HPI Hasso Plattner Institut

IT Systems Engineering | Universität Potsdam

Track 1 – Matrix Factorization

Final Presentation

Collaborative Filtering

Markus Freitag, Jan-Felix Schwarz

7 July 2011

Hasso Plattner Institut

Agenda

1. Recap: Matrix Factorization
2. Latest Improvements
   - Better Biases
   - Post Processing
   - Regularization
   - Parameter Tweaking
3. Statistical Analysis for Combination of Approaches
4. Contribution
5. Future Work
6. Lessons Learned
7. Summary

Agenda

1. **Recap: Matrix Factorization**
2. Latest Improvements
   - Better Biases
   - Post Processing
   - Regularization
   - Parameter Tweaking
3. Statistical Analysis for Combination of Approaches
4. Contribution
5. Future Work
6. Lessons Learned
7. Summary

Recap: Matrix Factorization

$$
\begin{pmatrix}
10 & 50 & & & 0 & 0 & & 25 \\
 & & 0 & & 100 & & & 0 \\
100 & & 80 & & & & & \\
0 & & & 50 & 30 & & & \\
 & & & & & 20 & & 0 \\
20 & 100 & & 0 & & 50 & 60 & \\
 & & 0 & & 90 & & & \\
10 & 50 & & 20 & & & & \\
 & & & & 90 & 100 & & 20 \\
 & & & 20 & & & 100 & 0 \\
0 & & 0 & & 40 & & & \\
 & & 70 & 10 & & & 10 &
\end{pmatrix}
$$

Recap: Matrix Factorization

item features · **user features**

$$\begin{pmatrix} 5.1 & 10.0 & 21.3 \\ 4.7 & 9.2 & 1.9 \\ 0.0 & 21.9 & 14.7 \\ 7.9 & 8.5 & 40.2 \\ 10.1 & 0.2 & 2.9 \\ 9.1 & 8.1 & 8.7 \\ 16.6 & 20.1 & 4.1 \\ 7.8 & 1.0 & 0.1 \end{pmatrix}^T \bullet \begin{pmatrix} 1.9 & 20.1 & 9.4 \\ 23.1 & 0.1 & 4.2 \\ 10.2 & 4.0 & 1.9 \\ 1.2 & 0.7 & 12.2 \\ 7.3 & 9.3 & 13.7 \\ 6.3 & 28.1 & 7.2 \\ 9.0 & 5.3 & 3.2 \\ 5.2 & 11.1 & 12.0 \\ 5.7 & 3.9 & 2.7 \\ 0.3 & 0.0 & 0.1 \\ 6.7 & 21.2 & 0.0 \\ 6.4 & 7.9 & 3.2 \end{pmatrix} = \begin{pmatrix} 10 & 50 & 80 & 90 & 0 & 0 & 100 & 25 \\ 70 & 55 & 0 & 90 & 100 & 15 & 10 & 0 \\ 100 & 76 & 80 & 90 & 10 & 30 & 20 & 0 \\ 0 & 90 & 10 & 50 & 30 & 90 & 100 & 10 \\ 0 & 10 & 100 & 70 & 40 & 20 & 10 & 0 \\ 20 & 100 & 100 & 0 & 10 & 50 & 60 & 90 \\ 80 & 20 & 0 & 80 & 90 & 76 & 0 & 10 \\ 10 & 50 & 90 & 20 & 10 & 90 & 100 & 10 \\ 0 & 0 & 10 & 50 & 90 & 100 & 40 & 20 \\ 60 & 70 & 50 & 20 & 90 & 10 & 100 & 0 \\ 0 & 10 & 0 & 90 & 40 & 20 & 50 & 30 \\ 40 & 80 & 70 & 10 & 100 & 0 & 10 & 10 \end{pmatrix}$$

Recap: SGD Algorithm

Stochastic Gradient Descent (SGD)

