On the Optimal Non-Personalized Recommendation: From the PRP to the Discovery False Negative Principle

Rocío Cañamares
Universidad Autónoma de Madrid
rocio.cannamares@uam.es

Pablo Castells
Universidad Autónoma de Madrid
pablo.castells@uam.es

ABSTRACT

We revisit the Probability Ranking Principle in the context of recommender systems. We find a key difference in the retrieval protocol with respect to query-based search, that leads to the identification of a different optimal ranking principle for non-personalized recommendation. Based on this finding, we explore the definition of practical ranking functions that may lean towards the optimal ranking. We run an experiment confirming and illustrating our theoretical analysis, and providing further observations and hints for reflection and future research.

1 INTRODUCTION

Robertson [7] put forward and discussed the Probability Ranking Principle (PRP) stating that under certain assumptions, the optimal ranking for a given information need is by decreasing probability of relevance of the documents to the information need. Robertson described and analyzed cases where the PRP may fail, and potential restatements of the principle in view of such limitations. A profuse line of research followed up extending or reexamining the PRP, seeking better, more complete, or more generalized principles [10], or aiming to fit the particularities of specific IR scenarios (such as interactive retrieval [5] or multimedia retrieval [9], to name a few). The PRP remains nonetheless a prominent notion today at the foundation of IR theory.

In this paper we analyze the recommendation task [1,4] as a new use case following up the spirit of this long strand of research: seeking and analyzing the definition of an optimal ranking, following a formal methodological approach. A particularity of recommendation compared to the search task is that item relevance is understood to be a fully personal and subjective matter, solely defined by each end-user's personal taste, whereas judging the relevance of a search result has a (non-null but) narrower scope for disagreement, limited by a specific information need and its explicit expression as a query. Yet the PRP analysis in the context of search [7] has similarly considered degrees of user-level subjectivity or disagreement (in particular, as a challenge to the PRP), whereby our present research can be connected to such prior work in more than one way.

In order to make the problem more tractable, we shall make a simplifying restriction: we shall consider non-personalized recommendation, where all users are delivered the same item ranking. This is somewhat against the essence of recommender systems, which are assumed to be personalized in order to find closer approximations to user satisfaction optima. This restriction does however not make the problem irrelevant: forms of non-personalized recommendation have been found to achieve suboptimal but non-negligible performance compared to state of the art personalized recommender systems [4]. Moreover, personalized algorithms have been shown to display a noticeable degree of correlation with non-personalized rankings [2,3,6]. Thus any theoretical finding we may learn for non-personalized recommendation might potentially generalize, in some form and to some degree, to personalized methods, and help understand their trends and limits.

A particularity of the recommendation task is that, in its most widespread statement, the system should avoid recommending items the target user has already been observed interacting with. This restriction applies in scenarios where the added-value of recommendation is tightly linked to a purpose of discovery, as a complement of what users can already have experienced by themselves, and the assistance that other information retrieval technologies such as search engines already provide. In terms of an evaluation experiment, the condition means that items with an observed interaction record for the target user should be excluded from the ranking delivered to this user.

This restriction substantially changes the frame for the optimal ranking analysis. To begin with, it means that a one-for-all ranking will eventually end up being personalized (albeit in a very limited way), since different items will be discarded from the ranking for different users, based on their respective individual prior experience with the item set. Moreover, item exclusion can potentially alter the optimality analysis, e.g. if dependencies are present between the probability of relevance and the probability of an item to be excluded from the ranking.

