An Analysis of Probabilistic Methods for Top-N Recommendation in Collaborative Filtering

European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases
(ECML PKDD) 2011

Nicola Barbieri Giuseppe Manco

Institute for High Performance Computing and Networks (ICAR)
Italian National Research Council

Outline

- Recommendation Problem: An overview
- Evaluating Recommendations
  - Precision vs Recommendation accuracy
- Probabilistic approaches to Recommendations
  - Probabilistic Item Ranking
- Experimental Evaluation
- Conclusion

Recommendation Problem

Items

Users	i_1	i_2	i_3	i_4	i_5	i_6	i_7	i_8	i_9	i_{10}
u_1	1		1	4	4		5		3	2
u_2	2	1			5	4	4	3	1	
u_3		1	1	5	4		4	2		3
u_4	2	2		2	1	3		3	5	
u_5			2		1		3		5	5
u_6	1	2			3	4	5	2	3	3
u_7	5		5	3	3	2			1	
u_8		5	5	3		3	2	1	1	
u_9		1	2	5	4		4		2	
u_{10}	1	1		5	5	3	5	3	2	3
u_{11}	2			2		3	3		5	5
u_{12}		2	2	2	1		3	3		5
u_{13}		4				3	3	1		1
u_{14}	5	4	5	3	3	3	3	1	1	
u_{15}			5	3	3	3			1	1

Rating Matrix

- RSs provide users with a list of products that will meet their interests
- As the volume of the catalog increases, Collaborative Filtering is becoming the most effective approach
  - Users' unobserved preferences are estimated by considering only past preference observations
- The recommendation list can be built by drawing upon the (predicted) highly-ranked items

The Recommendation Protocol:

- Let $\mathcal{C}_u^j$ a list of D candidate random items unrated by the user u in the past sessions $1, \ldots, j-1$;
- Associate to each item $i \in \mathcal{C}_u^j$ a score $p_i^{u,j}$ which represents the user's interest for i in session j;
- Sort $\mathcal{C}_u^j$ in descending order given the values $p_i^{u,j}$;
- Add the first N items from $\mathcal{C}_u^j$ to $\mathcal{L}_u^j$ and return the latter to the user.

Evaluating Recommendations

- <u>Predictive accuracy</u>: minimize statistical error metrics between observed and predicted preferences, such as the *Root Mean Squared Error (RMSE)*

$$RMSE = \sqrt{\frac{\sum_{(u,i)\,\in\,\mathcal{T}} (r_i^u - \hat{r}_i^u)^2}{|\mathcal{T}|}}$$

  - This approach adopts a *missing value prediction* perspective

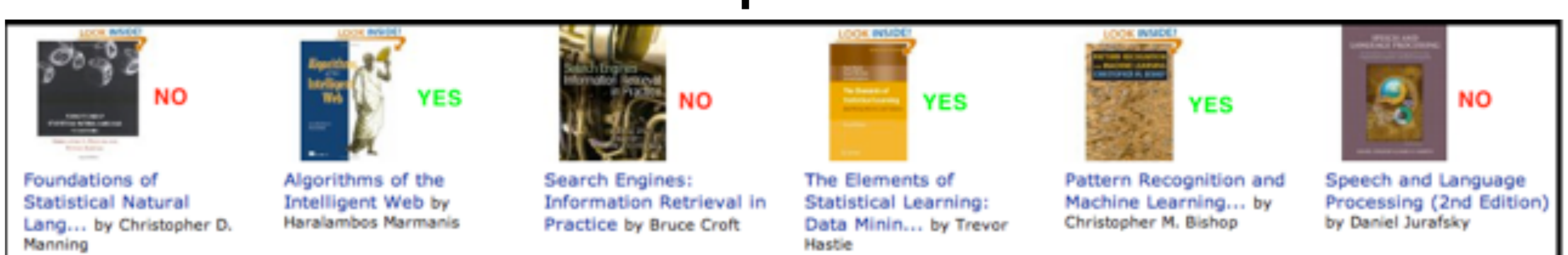
	i_1	i_2	i_3	i_4	i_5	i_6	i_7	i_8	i_9	i_{10}
u_1	1	**?**	1	4	4	**?**	5	**?**	3	2

→

	i_1	i_2	i_3	i_4	i_5	i_6	i_7	i_8	i_9	i_{10}
u_1	1	**2**	1	4	4	**4**	5	**3**	3	2

- <u>Recommendation Accuracy</u>: measure the accuracy (Precision & Recall) of the recommendation list

  - Focusing on the recommendation list, we can measure the quality of recommendations as users perceive them



Precision and Recall of the Recommendation List

- Let L_u denote the recommendation list provided to the user (u) during a generic session
- Let T_u denote the test-set entries for the user (u)
- N is the size of the recommendation list

$$\mathcal{T}_u = \mathcal{T}_u^r \bigcup \mathcal{T}_u^{\neg r}$$

Relevant items (rating higher than the avg)

Unrelevant items

$$Recall(N) = \frac{1}{M}\sum_{u=1}^{M}\frac{|\mathcal{L}_u \cap \mathcal{T}_u^r|}{|\mathcal{T}_u^r|}$$

$$Precision(N) = \frac{1}{M}\sum_{u=1}^{M}\frac{|\mathcal{L}_u \cap \mathcal{T}_u^r|}{N}$$

Evaluating User Satisfaction

- We assume that a recommendation meets user satisfaction if he/she can find in the recommendation list at least an item which meets his/her interests
- Following the methodology proposed by Cremonesi at Al.

