Predicting Performance in Recommender Systems

Alejandro Bellogín
Universidad Autónoma de Madrid, Escuela Politécnica Superior
Ciudad Universitaria de Cantoblanco, 28049 Madrid, Spain
alejandro.bellogin@uam.es

ABSTRACT

Performance prediction has gained growing attention in the Information Retrieval field since the late nineties and has become an established research topic in the field. Our work restates the problem in the area of Recommender Systems, where it has barely been researched so far, despite being an appealing problem, as it enables an array of strategies for deciding when to deliver or hold back recommendations based on their foreseen accuracy. We investigate the adaptation and definition of different performance predictors based on the available user and item features. The properties of the predictor are empirically studied by checking the correlation of the predictor output with a performance measure. Then, we propose to introduce the performance predictor in a recommender system to produce a dynamic strategy. Depending on how the predictor is introduced we analyze two different problems: dynamic neighbor weighting in collaborative filtering and dynamic weighting of ensemble recommenders.

Categories and Subject Descriptors: H.3.3 Information Search and Retrieval – *information filtering*.

General Terms: Algorithms, Performance, Experimentation, Theory.

Keywords: Recommender Systems, performance prediction, query clarity, hybrid recommender system.

1. INTRODUCTION

Current Recommender System (RS) and Information Retrieval (IR) technologies and scenarios are characterized by an increasing diversification of the types and sources of data, content, evidence, background knowledge, retrieval services and methods, available for a recommender system –or retrieval system at large– to make decisions and build its output. In such context, predicting the performance of a specific recommendation approach or component becomes an appealing problem, as it allows self-assessing or properly combining the available alternatives, and making the most of them by dynamically adapting the recommendation strategy to the situation at hand.

In IR, performance prediction has gained increasing attention since the late 90's and has become an established research topic in the field [2][3][10][11]. It has been mostly addressed as a query performance issue, which refers to the performance of an IR system in response to a specific query [2][3][10]. Particularly effective predictors have been defined based on language models by the so-called clarity score, which captures the ambiguity in a query with respect to the collection, or a specific result set [10].

Performance prediction finds a special motivation in Recommender Systems (RS). Contrary to query-based retrieval, as far as the initiative relies on the system, it may decide to produce recommendations or hold them back, depending on the expected level of performance on a per case basis, delivering only the sufficiently reliable ones. The performance of individual recommendation methods is highly sensitive to different conditions (data sparsity, quality, reliability, etc.) which in real settings are subject to a large dynamic variability. Being able to estimate in advance which recommenders are likely to provide the best output in a particular situation opens up an important window for performance enhancement. Moreover, if the performance of different recommendation methods can be foreseen, ensemble recommenders can be dynamically adjusted to favor the best expected method in each situation, thus optimizing the overall ensemble performance.

In this thesis, we explore the extension of the performance prediction problem to Recommender Systems. We investigate how to adapt the performance predictors defined in IR to the area of RS. We define and evaluate specific predictors, by analyzing their correlation with respect to specific recommender performance metrics. Once their predictive power is confirmed (with significant correlation values) we propose to introduce the predictors in a recommender system and evaluate their use in two different problems: dynamic ensemble recommendation and dynamic neighborhood in collaborative filtering.

In both problems, state of the art approaches typically implement a static solution, where all the components in the system (users, items, recommenders) are considered equally. We propose to make use of performance predictors and to dynamically favor those components that are expected to perform better. Our current experimental results are positive, validating the specific techniques implemented so far, and the feasibility of the overall research approach.

Research Questions

In this thesis, we address the following research questions:

- Is it possible to define a performance prediction theory for recommender systems in a sound, formal way?
- Is it possible to adapt query performance techniques to the recommendation task?
- What kind of evaluation should be performed, e.g., different from the one used in IR or not?
- What kind of problems can these models be applied to?

For such purposes, we investigate the adaptation of IR performance prediction techniques to RS, where performance prediction refers to the estimation of the performance of an IR system in response to a specific query [11]. More specifically, we research the adaptation of the notion of *query clarity* [10] as a basis for finding suitable predictors. We hypothesize that the amount of uncertainty involved in user and item data, captured by adaptations of query clarity predictor for RS, may correlate with the accuracy of a system's recommendations. In that case, we could introduce a performance predictor in a recommender system to produce an adaptive recommendation strategy. In this context, we analyze two problems: building dynamic ensemble recommenders and dynamic neighborhoods in CF by exploiting user or item clarity values.