2 BASIC CONCEPTS AND NOTATION

The recommendation task considers a set of users $\mathcal{U}$, a set of items $\mathcal{I}$, and a set of observed interaction records between users and items that can be interpreted as evidence of the user liking or disliking the item (i.e. relevance or non-relevance). As a widespread simplification, we may assume interaction data consist of a binary value $r: \mathcal{O} \subset \mathcal{U} \times \mathcal{I} \rightarrow \{0,1\}$ so that $r(u,i) = 1$ if the user $u \in \mathcal{U}$ likes the item $i \in \mathcal{I}$, and $r(u,i) = 0$ otherwise. Following common terminology, we shall refer to $r(u,i)$ as a *rating*, regardless of whether the datum has been explicitly introduced by the user as a literal rating, or is implicitly evidenced in her spontaneous interaction with the item. Ratings are available only for a subset $\mathcal{O}$ (typically a tiny fraction) of all user-item pairs $\mathcal{U} \times \mathcal{I}$ – there would otherwise not be any recommendation task to solve.

Taking the available rating data as input, the task of a recommender system is to compute a score for all user-item pairs where a rating is missing, and thus generate a ranking of unrated items to be delivered as a recommendation to each user in the system. The system output is evaluated using further user ratings on the recommended items, to be collected somehow, taken as relevance judgments. Such judgments can be obtained in different ways, depending on the evaluation setting. For instance, in offline evaluation, relevance judgments are sampled (as so-called test data) from the available rating dataset itself, hiding them from the rec-

ommender system to be evaluated, and the remaining ratings are supplied as input training data to the system. In our theoretical analysis we will to some extent abstract ourselves from the problem of obtaining judgments, and assume we will manage somehow to get the relevance information we need.

For the convenience of our formal analysis, we shall introduce two binary random variables $rated: \mathcal{U} \times \mathcal{I} \to \{0,1\}$ and $rel: \mathcal{U} \times \mathcal{I} \to \{0,1\}$, where $rated = 1$ iff a rating (be it positive or negative) by the user on the item is present in the input data, and $rel = 1$ iff the user likes the item, regardless of whether this is known to the system (by the presence of a rating) or not. With this notation we can express well-defined distributions, e.g. $p(rated|i)$ is the ratio of users in $\mathcal{U}$ who have rated item i, and $p(rel|i)$ is the fraction of users who like the item.

2 EXPECTED PRECISION

Whereas Robertson [7] considered a variety of evaluation metrics and cutoffs in his analysis, we shall focus here on $P@1$ as a simplest metric to make our analysis more tractable. Given a recommendation for a user, $P@1$ is a binary value that is equal to 1 if the target user likes the top ranked item, and 0 if she does not. This makes it easier to reason about the expected value of this metric. As a binary function, the expectation of $P@1$ for a given recommendation R is the probability of taking value 1: $\mathbb{E}[P@1|R] = p(P@1 = 1|R)$.

Now we need to be more precise with the computation of the metric: $P@1 = 1$ given a ranking R iff the first ranked *recommendable* item in R is relevant. Let this item be R_k, ranked in the k-th position of R. As stated in the introduction, recommendable means that R_k does not have a rating by the target user, and being the first means that all the items $R_1, R_2, \ldots, R_{k-1}$ above R_k are not recommendable because they do have a rating.

Let $rated_j$ represent the event that a rating $r(u, R_j)$ by the target user is present in the input data. Similarly, let rel_j mean R_j is relevant. If we marginalize $p(P@1 = 1|R)$ by the possibility that the k-th item is the first recommendable, we have:

$$\mathbb{E}[P@1|R] = \sum_{k=1}^{n} p(rel_k, rated_1, \ldots, rated_{k-1}, \neg rated_k|R)$$

where $n = |\mathcal{I}|$ is the total number of items in the system.

We shall now make the mild assumption that whether two items are rated or not by some user are mutually independent events. This assumption involves a (joint) probability overestimation that is negligible in the top ranking positions (the ones that determine $P@1$ for the most part). Now switching to the notation $p(rel|R_k)$ for $p(rel_k)$ and same for $rated$, under this assumption we have:

$$\mathbb{E}[P@1|R] \sim \sum_{k=1}^{n} p(rel, \neg rated|R_k) \prod_{j=1}^{k-1} p(rated|R_j) \quad (1)$$

We should note how this equation contrasts to what we would get without considering item exclusion, in which case we would simply have $\mathbb{E}[P@1|R] \sim p(rel|R_1)$ and $\mathbb{E}[P@N|R] \sim \sum_{k=1}^{N} p(rel|R_k)$ as in [7], and the PRP analysis would be similarly applicable here. The exclusion of rated items can thus make a difference in the computation of the metric and, potentially, in the outcome of a comparative evaluation of algorithms.

