Users and Noise: The Magic Barrier of Recommender Systems

Alan Said*, Brijnesh J. Jain*, Sascha Narr, and Till Plumbaum

Technische Universität Berlin
DAI Lab.
{alan,jain,narr,till}@dai-lab.de

Abstract. Recommender systems are crucial components of most commercial web sites to keep users satisfied and to increase revenue. Thus, a lot of effort is made to improve recommendation accuracy. But when is the best possible performance of the recommender reached? The *magic barrier*, refers to some unknown level of prediction accuracy a recommender system can attain. The magic barrier reveals whether there is still room for improving prediction accuracy, or indicates that any further improvement is meaningless. In this work, we present a mathematical characterization of the magic barrier based on the assumption that user ratings are afflicted with inconsistencies - noise. In a case study with a commercial movie recommender, we investigate the inconsistencies of the user ratings and estimate the magic barrier in order to assess the actual quality of the recommender system.

Keywords: Recommender Systems, Noise, Evaluation Measures, User Inconsistencies.

1 Introduction

Recommender systems play an important role in most top-ranked commercial websites such as Amazon, Netflix, Last.fm or IMDb [10]. The goal of these recommender systems is to increase revenue and present personalized user experiences by providing suggestions for previously unknown items that are potentially interesting for a user. With the growing amount of data in the Internet, the importance of recommender systems increases even more to guide users through the mass of data.

The key role of recommender systems resulted in a vast amount of research in this field, which yielded a plethora of different recommender algorithms [1, 4, 8]. An example of a popular and widely used approach to recommenders is *collaborative filtering*. Collaborative filtering computes user-specific recommendations based on historical user data, such as ratings or usage patterns [4, 7]. Other approaches include content-based recommenders (recommend items based on properties of a specific item), social recommenders (recommend things based on the past behavior of similar users in the social network) or hybrid combinations of several different approaches.

To select an appropriate recommender algorithm, and adapt it to a given scenario or problem, the algorithms are usually examined by testing their performance using either

* Both authors contributed equally to this work.

J. Masthoff et al. (Eds.): UMAP 2012, LNCS 7379, pp. 237–248, 2012.
© Springer-Verlag Berlin Heidelberg 2012

artificial or real test data reflecting the problem. The best performing algorithm and parameters among a number of candidate algorithms is chosen. To be able to compare performance, several different measures and metrics were defined. Common measures are precision and recall, normalized discounted cumulative gain (NDCG), receiver operating characteristic (ROC) or the root-mean-squared error (RMSE). RMSE is perhaps the most popular metric used to evaluate the prediction accuracy of a recommender algorithm [11]. It was the central evaluation metric used in the Netflix Prize competition[1]. For the RMSE as performance measure, a recommendation task is typically posed as that of learning a rating function that minimizes the RMSE on a given training set of user ratings. The generalization RMSE-performance of the learned rating function is then assessed on some independent test set of ratings, which is disjoint from the training set. One major drawback of measuring and comparing the performance using only static test data is that user behavior is not always reliable. According to studies conducted by [2, 3, 6] user ratings can be inconsistent (noisy) in the sense that a user may rate the same item differently at different points of time. Following these findings, Herlocker et al. [5] and other researchers coined the term *magic barrier*. The magic barrier marks the point at which the performance and accuracy of an algorithm cannot be enhanced due to noise in the data. Every improvement in accuracy might denote an over-fitting and not a better performance. Thus, comparing and measuring the expected future performance of algorithms based on static data might not work.

While investigations on the magic barrier are important for future recommendation research, only first evaluations on the inconsistency of ratings have been conducted so far. Most importantly, a mathematical characterization of the magic barrier is missing. In this paper, we will present such a mathematical characterization of the magic barrier, based on RMSE , which allows us to assess the actual performance of a recommender as well as the actual room for improvement. In particular, we can identify recommender algorithms that overfit the test set by chance or by peeking at the test set.[2]. We also conducted a first user study with a commercial recommender system, from the German movie recommendation website moviepilot[3], to substantiate our claims and findings in a real-world setting.