RecSys'11, October 23–27, 2011, Chicago, Illinois, USA.

Therefore, in this thesis, we investigate whether the hypothesis presented above, i.e., ambiguity in user's tastes correlates with accuracy of system's recommendations, holds in RS and can be useful in different situations. For this reason, we need to develop some predictors adapted from IR to RS by taking into account the unique characteristics present in RS data. After that, we plan to consider two complementary evaluations. First, we would compute the correlation between the predictor and the performance metric values, in order to check the predictive power of the model. Then, we would compare the final performance of the dynamic strategy against its static counterpart, to check whether the dynamic version outperforms or not the static one.

2. BACKGROUND

2.1 Performance Prediction

Query performance prediction in IR refers to the performance of an IR system in response to a specific query. It also relates to the appropriateness of a query as an expression for a user information need. In the literature, prediction methods have been classified into three groups depending on the available data used for prediction [11]: non-retrieval approaches, based on linguistic characteristics of the query, pre-retrieval approaches, which make the prediction before the retrieval stage by using statistical methods, and post-retrieval approaches, which use the rankings produced by the retrieval engine.

Non-retrieval and pre-retrieval approaches have the advantage that the prediction can be taken into account to improve the retrieval process itself. These predictors, however, have the potential handicap, with regards to their accuracy, that the extra retrieval effectiveness cues available after the system response are not exploited. Query scope [12] is an example of this type of predictors. It is a measure of the specificity of a query, which is quantified as the percentage of documents in the collection that contains at least one query term. Other examples such as statistical approaches based on Inverse Document Frequency (IDF), and variations thereof, have also been proposed in [2], [3], and [12]. These IDF-based predictors obtained moderate correlation with respect to the query performance. In [13], the authors investigate linguistic approaches, classified into morphological features, syntactic, and semantic. The authors found that many variables do not have a significant impact on any performance measure, only the more 'sophisticated' features, such as the semantic or syntactic ones.

On the other hand, post-retrieval predictors make use of the retrieved results. Broadly speaking, techniques in this category provide better prediction accuracy [2],[16]. However, computational efficiency is usually a problem for many of these techniques, and furthermore, the predictions cannot be used to improve the retrieval strategies, unless some kind of iteration is applied, as the output from the retrieval system is needed to compute the predictions in the first place. Effective predictors have been defined based on language models by the so-called clarity score, among others [11]. This predictor captures the (lack of) ambiguity in a query with respect to a specific result set, or the whole collection [10], [16] (the second case thus can be considered as a pre-retrieval predictor, since it would not make use of the result set).

2.2 Recommender Systems

Recommender Systems combine multiple data and strategies, and the balance is critical for the final performance. In this section, we briefly recall the main concepts in RS, including basic definitions and algorithms.

The aim of a recommender system is to assist users by suggesting "interesting" items from huge databases or catalogues, by taking into account (or inferring) the user's priorities or tastes. Three types of systems are commonly recognized, based on how recommendations are made [1]: content-based filtering (CBF), collaborative filtering (CF), and hybrid filtering (HF). CBF recommends the user items similar to the ones she preferred in the past, CF recommends the user items that people with similar tastes (or neighbors) liked in the past, and HF combines content-based and collaborative filtering approaches. In this context, although many alternatives are possible, the most common form of ground evidence of user preferences, upon which recommendations are generated, consists of explicit user ratings for individual items.

More formally, the recommender system has to find, for each user, the items maximizing a gain or utility function, i.e., $i^* = \arg\max_i g(u,i)$. Each type of RS estimates the utility of an item for a user differently. In this context, CBF typically builds a profile based on content features for the user, and similarly, another one for the item, after that, $g(u,i)$ is usually computed using the cosine function between both profiles. CF approaches, on the other hand, aggregate the values obtained by the set of more similar users (neighbors), such as follows: $g(u,i) = C \sum_{v \in N} sim(u,v) rat(v,i)$, where the similarity function is typically the Pearson correlation between the two users.

