On Over-Specialization and Concentration Bias of Recommendations: Probabilistic Neighborhood Selection in Collaborative Filtering Systems

Panagiotis Adamopoulos
padamopo@stern.nyu.edu

Alexander Tuzhilin
atuzhili@stern.nyu.edu

Department of Information, Operations, and Management Sciences
Leonard N. Stern School of Business, New York University

ABSTRACT

Focusing on the problems of over-specialization and concentration bias, this paper presents a novel probabilistic method for recommending items in the neighborhood-based collaborative filtering framework. For the probabilistic neighborhood selection phase, we use an efficient method for weighted sampling of k neighbors that takes into consideration the similarity levels between the target user (or item) and the candidate neighbors. We conduct an empirical study showing that the proposed method increases the coverage, dispersion, and diversity reinforcement of recommendations by selecting diverse sets of representative neighbors. We also demonstrate that the proposed approach outperforms popular methods in terms of item prediction accuracy, utility-based ranking, and other popular measures, across various experimental settings. This performance improvement is in accordance with ensemble learning theory and the phenomenon of "hubness" in recommender systems.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval - Information filtering, Selection process; H.4.m [**Information Systems Applications**]: Miscellaneous

Keywords

Collaborative Filtering; Probabilistic Neighborhood Selection; k-PN; Concentration Bias; Over-Specialization; Diversity; Mobility; Popularity Reinforcement; Long Tail

1. INTRODUCTION

Even though the broad social and business acceptance of recommender systems (RSes) has been achieved, a key underexplored dimension for further improvement is admittedly the usefulness of recommendations. Common recommenders, such as collaborative filtering (CF) algorithms, recommend products based on prior sales and ratings. Hence, they tend not to recommend products with limited historical data, even if these items would be rated favorably by the users. Therefore, RSes can create a rich-get-richer effect for popular items while this *concentration bias* can prevent what may otherwise be better consumer-product matches [19]. At the same time, common RSes usually recommend items very similar to what the users have already purchased or liked in the past [1]. However, this *over-specialization* of recommendations is often inconsistent with sales goals and consumers' preferences.

Aiming at alleviating the important problems of over-specialization and concentration bias and enhancing the usefulness of collaborative filtering RSes, we propose to generate recommendation lists based on a probabilistic neighborhood selection approach. In particular, we aim at providing personalized recommendations from a wide range of items in order to escape the obvious and expected recommendations, while avoiding significant predictive accuracy loss.

In this paper, we present a significant improvement of the classical k-NN method in which the estimation of an unknown rating of a user for an item is based not on the weighted average of the ratings of the k most similar (nearest) neighbors but on k probabilistically selected neighbors. The key intuition for this *probabilistic nearest neighbors* (k-PN) collaborative filtering method and for selecting *diverse neighbors* is three-fold. First, using the neighborhood with the most similar users to estimate unknown ratings and recommend candidate items, the generated recommendation lists usually consist of known items with which the users are already familiar. Second, because of the multi-dimensionality of user preferences, there are many items that the target user may like and are unknown to her k most similar users. Third, selecting very similar neighbors might have a detrimental effect on the performance of a model since such neighbors tend to capture the same predictive signals and information. Thus, we propose the use of a probabilistic neighborhood selection approach in order to alleviate the aforementioned problems of over-specialization and concentration bias and move beyond the limited focus of rating prediction accuracy.

To empirically evaluate the proposed approach, we conduct an experimental study and show that our method indeed alleviates the common problems of concentration bias and over-specialization by selecting diverse sets of neighbors. It also outperforms popular approaches by a wide margin, in terms of item prediction accuracy measures, across various experimental settings. Besides, we demonstrate that this performance gain is combined with further enhancements of other popular performance measures.

RecSys'14, October 6–10, 2014, Foster City, Silicon Valley, CA, USA.

ACM 978-1-4503-2668-1/14/10 ...$15.00.
http://dx.doi.org/10.1145/2645710.2645752.

Finally, the performance improvement can be attributed to carefully selecting diverse sets of representative neighbors. This is not only captured in the works and ideas of ancient philosophers and essayists, such as Plutarch, but it can also be theoretically motivated by the phenomenon of "hubness" and the ensemble learning theory and, in particular, the reduction of covariance among the selected neighbors and the more equal distribution of the number of times each user (or item) is included in a neighborhood.

In summary, the main contributions of this paper are:

- We propose a probabilistic neighborhood-based method (k-PN) as an improvement of the k-NN approach.
- We formulate the classical neighborhood-based CF approach as an ensemble method showing the potential suboptimality of k-NN in terms of predictive accuracy.
- We empirically show that the proposed method outperforms, by a wide margin, the classical CF algorithm and practically illustrate the suboptimality of k-NN in addition to providing theoretical justification.
- We show that the proposed k-PN method alleviates the common problems of over-specialization and concentration bias of recommendations, in terms of various popular metrics and a new metric that measures diversity reinforcement (mobility of recommendations).
- We identify a particular implementation of the k-PN method that performs consistently well across various experimental settings.

2. RELATED WORK

Since the introduction of the first CF systems in the mid-'90s [21, 26], there have been many attempts to improve their performance focusing primarily on error metrics [15]. Even though the rating prediction perspective is the prevailing paradigm in RSes, there are other perspectives that have been gaining significant attention in the field and try to alleviate problems pertaining to the narrow rating prediction focus [2]. This narrow focus has been evident in laboratory studies and real-world online experiments, which indicated that higher predictive accuracy does not always correspond to higher levels of user-perceived quality or increased sales [14, 29]. Two of the most important problems related to this narrow focus of many RSes that have been identified in the literature and hinder the user satisfaction are the over-specialization and concentration bias of recommendations.

Focusing on these two problems, various streams of research identify and discuss the phenomenon, discover its implications, and suggest methods in order to alleviate it. Aiming at verifying and measuring over-specialization bias, [31] employs a longitudinal data set and finds that RSes indeed expose the users to narrowing sets of items over time. Similarly, regarding the concentration bias of recommendations, [25] compares different RS algorithms with respect to aggregate diversity and their tendency to focus on certain parts of the product spectrum and shows that many popular algorithms may lead to an undesired popularity boost of already popular items. However, [24, 27] maintain that whether over-specialization and concentration bias will be enhanced or alleviated depends on the applied personalization technology. Thus, appropriate technical solutions are still needed in order to alleviate these problems of RSes.