Our main contributions are as follows:

- We present a mathematical characterization of the magic barrier. Based on the principle of empirical risk minimization form statistical learning theory, we show that the magic barrier reduces to the RMSE of the optimal (but unknown) rating function. Then we characterize the magic barrier in terms of the expected inconsistencies

[1] `http://www.netflixprize.com`

[2] This occurs when information about the test set is used for constructing a recommender algorithm. For example, when we devise a new recommendation algorithm based on some similarity measure. We train our new algorithm using the cosine similarity and assess its performance on a test set. Then we train the same algorithm using the Pearson correlation and measure its performance on the same test set. Finally, we report the result of the algorithm and the similarity measure with the best prediction performance on the test set rather than choosing the model with the best prediction performance on the training set. Such approaches lead to overly optimistic results.

[3] `http://www.moviepilot.de`

incurred in the ratings. Since the magic barrier cannot be computed directly, we derive a procedure to estimate it.
– We present and discuss a case study with moviepilot, a commercial recommender system for movies. The case study aims at investigating user inconsistencies in a real-world setting. In addition, we estimated the magic barrier of moviepilot to assess the quality of moviepilot's recommendation engine and to propose a limit on how much moviepilot's recommendations can be improved.

Based on our findings, we propose that a real-world recommender system should regularly interact with users by polling their opinions about items they have rated previously in order to audit their own performance and, where appropriate, to take measures to improve their system.

2 Related Work

Inconsistency in user behavior in the context of recommender systems is a known concept and has been studied on several occasions before. The first mention of inconsistencies in a scope similar to ours was made by Hill et al. [6] in their study on virtual communities. The authors questioned how reliable the ratings were and found a rough estimate by calculating the RMSE between two sets of ratings performed by 22 users on two occasions 6 weeks apart.

Similar reliability issues, e.g. the levels of noise in user ratings, were discussed by Herlocker et al. [5], coining the term *the magic barrier* as an upper level of recommender system optimization.

More recently, Amatriain et al. [2] performed a set of user trials on 118 users based on a subset of the Netflix Prize dataset. The authors attempted to find answers to whether users are inconsistent in their rating behavior, how large the inconsistencies are and what factors have an impact on the inconsistencies. They were able to identify a lower bound, a magic barrier, for the dataset used in the trials.

Following their user trials, Amatriain et al. [3] successfully increased the accuracy of a recommender system by implementing a de-noising step based on re-ratings collected in a study. They presented two re-rating strategies (user-based and data-based) in order to find the *ground truth values* of ratings for the purpose of maximizing accuracy improvements in a recommender system. They concluded that re-rating previously rated items could, in some circumstances, be more beneficial than rating previously unrated items.

Some of the inconsistencies in users' rating behavior can be mitigated by temporal aspects, as Lathia et al. show [8]. This mitigation does however not compensate for all inconsistencies, which Amatriain et al. [3] showed by having different time spans between re-ratings.

The problems of noisy user behavior is connected to the type of evaluation used. Pu et al. [9] present a *user-centric* (as opposed to *data-centric*) evaluation framework which measures the quality of recommendations in terms of *usability*, *usefulness*, *interaction quality* and *user satisfaction*, which allows for optimization of recommender systems based on direct user interaction instead of *offline* accuracy metrics such as RMSE, NDCG, etc. User-centric evaluation does however come with a cost in terms of

time, additionally it requires a set of users to be available for the evaluation process. Given these drawbacks, most recommender system evaluation still uses traditional information retrieval measures and methods, even though these might not always reflect the actual quality of the recommendation [11] due to the aforementioned inconsistencies in users' rating behavior.

If a model for the maximum level of measure-based optimization would be available, a magic barrier, it could serve as the cut-off point between data-centric and user-centric evaluation.

3 The Empirical Risk Minimization Principle

We pose the recommendation task as that of a function regression problem based on the empirical risk minimization principle from statistical learning theory [12]. This setting provides the theoretical foundation to derive a lower bound (henceforth referred to as the magic barrier) on the root-mean-square error that can be attained by an optimal recommender system.

3.1 The Traditional Setting of a Recommendation Task

We begin by describing the traditional setting of a recommendation task as presented in [4].

Suppose that $\mathcal{R}$ is a set of ratings r_{ui} submitted by users $u \in \mathcal{U}$ for items $i \in \mathcal{I}$. Ratings may take values from some discrete set $\mathcal{S} \subseteq \mathbb{R}$ of rating scores. Typically, ratings are known only for few user-item pairs. The recommendation task consists of suggesting new items that will be rated high by users.