- For each user *u* and each item *i* in $\mathcal{T}_u^r$
  - Generate a candidate list $\mathcal{C}_u$
  - Add *i* to $\mathcal{C}_u$ and sort the list according to the scoring function
  - Record the position of item *i* in the sorted list:
    - if *i* belongs to the top-N items, we have a *hit*
    - otherwise, we have a *miss*

$$US_{Recall(N)} = \frac{\#hits}{|\mathcal{T}_u^r|}$$

$$US_{Precision(N)} = \frac{\#hits}{N \cdot |\mathcal{T}_u^r|} = \frac{US_{Recall(N)}}{N}$$

Prediction vs Recommendation Accuracy

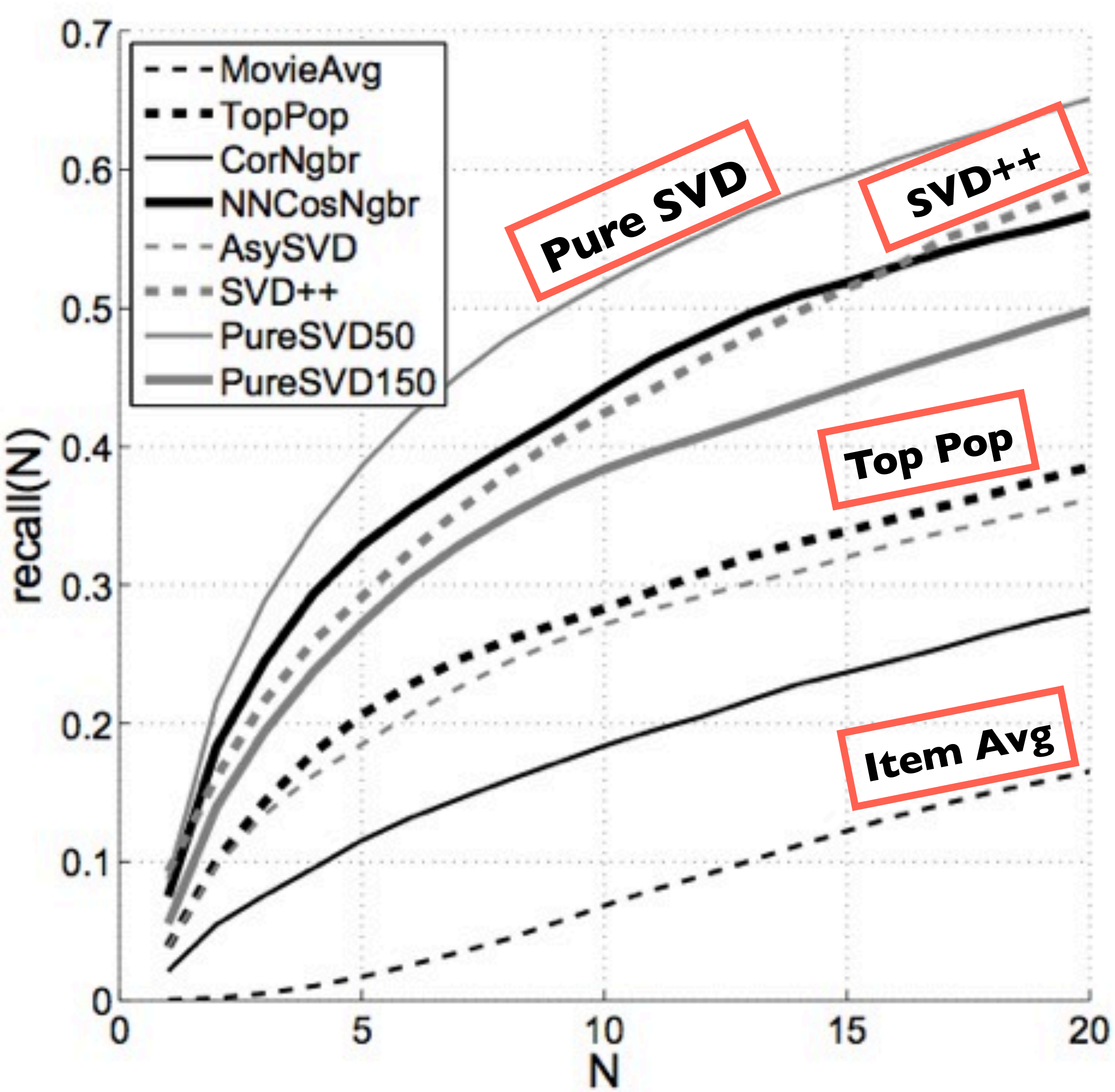
- The recommendation problem has been traditionally interpreted as a *missing value prediction* problem (matrix completion)
- Standard approach: minimize statistical error metrics (MSE, RMSE)
    - The common belief is that small improvements in prediction accuracy would reflect an increase of the accuracy of the recommendation lists
- However, recent works have shown that there is no monotonic relation between error metrics and accuracy metrics:

lower RMSE does not imply higher recommendation accuracy

Performance of Recommender Algorithms on Top-N Recommendation Tasks

[Cremonesi et Al. 2010]

- Pure-SVD, despite its poor performance in prediction accuracy, consistently outperforms the best latent factor models, such as *SVD++*



- Test on Movielens1M data (1.4% of training data as test set)

Probabilistic Approaches to Recommendation: Advantages

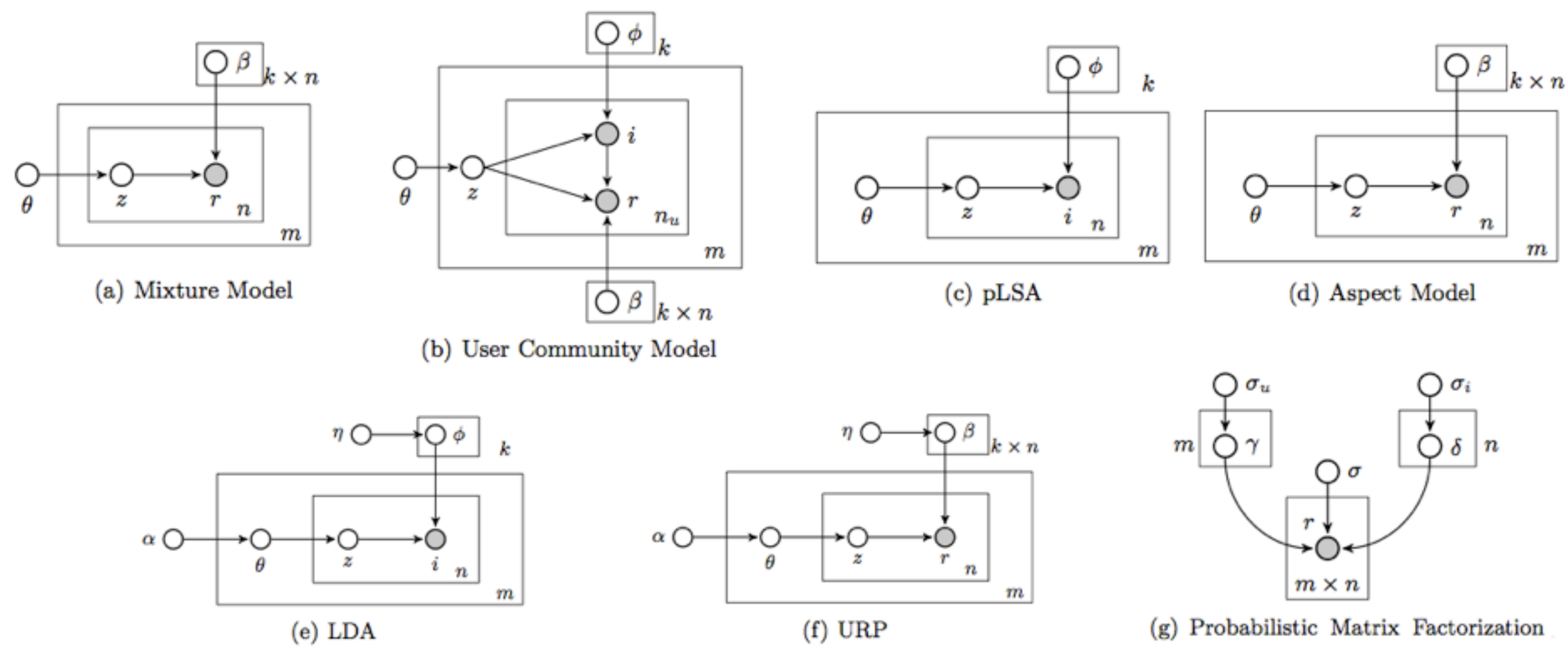
- In this work we aim at evaluating the performance in terms of recommendation accuracy achieved by state-of-art probabilistic models
- Main Advantages:
    - Representation via graphical model
    - They do not focus on a particular error metric: parameters are determined by maximizing the likelihood of the data (which is a more general approach)
    - They model a distribution over rating values which can be used to determine the confidence of the model in predicting each preference value
    - Possibility to plug prior knowledge into the generative process
    - No regularization terms to avoid overfitting
    - They provide an unified framework for combining collaborative and content features

