A Performance Prediction Approach to Enhance Collaborative Filtering Performance

Alejandro Bellogín and Pablo Castells

{**alejandro.bellogin**, pablo.castells}@uam.es

Universidad Autónoma de Madrid
Escuela Politécnica Superior

European Conference on Information Retrieval 2010
March 28-31, Milton Keynes, United Kingdom

Introduction: Recommender Systems

$$i_u^* = \arg\max_{i \in I} \text{utility}(u, i)$$

(Adomavicius & Tuzhilin 2005)

Introduction: Recommender Systems

$$i_u^* = \arg\max_{i \in I} \text{utility}(u,i)$$

- Collaborative filtering (Adomavicius & Tuzhilin 2005)

$$\text{utility}(u,i) = k \sum_{v \in N(u)} \text{sim}(u,v) \times r_{v,i}$$

Is similarity enough?

- No, we propose the following modification

$$\mathrm{utility}(u,i) = k \sum_{v \in N(u)} \boxed{\gamma(v)}\,\mathrm{sim}(u,v) \times r_{v,i}$$

- Related work:
  - Experts (Amatriain et al. 2009)
  - Power users (Lathia et al. 2008)
  - Trust (Kwon et al. 2009, O'Donovan & Smyth 2005)
  - Dealing with users with little overlapping
    - Significance weighting: n/50 (Herlocker et al. 2002)
    - Confidence (Clements et al. 2007)

Our approach

- Predict "neighbor performance" $\gamma(\cdot)$
- Adaptation of query performance prediction techniques
  - User / item clarity
- Check predictive power
  - Correlation against "neighbor goodness"
- Enhance CF performance with dynamic weights on neighbors

Performance prediction in IR

- Mostly addressed as query performance (Hauff et al. 2008)
- Query clarity (Cronen-Townsend et al. 2002)
  - Distance (relative entropy) between query and collection language models

$$\text{clarity}(q) = \sum_{w \in V} P(w|q) \log_2 \frac{P(w|q)}{P_{coll}(w)}$$

$$P(w|q) = \sum_{d \in R} P(w|d) P(d|q), \quad P(q|d) = \prod_{w_q \in q} P(w_q|d)$$

$$P(w|d) = \lambda P_{ml}(w|d) + (1-\lambda) P_{coll}(w)$$

- Query clarity captures the (lack of) ambiguity in a query with respect to the collection
  - Queries whose likely relevant documents are a mix of disparate topics receive a lower score than those with a topically-coherent result set.
  - Strong correlation between query clarity and the performance (average precision) of the result set

Predicting good neighbors

- User "clarity", item "clarity"…?
- Many possible ways to map query clarity to elements in CF
- For instance, for user clarity:

$$\boxed{\begin{aligned}\gamma(u) = clarity(u) &= \sum_{v \in U} p(v \mid u) \log_2 \frac{p(v \mid u)}{p_c(v)} \\ p(v \mid u) &= \sum_{i : rat(u,i) \neq \emptyset} p(v \mid i)\, p(i \mid u) \\ p_c(v) &= \frac{1}{|U|}\end{aligned}}$$

Evaluation

- Correlation between predictor and performance metric
  - How do we define the "performance" of a neighbor?
- Final performance improvements when dynamic weights are introduced
  - Metric: RMSE

- Dataset:
  - MovieLens (100K)

- Two variables:
  - Neighborhood size
  - Sparsity (number of available ratings)

- Baseline:
  - Standard user-based kNN CF with Pearson similarity

Assessing predictive power

- A neighbor performance metric is needed

- Proposed approximation to "neighbor goodness"
  How does a user affect the total MAE of the system