Discussing the business implication of over-specialization and concentration, [19] shows that these phenomena direct users towards a common experience, in contrast to the potential goals of RSes, and lead to a significant reduction in profits and sales diversity. Similar implications have also been observed in various other domains (e.g. [17]).

Finally, a stream of research in RSes attempts to alleviate these problems by proposing technical methods. Over-specialization is often practically addressed by injecting randomness in the recommendation procedure [10], filtering out items that are too similar to items the user has rated in the past [11], or increasing the individual diversity of recommendations [39]. Interestingly, [34, 35] present an inverted neighborhood model, k-furthest neighbors, to identify less ordinary neighborhoods for the purpose of creating more diverse recommendations by recommending items disliked by the least similar users. Finally, the concentration problem is typically addressed by re-ranking the list of candidate items taking into consideration their popularity [8] or using sophisticated graph-theoretic approaches [7].

3. MODEL

In this section, we present the proposed approach in the context of the classical user-based collaborative filtering (CF) method. However, the proposed approach is not specific to this algorithm and can be easily extended to any neighborhood-based CF method, including item-based approaches.

User-neighborhood based recommendation methods predict the rating $r_{u,i}$ of user u for item i using the ratings given to i by users most similar to u, called nearest neighbors and denoted by $\mathcal{N}_i(u)$. Taking into account the fact that the neighbors can have different levels of similarity, $w_{u,v}$, and considering the k users v with the highest similarity to u, the predicted rating is:

$$\hat{r}_{u,i} = \bar{r}_u + \frac{\sum_{v \in \mathcal{N}_i(u)} w_{u,v} * (r_{v,i} - \bar{r}_v)}{\sum_{v \in \mathcal{N}_i(u)} |w_{u,v}|}, \quad (1)$$

where $\bar{r}_u$ is the average of the ratings given by user u. However, the ratings given by the nearest neighbors of user u can be combined into a single estimation using various combining (or aggregating) functions [9]. Examples of such functions include majority voting, distance-moderated voting, adjusted weighted average, and percentiles [5].

In the same way, the neighborhood used for estimating the unknown ratings and recommending items can be formed in different ways. Instead of using the k users with the highest similarity to the target user, any approach or procedure that selects k of the candidate neighbors can be used, in principle.

In this paper, we propose a novel k-NN CF method (k-PN) using a *probabilistic neighborhood selection technique that*, instead of the most similar neighbors, *systematically selects a set of diverse neighbors in order to alleviate the over-specialization and concentration problems.* The proposed approach uses a general algorithm for efficient sampling [38] that can also take into consideration the similarity levels between the target user and the n candidate neighbors.

3.1 Probabilistic Neighborhood Selection

For the probabilistic neighborhood selection phase of the proposed algorithm, we allow the neighbors to represent the whole spectrum of candidates, while focusing on specific areas of this spectrum. Selecting such diverse neighbors, the proposed method aims at alleviating the problems of over-specialization and concentration bias (see Section 3.2).

In a nutshell, for the neighborhood selection phase of the k-PN approach, an initial weight is assigned to each candi-

date neighbor and then the candidates are sampled, without replacement, proportionally to their assigned weights. These initial weights can be derived based on popular distance metrics (e.g. Cosine similarity, Pearson correlation, etc.), probability distributions, or other strategies and techniques. For instance, in order to use certain probability distributions aiming at specific areas of the spectrum of candidates, the initial weight w_i for each candidate i can be generated using some function of its distance from the target u or its ranking (based on the distance metric) and the corresponding probability density function (e.g. $w_i = \mathcal{P}(\text{rank}_i)$ or $w_i = \mathcal{P}(\text{sim}(u,i))$; for a complete example see Section 4.2). Based on the selection of the initial weights, the algorithm will select different neighborhoods and, thus, generate different recommendations. We should note here that including all the available candidates in a neighborhood, instead of a diverse set, does not alleviate the problems under study, as discussed in Section 3.2.

For implementing the proposed approach, we suggest an efficient method (based on [18, 38]) for weighted sampling of k neighbors without replacement that takes into consideration the similarity levels between the target user and the population of n candidate neighbors. In particular, the set of candidate neighbors at any time is described by values $\{w_1^{'}, w_2^{'}, \ldots, w_n^{'}\}$. In general, if user i is still a candidate for selection, then $w_i^{'} = w_i$ (where w_i is generated as previously described), whereas $w_i^{'} = 0$ if the user has been already selected in the neighborhood and, hence, removed from the set of candidates. Denote the sum of the weights of the first j candidates by $S_j = \sum_{i=1}^{j} w_i^{'}$, where $j = 1, \ldots, n$, and let $Q = S_n$ be the sum of the weights $\{w_i^{'}\}$ of all the candidates. In order to draw a neighbor, choose x with uniform probability from $[0, Q]$ and find l such that $S_{l-1} \leq x \leq S_l$. Then, add l to the neighborhood and remove it from the set of candidates while setting $w_l^{'} = 0$. After a candidate has been selected into the neighborhood, this neighbor is no longer available for later selection.

This method can be easily implemented using a binary search tree having all n candidate neighbors as leaves with values $\{w_1, w_2, \ldots, w_n\}$, whereas the value of each internal node of the tree is the sum of the values of the corresponding immediate descendant nodes. This sampling method requires $O(n)$ initialization operations, $O(k \log n)$ additions and comparisons, and $O(k)$ divisions and random number generations [38]. The suggested method can be used with any distance metric and valid probability distribution including the empirical distribution of users' similarity (see Section 4.2). Algorithm 1 summarizes the method for efficient weighted sampling without replacement [38].

Note that the same approach can also be used for item-based neighborhood methods by simply sampling diverse neighborhoods of items, instead of users.

