Predicting Performance in Recommender Systems

Alejandro Bellogín

Supervised by **Pablo Castells** and **Iván Cantador**

Information Retrieval Group, Universidad Autónoma de Madrid, Spain

alejandro.bellogin@uam.es

Motivation

Is it possible to predict the accuracy of a recommendation?

E.g., we can decide *whether to deliver* a recommendation or not, depending on such prediction. Or, even, to *combine* different recommenders *according to the expected performance* of each one.

Hypothesis

Data that are commonly available to a Recommender System could contain **signals** that enable an *a priori* estimation of the **success** of the recommendation

Research questions

1. Is it possible to define a performance prediction **theory** for recommender systems in a sound, formal way?
2. Is it possible to **adapt** query performance techniques (from IR) to the recommendation task?
3. What kind of **evaluation** should be performed? Is IR evaluation still valid in our problem?
4. What kind of recommendation **problems** can these models be applied to?

Research question 1

Is it possible to define a performance prediction theory for recommender systems in a sound, formal way?

a) Define a predictor of performance

$\gamma = \gamma(u, i, r, \ldots)$

b) Agree on a performance metric

$\mu = \mu(u, i, r, \ldots)$

c) Check predictive power by measuring correlation

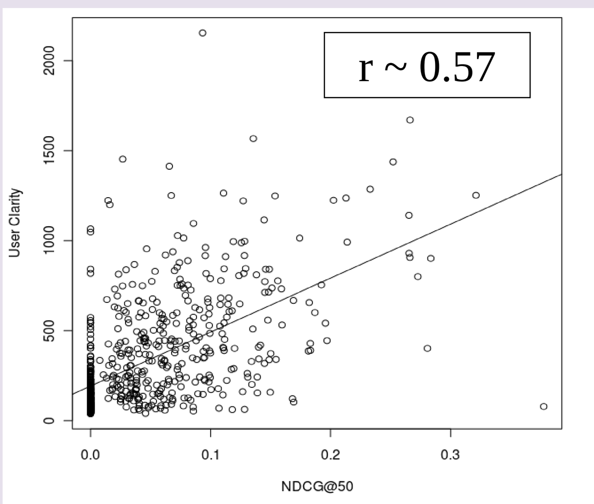
$\mathrm{corr}([\gamma(x_1), \ldots, \gamma(x_n)], [\mu(x_1), \ldots, \mu(x_n)])$

d) Evaluate final performance: dynamic vs static

Correlation coefficients

Pearson: linear correlation

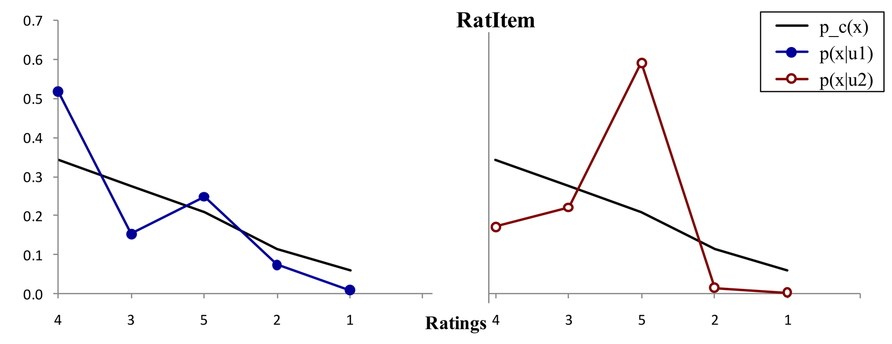
Spearman: rank correlation



Research question 2

Is it possible to adapt query performance techniques (from IR) to the recommendation task?

- In Information Retrieval: *"Estimation of the system's performance in response to a specific query"*
- Several predictors proposed
- We focus on query clarity → **user clarity**



User clarity

It captures the uncertainty in user's data

$$\mathrm{clarity}(u) = \sum_{x \in X} p(x|u) \log\left(\frac{p(x|u)}{p_c(x)}\right)$$

user's model

system's model

Distance between the user's and the system's probability model

We propose 3 formulations (for space *X*):

- **Based on ratings**
- **Based on items**
- **Based on ratings and items**

Research question 3

What kind of evaluation should be performed? Is IR evaluation still valid in our problem?

- In IR: Mean Average Precision + correlation
  - 50 points (queries) vs 1000+ points (users)
- Performance metric is not clear:
  - error-based?
  - precision-based?
- What is performance?
  - It may depend on the final application
- Possible bias
  - E.g., towards users or items with larger profiles