Thirdly, in [9] a detailed taxonomy is presented, where HF approaches are classified into different categories. Proliferation of new recommendation strategies are giving rise to an increasing variety of available options for the development of recommender systems. Researchers in Machine Learning have known for long that the combination of methods usually achieves better results than each method separately, which is also true in RS –the Netflix prize has been a paradigmatic example of this, where all the top classified teams used large ensemble recommenders, a specific type of HF approaches [5].

In this thesis, we focus on weighted hybrid approach, where scores are aggregated using linear combination or voting schemes, as an option that begets a simple and general formulation of the dynamic balance of the combined methods by just setting the weights of each method in the hybrid combination. Specifically, this approach can be defined as follows:

$$g(u,i) = \lambda * g_{R1}(u,i) + (1-\lambda) * g_{R2}(u,i)$$

where the weighting factor λ which aggregates the output from each recommender R1 and R2 is usually the same for every user u and item i. Because of that, we denote such a recommender as static hybrid.

3. PREDICTING PERFORMANCE IN RECOMMENDER SYSTEMS

Performance prediction provides tools that can be useful in many ways with dealing effectively with poorly-performing queries in IR. In this thesis, we take the perspective of an IR system and apply it to RS, where performance prediction helps addressing the problem of retrieval consistency, that is, a retrieval system can invoke alternative retrieval strategies for different queries according to their expected performance (e.g., query expansion or different ranking functions based on the predicted difficulty).

More specifically, we use the clarity score, which is measured as the following Kullback-Leibler divergence:

$$\text{clarity}(q) = \sum_{w \in \mathcal{V}} p(w|q) \log_2 \frac{p(w|q)}{p_c(w)}$$

$$p(w|q) = \sum_{d \in R} p(w|d)p(d|q)$$

$$p(d|q) = p(q|d)p(d);\ p(q|d) = \prod_{w_q \in q} p(w_q|d)$$

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

The clarity value can be reduced, thus, to an estimation of the prior $p_c(w)$ and the posterior $p(w|q)$ of query terms w over document d, based on term frequencies and smoothing. Cronen-Townsend et al [10] showed that clarity is correlated with performance, demonstrating that the result quality is largely influenced by the amount of uncertainty involved in the inputs the system takes. In this sense, queries whose likely documents are a mix of documents from disparate topics receive lower score than if they result in a topically-coherent retrieved set.

In this thesis, we investigate the adaptation of IR performance techniques to RS. More specifically, we draw from the notion of query clarity [10] as a basis for finding suitable predictors. In analogy to query clarity, we hypothesize that the amount of uncertainty involved in user and item data (reflecting ambiguity in the user's tastes, and item popularity patterns) may correlate as well with the accuracy of the system's recommendations. This uncertainty can be captured as the *clarity of users* and the *clarity of items* by an adaptation of the query clarity formulation. This adaptation, however, is not straightforward, since RS rank and recommend items without an explicit user query, using other inputs instead. We explore different formulations of the user clarity under different models in [6] and [8]. Different assumptions for the random variables derive different models, all of them grounded on previous probability formulations in RS such as [15].

4. EMPIRICAL STUDY

At this moment, we have started to analyze empirically the proposed predictors and their applications. With our experiments, we aim to, first, study the properties of the predictors, in terms of their correlation to actual recommendation performance. After that, we want to test the usefulness of the predictors with higher correlation values for weight adjustment in two different problems: dynamic ensemble weighting and dynamic CF neighborhood weighting. In this test, we compare the performance (in terms of recommendation accuracy) of dynamic against static strategies. All the experiments presented in this section use Movielens 100K[1] dataset, additional experiments are currently in progress using Last.fm[2] and Filmtipset[3] data.

4.1 Dynamic neighborhood

Different works have experimented with adaptive neighborhoods in order to improve CF performance [4][14]. In [8], we investigate whether CF results can be further enhanced by introducing, in addition to a similarity function, further effectiveness predictors, such as the user clarity value defined in the previous section, into the weights of the linear combination of neighbor ratings. The idea can be expressed as rewriting the general equation presented in Section 2.2 for rating prediction in CF as:

$$g(u,i) = C \sum_{v \in N} \gamma(v;u,i) sim(u,v) rat(v,i)$$

where $\gamma(v;u,i)$ is a predictor of the performance of neighbor v. In the general case, γ can be sensitive to the specific target user u, the item i, and in general it could even take further inputs from the recommendation space and context. As a first step, we explore the simple case when the predictor only examines the data related to the neighbor user v, and in particular, we consider $\gamma(v;u,i) = clarity(v)$. Figure 1 shows the effect of the introduction of this predictor in the computation of collaborative recommendations.