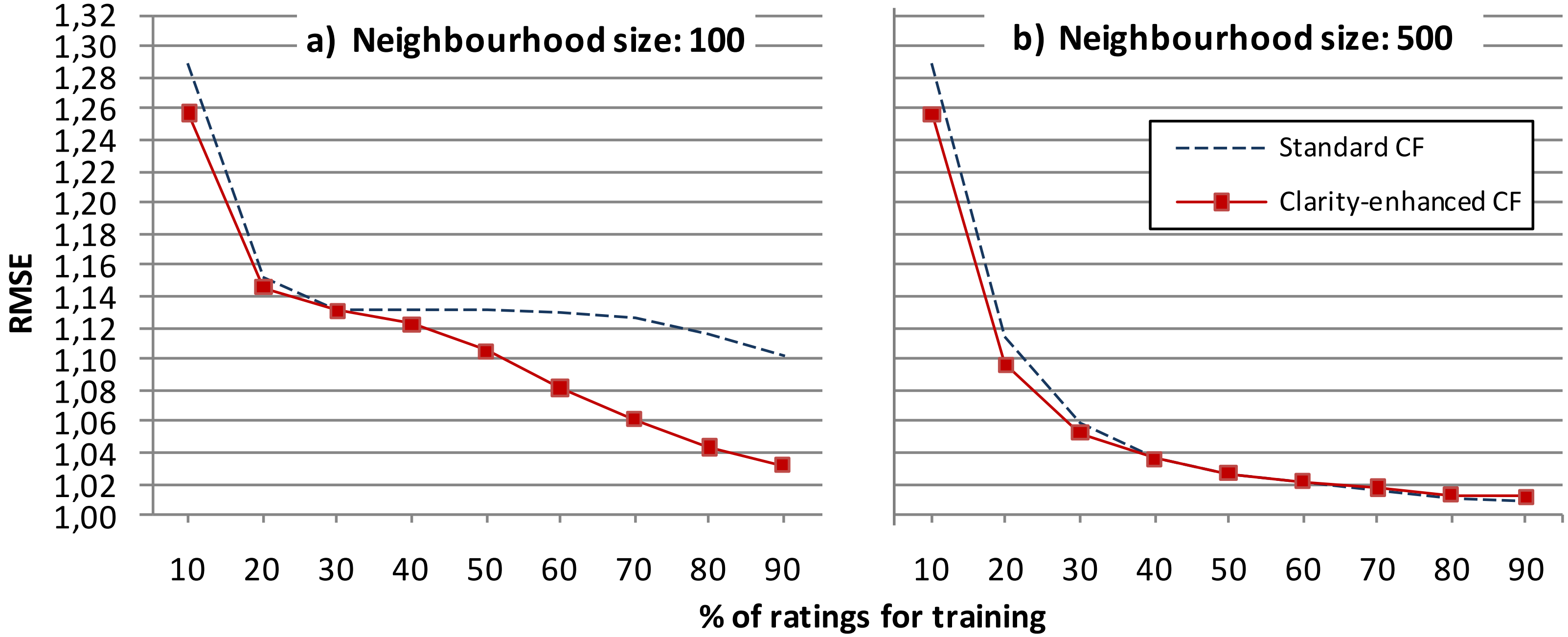
NG(u) ~ "total MAE reduction by u" ~ "MAE without u" – "MAE with u"

$$= \frac{1}{|R_{U-\{u\}}|} \sum_{v \in U-\{u\}} \mathrm{CE}_{U-\{u\}}(v) - \frac{1}{|R_{U-\{u\}}|} \sum_{v \in U-\{u\}} \mathrm{CE}_U(v)$$

$$\mathrm{CE}_X(v) = \sum_{i:rat(v,i) \neq \emptyset} |\tilde{r}_X(v,i) - r(v,i)|$$