Research question 4

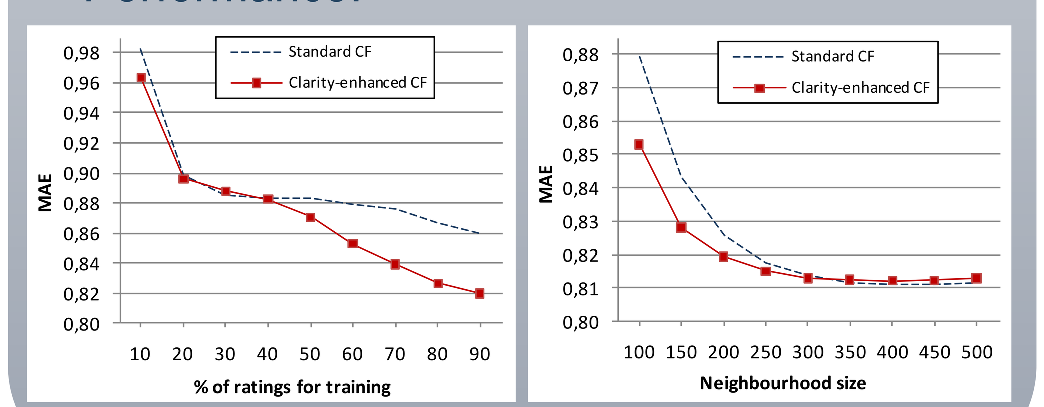
What kind of recommendation problems can these models be applied to?

- Whenever a combination of strategies is available

- Example 1: dynamic neighbor weighting
- The user's neighbors are weighted according to their similarity
- Can we take into account the uncertainty in neighbor's data?
- User neighbor weighting:
  - Static: $g(u,i) = C \sum_{v \in N[u]} \mathrm{sim}(u,v) \times rat(v,i)$
  - Dynamic: $g(u,i) = C \sum_{v \in N[u]} \gamma(v) \times \mathrm{sim}(u,v) \times rat(v,i)$
- Correlation analysis:

% training	10%	20%	30%	40%	50%	60%	70%	80%	90%
correlation	-0.10	0.10	0.18	0.18	0.18	0.17	0.17	0.15	0.15

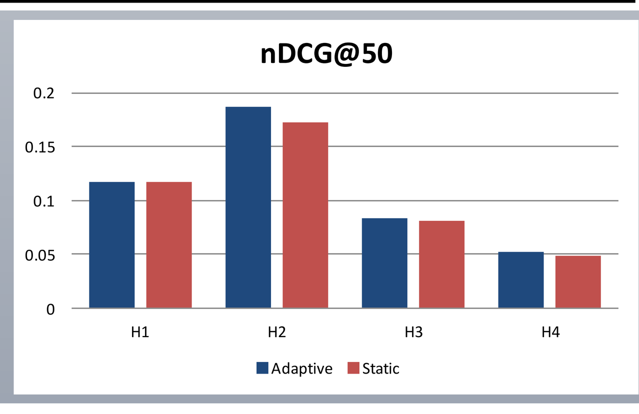
- Performance:



- Example 2: dynamic ensemble recommendation
- Weight is the same for every item and user (learnt from training)
- What about boosting those users predicted to perform better for some recommender?
- Hybrid recommendation:
  - Static: $g(u,i) = \lambda \times g_{R1}(u,i) + (1-\lambda) \times g_{R2}(u,i)$
  - Dynamic: $g(u,i) = \gamma(u) \times g_{R1}(u,i) + (1-\gamma(u)) \times g_{R2}(u,i)$
- Correlation analysis:

Predictor	CBF	IB	TF-L1	TF-L2	UB	**Median**	**Mean**
ItemSimple	0.257	0.146	0.521	0.564	0.491	0.491	0.396
ItemUser	0.252	0.188	0.534	0.531	0.483	0.483	0.398
RatUser	0.234	0.182	0.507	0.516	0.469	0.469	0.382
RatItem	0.191	0.184	0.442	0.426	0.395	0.395	0.328
IRUser	0.171	-0.092	0.253	0.399	0.257	0.253	0.198
IRItem	0.218	0.152	0.453	0.416	0.372	0.372	0.322
IRUserItem	0.265	0.105	0.523	0.545	0.444	0.444	0.376

- Performance:



Future Work

Explore other input sources

- Implicit data (with time)
- Item predictors
  - Different recommender behavior depending on item attributes
  - They would allow to capture popularity, diversity, etc.
- Social links
  - Use graph-based measures as indicators of user strength
  - First results are positive

We need a theoretical background

Why do some predictors work better?

Predictor	CBF	IB	TF-L1	TF-L2	UB
ItemSimple	0.257	0.146	0.521	0.564	0.491
ItemUser	0.252	0.188	0.534	0.531	0.483
RatUser	0.234	0.182	0.507	0.516	0.469
RatItem	0.191	0.184	0.442	0.426	0.395
IRUser	0.171	-0.092	0.253	0.399	0.257
IRItem	0.218	0.152	0.453	0.416	0.372
IRUserItem	0.265	0.105	0.523	0.545	0.444

Larger datasets

Publications

- A Performance Prediction Aproach to Enhance Collaborative Filtering Performance. A. Bellogín and P. Castells. In ECIR 2010.
- Predicting the Performance of Recommender Systems: An Information Theoretic Approach. A. Bellogín, P. Castells, and I. Cantador. In ICTIR 2011.
- Self-adjusting Hybrid Recommenders based on Social Network Analysis. A. Bellogín, P. Castells, and I. Cantador. In SIGIR 2011.

5th ACM Conference on Recommender Systems (RecSys2011) – Doctoral Symposium

Chicago, USA, 23-27 October 2011

Acknowledgments to the **National Science Foundation** for the funding to attend the conference