3 OPTIMAL RANKING PRINCIPLE

We can now set forth the following result on the optimal non-personalized ranking for expected precision.

Lemma – Discovery False Negative Principle. Assuming pairwise item rating independence, the optimal non-personalized recommendation R that maximizes the expected $P@1$ ranks items by non-increasing value of $p(rel|\neg rated, R_k)$.

Proof. It suffices to show that a swap against $p(rel|\neg rated, R_k)$ in a ranking produces a smaller value for $\mathbb{E}[P@1|R]$. Given that any ranking can be generated by a sequence of pairwise counter-order swaps on any other ranking (as per e.g. the proof of correction of bubble sort), we would have proven our point.

Let R be some ranking so that $p(rel|\neg rated, R_k) \geq p(rel|\neg rated, R_{k+1})$ for some k, and let us consider a ranking R' consisting of swapping R_k and R_{k+1} in R. Using equation 1 it is easy to see that, by trivial algebraic cancellation and rearrangement of terms, we have:

$$\begin{aligned}
&\mathbb{E}[P@1|R] \geq \mathbb{E}[P@1|R'] \\
&\quad \Leftrightarrow p(rel, \neg rated|R_k) + p(rel, \neg rated|R_{k+1}) p(rated|R_k) \\
&\qquad \geq p(rel, \neg rated|R_{k+1}) + p(rel, \neg rated|R_k) p(rated|R_{k+1}) \\
&\quad \Leftrightarrow \frac{p(rel, \neg rated|R_k)}{1 - p(rated|R_k)} \geq \frac{p(rel, \neg rated|R_{k+1})}{1 - p(rated|R_{k+1})} \\
&\quad \Leftrightarrow p(rel|\neg rated, R_k) \geq p(rel|\neg rated, R_{k+1})
\end{aligned}$$

Which is true by description of R. That is, swapping R_k and R_{k+1} decreases $\mathbb{E}[P@1|R]$. □

The scope of the lemma is non-personalized recommendation because we are considering a single ranking R for all users, and the user variable is missing in the above statement and the subsequent proof all along.

We thus see we get a variation of the PRP, stating we should rank items by decreasing value of $p(rel|\neg rated, R_k)$ rather than $p(rel|R_k)$. The probability $p(rel|\neg rated, R_k)$ corresponds to the fraction of unobserved (unrated) user tastes that are positive: the ratio of positive missing ratings. This means that the best items to be recommended are not exactly the ones that please most people, but the ones for which most unobserved preferences by the system (or undiscovered by users themselves) are positive. If we look at preference discovery as a retrieval process (prior to recommendation) in its own, $p(rel|\neg rated, R_k)$ represents the false negative ratio of this process. We may thus label this finding as the Discovery False Negative Principle (DFNP).

This principle makes natural sense in the recommendation context. An item that many people like (pure probability of relevance), but that most people have already interacted with, is of little use for recommendation, as it will be excluded from the rankings delivered to their potential "likers", and will be recommended to people who have not yet interacted with the item, but who will most probably not like it. Items with a high positive ratio in their missing ratings, in contrast, have a safe unexploited potential market –be it small or large– to make profit from.

Ratings come to be by users becoming aware of the existence of an item in the first place (by searching, browsing, advertisement, advice from a friend, random chance, etc.) and, second, by the system witnessing the encounter between the user and the item. Thus recommendation should favor items for which prior

discovery has most failed, which to much extent describes the raison d'être of recommendation: complementing and filling the gaps left by other means for discovery and retrieval.