ALGORITHM 1: Weighted Sampling Without Replacement

Input: Initial weights $\{w_1, \ldots, w_n\}$ of candidates for neighborhood $\mathcal{N}_i(u)$
Output: Neighborhood of user u, $\mathcal{N}_i(u)$

k: Number of users in the neighborhood of user u, $\mathcal{N}_i(u)$
$L(v)$: The left-descendent of node v
$R(v)$: The right-descendent of node v
G_v: The sum of weights of the leaves in the left subtree from node v
Q: The sum of weights of the nodes in the binary tree

Build binary search tree with n leaves labeled $1, 2, \ldots, n$;
Assign to leaves corresponding values $w_1, w_2, \ldots, w_n$;
Associate values G_v with internal nodes;
Set $Q = \sum_{v=1}^{n} w_v$;
Set $\mathcal{N}_i(u) = \emptyset$;
for $j \leftarrow 1$ **to** k **do**
    Set $C = 0$;
    Set $v =$ the root node;
    Set $\mathcal{D} = \emptyset$;
    Select x uniformly from $[0, Q]$;
    repeat
        if $x \leq G_v + C$ **then**
            Set $\mathcal{D} = \mathcal{D} \uplus \{v\}$;
            Move to node/leaf $L(v)$;
        else
            Set $C = C + G_v$;
            Move to node/leaf $R(v)$;
        end
    until *a leaf is reached*;
    Set $\mathcal{N}_i(u) = \mathcal{N}_i(u) \uplus \{v\}$;
    for *each node* $d \in \mathcal{D}$ **do**
        Set $G_d = G_d - w_v$;
    end
    Set $Q = Q - w_v$;
    Set $w_v = 0$;
end

3.2 Theoretical Motivation

In this section, we present the theoretical motivation for the proposed approach and the connections to the phenomenon of "hubness" as well as the ensemble learning theory. In particular, we discuss a major implication of selecting just the most similar candidates (or even all the candidates) as neighbors and we motivate how the proposed method can alleviate the over-specialization and concentration problems without significantly reducing, and even increasing, the predictive accuracy, demonstrating that *similar but diverse neighbors should be used in neighborhood-based methods.*

It should be clear by now that selecting neighborhoods using underlying probability distributions can result in very different recommendations from those generated based on the standard neighborhood-based approaches. For the sake of brevity, we focus on the phenomenon of "hubness" and the effect of selecting diverse neighbors on the predictive accuracy of the proposed approach.

The phenomenon of "hubness" is related to a new aspect of the dimensionality curse and affects the distribution of k-occurrences: the number of times a point occurs among the k nearest neighbors of the other points in a data set, according to some distance measure [32]. This distribution becomes considerably skewed as dimensionality increases, causing the emergence of hubs, that is, points which appear in many more k-NN lists than other points, effectively making them "popular" nearest neighbors. This is an inherent property that depends on the intrinsic, rather than embedding, dimensionality of data and, thus, dimensionality reduction techniques, such as matrix factorization, do not alleviate the problem effectively. For the same reason "hubness" occurs even for small values of k and for all cosine-like measures, such as Pearson correlation, cosine similarity, and adjusted cosine. Besides, "hubness" is unrelated to other data properties like sparsity or skewness of the distribution of ratings [30]. Nevertheless, this phenomenon is part of the problem of concentration bias of recommendations. In particular, [36] shows that hubness reduces coverage and reachability, especially of long-tail items, in both content-based

and CF systems. Thus, these problems can be alleviated by selecting *neighbors other than the most similar to the target.*

Moreover, in order to further theoretically motivate the proposed approach, we focus on ensemble learning theory; a more comprehensive discussion of neighborhood-based methods and ensembles can be found in [4]. According to ensemble learning theory, in addition to the bias and variance of the individual estimators, the generalization error of an ensemble also depends on the covariance between the individuals; an ensemble is controlled by a three-way trade-off. Hence, if two estimators f_i and f_j that are members of the ensemble are positively correlated, then the correlation increases the generalization error, whereas if they are negatively correlated, then the correlation contributes to a decrease in the generalization error. Thus, a diverse set of estimators is preferable for an ensemble.

In the context of neighborhood-based CF methods in RSes, we can conceptualize the i^{th} most similar neighbor to the target user as corresponding to a single estimator f_i that simply predicts the rating of this specific neighbor; the different predictions can then be combined into a single estimation using a combining function. Hence, reducing the aggregated pairwise covariance of the neighbors (estimators) can decrease the generalization error of the model; at the same time, it may increase the bias or variance of the estimators and the generalization error. Therefore, one way to reduce the covariance is not to restrict the k estimators only to the k nearest (most similar) neighbors but to select a *diverse set of neighbors* (estimators).[1,2]

4. EXPERIMENTS

To empirically validate the k-PN method presented in Section 3 and evaluate the generated recommendations, we conduct a large number of experiments on "real-world" data sets and compare our results to different baselines. For an apples-to-apples comparison, the selected baselines include the user-based k-NN CF approach, which we promise to improve in this study. Compared to other popular algorithms, user-based k-NN generates recommendations that suffer less from concentration bias and over-specialization [15, 25] and has also been found to perform well in terms of other performance measures [12, 14, 3, 6]. Nevertheless, the proposed approach can be applied to any neighborhood-based method and it is not specific to the user-based approach, which has been selected for increased compatibility as well as interpretability of the results. Additionally, we also compare our results against furthest neighbors models (k-FN) [34, 35]. Finally, we also compare our experimental results against matrix factorization (MF) [20].

4.1 Data Sets

The data sets that we used are the MovieLens [22] and MovieTweetings [16] as well as a snapshot from Amazon [28]. The RecSys HetRec 2011 MovieLens (ML) data set [22] contains 855,598 ratings (on a 1-5 scale) from 2,113 users on 10,197 movies. Moreover, the MovieTweetings (MT) data set is described in [16] and consists of ratings included in well-structured tweets on Twitter. Owing to the extreme sparsity of the data set, we decided to condense the data set in order to obtain more meaningful results from collaborative filtering algorithms. In particular, we removed items and users with fewer than 10 ratings. The resulting data set contains 12,332 ratings (on a 0-10 scale) from 839 users on 836 movies. Finally, the Amazon (AMZ) data set is described in [28] and consists of reviews of fine foods during a period of more than 10 years. After removing items with fewer than 10 ratings and reviewers with fewer than 25 ratings each, the data set consists of 15,235 ratings (on a 1-5 scale) from 407 users on 4,316 items.