It is common practice to pose the recommendation task as that of learning a rating function

$$f : \mathcal{U} \times \mathcal{I} \to \mathcal{S}, \quad (u, i) \mapsto f(u, i)$$

on the basis of a set of training examples from $\mathcal{R}$. Given a user u, the learned rating function f is then used to recommend those items i that have largest scores $f(u, i)$. The accuracy of a rating function f is evaluated on a test set, which is a subset of $\mathcal{R}$ disjoint from the training set.

A popular and widely used measure for evaluating the accuracy of f on a set $\mathcal{R}$ of ratings is the root-mean-square error (RMSE) criterion

$$E(f|\mathcal{R}) = \sqrt{\frac{1}{|\mathcal{R}|} \sum_{(u,i)\in\mathcal{R}} \big(f(u, i) - r_{ui}\big)^2}, \tag{1}$$

where the sum runs over all user-item pairs (u, i) for which $r_{ui} \in \mathcal{R}$.[4]

[4] For the sake of brevity, we abuse notation and write $(u, i) \in \mathcal{R}$ for user-item pairs (u, i) for which $r_{ui} \in \mathcal{R}$.

3.2 Recommendation as Risk Minimization

Learning a rating function by minimizing the RMSE criterion can be justified by the inductive principle of empirical risk minimization from statistical learning theory [12]. Within this setting we describe the problem of learning a rating function as follows: We assume that

- user-item pairs (u, i) are drawn from an unknown probability distribution $p(u, i)$,
- rating scores $r \in \mathcal{S}$ are provided for each user-item pair (u, i) according to an unknown conditional probability distribution $p(r|u, i)$,
- $\mathcal{F}$ is a class of rating functions.

The probability $p(u, i)$ describes how likely it is that user u rates item i. The conditional probability $p(r|u, i)$ describes the probability that a given user u rates a given item i with rating score r. The class $\mathcal{F}$ of functions describes the set from which we choose (learn) our rating function f for recommending items. An example for $\mathcal{F}$ is the class of nearest neighbor-based methods.

The goal of learning a rating function is to find a function $f \in \mathcal{F}$ that minimizes the expected risk function

$$R(f) = \sum_{(u,i,r)} p(u, i, s)\big(f(u, i) - r\big)^2, \tag{2}$$

where the sum runs over all possible triples $(u, i, r) \in \mathcal{U} \times \mathcal{I} \times \mathcal{S}$ and $p(u, i, r) = p(u, i)p(r|u, i)$ is the joint probability.

The problem of learning an optimal rating function is that the distribution $p(u, i, s)$ is unknown. Therefore, we can not compute the optimal rating function

$$f_* = \arg\min_{f \in \mathcal{F}} R(f).$$

directly. Instead, we approximate f_* by minimizing the empirical risk

$$\widehat{R}(f|\mathcal{X}) = \frac{1}{|\mathcal{X}|} \sum_{r_{ui} \in \mathcal{X}} \big(f(u, i) - r_{ui}\big)^2,$$

where $\mathcal{X} \subseteq \mathcal{R}$ is a training set consisting of ratings r_{ui} given by user u for item i. Observe that minimizing the empirical risk is equivalent to minimizing the RMSE criterion.

A theoretical justification of minimizing the RMSE criterion (or the empirical risk) arises from the following result of statistical learning theory [12]: under the assumption that the user ratings from $\mathcal{R}$ are independent and identically distributed, the empirical risk is an unbiased estimate of the expected risk.[5]

[5] The set of users and items are both finite. In order to apply the law of large numbers, we may think of $\mathcal{R}$ as being a set of ratings obtained by randomly selecting triples (u, i, s) according to their joint distribution.

4 The Magic Barrier

This section derives a magic barrier (lower bound) on the RMSE that can be attained by an optimal recommender system. We show that the magic barrier is the standard deviation of the inconsistencies (noise) inherent in user ratings. To this end, we first present a noise model and then derive the magic barrier.

4.1 A Statistical Model for Users' Inconsistencies

As shown in user studies [2,3,6], users' rating tend to be inconsistent. Inconsistencies in the ratings could be due to, for example, change of taste over time, personal conditions, inconsistent rating strategies, and/or social influences, just to mention a few.