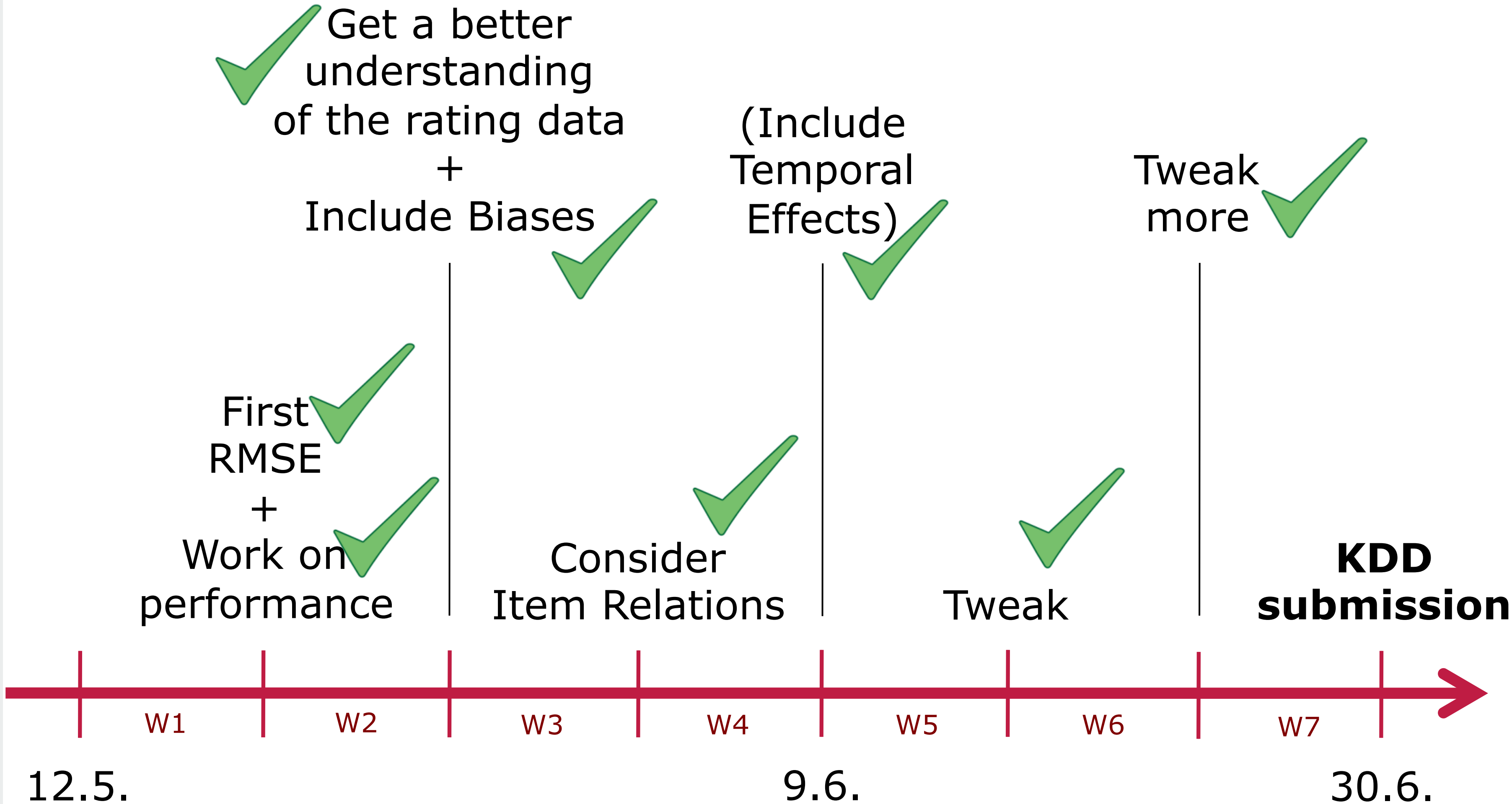
- Approximation procedure for learning one feature

- For each rating in the training set the feature values are modified relative to the prediction error
  - *User value += Learning Rate * Error * Item value*
  - *Item value += Learning Rate * Error * User Value*

- Iterate over the training set until the sum of squared errors (SSE) converges

- Training set split into 4 subsets (track, album, artist, genre)
  - Don't presume a common underlying model

