Performance Prediction in Recommender Systems

Alejandro Bellogín
Supervisor: Pablo Castells
Co-supervisor: Iván Cantador
Escuela Politécnica Superior
Universidad Autónoma de Madrid

@abellogin
alejandro.bellogin@uam.es

Recommender Systems

- Content-based filtering (CB), Collaborative Filtering (CF), Hybrid Filtering (HF)
- For example: User-based Collaborative Filtering

	i_1		i_k		i_m
u_1	rat_{11}		rat_{1k}		rat_{1m}
u_j	rat_{j1}		?		r_{jm}
u_n	rat_{n1}		rat_{nk}		rat_{nm}

$$\boxed{g(u_j, i_k) = C \sum_{v \in N[u]} \mathrm{sim}(u_j, v) \times rat(v, i_k)}$$

$\mathrm{sim}(u, v)$: Pearson

$N[u]$: most similar

Motivation

Can we detect **ambiguous** users?

In fact, when is a user considered ambiguous?

Hypothesis

The amount of **uncertainty** (ambiguity) in user data *may* correlate with the **accuracy** of a system's recommendations

Hypothesis

The amount of **uncertainty** (ambiguity) ⬇ in user data *may* correlate with the ⬆ **accuracy** of a system's recommendations

Research Question

How to **dynamically** adapt a recommendation strategy to the user's preference information available at a certain time?

Research Question

How to **dynamically** adapt a recommendation strategy to the user's preference information available at a certain time?

Or, if we <u>predict</u> which are the ambiguous users, can we treat them in a way such the system's <u>performance</u> increases?

Proposal

1. Define a predictor of performance $\boldsymbol{\gamma} = \boldsymbol{\gamma}(u, i, r, \ldots)$

2. Introduce the predictor in an adaptive strategy:
   a) Evaluate its **predictiveness** using correlation with performance measure
   b) Evaluate final **performance**: static vs adaptive strategy

Predictor definition

- Based on performance prediction from Information Retrieval
  - "Estimation of the system's performance in response to a specific query"

- User clarity: captures uncertainty in user data
  - Distance between the user's and the system's probability model

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

user's model

system's model

  - *X* may be: users, items, ratings, or a combination

UNIVERSIDAD AUTONOMA DE MADRID

IR Group @ UAM

Applications

- Neighbour weighting in Collaborative Filtering
  - User's neighbours are weighted according to their similarity
  - Can we take into account the neighbour's confidence/ambiguity?

- Hybrid recommendation
  - Weight is the same for every item and user (learnt from training)
  - What about boosting those users predicted to perform better for some method?

Adaptive Strategies

- User neighbour weighting
  - Static: $$g(u,i) = C \sum_{v \in N[u]} \text{sim}(u,v) \times rat(v,i)$$

Adaptive Strategies

- User neighbour weighting [1]
  - Static: $$g(u,i) = C \sum_{v \in N[u]} \mathrm{sim}(u,v) \times rat(v,i)$$
  - Adaptive: $$g(u,i) = C \sum_{v \in N[u]} \gamma(v) \times \mathrm{sim}(u,v) \times rat(v,i)$$

Adaptive Strategies

- User neighbour weighting [1]
  - Static: $$g(u,i) = C \sum_{v \in N[u]} sim(u,v) \times rat(v,i)$$
  - Adaptive: $$g(u,i) = C \sum_{v \in N[u]} \gamma(v) \times sim(u,v) \times rat(v,i)$$
- Hybrid recommendation
  - Static: $$g(u,i) = \lambda \times g_{R1}(u,i) + (1-\lambda) \times g_{R2}(u,i)$$