For the sake of convenience, we regard inconsistencies in user ratings as noise. The following fictitious scenario illustrates the basic idea behind our noise model: Consider a movie recommender with n movies and a rating scale from zero to five stars, where zero stars refers to a rating score reserved for unknown movies only. Users are regularly asked to rate m randomly selected movies. After a sufficiently long period of time, each user has rated each movie several times. The ratings may vary over time due to several reasons ([8]) such as change of taste, current emotional state, group-dynamic effects, and other external as well as internal influences.

Keeping the above scenario in mind, the *expected rating* of a user u on movie i is defined by the expectation

$$\mathbb{E}[R_{ui}] = \mu_{ui},$$

where R_{ui} is a random variable on the user-item pair (u, i) and takes on zero to five stars as values. Then a rating r_{ui} is composed of the expected rating μ_{ui} and some error term ε_{ui} for the noise incurred by user u when rating item i. We occasionally refer to the error ε_{ui} as user-item noise. Thus, user ratings arise from a statistical model of the form

$$r_{ui} = \mu_{ui} + \varepsilon_{ui}, \tag{3}$$

where the random error ε_{ui} has expectation $\mathbb{E}[\varepsilon_{ui}] = 0$.

4.2 Deriving the Magic Barrier

Suppose that f_* is the true (but unknown) rating function that knows all expected ratings μ_{ui} of each user u about any item i, that is

$$f_*(u, i) = \mu_{ui} \tag{4}$$

for all users $u \in \mathcal{U}$ and items $i \in \mathcal{I}$. Then the optimal rating function f_* minimizes the expected risk function Eq. (2). Substituting Eq. (3) and Eq. (4) into the expected risk function Eq. (2) and using $p(u, i, s) = p(u, i)p(s|u, i)$ gives

$$R(f_*) = \sum_{(u,i)} p(u, i)\mathbb{E}\left[\varepsilon_{ui}^2\right] = \sum_{(u,i)} p(u, i)\mathbb{V}\left[\varepsilon_{ui}\right], \tag{5}$$

where the sum runs over all possible user-item pairs $(u, i) \in \mathcal{U} \times \mathcal{I}$ and $\mathbb{V}[\varepsilon_{ui}]$ denotes the variance of the user-item noise ε_{ui}. Eq. (5) shows that the expected risk of an optimal rating function f_* is the mean variance of the user-item noise terms.

Expressed in terms of the RMSE criterion, the magic barrier $B_{\mathcal{U}\times\mathcal{I}}$ of a recommender system with users $\mathcal{U}$ and items $\mathcal{I}$ is then defined by

$$B_{\mathcal{U}\times\mathcal{I}} = \sqrt{\sum_{(u,i)} p(u,i)\mathbb{V}\left[\varepsilon_{ui}\right]}.$$

The magic barrier is the RMSE of an optimal rating function f_*. We see that even an optimal rating function has a non-zero RMSE *unless all users are consistent with their ratings*.

Observe that an optimal rating function needs not to be a member of our chosen function class $\mathcal{F}$ from which we select (learn) our actual rating function f. Thus the RMSE of f can be decomposed into the magic barrier $B_{\mathcal{U}\times\mathcal{I}}$ and an error E_f due to model complexity of f giving

$$E_{RMSE}(f) = B_{\mathcal{U}\times\mathcal{I}} + E_f > B_{\mathcal{U}\times\mathcal{I}}.$$

4.3 Estimating the Magic Barrier

As for the expected risk, we are usually unable to directly determine the magic barrier $B_{\mathcal{U}\times\mathcal{I}}$. Instead we estimate the magic barrier according to the procedure outlined in Algorithm 1.

Algorithm 1. Procedure for estimating the magic barrier.

Procedure: Let $\mathcal{X} \subseteq \mathcal{U} \times \mathcal{I}$ be a randomly generated subset of user-item pairs.