4 OPTIMIZING NON-PERSONALIZED RECOMMENDATION

Considering the principle that drives the best possible recommendation, we may wonder if we could use it to the benefit of designing the best possible recommendation algorithms, namely by managing to obtain some approximation to $p(rel|\neg rated, R_k)$. A proper estimation of this probability involves two random variables, for one of which we have full knowledge ($rated$), but not for the other (rel). For relevance, we only have a sample, namely, the relevance that is observed by ratings. Unfortunately using this sample is incompatible with the estimation of a probability that negates the presence of ratings as a condition.

We can however consider combinations of probabilities that may partially match the optimal ranking function, and use them to rank items for recommendation, taking positive ratings as an observed sample of the relevance data, in the hope that such functions may produce rankings that are, in practice, not that far from the optimal. Reading the optimal ranking function as $p(rel|\neg rated, R_k) = p(rel, \neg rated|R_k)/p(\neg rated|R_k)$, we may consider related expressions such as:

$$f_1(R_k) = p(rel, rated|R_k)/p(\neg rated|R_k)$$
$$f_2(R_k) = p(rel, rated|R_k)/p(rated|R_k)$$
$$f_3(R_k) = p(rel, rated|R_k)$$

It is easy to see that f_2 is the average rating value of an item (i.e. the ratio of positive ratings when we consider binary values), and f_3 is equivalent to the number of positive ratings of the item, which amounts to what is referred to as item popularity in the recommender systems literature [2,3,4,6]. f_1 does not have such a natural interpretation but might be a fair candidate as well.

Even if such functions are not a proper analytical match of the optimal ranking function, we may hope they may produce as good rankings as we can get without further relevance knowledge beyond the available ratings. We explore this possibility empirically, setting up a special purpose dataset.

5 EXPERIMENT

To match the implicit assumptions of our theoretical analysis, we take a crowdsourced dataset that provides the opportunity to get ratings in the way users might produce through spontaneous activity, but at the same time includes further relevance knowledge that would not be obtained in the natural process.

The dataset[1] [3] was built using the CrowdFlower[2] platform, and includes preference judgments entered by 1,000 people for 1,000 music tracks randomly sampled from the Deezer database.[3] A judgment declares whether or not the user likes the music, after listening to a short clip of the track. Each user is assigned 100 tracks, sampled uniformly at random, in such a way that each track gets about 100 judgments, amounting to a total of 100,000 judgments in the dataset. In addition to her taste, the user is asked whether or not she knew the music before this survey. Fig. 1a shows the user interface where the CrowdFlower workers enter their input for a music track, and Fig. 1b shows the resulting distributions of the total number of judgments, positive judgments, and prior awareness for each item.

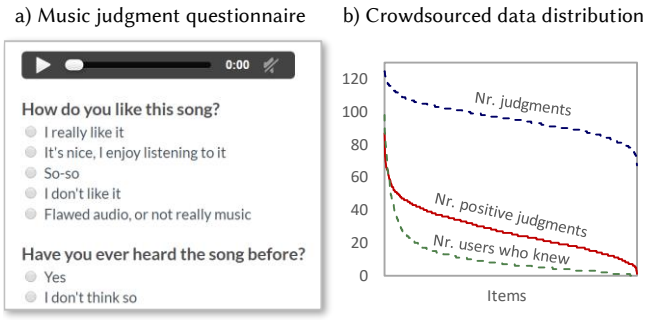

Figure 1: Music track judgment questionnaire (left) and data distribution in the obtained dataset (right). We take the top two answers to the first question in the judgment form ("how do you like this song") as indicating relevance, and the next two as non-relevance. We used the last answer ("flawed audio") to help curate the set of 1,000 sampled tracks and discard flawed ones, as well as to filter out unreliable users (we intentionally introduce flawed music at a random position every 12 tracks and discard users who fail to properly identify it). The questionnaire does not show the song title or artist in order to get as much spontaneous and unbiased answers from users as possible. In the data distribution graph (right), the x axis for each curve is sorted by decreasing value of the y coordinate (each curve has therefore a different order in the x axis).