Adaptive Strategies

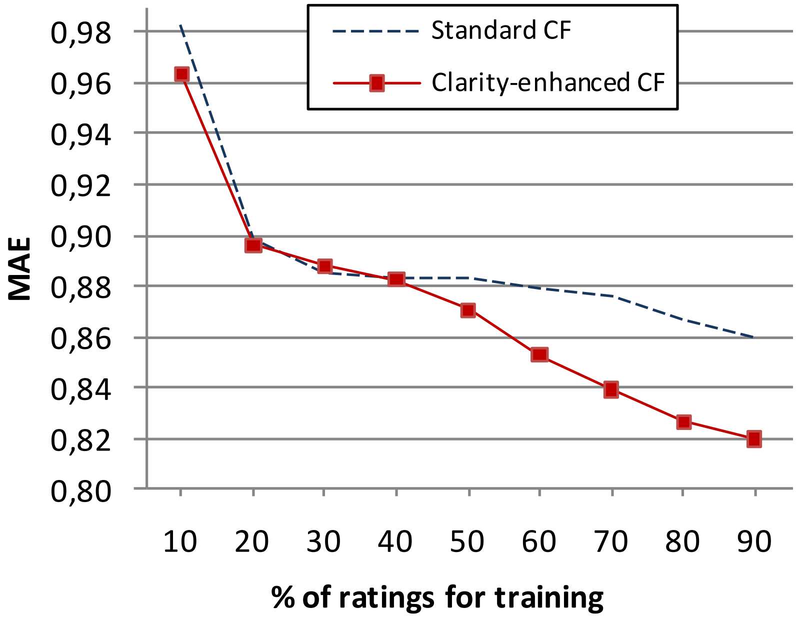
- User neighbour weighting [1]
  - Static: $$g(u,i) = C \sum_{v \in N[u]} \mathrm{sim}(u,v) \times rat(v,i)$$
  - Adaptive: $$g(u,i) = C \sum_{v \in N[u]} \gamma(v) \times \mathrm{sim}(u,v) \times rat(v,i)$$
- Hybrid recommendation [3]
  - Static: $$g(u,i) = \lambda \times g_{R1}(u,i) + (1 - \lambda) \times g_{R2}(u,i)$$
  - Adaptive: $$g(u,i) = \gamma(u) \times g_{R1}(u,i) + (1 - \gamma(u)) \times g_{R2}(u,i)$$

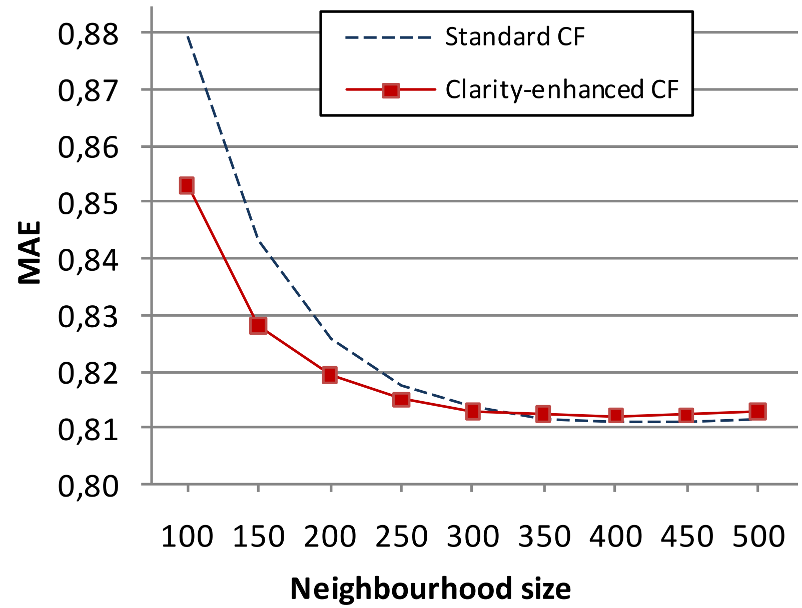
Results – Neighbour weighting

- Correlation analysis [1]
  - With respect to Neighbour Goodness metric: “how good a neighbour is to her vicinity”

% 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 [1] (MAE = Mean Average Error, the lower the better)