Probabilistic Approaches to Recommendation

- Each triple $\langle u, i, r \rangle$ is considered as the output of a random observation drawn for the joint distribution of the random variables U, I and R
- Two main modeling perspectives:
  - <u>Forced Prediction</u>: focus on the estimate of $P(r|u, i)$
  - <u>Free Prediction</u>: the item selection process is included in the model, which is typically based on the estimate of $P(r, i|u)$
    - We are interested in predicting both the item selection and the preference of the user for each selected item
    - If we assume that the selection is independent from the rating, $P(r, i|u)$ can be factorized as $P(r|i, u)P(i|u)$

Probabilistic Modeling of Preference Data: An Overview

- <u>Latent factors modeling</u>: the state of the hidden variable associated to each preference observation ⟨u,i⟩ models the underlying reason why u has chosen/rated i



(a) Mixture Model (b) User Community Model (c) pLSA (d) Aspect Model (e) LDA (f) URP (g) Probabilistic Matrix Factorization

Exploiting Probabilities for Item Ranking

- The underlying probabilistic framework provides high flexibility in the choice of the item ranking function
- *Predicted Preference*

$$p_i^u = E[R|u, i]$$

- *Item Selection*

$$p_i^u = P(i|u) = \sum_z P(z|u)P(i|z)$$

- *Item Selection and Relevance*

$$\begin{aligned} p_i^u &= P(i, r > \bar{r}_{\mathbf{T}}|u) \\ &= P(i|u)P(r > \bar{r}_{\mathbf{T}}|u, i) = \sum_z P(z|u)P(i|z)P(r > \bar{r}_{\mathbf{T}}|i, z) \end{aligned}$$

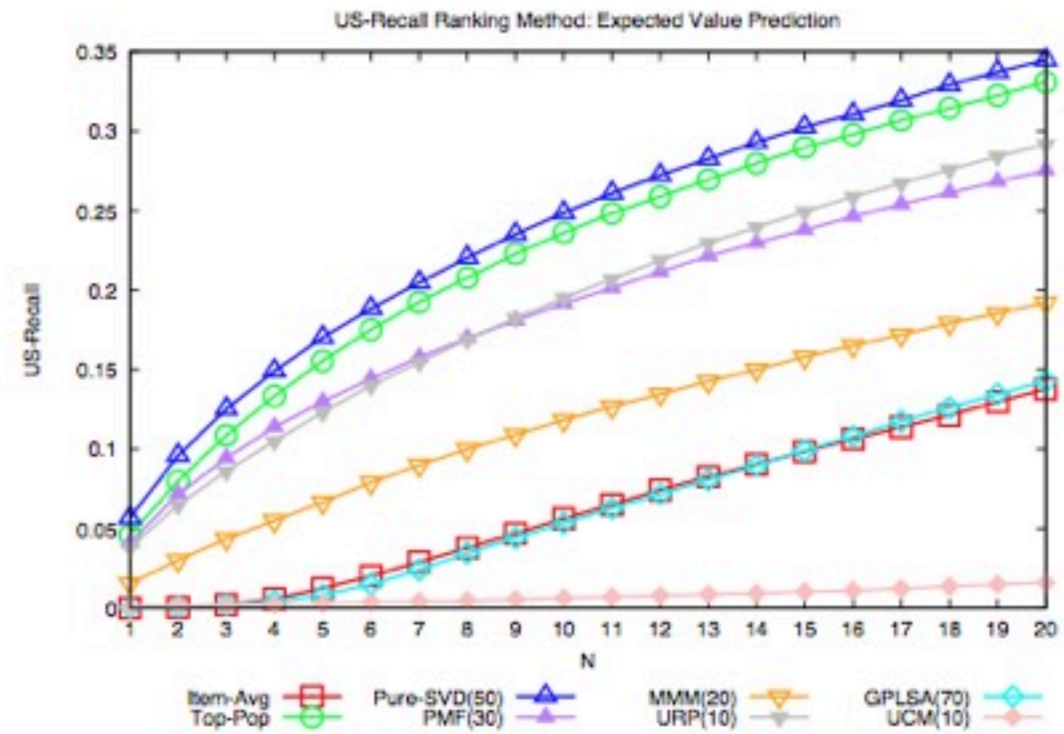
Experimental Evaluation

- We study the effects of the ranking function on the accuracy of the recommendation list, by employing a MonteCarlo 5-folds validation on Movielens1M data