Our method clearly improves the baseline (by up to 5% for 60-80% cuts) when a small neighborhood is used. When larger neighborhoods are used (see [8] for more details) our method gets almost equal performance, which means that our approach get better results out of more economic neighborhood sizes.

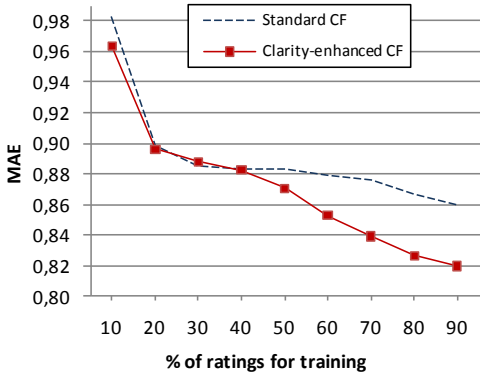

Figure 1. Performance comparison of CF with dynamic neighbor weighting, and standard CF (neighborhood size is 100).

Complementary, correlation analysis in this problem is not straightforward, since we need a measurable definition of what neighbor performance means, in order to check the correlation between predicted outcomes and objective measurements. In [8], we propose and define such a metric, which measures how "good a neighbor" a user is to her surrounding, by computing the difference in performance when including vs. excluding the user from the dataset. Pearson's correlation between the predictor and this performance metric is between 0.15 and 0.20, moderate correlations which, however, provide significant performance improvements as we can observe in Figure 1.

4.2 Dynamic ensemble recommendation

When using large ensemble recommenders, a specific type of HF approaches, the most important decision is how to combine information coming from the different recommenders, which, typically, are CBF and CF. We assume a weighted hybrid approach is used (see Section 2.2). The key point in this situation is that the optimal combination parameter value will be (potentially) different for each pair (user, item), instead of the typical scenario where static linear combination is used. That is, instead of the formula presented in Section 2.2, we would have:

$$g(u,i) = \gamma_{R1}(u,i) * g_{R1}(u,i) + \gamma_{R2}(u,i) * g_{R2}(u,i)$$

where γ_R is a combination parameter which may depend on the current user, item, or both.

In this thesis, we investigate how performance predictors may be used for building dynamic ensemble recommenders. In particular, we would like to consider $\gamma_{R1}(u,i) = clarity(u)$ under some conditions such as when positive correlation is found between the predictor and the recommender performance. Currently, we have

[1] Available at http://www.grouplens.org/node/73
[2] Last.fm, Social music service, http://www.last.fm
[3] Filmtipset, Social movie service, http://nyheter24.se/filmtipset

found that our predictors are able to obtain strong (around 0.5) positive correlation with some recommenders [6]. We believe this result would allow for proposing a framework in which decisions about when and how predictors should be used for ensemble recommenders can be taken.

5. DISCUSSION

To the best of our knowledge, this is the first research work addressing performance prediction in Recommender Systems. Solutions to this problem may find several uses, such as deciding whether or not a recommendation should be sent to a user, weighting ensemble recommenders, and weighting neighbors in collaborative filtering. The first results in our experiments are positive, as shown in the previous section.

We investigate different adaptations of query clarity techniques from ad-hoc Information Retrieval to define performance predictors in the context of Recommender Systems. For this adaptation, we use different formulations and assumptions, most of them grounded on studies from RS and IR, such as [15] and [10].

The first problem presented herein (dynamic neighborhood) supports the predictive power of clarity-based techniques in Collaborative Filtering as a basis for the adjustment of weights in the combination of neighbor ratings. Our approach successfully assigns higher weights to those neighbors expected to perform better. The results are particularly positive for small neighborhood situations. We plan nonetheless to study the proposed approach under alternative baseline CF formulation, such as item-based and factor models.