Roadmap & Implementation Status



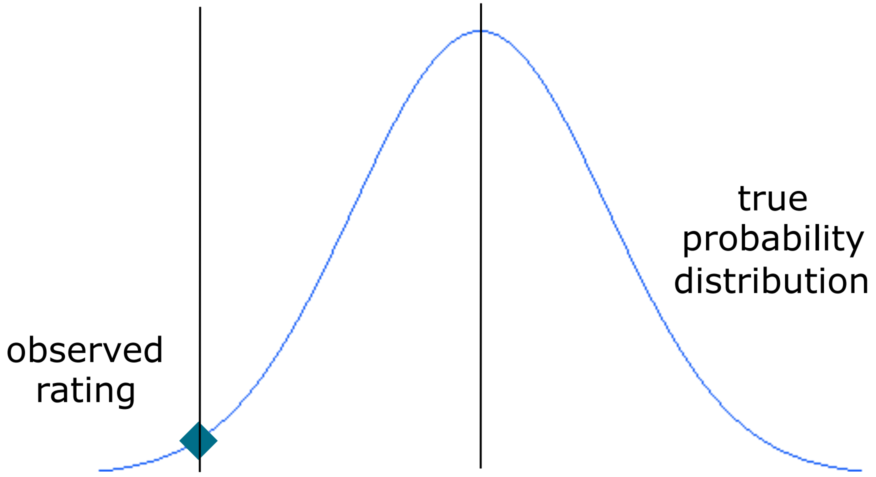
Agenda

1. Recap: Matrix Factorization
2. **Latest Improvements**
   - **Better Biases**
   - Post Processing
   - Regularization
   - Parameter Tweaking
3. Statistical Analysis for Combination of Approaches
4. Contribution
5. Future Work
6. Lessons Learned
7. Summary

Better Biases

- Naive Bias: Item Avg – Global Avg
  - But: What if there is only one observed rating?
    → See it as a draw from the true probability distribution

- Best guess for actual mean: linear blend of the observed mean and the global mean
  - Blending ratio equal to the ratio of variances



Better Biases

- *V_a: Variance of all the items' average ratings*
- *V_b: Variance of individual item ratings*
- *$K = V_a / V_b$*

$$BetterItemAvg = K * GlobalAvg + \frac{sum(ObservedRatings)}{K * count(ObservedRating)}$$

- S. Funk used a constant K
- *"But in fact K=25 seems to work well so I used that instead. :)"*
- We also tried 25 and in fact it works

» *http://sifter.org/~simon/journal/20061211.html*

Better Biases

RMSE Improvement 

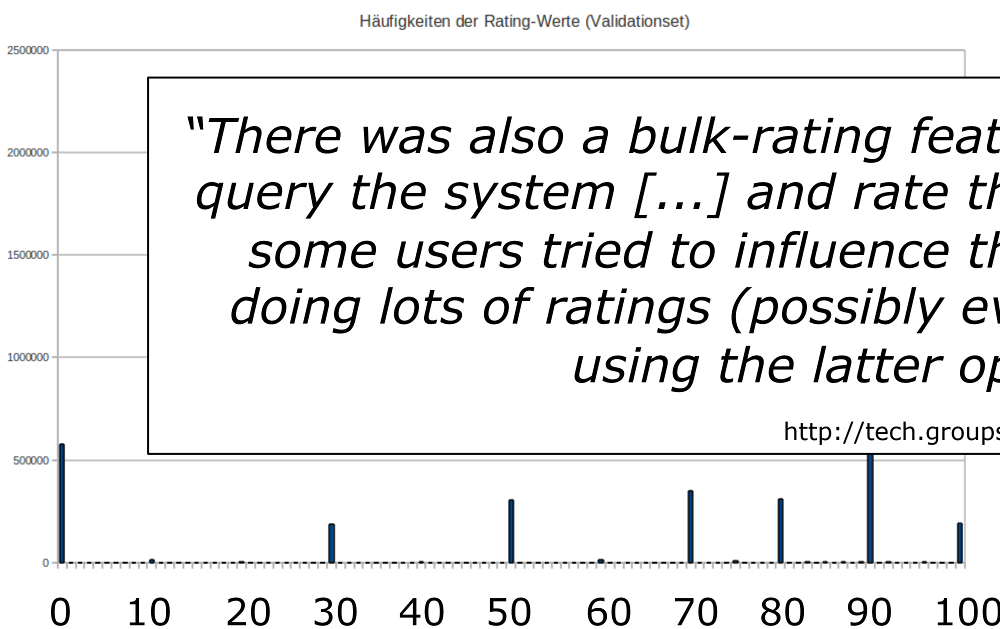
Validation	**Test**
24.2480	27.3456
↓	↓
24.2540	26.8627

Agenda

1. Recap: Matrix Factorization
2. **Latest Improvements**
   - Better Biases
   - **Post Processing**
   - Regularization
   - Parameter Tweaking
3. Statistical Analysis for Combination of Approaches
4. Contribution
5. Future Work
6. Lessons Learned
7. Summary