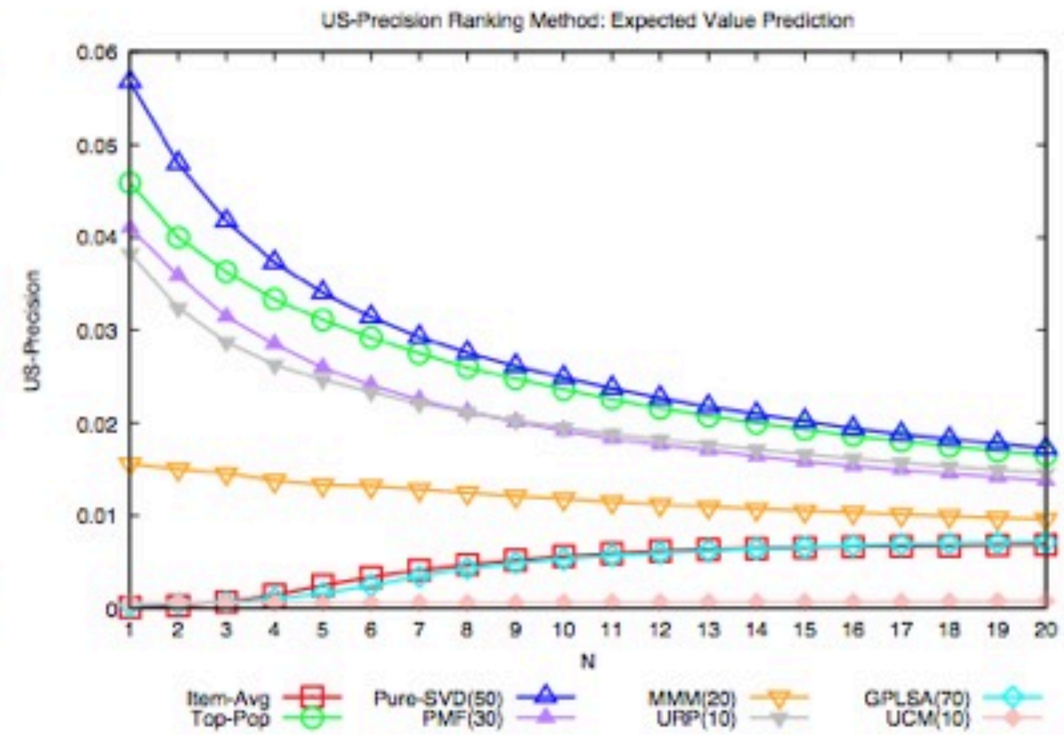
	Training Set	Test Set
Users	6,040	6,032
Items	3,706	3,444
Ratings	800,729	199,480
Avg ratings (user)	132	33
Avg ratings (item)	216	57
Sparseness Coeff	96%	99%

- We consider *Top-Pop* and *Item-Avg* algorithms as baseline, *and Pure-SVD* as a main competitor
- The following results are obtained by varying the length of the recommendation list in the range [1,20] and the dimension of the random candidate list is fixed to 1000