1. For each user-item pair $(u, i) \in \mathcal{X}$ do
   (a) Sample m ratings $r_{ui}^1, \ldots, r_{ui}^m$ on a regular basis
   (b) Estimate the expectation μ_{ui} by the sample mean

$$\widehat{\mu}_{ui} = \frac{1}{m}\sum_{t=1}^{m} r_{ui}^t$$

   (c) Estimate the variance of the ratings

$$\widehat{\varepsilon}_{ui}^2 = \frac{1}{m}\sum_{t=1}^{m}\left(\widehat{\mu}_{ui} - r_{ui}^t\right)^2$$

2. Estimate the magic barrier by taking the average

$$\widehat{B}_{\mathcal{X}} = \sqrt{\frac{1}{|\mathcal{X}|}\sum_{(u,i)\in\mathcal{X}} \widehat{\varepsilon}_{ui}^2}. \tag{6}$$

We postulate that all rating functions $f \in \mathcal{F}$ with an empirical risk of the form

$$\widehat{R}(f|\mathcal{X}) \leq \widehat{B}_{\mathcal{X}}^2$$

are likely to overfit on the set $\mathcal{X}$ and consider further improvements on the RMSE below $\widehat{B}_{\mathcal{X}}$ as *meaningless*.

5 Case Study Using a Commercial Movie Recommender

Our experimental case study serves to validate the noise model and to investigate the relationship between the estimated magic barrier and the prediction accuracy of moviepilot. Due to limited resources, we conducted a moderately scaled user study in a real-world setting.

5.1 Moviepilot

moviepilot is a commercial movie recommender system having more than one million users, $55,000$ movies, and over 10 million ratings. Movies are rated on a 0 to 10 scale with step size 0.5 (0 corresponding to a rating score of 0, not an unknown rating). The two most common ways to rate movies are either through the "discover new movies" page, shown in Fig. 1(a) or through the "100 movies of your lifetime" page. The former presents a combination of new, popular and recommended movies whereas the latter one presents, like the title suggests, the 100 previously unrated movies deemed most probable to be liked by the user.

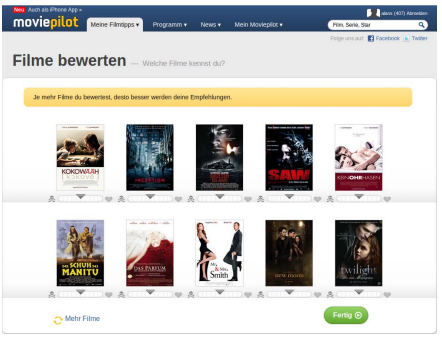
The recommendation engine uses a neighborhood-based collaborative filtering approach, with a similarity measure inspired by the cosine similarity, and is retrained regularly, so as to always be able to recommend movies based on an up-to-date model of users' rating histories.

5.2 Data

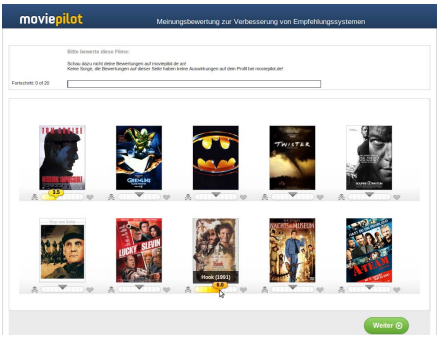
To estimate the magic barrier, we created a Web-based study for collecting users' *opinions* on movies. An opinion is a score in the same rating scale as standard user ratings. The difference between the two is that ratings are stored in the user profile and used for predictions, whereas opinions do not show up in users' profiles, are only stored in the survey and do, subsequently, not affect the recommendations users are given.

The opinion collection study was implemented as a Web application, Fig. 1(b), mirroring the look-and-feel of moviepilot's rating interface as closely as possible. By emulating the rating interface of moviepilot for opinion polling, we aimed at mitigating any potential distortion of the data due to different interface elements. A comparison of both is shown in Fig. 1.

Users were notified about the study by announcements in newsletters and posts in the community forums. The announcements provided information on timing, duration, process of the opinion collection, the collector, the URL, etc. Users were asked to not to peek at their old ratings when taking part in the study. They were also informed that submitted opinions would not be stored in their profiles.



(a) moviepilot's find new movies page



(b) Opinion interface

Fig. 1. The moviepilot rating interface and the opinion interface used in the user study. The opinion interface mimics the look-and-feel of moviepilot in order to give users a feeling of familiarity lowering the level UI-induced noise.

Whilst taking part in the study, users were presented with a number of movies randomly drawn from the complete set of their rated movies. Each user could submit at most one opinion on each movie. A user could skip any number of movies without providing any opinion. After at least a 20 opinions had been given, the active user could complete the study.