The second problem presented (dynamic ensemble recommendation) aims to provide a framework where decisions about when and how dynamic hybridization should be performed and improvements should be expected. Current experiment results indicate that it is possible to define predictors which obtain strong correlation with recommender performance. We plan to evaluate our framework with more than two recommenders in the ensemble, and more than one performance predictor, eventually, one for each recommender.

6. FURTHER WORK IN PROGRESS

The main focus of our research concerns the definition of user performance predictors, as well as different applications where they can be used. In this line, this thesis needs to a) find a theoretical background about why some predictors work better than others, i.e., have stronger correlations; and b) explore other input sources apart from ratings, such as the time dimension or the social links between users.

Regarding the first aspect, there are open issues which need further investigation, like for example the real meaning of the random variables introduced in our probabilistic models. Furthermore, we want to analyze why some recommenders are more prone to stronger correlation with respect to different formulations of the same predictor than others. We are working currently on this issue, specifically, by taking into account the final performance of the recommender, building unbiased correlation estimators, and also by analyzing relations between components of recommenders and predictors on a user-basis.

In order to explore other input sources, the next step will be to obtain more heterogeneous datasets where not only ratings, but implicit feedback, time and social relations are available. New performance predictors using this data should be defined and evaluated accordingly. At this point, we have presented in [7] a first experiment where social predictors are defined by using link-analysis metrics over a social network.

Acknowledgements. Special thanks to Prof. Pablo Castells and Dr. Iván Cantador for their support and guidance.

7. REFERENCES

[1] G. Adomavicius and A. Tuzhilin. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Trans. on Knowl. and Data Eng.*, 17 (6): 734–749, 2005.

[2] G. Amati, C. Carpineto, G. Romano. Query difficulty, robustness, and selective application of query expansion. In: ECIR, 127–137. Springer, Heidelberg, 2004.

[3] J.A. Aslam, V. Pavlu. Query hardness estimation using Jensen-Shannon divergence among multiple scoring functions. In: ECIR, 198–209. Springer, Heidelberg, 2007.

[4] L. Baltrunas and F. Ricci. Locally adaptive neighborhood selection for collaborative filtering recommendations. In: AH, 22–31. Springer, Heidelberg, 2008.

[5] R. M. Bell and Y. Koren. Lessons from the Netflix prize challenge. *SIGKDD Explor. Newsl.*, 9(2): 75–79, 2007.

[6] A. Bellogín, P. Castells, I. Cantador. Predicting the Performance of Recommender Systems: An Information Theoretic Approach. In: ICTIR, 27–39. Springer, Heidelberg, 2011.

[7] A. Bellogín, P. Castells, I. Cantador. Self-adjusting Hybrid Recommenders Based on Social Network Analysis. In: SIGIR, 1147–1148. ACM Press, NY, 2011.

[8] A. Bellogín and P. Castells. A Performance Prediction Approach to Enhance Collaborative Filtering. In: ECIR, 382–393. Springer, Heidelberg, 2010.

[9] R. Burke. Hybrid recommender systems: Survey and experiments. *User Modeling and User-Adapted Interaction*, 12(4): 331–370, 2002.

[10] S. Cronen-Townsend, Y. Zhou, and B.W. Croft. Predicting query performance. In: SIGIR, 299–306. ACM Press, NY, 2002.

[11] C. Hauff. *Predicting the Effectiveness of Queries and Retrieval Systems*. PhD thesis, University of Twente, Enschede, 2010.

[12] B. He and I. Ounis. Inferring query performance using pre-retrieval predictors. In: SPIRE, 43–54. Springer, Heidelberg, 2004.

[13] J. Mothe and L. Tanguy. Linguistic features to predict query difficulty. In: *SIGIR Workshop on Predicting Query Difficulty – Methods and Applications*, 2005.

[14] O'Donovan, J., Smyth, B.: Trust in recommender systems. In: IUI, 167–174. ACM Press, NY, 2005.

[15] J. Wang, A.P. de Vries, M.J.T. Reinders. Unifying user-based and item-based collaborative filtering approaches by similarity fusion. In: SIGIR, 501–508. ACM Press, NY, 2006.

[16] Y. Zhou and B.W Croft. Query performance prediction in web search environments. In: SIGIR, 543–550, ACM Press, NY, 2007.