Table 1: Probability Distributions and Density Functions for Neighborhood Selection.

Label	Probability Distribution	Probability Density Function (weights)	Location and Shape Parameters	
k-NN	-	$\begin{cases}1/k, \text{if } x \leq n-k \\ 0, \text{otherwise}\end{cases}$	-	
E	Empirical Similarity	$w_x / \sum_{i=1}^{n} w_i$	-	
U	Uniform	$1/n$	-	
N_1 N_2	Normal	$\frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$	$\mu = 0$ $\mu = (2.0/15.0)n$	$\sigma = (0.25/15.0)n$ $\sigma = (0.5/15.0)n$
Exp_1 Exp_2	Exponential	$\lambda e^{-\lambda x}$	$\lambda = 1/k$ $\lambda = 2/k$	
W_1 W_2	Weibull	$\frac{\mu}{\lambda}\left(\frac{x}{\lambda}\right)^{\mu-1} e^{-(x/\lambda)^{\mu}}$	$\mu = 0.25$ $\mu = 0.50$	$\lambda = n/20$ $\lambda = n/20$
FN_1 FN_2	Folded normal	$\frac{\sqrt{2}}{\sigma\sqrt{\pi}} e^{-\frac{\theta^2}{2}} e^{-\frac{x^2}{2\sigma^2}} \cosh\left(\frac{\theta x}{\sigma}\right)$	$\theta = 1$ $\theta = 1$	$\sigma = k$ $\sigma = k/2$
k-FN	-	$\begin{cases}1/k, \text{if } x \geq n-k \\ 0, \text{otherwise}\end{cases}$	-	

4.2 Experimental Settings

Using the ML, MT, and AMZ data sets, we conducted a large number of experiments and compared the results against the standard user-based k-NN approach, different k-FN methods, and matrix factorization. In order to test the proposed approach of probabilistic neighborhood selection under various experimental settings, we used different sizes of neighborhoods ($k \in \{20, 30, \ldots, 80\}$) and different probability distributions ($\mathcal{P} \in \{$normal, exponential, Weibull, folded normal, uniform$\}$), with various specifications (i.e. location and scale parameters), as well as the empirical distribution of user similarity, described in Table 1. The uniform distribution is used in order to compare the proposed method against randomly selecting neighbors. The specific distributions were selected because they focus on different areas of the spectrum of candidate neighbors and they constitute common but flexible examples that can be easily reproduced. Additionally, we used two k-FN models [34, 35]; the second furthest neighbor model (k-FN$_2$) employed in this study corresponds to recommending the least liked items of the furthest neighbors instead of the most liked ones (k-FN$_1$). We should note here that because of the strict deterministic nature of both k-NN and k-FN, it is not possible to interpolate between these two methods and select diverse neighbors that approximate the results of k-PN. In addition, we generated recommendation lists of different sizes ($l \in \{1, 3, 5, 10, 20, \ldots, 100\}$). In summary, we used 3 data sets, 7 different sizes of neighborhoods, 12 probability distributions, and 13 different lengths of recommendation lists, resulting in $3,276$ experiments in total.

For the probabilistic neighborhood selection, we used the method described in Section 3.1. In order to estimate the initial weights $\{w_i\}$ of the procedure, we used the probability density functions illustrated in Table 1. Without loss of gen-

[1] Let $r_{u,i}$ and $r_{u,j}$ be the correlation of target user u and candidate neighbors i and j respectively, then the correlation $r_{i,j}$ of neighbors i and j is bounded by the following expression: $r_{u,i}r_{u,j} - \sqrt{1-r_{u,i}^2}\sqrt{1-r_{u,j}^2} \leq r_{i,j} \leq r_{u,i}r_{u,j} + \sqrt{1-r_{u,i}^2}\sqrt{1-r_{u,j}^2}$.

[2] For a formal argument why the proposed probabilistic approach can result in very different recommendations from those generated based on the standard k-NN approach and how the item predictive accuracy can be affected, a 0/1 loss can be used in the context of classification ensemble learning. For a rigorous derivation of the generalization error in ensemble learning using the bias-variance-covariance decomposition and a 0/1 loss function see [33, 37].

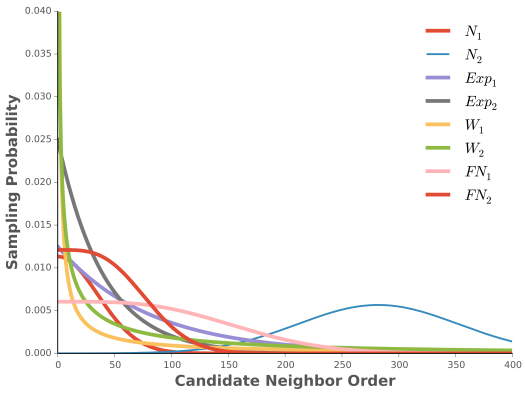


Figure 1: Sampling probability for the nearest candidate neighbors using different probability distributions for the ML data set.

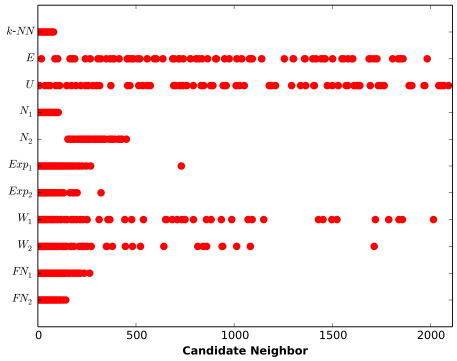


Figure 2: Sampled Neighborhoods using the different probability distributions for the MovieLens data set.

erality, in order to take into consideration the similarity levels of the candidate neighbors, the candidates can be ordered and re-labeled such that $s_{u,1} \geq s_{u,2} \geq \ldots \geq s_{u,n}$, where $s_{u,j}$ is the similarity level of target u and candidate j based on some distance metric. Then, the initial weight w_j for each candidate can be generated using its ranking and a probability density function. For instance, using the Weibull probability distribution (i.e. W_1 or W_2), the weight of the most similar candidate (i.e. $j = 1$) is $w_1 = \frac{\mu}{\lambda}\left(\frac{1}{\lambda}\right)^{\mu-1} e^{-(1/\lambda)^{\mu}}$, where μ and λ are the shape and scale parameters of the distribution and n is the total number of all the candidate neighbors.[3] In contrast to the deterministic k-NN and k-FN approaches, depending on the parameters of the employed probability density function, this candidate neighbor (i.e. the most similar to the target) may or may not have the highest weight w_j.[4] Figure 1 shows the likelihood of sampling each candidate neighbor using different probability distributions for the MovieLens data set and $k = 80$ and Figure 2 shows the sampled neighborhoods for a randomly selected target user using the different distributions; the candidate neighbors for each target user and item in the x axis are ordered based on their similarity to the target user with 0 corresponding to the nearest (i.e. most similar) candidate. As we can see, the selected distributions focus on different areas of the spectrum of candidate neighbors. We should note here that using the empirical distribution of user similarity resulted in *more diverse neighborhoods*.