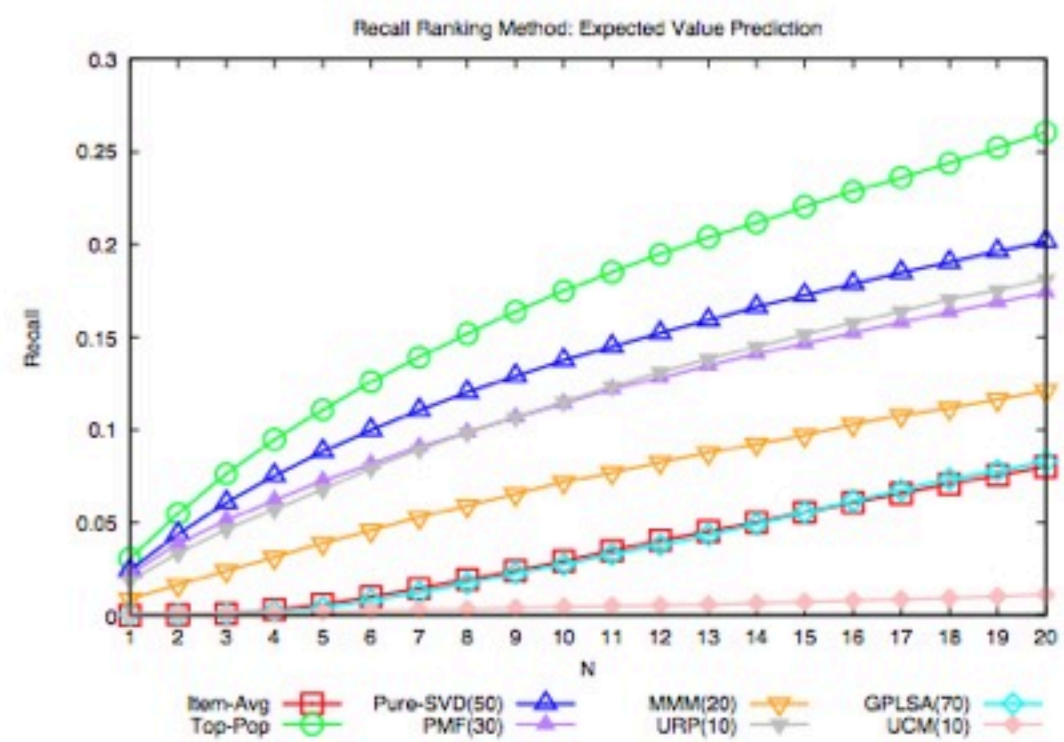
Predicted Preference



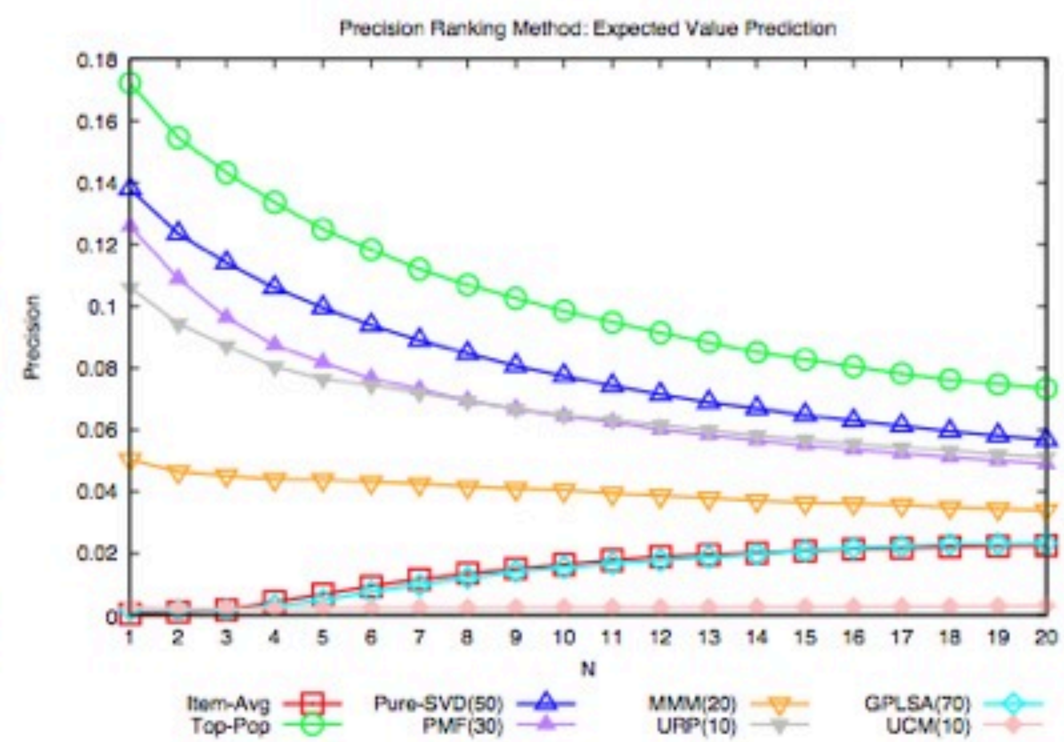
(a) US-Recall



(b) US-Precision



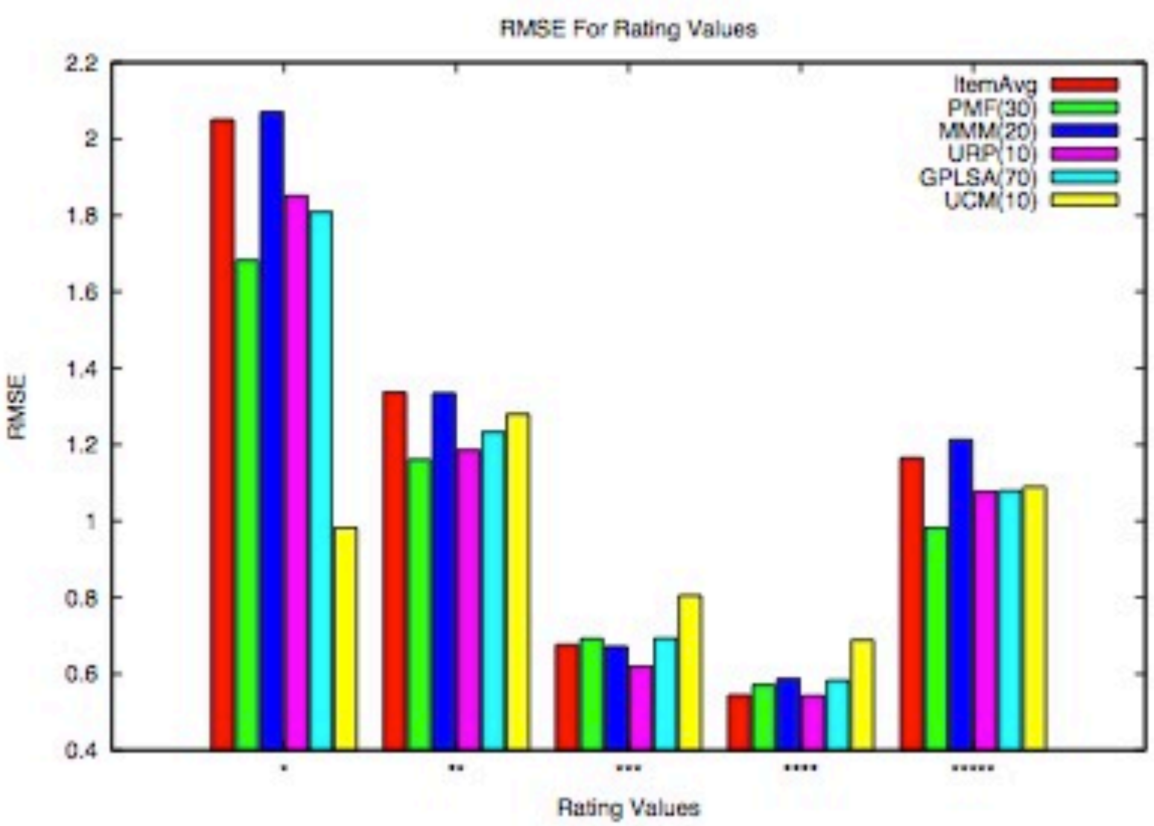
(c) Recall

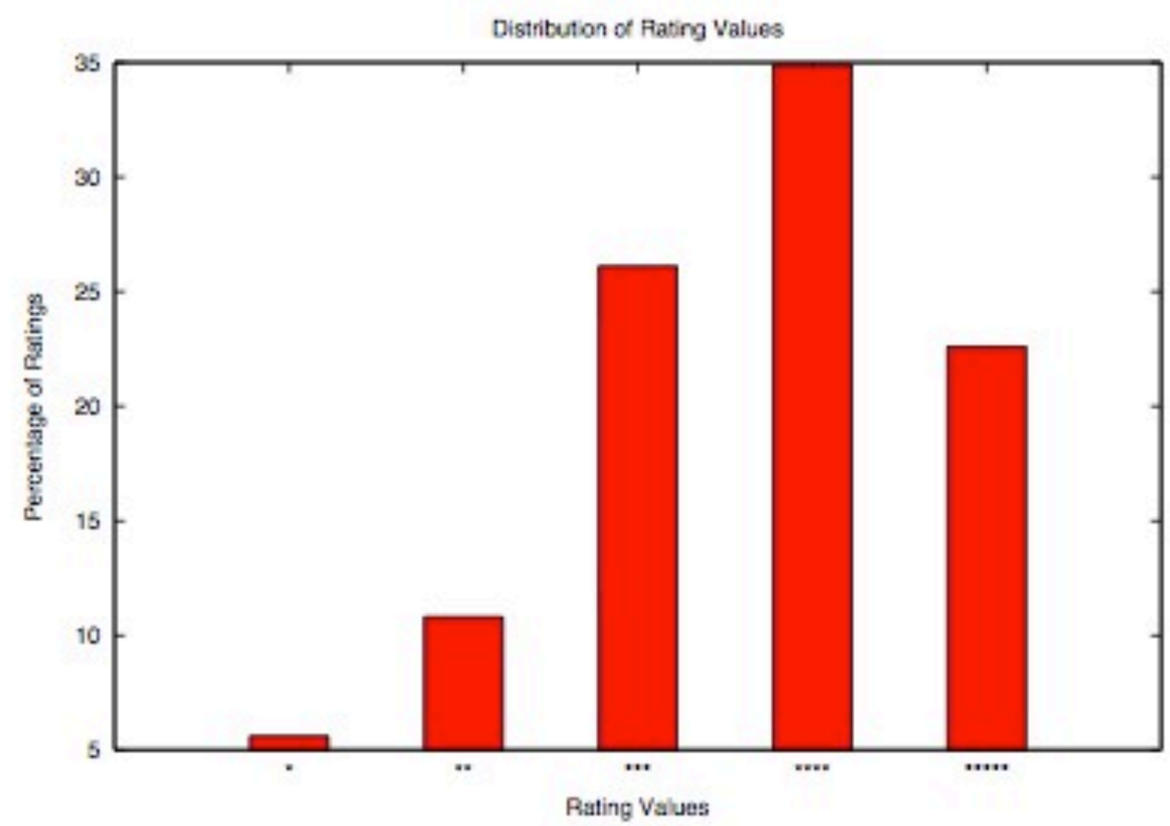