Now we use this offline dataset to reproduce an online recommendation scenario as follows. The judgments for music that users declare having already heard before can be taken to reasonably represent ratings that users might have entered spontaneously in a system, had they come to find such items within such a system. These judgments therefore make up a reasonable representation of the input data that a recommender system is commonly supplied with. And the remaining judgments, for music that users had never heard before the survey, can be used as relevance judgments for evaluation. These relevance judgments thus apply to unrated items, the ones that are recommendable for each user. This relevance knowledge is not complete: our crowdsourced survey only covers about 10% of all items for each user. But since the user-item pairs are sampled uniformly at random, the judgments provide an unbiased estimate of the full relevance information.

Furthermore, to represent the design of an offline experiment, we randomly split the rating data into training and test subsets, with a ratio $\rho \in (0,1]$ of training data. The recommendation algorithms are only supplied with the training ratings, and the test data are put together with the unrated item judgments to form the set of relevance judgments for evaluation.

Note that the higher the training ratio ρ, the more items shall be discarded from recommendations (because of having training ratings for more users). Thus ρ sets the transition from an offline setting with different split ratios, to an online setting experiment

[1] The dataset is available at http://ir.ii.uam.es/cm100k.
[2] http://crowdflower.com.
[3] http://deezer.com.

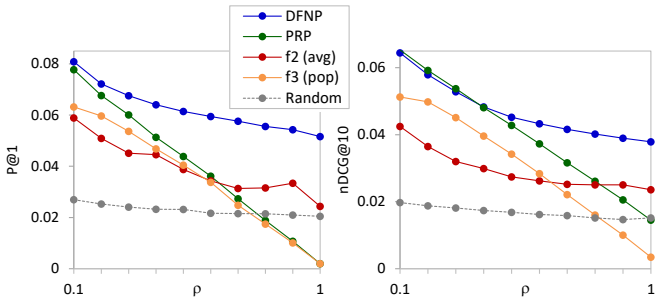

Figure 2: Experiment results. The curves show the evolution of the recommendations accuracy for different rating data split ratios by steps of 0.1. For f_2 (average rating) we use Dirichlet smoothing with $\mu = 1$ in the probability estimation, as it is highly sensitive to the large variance of the average value in the items with fewest ratings. The results are averaged over 100 repetitions of the data splits.

at $\rho = 1$ where no available input is spared for evaluation. We use this possibility to test and observe how the experiment results may change through this transition, and see in particular how the outcome of online vs. offline experiments may agree or differ.

Fig. 2 shows the results for ρ ranging from 0.1 to 1. Alongside the non-personalized recommendations, we evaluate the PRP and DFNP as oracle rankings that are given access to all the available relevance information. We see that for low values of ρ, the PRP and DFNP are not far from each other. However, for higher values of ρ the disagreement grows considerably due to the increasing effect of item exclusion, and reaches a quite extreme point at $\rho = 1$. We see that the PRP completely fails to represent an optimal ranking at $\rho = 1$, to the point of being even substantially worse than a random recommendation. In contrast, the DFNP seems quite robust to the split ratio. A general decrease in precision with the split ratio for DFNP, as for any recommendation, is natural since increasing ρ means preserving less positive relevance judgments for evaluation (which are left as training data).

The non-personalized attempts at approximating the optimal ranking seem to be somewhat effective for low values of ρ, but are increasingly ineffective for higher split ratios. Popularity-based recommendation (f_3) seems to follow the PRP rather than the DFNP ranking. The f_1 ranking gives almost equal results to popularity, and is hence omitted from the figure. The lack of difference between f_1 and f_3 is due to the variations in $p(\neg rated|R_k)$ being negligible in relative terms, compared to the differences in $p(rel, rated|R_k)$ between items. In contrast, recommendation by average rating seems to be more robust and consistent than the popularity ranking to variations in the split ratio, and possibly a better approximation to the DFNP. It is the only ranking that stands above random recommendation for $\rho = 1$.