In all the conducted experiments, in order to measure the similarity among the candidate neighbors, we used the Pearson correlation; similar results were also obtained using the cosine similarity. Also, we used significance weighting as in [23], in order to penalize for similarity based on few common ratings, and filtered any candidate neighbors with zero weight [15]. For the similarity estimation of the candidates in the k-furthest neighbor algorithm, we used the approach described in [34, 35]. Besides, we used the standard combining function as in Eq. (1). Similar results were also obtained using a combining function without a first-order bias approximation: $\hat{r}_{u,i} = \sum_{v \in \mathcal{N}_i(u)} w_{u,v} r_{v,i} / \sum_{v \in \mathcal{N}_i(u)} |w_{u,v}|$;

[3]For continuous probability distributions, the cumulative distribution function can also be used such as $w_i = F(i + 0.5) - F(i - 0.5)$ or $w_i = F(i) - F(i - 1)$.

[4]For a probabilistic furthest neighbors model the candidates can be ordered in reverse similarity order such that $s_{u,1} \leq s_{u,2} \leq \ldots \leq s_{u,n}$. Initial experiments illustrated that such models underperform the proposed approach.

any differences are explicitly discussed in the following section. In addition, we used a holdout validation scheme in all of our experiments with 80/20 splits of the rating tuples to the training/test parts in order to avoid overfitting. Finally, the evaluation of the various approaches in each experimental setting is based on users with more than k candidate neighbors, where k is the corresponding neighborhood size; if a user has k or fewer available candidate neighbors, then the same neighbors are always selected and the results for the specific user are in principle identical for all the examined approaches, apart from the inverse k-FN (k-FN$_2$) method. Similarly, the generated recommendation lists were also evaluated using a subset of the test set containing only highly rated items as well as only long-tail items [13].

5. RESULTS

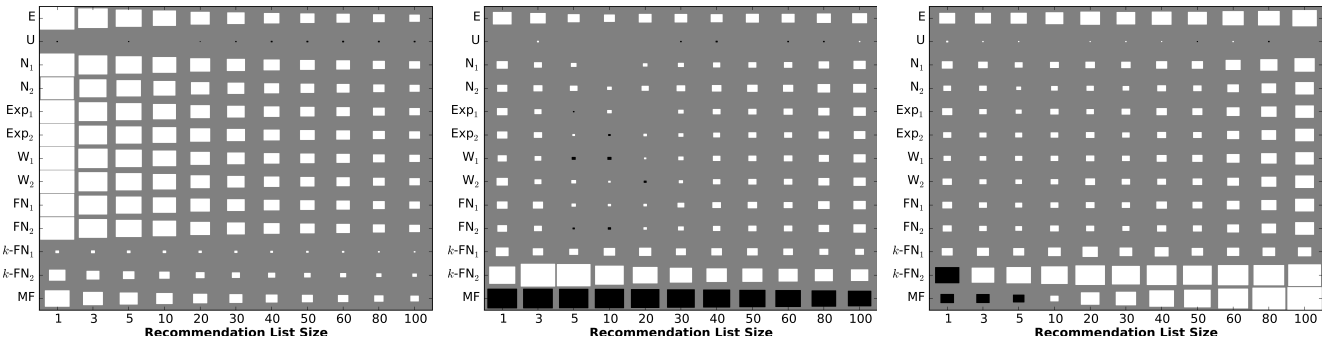
The aim of this study is to demonstrate that the proposed method indeed effectively generates recommendations that alleviate the over-specialization and concentration problems while performing well in terms of other important metrics of RSes. Therefore, we conduct a comparative analysis of our method and the standard baseline (k-NN), matrix factorization, and the k-furthest neighbor approaches, in different experimental settings.

Given the number and the diversity of experimental settings, the presentation of the results constitutes a challenging problem. A reasonable way to compare the results across the different settings is by computing the relative performance differences and discussing only the most interesting dimensions. Due to space limitations, detailed results, supplementary graphs and tables, and tests of statistical significance about all the conducted experiments as well as additional performance metrics measuring orthogonality of recommendations and predictive accuracy are included in [4].

Overall, *the proposed method generates recommendations that are very different from the classical CF approaches and alleviates the over-specialization and concentration problems*, based on metrics of coverage, dispersion, and diversity reinforcement (mobility of recommendations), while *avoiding any significant accuracy loss.* Fig. 3 illustrates the aforementioned findings. It shows an overview of the performance of all the methods on the ML data set across various metrics for recommendation lists of size $l = 10$.

5.1 Coverage and Diversity

In this section, we investigate the effect of the proposed method on coverage and aggregate diversity, two important metrics which in combination with other measures discussed in this study show whether the proposed approach alleviates



(a) MovieLens (ML) (b) MovieTweetings (MT) (c) Amazon (AMZ)

Figure 4: Increase in aggregate diversity performance for the different data sets and recommendation list sizes.