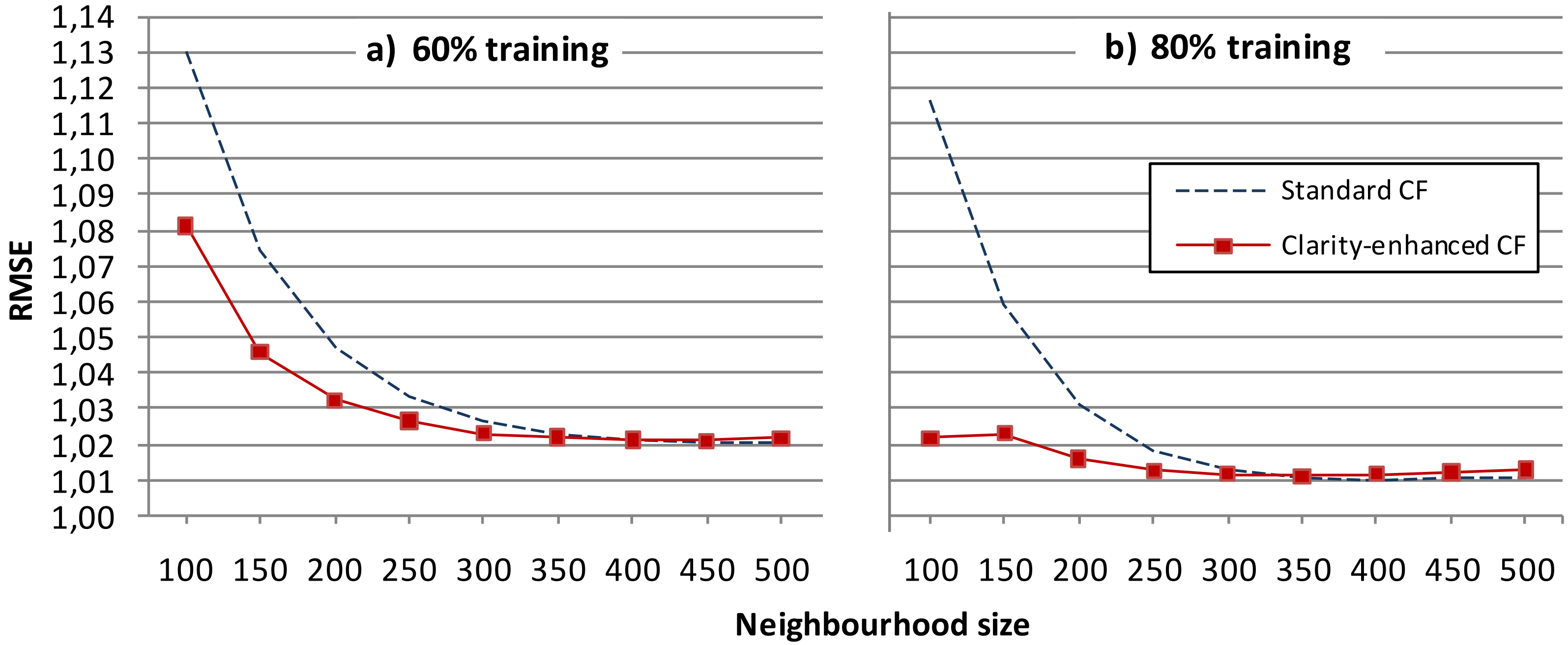
- Observed results
  - Pearson correlation of **0.18** (50% sparsity, p-value < 0.05)

Dynamic neighbor weights in CF



Performance comparison for different rating density

Dynamic neighbor weights in CF



Performance comparison for different neighbourhood sizes

Conclusions

- Performance prediction for neighbor selection in CF
- Positive though moderate correlations values
  - Revise NG: is it an adequate metric?
  - Improve predictor
- Performance improvements using dynamic weights for neighbors
  - Higher difference for small neighborhoods

Future work

- Alternative variants of clarity based predictor
  - Even $\gamma(u, v, i, \ldots)$
- Analysis of user performance metric
- Further comparison with other predictors: variance, social-based, time-based
- Predicting performance can be useful in many recommendation and personalization scenarios
  - Hybrid recommender systems, personalized IR, rank fusion

Thank you

Bibliography

- Adomavicius, G., Tuzhilin, A.: Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. IEEE Transactions on Knowledge and Data Engineering 17(6), 734--749 (2005)
- Amatriain, X., Lathia, N., Pujol, J. M., Kwak, H., and Oliver, N: The wisdom of the few: a collaborative filtering approach based on expert opinions from the web. In SIGIR 2009, pp. 532-539 (2009)
- Clements, M., de Vries, A. P., Pouwelse, J. A., Wang, J., and Reinders, M. J. T. "Evaluation of neighbourhood selection methods in decentralized recommendation systems," in Workshop on Large Scale Distributed Systems for Information Retrieval (LSDS-IR) (2007)
- Cronen-Townsend, S., Zhou, Y. & Croft, B. W. (2002), Predicting query performance, *in* 'SIGIR '02: Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval', ACM Press, New York, NY, USA, pp. 299–306.
- Hauff, C., Hiemstra, D., de Jong, F.: A survey of pre-retrieval query performance predictors. In: 17th ACM conference on Information and knowledge management (CIKM 2008), pp. 1419—1420. ACM Press, New York (2008)
- Herlocker, J., Konstan, J. A., and Riedl, J. An empirical analysis of design choices in neighborhood-based collaborative filtering algorithms. Inf. Retr., 5(4), 287-310 (2002).
- Kwon, K., Cho, J., and Park, Y. Multidimensional credibility model for neighbor selection in collaborative recommendation. Expert Syst. Appl., 36(3):7114-7122 (2009)
- Lathia, N., Hailes, S., and Capra, L: kNN CF: a temporal social network. In RecSys '08, pp. 227-234 (2008)
- O'Donovan, J. and Smyth, B. Trust in recommender systems. In IUI '05, pp 167-174 (2005)