Post Processing

- Idea: Identify bulk ratings
  - Unify predictions in the bulk rating
- Round to most frequent rating values (decades)



> *"There was also a bulk-rating feature where users could query the system [...] and rate them in one go. Lastly, some users tried to influence the recommender by doing lots of ratings (possibly even semi-automatic) using the latter option."*
>
> http://tech.groups.yahoo.com/group/kddcup2011/message/68

user	item	rating	day	time
1	23521	90	5424	15:27:00
1	25311	90	5424	15:27:00
1	34525	90	5424	15:27:00
1	34947	90	5424	15:27:00
				15:27:00
				15:27:00
				15:27:00
				15:27:00
				15:27:00
				15:27:00
				15:27:00
				15:27:00
				15:27:00
				15:27:00
				15:27:00
				15:27:00
				15:27:00
1	74262	90	5424	15:27:00
1	75445	90	5424	15:27:00
1	75757	90	5424	15:27:00
1	79902	90	5424	15:27:00
1	85222	90	5424	15:27:00
1	95689	90	5424	15:27:00

Post Processing

- There are many sets of simultaneous ratings by one user
  - Only few sets have a constant rating value!
  - But many sets have only one or two “outliers“
- Identify users which tend to bulk rate using metrics and thresholds
  - Average count of distinct rating values in potential bulks
  - Average span between minimum and maximum rating value in potential bulks
- If a “bulk rating user” has simultaneous ratings in the test set assume a bulk rating

Post Processing

RMSE Worsening 

Validation	**Test**
24.9571	25.3100
24.7436	25.3909

→ Maybe tweaking parameters might have yielded a slight improvement

Agenda

1. Recap: Matrix Factorization
2. **Latest Improvements**
   - Better Biases
   - Post Processing
   - **Regularization**
   - Parameter Tweaking
3. Statistical Analysis for Combination of Approaches
4. Contribution
5. Future Work
6. Lessons Learned
7. Summary

Regularization

- Regularization tries to prevent overfitting when learning a single feature
- A certain factor *K* damps the learning effect

*userValue += lrate * (err * movieValue - K * userValue)*

*movieValue += lrate * (err * userValue - K * movieValue)*

Regularization

RMSE Worsening 

Validation	Test
24.2434	26.9824
24.9700	27.3153

→ Regularization probably only makes sense when more features are used

Agenda

1. Recap: Matrix Factorization
2. **Latest Improvements**
   - Better Biases
   - Post Processing
   - Regularization
   - **Parameter Tweaking**
3. Statistical Analysis for Combination of Approaches
4. Contribution
5. Future Work
6. Lessons Learned
7. Summary

Parameter Tweaking

- Tested the effect of every parameter
- Challenge: Parameters affect each other

- Learning Rate set to 0.002
  - Lower: takes too long
  - Higher: risk of shooting over, underfitting
- Feature count set to 14
  - Lower: underfitting
  - Higher: overfitting
- Improvement threshold set to 0,01% (break condition)
  - Higher: single features get less “detailed”
  - Lower: takes too long

Submissions & RMSEs

Submission Results

#	Description	LR	Features	RMSE
1	Test with 50	-	-	37.8262
2	First complete run	0.01	1	28.3295
7	Test with more f.	0.002	10	27.3462
12	Used validation set	0.002	10	26.5217
21	Better bias	0.002	10	26.8627
25	Regularization	0.002	14	27.3153
27	Best values	0.002	14	26.1261

Agenda

1. Recap: Matrix Factorization
2. Latest Improvements
   - Better Biases
   - Post Processing
   - Regularization
   - Parameter Tweaking
3. **Statistical Analysis for Combination of Approaches**
4. Contribution
5. Future Work
6. Lessons Learned
7. Summary

Overlaps of Good and Bad Predictions

	item good	item bad
matrix good	33%	2%
matrix bad	<1%	65%

	item good	item bad
hierarchie good	58%	<1%
hierarchie bad	<1%	74%

	matrix good	matrix bad
hierarchie good	52%	<1%
hierarchie bad	1%	59%

Performance on Different Item Types

"portion of a type in the set"
–
"portion of that type in the whole validation set"

	Matrix		Item-based		Hierarchy	
	good	bad	good	bad	good	bad
album	-3%	+3%	-4%	0%	-8%	-1%
artist	**+12%**	**-11%**	**+16%**	**-8%**	**34%**	**-5%**
genre	0%	-5%	0%	-5%	-5%	-6%
track	**-9%**	**14%**	**-12%**	**+13%**	**-21%**	**+12%**