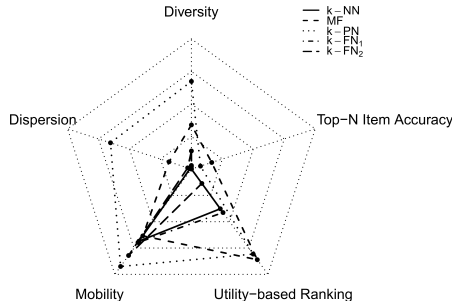


Figure 3: Summary of performance for the ML data set.

the over-specialization and concentration bias problems of common RSes. The results obtained using the *catalog coverage* metric are equivalent to those using the *diversity-in-top-N* metric for aggregate diversity; henceforth, only one set of results is presented. Fig. 4 presents the results obtained by applying our method to the ML, MT, and AMZ data sets. In particular, the Hinton diagram in Fig. 4 shows the percentage increase/decrease in performance compared to the k-NN baseline for each probability distribution and recommendation lists of size $l \in \{1, 3, 5, 10, 20, \ldots, 100\}$ over seven neighborhood sizes, $k \in \{20, 30, \ldots, 80\}$. Positive and negative values are represented by white and black squares, respectively, and the size of each square represents the magnitude of each value.

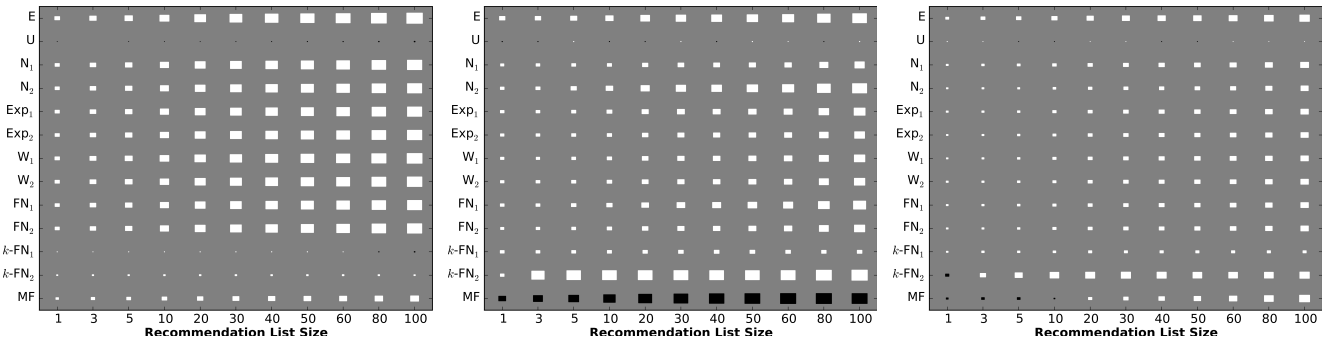
Fig. 4 demonstrates that *the proposed method in most cases performs better than the user-based k-NN, matrix factorization, and the k-FN methods*. The more diverse recommendations were achieved using the empirical distribution of user similarity and the inverse k-furthest neighbors approach (k-FN$_2$). In particular, the average aggregate diversity across all the probability distributions, neighborhoods, and recommendation list sizes was 22.10%, 46.09%, and 13.52% for the ML, MT, and AMZ data sets, respectively; the corresponding diversity using only the empirical distribution was 24.20%, 50.55%, and 17.04% for the different data sets. The corresponding performance of MF [20] was measured as 14.17%, 10.70%, and 14.39%, respectively.

Furthermore, the performance was increased both in the experiments where the k-NN method, because of the specifics of the particular data sets, resulted in low aggregate diversity (e.g. Amazon) and high diversity performance (e.g. MovieTweetings). In addition, the experiments conducted using the same probability distribution (e.g. Exp_1 and Exp_2) exhibit very similar performance. As one would expect, in most cases the aggregate diversity increased, whereas the magnitude of the difference in performance decreased, with increasing recommendation list size l. Without using the first-order bias approximation in the combining function, the standard k-NN method resulted in higher aggregate diversity and catalog coverage but the proposed approach still outperformed the classical algorithm in most of the cases by a narrower margin; using the inverse k-FN method (k-FN$_2$) without the first-order bias approximation resulted in decrease in performance for the Amazon data set. The performance of empirical distribution was 33.37%, 61.06%, and 41.50% for the different data sets. Nevertheless, *using all the candidates, instead of probabilistically selecting a diverse neighborhood, underperforms the proposed approach* since the overall contribution of neighbors other than the most similar is significantly discounted.

In terms of statistical significance, using the Friedman test and performing post hoc analysis, the differences among the employed baselines (i.e. k-NN, MF, k-FN$_1$, and k-FN$_2$) and all the proposed specifications are statistically significant ($p < 0.001$) for the ML data set. For the MT and AMZ data sets, all the proposed specifications (i.e. E, N_1, N_2, Exp_1, Exp_2, W_1, W_2, FN_1, and FN_2) significantly outperform the k-NN and matrix factorization algorithms; the empirical distribution significantly outperforms also the k-FN$_1$ method.

5.2 Dispersion and Diversity Reinforcement

In order to conclude whether the proposed approach alleviates the over-specialization and concentration bias, the generated recommendation lists should also be evaluated for the inequality across items using the Gini coefficient. Fig. 5 shows the percentage increase (white squares) or decrease (black squares) in dispersion of recommendations compared to the k-NN baseline. The Gini coefficient was on average improved by 6.81%, 3.67%, and 1.67% for the ML, MT, and AMZ data sets, respectively; the corresponding figures using only the empirical distribution were 7.48%, 6.73%, and 3.45% for the different data sets, which implies an improvement of 7.41%, 16.76%, and 1.54% over MF and 9.69%, 5.34%, and 2.84% over k-FN. The more uniformly distributed recommendation lists were achieved using the empirical distribution of user similarity and the inverse k-FN approach. Moreover, the larger the size of the recommendation lists, the larger the improvement in the Gini coefficient. Similarly, without using the first-order bias approximation in the rating combining function, the average dispersion was further improved by 6.48%, 6.83%, and 20.22% for the ML, MT, and AMZ data sets, respectively. This implies an improvement of 14.91%, 22.83%, and 21.19% against MF and 16.94%, 5.90%, and 2.38% against k-FN. As we can conclude, *in the*



(a) MovieLens (ML) (b) MovieTweetings (MT) (c) Amazon (AMZ)

Figure 5: Increase in dispersion of recommendations for the different data sets and recommendation list sizes.

recommendation lists generated from the proposed method, the number of times an item is recommended is more equally distributed compared to other CF methods. In terms of statistical significance, all the proposed specifications (apart from the N_1, Exp_2, and FN_2 for the MT data set and the N_2, Exp_2 for the AMZ data set) significantly outperform the k-NN, matrix factorization, and k-FN$_1$ methods ($p < 0.001$). The empirical distribution also significantly outperforms the k-FN$_2$ method for the ML data set; the differences are not statistically significant for the other data sets.