Predicting good neighbors

- Many possible ways for the mapping

- User clarity:

$$\gamma(v) = clarity(v) = \sum_{w \in U} p(w|v) \log_2 \frac{p(w|v)}{p_c(w)}$$

$$p(w|v) = \sum_{i:rat(v,i) \neq 0} p(w|i)\, p(i|v)$$

$$p(w|i) = \lambda p_{ml}(w|i) + (1-\lambda) p_c(w)$$

$$p(i|v) = \lambda p_{ml}(i|v) + (1-\lambda) p_c(i)$$

$$p_{ml}(w|i) = \frac{r(w,i)}{\sum_{w \in U} r(u,i)}$$

$$p_{ml}(i|v) = \frac{r(v,i)}{\sum_{j \in I} r(v,j)}$$

$$p_c(v) = \frac{1}{|U|},\ p_c(i) = \frac{1}{|I|}$$

Correlation

- Pearson correlation: user clarity vs neighbor goodness

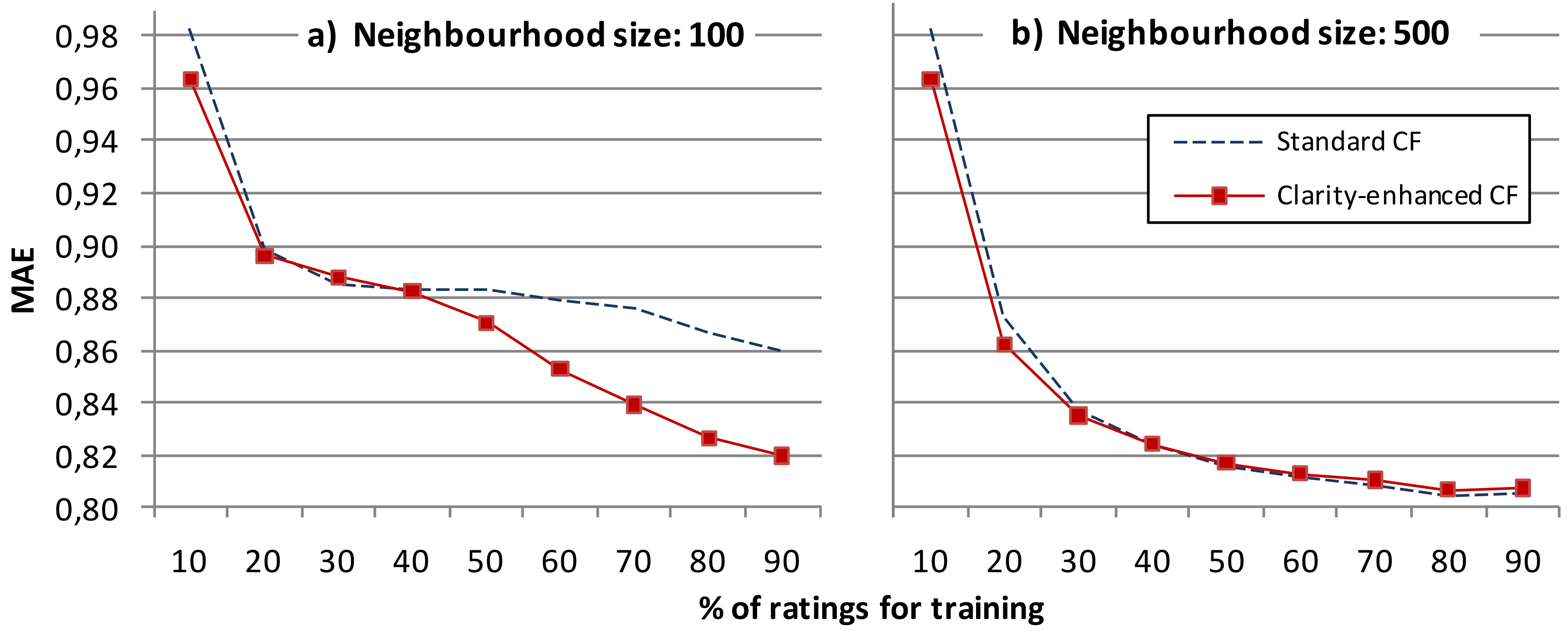
% training	10%	20%	30%	40%	50%	60%	70%	80%	90%
correlation	-0.10	0.10	0.18	0.18	0.18	0.17	0.17	0.15	0.15