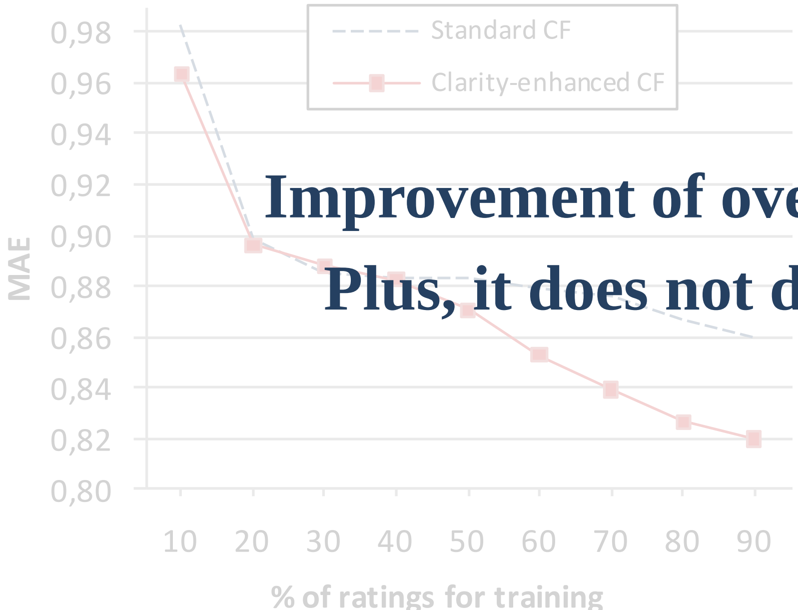
Results – Neighbour weighting

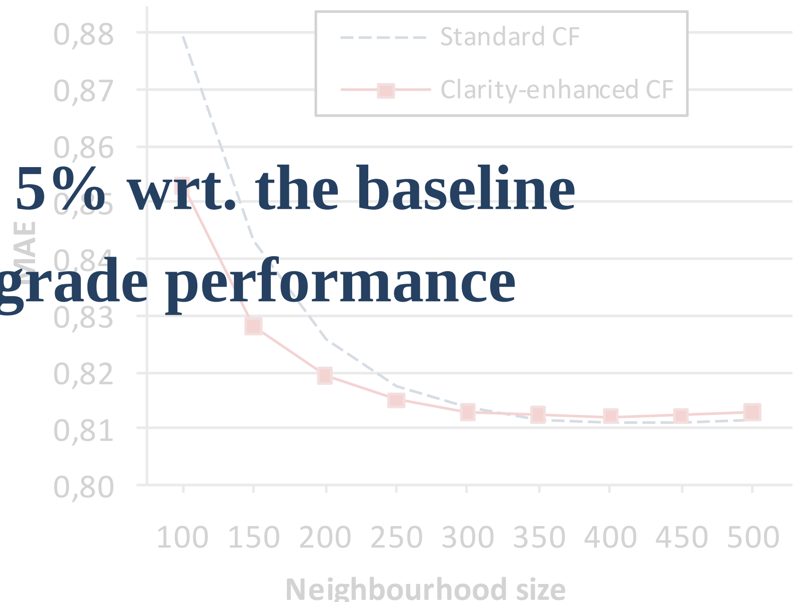
- Correlation analysis [1]
  - With respect to Neighbour Goodness metric: "how good a neighbour is to her vicinity"

Positive, although not very strong correlations

% 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 [1] (MAE = Mean Average Error, the lower the better)





Improvement of over 5% wrt. the baseline

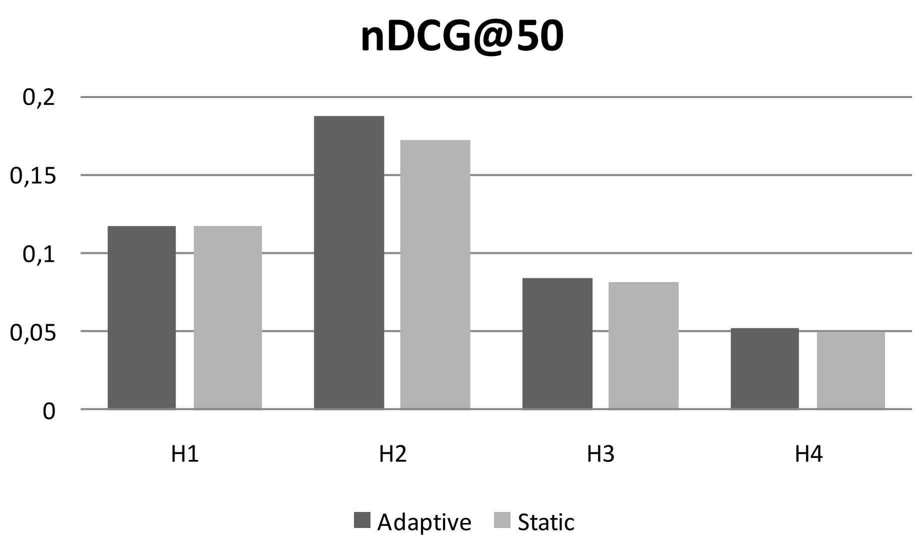
Plus, it does not degrade performance

Results – Hybrid recommendation

- Correlation analysis [2]
  - With respect to nDCG@50 (nDCG, normalized Discount Cumulative Gain)

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 [3]



Results – Hybrid recommendation

- Correlation analysis [2]
  - With respect to nDCG@50 (nDCG, normalized Discount Cumulative Gain)

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
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
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

In average, most of the predictors obtain positive, strong correlations

- Performance [3]

Adaptive strategy outperforms static for different combination of recommenders

Contributions

- Inferring user's performance in a recommender system

- Building adaptive recommendation strategies
  - Dynamic neighbour weighting: according to expected goodness of neighbour
  - Dynamic hybrid recommendation: based on predicted performance

- Encouraging results
  - Adaptive strategies obtain better (or equal) results than static
  - Positive predictive power (good correlations between predictors and metrics)

Related publications

- [1] A Performance Prediction Aproach to Enhance Collaborative Filtering Performance. A. Bellogín and P. Castells. In ECIR 2010.