However, simply evaluating the recommendation lists in terms of dispersion and inequality does not provide any information about the (popularity-based) diversity reinforcement and mobility of the recommendations (i.e. whether popular or long-tail items are more likely to be recommended) since these metrics do not consider the prior state of the system. Hence, we employ a *diversity reinforcement* measure M to assess whether the proposed recommender system approach follows or changes the prior popularity of items when recommendation lists are generated. Thus, we define M, which equals the proportion of items that are "mobile" (e.g. changed from popular in terms of number of ratings to "long tail" in terms of recommendation frequency), as follows:

$$M = 1 - \sum_{i=1}^{K} \pi_i \rho_{ii}$$

where the vector π denotes the initial distribution of each of the K (popularity) categories and ρ_{ii} the probability of staying in category i, given that i was the initial category.[5] A score of zero denotes no change (i.e. the number of times an item is recommended is proportional to the number of ratings it has received), whereas a score of one denotes that the RS recommends only the long-tail items (i.e. the number of times an item is recommended is proportional to the inverse of the number of ratings it has received).

In the conducted experiments, based on the 80-20 rule or Pareto principle, we use two categories, labeled as "head" and "tail", where the former category contains the top 20% of items (in terms of ratings or recommendations frequency) and the latter category the remaining 80%. The experimental results demonstrate that *the proposed method generates recommendation lists that exhibit in most cases higher diversity reinforcement compared to the k-NN, MF, and k-FN methods.* In particular, the performance was increased by 0.91%, 0.95%, and 0.19% for the ML, MT, and AMZ data sets, respectively; the corresponding improvement using only the empirical distribution was 1.29%, 1.46%, and 0.45% for the different data sets, which implies an improvement of 1.52%, 1.12%, and 2.11% over MF and 1.01%, 1.31%, and 0.29% over k-FN. We also note that recommendation lists of larger size resulted on average in even larger improvements. Besides, considering a smaller number of items as popular also resulted in larger improvements. Similarly, without the first-order bias approximation the average diversity reinforcement was further increased by 0.69%, 0.53%, and 3.28% for the ML, MT, and AMZ data sets, respectively. This implies an improvement of 2.40%, 1.82%, and 1.65% against MF and 1.83%, 1.44%, and 1.82% against k-FN. Fig. 6 shows the transition probabilities of each category for recommendation lists of size $l = 100$ using the empirical distribution of similarity and the MovieTweetings data set. In terms of statistical significance, in most of the cases all the proposed specifications significantly outperform the baseline methods ($p < 0.005$) [4].

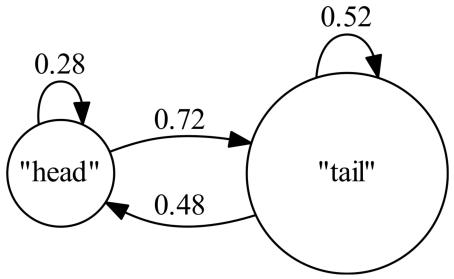


Figure 6: Diversity reinforcement for the MT data set.

5.3 Utility-based Ranking and Accuracy

Further, in order to better assess the quality of the proposed approach, the recommendation lists should also be evaluated for the ranking of the items that are presented to the users, taking into account the rating scale of the selected data sets. Assuming that the utility of each recommendation is the rating of the recommended item discounted by a factor that depends on its position in the list of recommendations, we evaluate the generated recommendation lists based on the normalized Cumulative Discounted Gain (nDCG), where positions are discounted logarithmically; similar results were also obtained for item prediction accuracy.

The highest performance was again achieved using the empirical distribution of user similarity, the normal, or the Weibull distribution. In particular, the average increase of the nDCG score across all the examined probability dis-

[5]The proposed diversity reinforcement score can be easily adapted in order to differentiate the direction of change and the magnitude.

tributions, neighborhoods, and recommendation list sizes was 100.06%, 20.05%, and 89.85% for the ML, MT, and AMZ data sets, respectively; the corresponding increase using only the empirical distribution was 117.65%, 23.01%, and 383.99% for the different data sets resulting on average in a 2-fold increase. The absolute performance of the empirical distribution for the different datasets was 73.54%, 74.62%, and 42.63%, respectively. Even though k-PN on average underperforms MF, it performs very well on both ML and MT, especially given the goals of this method. Without using the first-order bias approximation in the rating combining function, the proposed approach outperformed in most of the cases the classical k-NN algorithm and the k-FN methods by an even wider margin. The same wide margin was also observed focusing on long-tail items, except for MT. In terms of statistical significance, the differences among the employed baselines and all the proposed specifications (apart from the FN_1 for the MT data set and the N_1, Exp_2, W_2, and FN_2 for the AMZ data set) are statistically significant.

6. DISCUSSION AND CONCLUSIONS

In this paper, studying the problems of concentration bias and over-specialization, we present a novel *probabilistic neighborhood selection* method for generating recommendations in CF systems. We illustrate the practical implementation of the proposed approach in the context of memory-based systems adapting and improving the standard k-nearest neighbors (k-NN) method. In the proposed approach, the neighborhood is selected based on an underlying probability distribution, instead of just the k neighbors with the highest similarity level to the target. For the probabilistic neighborhood selection (k-PN) approach, we use an efficient method for weighted sampling of k neighbors that takes into consideration the similarity levels between the target and all the candidate neighbors. In addition, we conduct an empirical study showing that the proposed method, by selecting *diverse representative neighborhoods*, generates recommendations that are very different from the classical CF approaches and alleviates the over-specialization and concentration problems while outperforming k-NN, k-FN, and matrix factorization methods. We also demonstrate that using specific probability distributions the proposed method outperforms, by a wide margin in most cases, both the standard k-nearest neighbors and the k-furthest neighbors approaches in terms of both item prediction accuracy and utility-based ranking. The probabilistic nature of the proposed approach is a virtue since sampling different neighbors at each recommendation instance generates different recommendation lists that possess the same properties. The experimental results are also in accordance with the phenomenon of "hubness" and the ensemble learning theory that we employ in the neighborhood-based CF framework. Besides, we show that the performance improvement is not achieved at the expense of other popular performance measures.