✓ Direct correlation

- When calculated with significant data

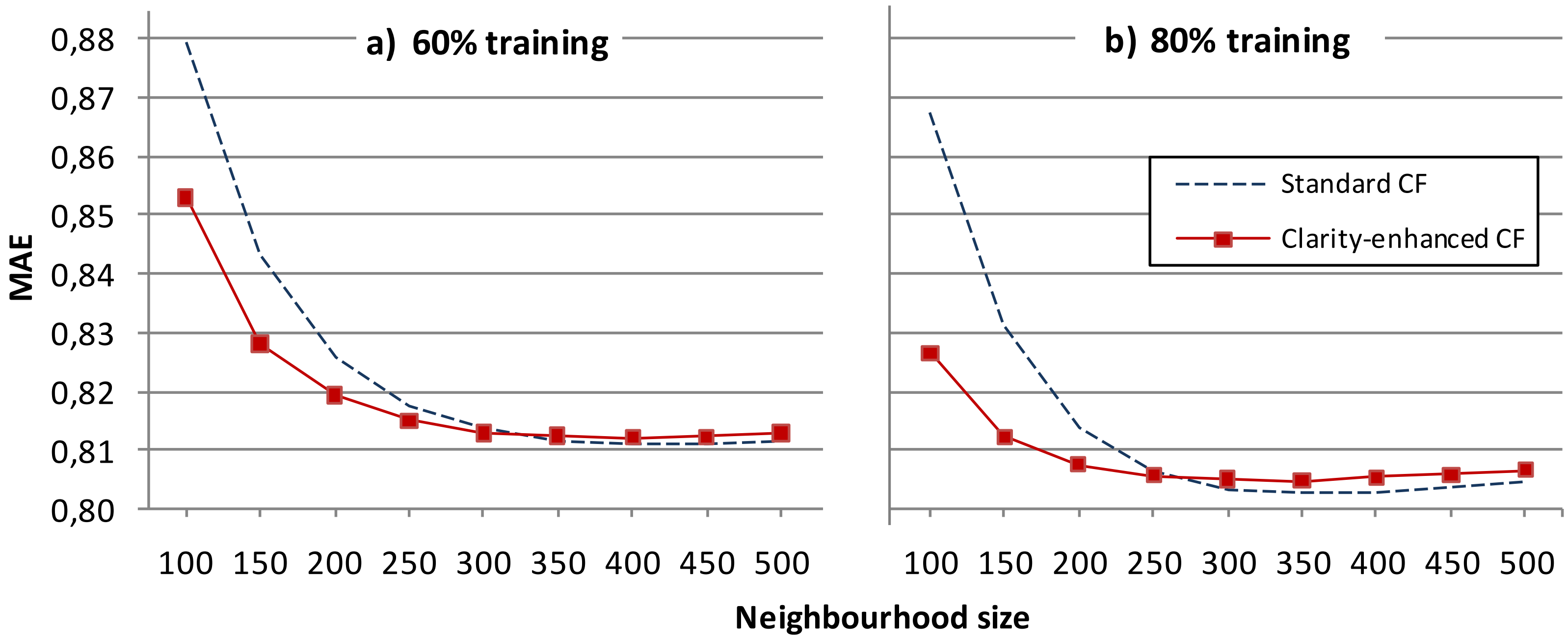
✗ Not strong values

Performance results I



Performance comparison for different rating density

Performance results II



Performance comparison for different neighbourhood sizes