- [2] Predicting the Performance of Recommender Systems: An Information Theoretic Approach. A. Bellogín, P. Castells, and I. Cantador. In ICTIR 2011.

- [3] Performance Prediction for Dynamic Ensemble Recommender Systems. A. Bellogín, P. Castells, and I. Cantador. In press.

Future Work

- What is performance?

- We need a theoretical background
  - Why do some predictors work better?

- Explore other input sources
  - Implicit data (with time)
  - Social links

- Larger datasets

FW – Performance definition

- What is performance?

RMSE?

Precision?

User satisfaction?

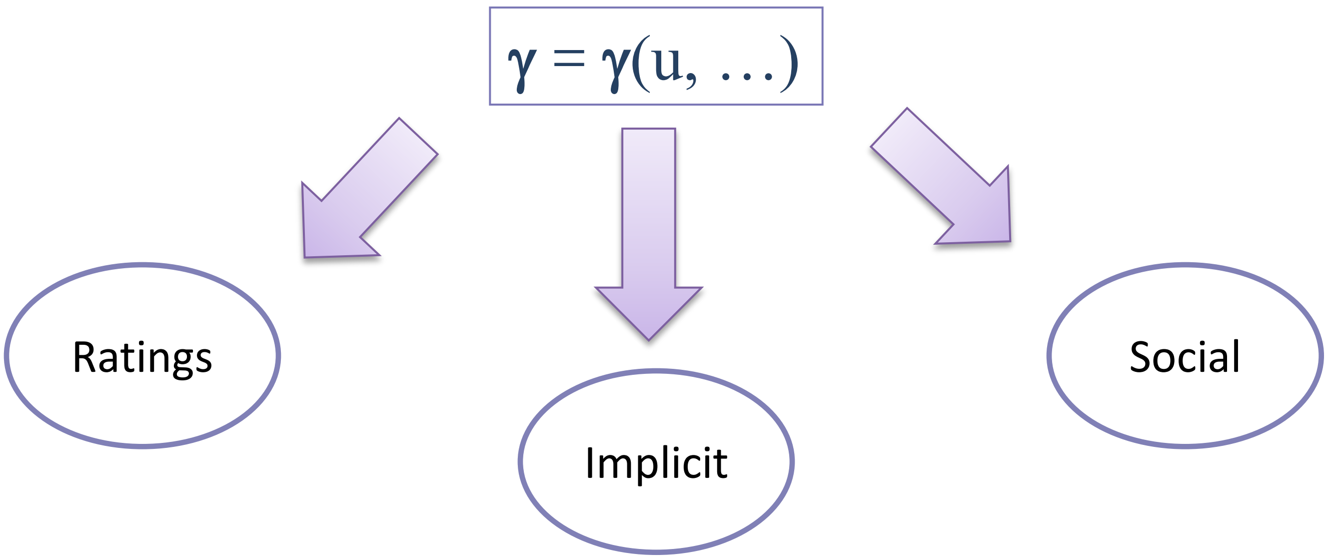
FW – Theoretical background

- We need a theoretical background
  - Why do some predictors work better?