(d) Precision

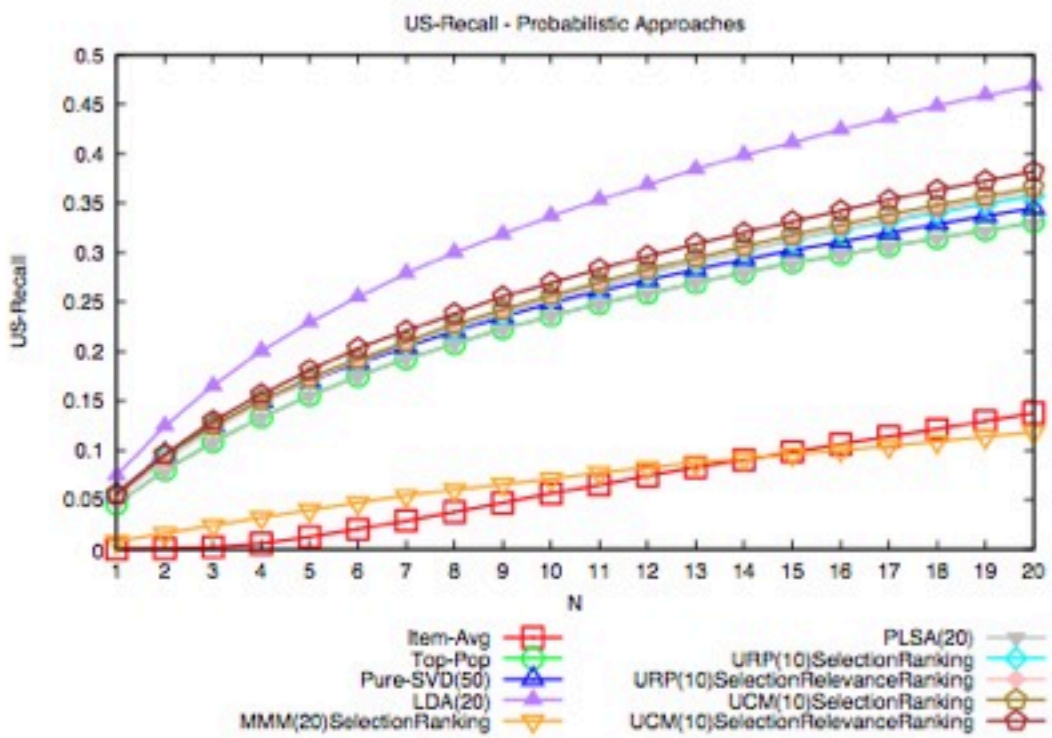
Predicted Preference: Discussion

- Prediction rating flats on the average rating, which causes errors on the extremes
- One and five stars are more interesting from a recommendation perspective