The study ran from mid April 2011 to early May 2011. We recorded only opinions of users that provided opinions on at least 20 movies. A total of 306 users provided $6,299$ opinions about $2,329$ movies.

5.3 Experimental Setup

We estimated the magic barrier according to the procedure described in Algorithm 1. For this, we used the ratings and opinions of those user-item pairs for which the $6,299$ opinions had been recorded.[6] This setup corresponds to sampling two ratings for each of the $6,299$ user-item pairs. The estimate of the magic barrier is the average of the squared sample noise over all $6,299$ user-item pairs.

5.4 Results

Fig. 2 summarizes the outcome of the opinion polling study, it shows the RMSE of moviepilot's recommendation engine and an estimated magic barrier taken over all $6,299$ user-item pairs. The plot also shows the estimated magic barrier restricted to the following subsets of the $6,299$ user-item pairs: (1) user-item pairs with above average rating, (2) user-item pairs with above average opinion, (3) user-item pairs with below average rating, and (4) user-item pairs with below average opinion. The user-specific averages were taken over all ratings given by the respective user.

[6] Though opinions differ from ratings conceptually, we treat them equally when estimating the magic barrier.

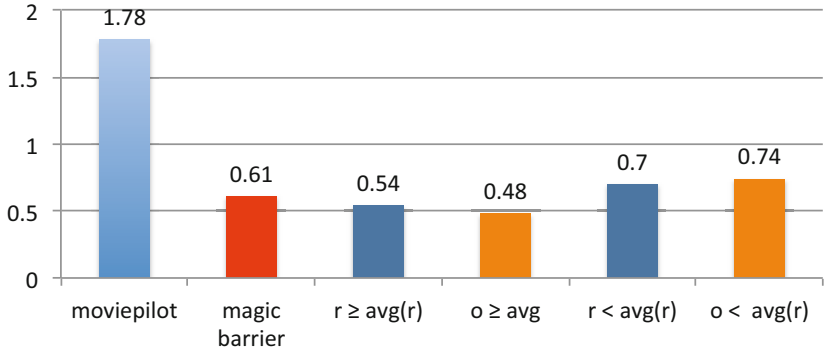


Fig. 2. RMSE of moviepilot and estimated magic barrier, where *all* refers to the estimation computed over all $6,299$ user-item pairs, $r \geq avg(r)$ and $o \geq avg(r)$, resp. refer to the magic barrier restricted to all user-item pairs with ratings and opinions, respectively, *above or equal to* the user-specific average over all user ratings. Similarly, $r < avg(r)$ and $o < avg(r)$, respectively, refer to the magic barrier restricted to all user-item pairs with ratings and opinions, resp., *below* the user-specific average over all user ratings and opinions.

The first observation to be made is that the estimated magic barrier of moviepilot is circa 0.61, which is slightly more than one step in moviepilot's rating scale (± 0.61). In contrast, the RMSE of moviepilot's recommendation engine is about 1.8 which is between three and four rating steps. Under the assumption that the estimated magic barrier is a good estimate of the unknown magic barrier, improvements of a recommender method close to or below the estimated magic barrier are meaningless. Under the same assumptions, there is room for improving the prediction accuracy of moviepilot. These assumptions, however, have to be taken with care due to the limited amount of data for estimating the expected rating of each user-item pair.

The second observation to be made is that our estimate of the magic barrier is lower when restricted to user-item pairs with ratings/opinions above average ratings than for below average ratings. We hypothesize that users tend to be more consistent with their ratings/opinions for movies that they have rated above average. This finding complements the observations of Amatriain et al. [2], i.e. that user ratings seem to be more consistent at the extreme ends of the rating scale.

The results obtained in this as well as in other studies and the theoretical treatment on magic barriers give a strong case for collecting opinions (or re-ratings) in order to

1. estimate the magic barrier for performance evaluation, and
2. improve recommendations based on a set of ratings for each user-item pair rather than on a single rating.

A good estimate of the magic barrier is useful for assessing the quality of a recommendation method and for revealing room for improvements. Recommenders with a prediction accuracy close to the estimated magic barrier can be regarded as 'optimal'. Further improvements of such recommenders are meaningless.

