uv} ... Pearson correlation between users u and v

$$r_{ui} = \frac{\sum_{v \in N(u,i)} c_{uv} r_{vi}}{\sum_{v \in N(u,i)} c_{vi}} \qquad (1)$$

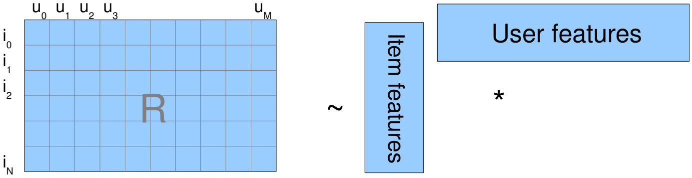
Notes on user similarity models

- In general two users have very few common ratings
- 500.000 users in the Netflix dataset
- Precomputation of all user/user correlations is not possible (takes over 1TByte storage)

Item/Item Correlations

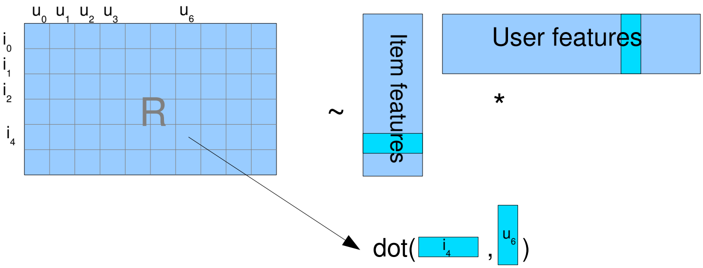
- In general well defined
- 18.000 movies in the Netflix dataset
- Precomputable
- Predictions are more accurate

Rating Matrix factorization



- Low-rank approximation of the rating matrix
- Prediction given by inner product of item and user feature
- Fast and good performance
- Problem: R is very sparse, 1% filled in the Netflix dataset

Rating Matrix factorization: Rating Prediction

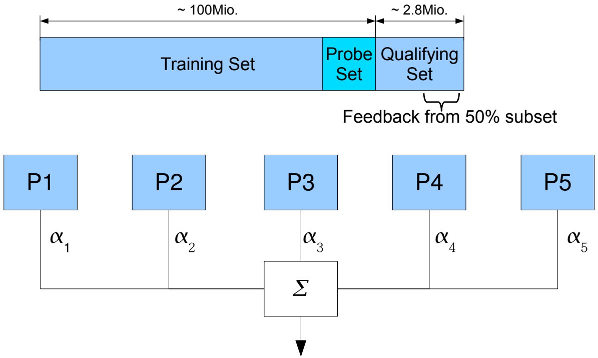


- Rating prediction is given by a dot product from an item and an user feature

Notes on Matrix Factorization

- Fast and good performance
- Training can be done with gradient descent
- Regularization is important
- A single matrix factorization can achieve an improvement of more than 5%
- Very accurate model for users with many votes
- A restriction to non negative features can also be used

Combination of Predictions



- Combination with linear blending
- Simple and efficient
- α_n can be calculated with pseudo-inverse

Overview

- Recommender systems can be used in
  - Webshops
  - Search engines
  - Social networks
  - Personalized advertisement
  - Everywhere a back-channel exists

The secret behind our success

- Knowledge in machine learning
- Independent exploration
- Different viewpoints
- Communication is "all"
  - Many skype-hours of talking
- Good intuition
  - How can we squeeze out max. information out of the dataset?
- Good programming skills to generate fast code
- Computing power
  - 8x DualCore >3GHz, 8GB RAM

Netflixprize Leaderboard 23.06.2008

Netflix Prize

Home | Rules | Leaderboard | Register | Update | Submit | Download

Leaderboard

Display top 40 leaders.

Rank	Team Name	Best Score	% Improvement	Last Submit Time
--	No Grand Prize candidates yet	--	--	--
Grand Prize - RMSE <= 0.8563				
--	No Progress Prize candidates yet	--	--	--
Progress Prize - RMSE <= 0.8625				
1	BellKor	0.8643	9.15	2008-05-30 13:25:58
2	BigChaos	0.8667	8.90	2008-06-23 17:34:44
3	When Gravity and Dinosaurs Unite	0.8675	8.82	2008-05-09 13:25:30
4	Gravity	0.8687	8.69	2008-06-06 20:45:22
5	PragmaticTheory	0.8697	8.59	2008-06-22 23:07:00
6	Just a guy in a garage	0.8704	8.51	2008-06-23 08:03:09
7	acmehill	0.8709	8.46	2008-06-09 23:18:29
Progress Prize 2007 - RMSE = 0.8712 - Winning Team: KorBell				
8	KorBell	0.8712	8.43	2007-10-01 23:25:23
9	basho	0.8714	8.41	2008-05-21 22:06:00

- Leading team from AT&T Research
- We are currently on 2^{nd} place (team BigChaos)

Thank you for your attention.