Agenda

Contribution

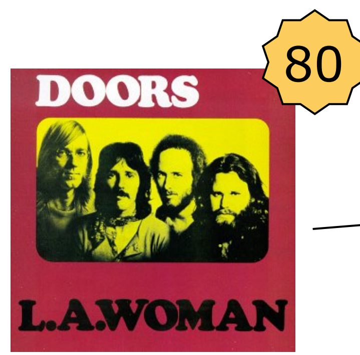
- Implemented a matrix factorization model for a collaborative filtering use case
- Made the implementation work on a very large data set
- Various extensions of the model
  - Biases
  - Hierarchy
  - Temporal Effects (Post Processing)
- Analysis to get a better understanding of the data set
- Combination with other approaches (Item-based, Hierarchy)
- Close to the ladder: Best RMSE 25.3100 (~0.5 to #100)

Agenda

Future Work: Item Relations

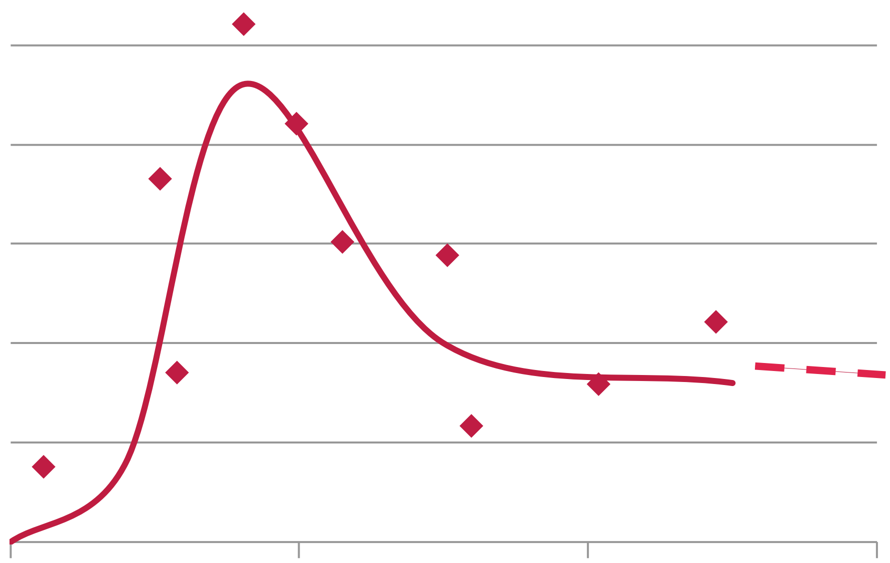
- We have 4 different prediction models
- Combined these models to improve predictions

- Blend prediction with associated predictions
  - of the same user
  - for related items



Future Work: Temporal Effects

- Express biases and vectors as functions over time
- Model will be able to reflect trends in item popularity and user preferences
- Regression problem to learn functions



Agenda

1. Recap: Matrix Factorization
2. Latest Improvements
   - Better Biases
   - Post Processing
   - Regularization
   - Parameter Tweaking
3. Statistical Analysis for Combination of Approaches
4. Contribution
5. Future Work
6. **Lessons Learned**
7. Summary

Lessons Learned

- Matrix factorization is a working model for collaborative filtering
- Squeezing out the last points of improvements for one model gets harder and harder
- Combination of different models is effective
- Implement various approaches and combine them instead of optimizing one algorithm to extremes!
- Be pragmatic!

Hands-on experience

- Collaborative Filtering, Machine Learning, and statistics
- Efficient use of memory, performance optimization, profiling

Agenda

1. Recap: Matrix Factorization
2. Latest Improvements
   - Better Biases
   - Post Processing
   - Regularization
   - Parameter Tweaking
3. Statistical Analysis for Combination of Approaches
4. Contribution
5. Future Work
6. Lessons Learned
7. **Summary**

Summary

- Implemented various extensions for the matrix factorization approach
  - Not everything improved the RMSE
- Optimized our algorithm by tweaking parameters, submissions for testing effects on the test set
- Statistic analysis to compare our different approaches (Matrix, Item-based, Hierarchy)
- Time was the limiting factor. There are still numerous extension possibilities.

Questions?