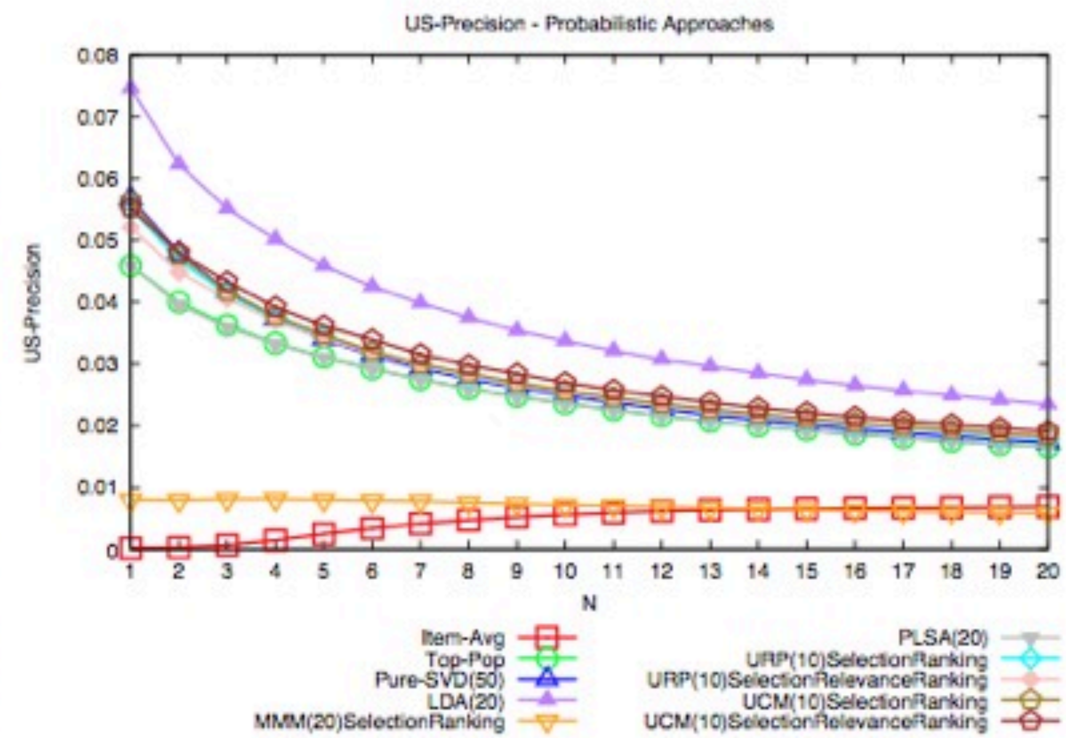




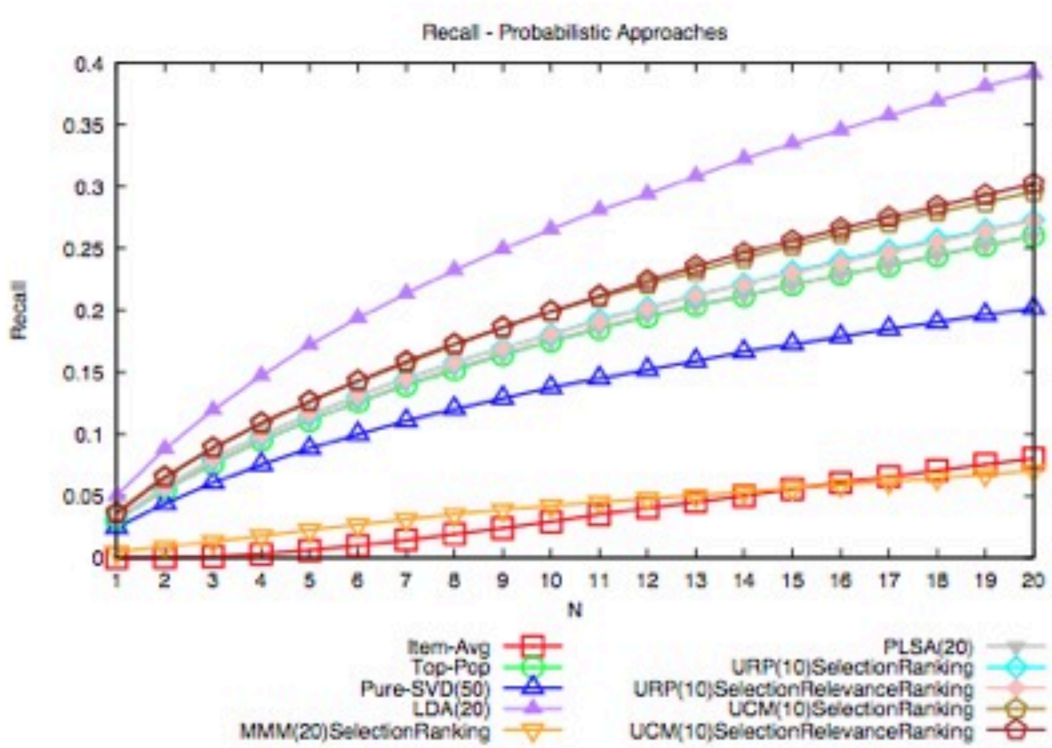
Item Selection and Relevance



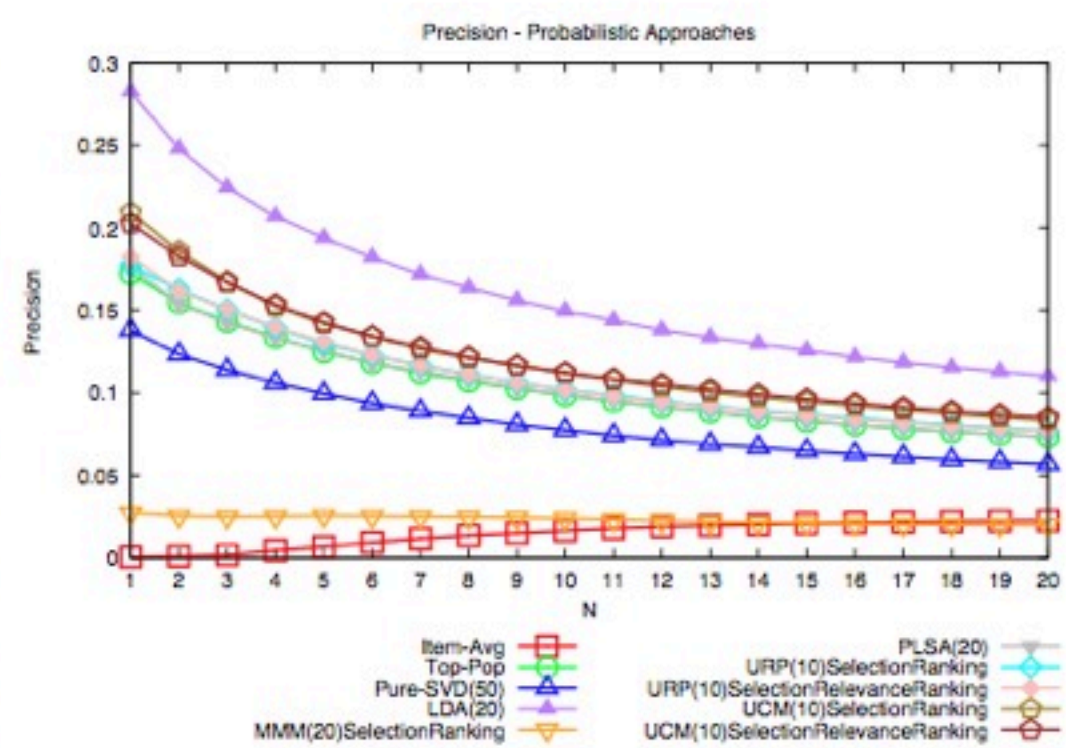
(a) US-Recall



(b) US-Precision



(c) Recall



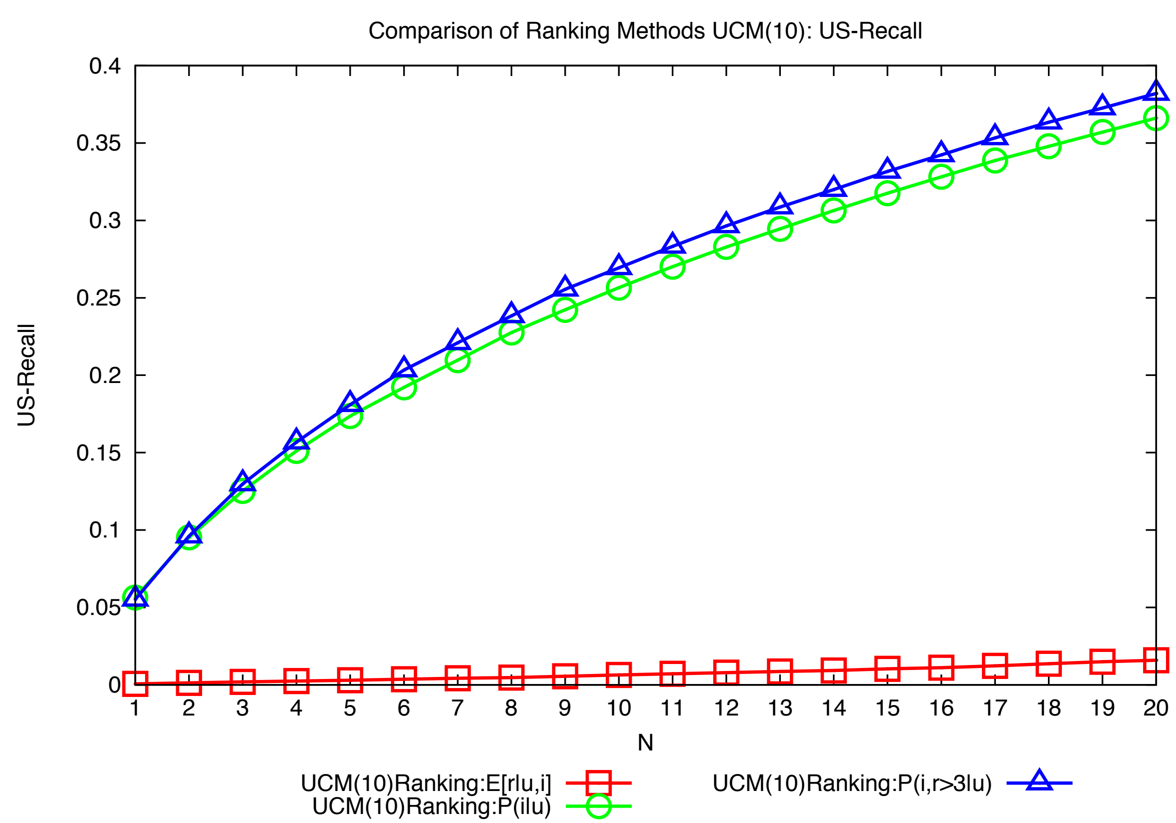
(d) Precision

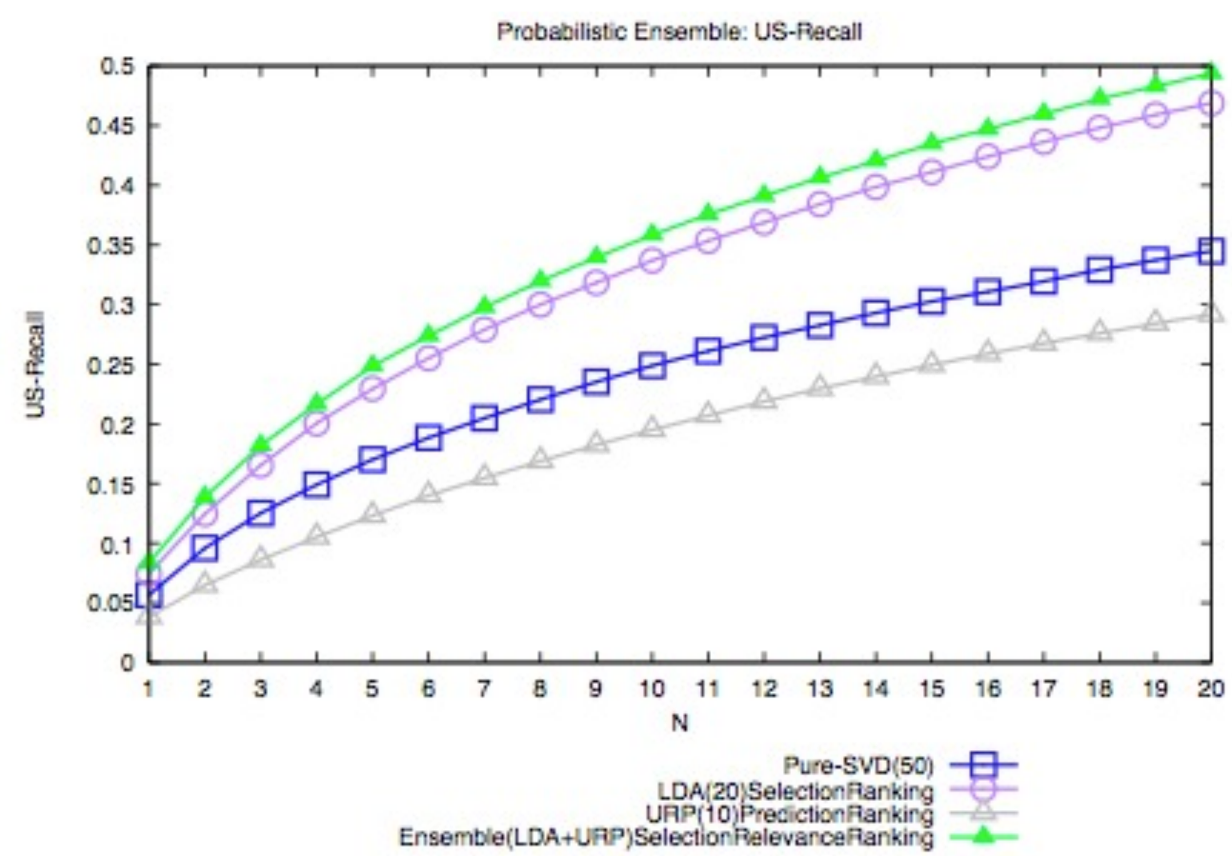
Conclusion

- Probabilistic models, equipped with the proper ranking function, exhibit competitive advantages over state-of-the-art RS in terms of recommendation accuracy

- Whereas the predicted preference item ranking provides poor accuracy results, strategies based on item selection guarantee significant improvements

- Item selection component plays the most important role in recommendation ranking. Better results can be achieved by considering also a rating prediction component





Thanks!!

Question?

For further discussion
nicolabarbieri1@gmail.com