6 Conclusion

The magic barrier is the RMSE of an optimal rating function, and as such, it provides a lower bound for the RMSE an arbitrary rating function can attain. In terms of noise incurred when users rate items, the magic barrier is the square root of the expected variance of the user-item noise. Using this characterization, it is straightforward to derive a procedure for estimating the magic barrier.

In an experimental case study using moviepilot, a commercial movie recommender system, we investigated inconsistencies of user ratings and estimated the magic barrier for assessing the actual prediction accuracy of moviepilot. The results confirm that users are inconsistent in their ratings and that they tend to be more consistent for above average ratings. Our estimate of the magic barrier reveals that there is room to improve moviepilot's recommendation algorithm.

On the basis of our findings we suggest that regularly polling ratings for previously rated items can be useful to audit the performance of the recommendation engine and may, where appropriate, lead to measures taken for improving the existing system.

To obtain statically sound results, a large-scale user study is imperative. In order to regularly poll opinions/ratings of previously rated items, the following issues should be addressed: (1) How to implement a user-friendly interface for polling opinions/ratings without having a deterrent effect on users and unbiased results at the same time? (2) How to present items and sample opinions/ratings to obtain a good estimate of the magic barrier?

Acknowledgments. The authors would like to express their gratitude to the users of moviepilot who took their time to conduct the survey and the moviepilot team who contributed to this work with dataset, relevant insights and support.

The work in this paper was conducted in the scope of the KMulE project which was sponsored by the German Federal Ministry of Economics and Technology (BMWi).

References

1. Adomavicius, G., Tuzhilin, A.: 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, 734–749 (2005)
2. Amatriain, X., Pujol, J.M., Oliver, N.: I Like It.. I Like It Not: Evaluating User Ratings Noise in Recommender Systems. In: Houben, G.-J., McCalla, G., Pianesi, F., Zancanaro, M. (eds.) UMAP 2009. LNCS, vol. 5535, pp. 247–258. Springer, Heidelberg (2009)
3. Amatriain, X., Pujol, J.M., Tintarev, N., Oliver, N.: Rate it again: increasing recommendation accuracy by user re-rating. In: Proceedings of the Third ACM Conference on Recommender Systems, RecSys 2009, pp. 173–180. ACM, New York (2009)
4. Desrosiers, C., Karypis, G.: A Comprehensive Survey of Neighborhood-based Recommendation Methods. In: Ricci, F., Rokach, L., Shapira, B., Kantor, P.B. (eds.) Recommender Systems Handbook, pp. 145–186. Springer, US (2011)
5. Herlocker, J.L., Konstan, J.A., Terveen, L.G., Riedl, J.T.: Evaluating collaborative filtering recommender systems. ACM Trans. Inf. Syst. 22, 5–53 (2004)
6. Hill, W., Stead, L., Rosenstein, M., Furnas, G.: Recommending and evaluating choices in a virtual community of use. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 194–201. ACM Press/Addison-Wesley Publishing Co. (1995)

7. Koren, Y., Bell, R.: Advances in Collaborative Filtering. In: Ricci, F., Rokach, L., Shapira, B., Kantor, P.B. (eds.) Recommender Systems Handbook, pp. 145–186. Springer, US (2011)
8. Lathia, N., Hailes, S., Capra, L., Amatriain, X.: Temporal diversity in recommender systems. In: Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2010, pp. 210–217. ACM, New York (2010)
9. Pu, P., Chen, L., Hu, R.: A user-centric evaluation framework for recommender systems. In: Proceedings of the Fifth ACM Conference on Recommender Systems, RecSys 2011, pp. 157–164. ACM, New York (2011)
10. Ricci, F., Rokach, L., Shapira, B.: Introduction to Recommender Systems Handbook. In: Ricci, F., Rokach, L., Shapira, B., Kantor, P.B. (eds.) Recommender Systems Handbook, pp. 1–35. Springer, US (2011)
11. Shani, G., Gunawardana, A.: Evaluating Recommendation Systems. In: Ricci, F., Rokach, L., Shapira, B., Kantor, P.B. (eds.) Recommender Systems Handbook, pp. 257–297. Springer, US (2011)
12. Vapnik, V.N.: The nature of statistical learning theory. Springer-Verlag New York, Inc., New York (1995)