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

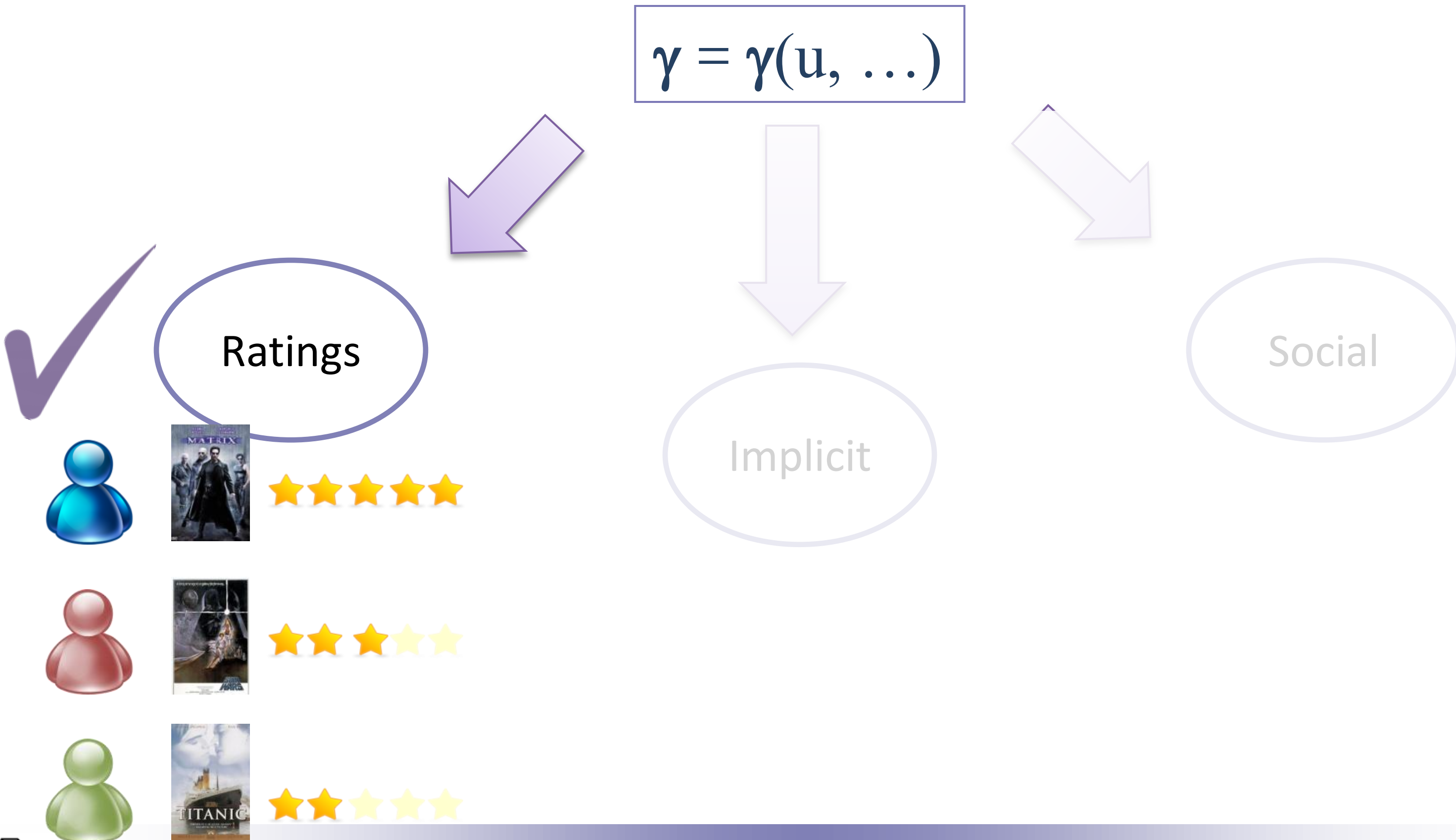
FW – Other input sources

- Explore other input sources
  - Implicit data (with time)
  - Social links



FW – Other input sources

- Explore other input sources
  - Implicit data (with time)
  - Social links

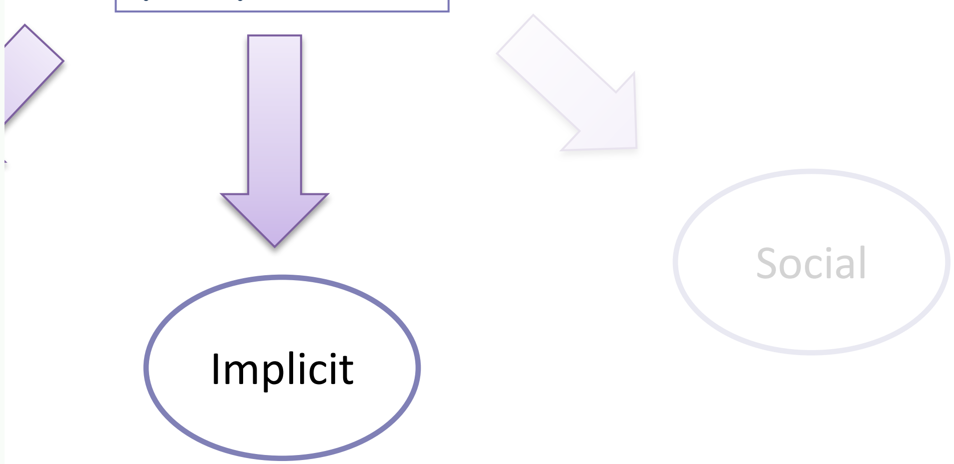


FW – Other input sources

- Explore other input sources
  - Implicit data (with time)
  - Social links

Time	User	Item
1108335607000	387	1330
1108335638000	863	75655
1108335704000	701	3818
1108335730000	283	962
1108335760000	141	7557
1108335784000	518	720
1108335840000	863	75655
1108335944000	701	9905
1108335960000	784	4941
1108336036000	518	720
1108336081000	141	7557
1108336204000	701	1508
1108336209000	958	762
1108336252000	518	720
1108336297000	387	1525
1108336454000	701	3018
1108336458000	518	720
1108336514000	863	75655
1108336655000	863	75655
1108336696000	518	720

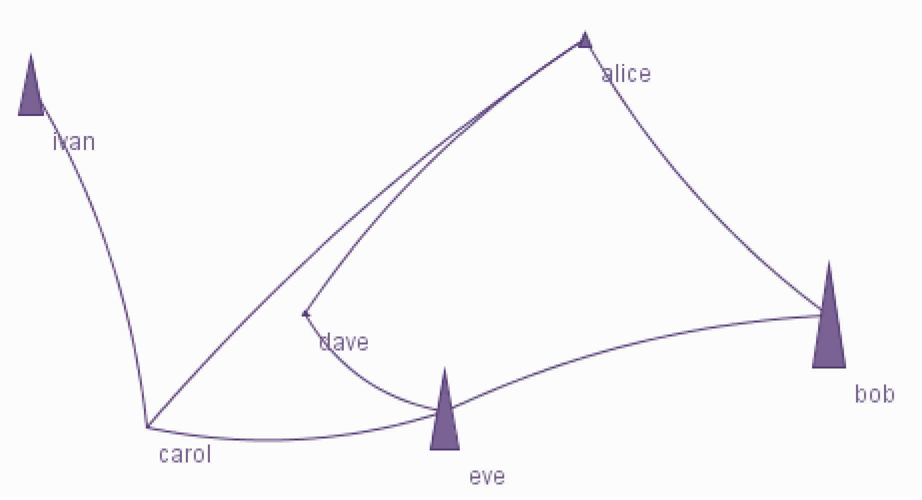
$$\gamma = \gamma(u, \ldots)$$

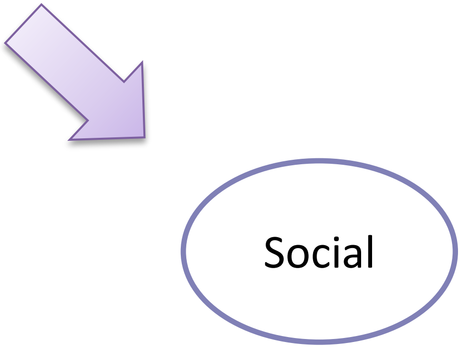


FW – Other input sources

- Explore other input sources
  - Implicit data (with time)
  - Social links

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





Thank you!

Performance Prediction in Recommender Systems

Alejandro Bellogín

Supervisor: Pablo Castells

Co-supervisor: Iván Cantador

Escuela Politécnica Superior
Universidad Autónoma de Madrid

@abellogin
alejandro.bellogin@uam.es

Questions to the committee

- In the same way as we have translated the performance prediction concept from IR to RS, **is there any concept from the User Modelling area which infers the ambiguity in a user profile and can be incorporated in a similar way into RS?**

- Up to now, we have focused our research on user-based CF and ensemble recommenders. We believe this idea may also be useful in a **personalisation scenario**, where depending on how ambiguous a user is predicted to be, the personalisation should receive more or less weight than the query. **Could this be interesting for the UMAP community?** Moreover, **is there any other application where the proposal may also be relevant?**

- In theory, correlation values between a predictor and a performance metric should uncover some aspects of the user, such as her ambiguity and uncertainty. At this moment, we have checked that performance predictors are able to capture rating noise (as in Amatriain et al., UMAP 2009). **If a user study could be conducted, which variables should be measured in order to validate our predictors?**

Answers (from reviews)

- Concepts from User Modelling area which infers the ambiguity in a user profile
  - More general: context
  - Goal: how to find the best fit of the conditions for a particular user goal

- Useful for personalisation? Or any other application?
  - It could be, but the model might be much more complex

- Variables to measure in a hypothetical user study
  - It depends on the user profile representation