The poor outcome for PRP, and the rankings that seem to follow it, is due to the fact that the top few music tracks that most people like in the survey ("I will survive" by Gloria Gaynor, Beethoven's "Fur Elise", Mozart's "Rondo alla Turca") are known to almost everyone who was asked to judge them. As a consequence, the only users left for whom the items are not excluded are mainly those who were not asked to judge them. Since we take the absence of judgment as non-relevance, this badly hurts the performance of the PRP ranking. This may be to some extent unfair, as these items might actually please some users for whom we have no judgment. However, these users might in fact already know the items if they were asked, and again, the items would be excluded. Further research would be needed to try to elucidate what is the true situation.

Be that as it may, it becomes clear that the PRP is vulnerable to the overlap between relevance and rating, and can largely diverge from an optimal ranking when these two conditions strongly correlate (i.e. when relevance mostly falls on rated items).

6 CONCLUSIONS

We have found that the common recommender system task, where items should not be recommended to users who have already discovered them, motivates a revision of the Probability Ranking Principle [7]. Our analysis finds a simple principle for the optimal ranking in this context. We empirically confirm the divergence between this principle and the PRP in a small experiment, where the former shows a more consistent behavior over variations in the experimental setting for recommender system evaluation.

Recent research in the field has shown that most collaborative filtering algorithms are biased towards recommending popular items [2,6]. More recently, we found that certain algorithms are rather biased to the average rating instead [3], and such algorithms apparently show worse results in common experiments on public datasets. Interestingly, our present exploration raises the question whether the average rating might be a better signal than the number of positive ratings under certain experimental conditions, incidentally the ones that may more closely represent a live setting. This may call for a second look at the outcomes of offline experiments, under the light of further angles in the experimental design, involving e.g. the relevance judgment collection procedure, or reproducing the conditions of an online setting.

ACKNOWLEDGMENTS

This work was partially supported by the national Spanish Government (grant nr. TIN2016-80630-P).

REFERENCES

[1] G. Adomavicius and A. Tuzhilin. 2005. 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 (June 2005), 734–749.

[2] A. Bellogín, P. Castells, and I. Cantador. Statistical Biases in Information Retrieval Metrics for Recommender Systems. *Information Retrieval*, in press.

[3] R. Cañamares and P. Castells. 2017. A Probabilistic Reformulation of Memory-Based Collaborative Filtering – Implications on Popularity Biases. In *Proceedings of the 40th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (SIGIR 2017).

[4] P. Cremonesi, Y. Koren, and R. Turrin. 2010. Performance of recommender algorithms on top-n recommendation tasks. In *Proceedings of the 4th ACM Conference on Recommender Systems* (RecSys 2010), 39–46.

[5] N. Fuhr. 2008. A probability ranking principle for interactive information retrieval. *Information Retrieval* 11, 3 (June 2008), 251–265.

[6] D. Jannach, L. Lerche, I. Kamehkhosh, and M. Jugovac. 2015. What recommenders recommend: an analysis of recommendation biases and possible countermeasures. *User Modeling and User-Adapted Interaction* 25, 5 (Dec 2015), 427–491.

[7] S. E. Robertson. 1977. The Probability Ranking in IR. *Journal of Documentation* 33, 4 (Jan 1977), 294–304.

[8] G. Shani and A. Gunawardana, 2015. Evaluating Recommendation Systems. In: *Recommender Systems Handbook, 2nd edition*, F. Ricci, L. Rokach, and B. Shapira (Eds.). Springer, New York, NY, USA, Chapter 8, 265–308.

[9] M. Wechsler and P. Schäuble. 2000. The Probability Ranking Principle Revisited. *Information Retrieval* 3, 3 (Oct 2000), 217–227.

[10] C. Zhai and J. Lafferty. 2006. A risk Minimization Framework for Information Retrieval. Information Processing & Management 42, 1 (Jan 2006), 31–55.