7. REFERENCES

[1] Z. Abbassi, S. Amer-Yahia, et al. Getting recommender systems to think outside the box. In *RecSys '09*. ACM, 2009.

[2] P. Adamopoulos. On discovering non-obvious recommendations: Using unexpectedness and neighborhood selection methods in collaborative filtering systems. In *WSDM '14*. ACM, 2014.

[3] P. Adamopoulos and A. Tuzhilin. On Unexpectedness in Recommender Systems: Or How to Expect the Unexpected. In *DiveRS 2011 at RecSys '11*. ACM, 2011.

[4] P. Adamopoulos and A. Tuzhilin. Probabilistic neighborhood selection in collaborative filtering systems. *Working Paper: CBA-13-04, NYU*, 2013. http://hdl.handle.net/2451/31988.

[5] P. Adamopoulos and A. Tuzhilin. Recommendation opportunities: Improving item prediction using weighted percentile methods in collaborative filtering systems. In *RecSys '13*. ACM, 2013.

[6] P. Adamopoulos and A. Tuzhilin. On unexpectedness in recommender systems: Or how to better expect the unexpected. *ACM TIST*, 2015. http://tist.acm.org/papers/TIST-2012-09-0141.R2.pdf.

[7] G. Adomavicius and Y. Kwon. Maximizing Aggregate Recommendation Diversity: A Graph-Theoretic Approach. In *DiveRS at RecSys '11*. ACM, 2011.

[8] G. Adomavicius and Y. Kwon. Improving aggregate recommendation diversity using ranking-based techniques. *IEEE TKDE*, 24(5):896 –911, 2012.

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

[10] M. Balabanović and Y. Shoham. Fab: content-based, collaborative recommendation. *CACM*, 40(3):66–72, 1997.

[11] D. Billsus and M. J. Pazzani. User modeling for adaptive news access. *UMUAI*, 10(2-3):147–180, 2000.

[12] R. Burke. Hybrid recommender systems: Survey and experiments. *UMUAI*, 12(4):331–370, 2002.

[13] P. Cremonesi et al. Performance of recommender algorithms on top-n recommendation tasks. In *RecSys '10*. ACM, 2010.

[14] P. Cremonesi, F. Garzotto, et al. Looking for "good" recommendations: A comparative evaluation of recommender systems. In *INTERACT '11*. 2011.

[15] C. Desrosiers and G. Karypis. A comprehensive survey of neighborhood-based recommendation methods. In *Recommender systems handbook*. Springer, 2011.

[16] S. Dooms et al. Movietweetings: a movie rating dataset collected from twitter. In *CrowdRec at RecSys '13*, 2013.

[17] J. A. Evans. Electronic publication and the narrowing of science and scholarship. *Science*, 321(5887):395–399, 2008.

[18] R. Fagin and T. G. Price. Efficient calculation of expected miss ratios in the independent reference model. *SICOMP*, 7(3), 1978.

[19] D. Fleder and K. Hosanagar. Blockbuster culture's next rise or fall: The impact of recommender systems on sales diversity. *Manage. Sci.*, 55(5):697–712, 2009.

[20] Z. Gantner, S. Rendle, et al. MyMediaLite: A free recommender system library. In *RecSys '11*. ACM, 2011.

[21] D. Goldberg, D. Nichols, et al. Using collaborative filtering to weave an information tapestry. *CACM*, 35(12):61–70, 1992.

[22] GroupLens research group, 2011. http://www.grouplens.org.

[23] J. L. Herlocker, J. A. Konstan, et al. An algorithmic framework for performing collaborative filtering. In *SIGIR*. ACM, 1999.

[24] O. Hinz, J. Eckert, et al. Drivers of the long tail phenomenon: an empirical analysis. *JMIS*, 27(4):43–70, 2011.

[25] D. Jannach, L. Lerche, et al. What recommenders recommend–an analysis of accuracy, popularity, and sales diversity effects. In *UMAP*. Springer, 2013.

[26] J. A. Konstan, B. N. Miller, et al. Grouplens: applying collaborative filtering to usenet news. *CACM*, 40(3), 1997.

[27] C. Matt, T. Hess, et al. The differences between recommender technologies in their impact on sales diversity. In *ICIS*, 2013.

[28] J. McAuley and J. Leskovec. Hidden factors and hidden topics: understanding rating dimensions with text. In *RecSys'13*, 2013.

[29] S. M. McNee, J. Riedl, et al. Being accurate is not enough: how accuracy metrics have hurt recommend. systems. In *CHI*, 2006.

[30] A. Nanopoulos et al. How does high dimensionality affect collaborative filtering? In *RecSys '09*. ACM, 2009.

[31] T. T. Nguyen, P.-M. Hui, et al. Exploring the filter bubble: the effect of using rec. sys. on content diversity. In *WWW*, 2014.

[32] M. Radovanović, A. Nanopoulos, et al. Hubs in space: Popular nearest neighbors in high-dimensional data. *JMLR*, 2010.

[33] F. Roli and G. Fumera. Analysis of linear and order statistics combiners for fusion of imbalanced classifiers. In *MCS*, 2002.

[34] A. Said, B. Fields, et al. User-centric evaluation of a k-furthest neighbor CF recommender algorithm. In *CSCW*. ACM, 2013.

[35] A. Said, B. J. Jain, et al. Increasing diversity through furthest neighbor-based recommendation. In *DDR at WSDM '12*, 2012.

[36] K. Seyerlehner, A. Flexer, et al. On the limitations of browsing top-n recommender systems. In *RecSys '09*. ACM, 2009.

[37] K. Tumer and J. Ghosh. Error correlation and error reduction in ensemble classifiers. *Connection science*, 8(3-4), 1996.

[38] C.-K. Wong and M. C. Easton. An efficient method for weighted sampling without replacement. *SICOMP*, 9(1):111–113, 1980.

[39] C.-N. Ziegler, S. M. McNee, et al. Improving recommendation lists through topic diversification. In *WWW '05*. ACM, 2005.