Recommendation Diversification Using Explanations

Cong Yu#, Laks V.S. Lakshmanan*, Sihem Amer-Yahia#

#*Yahoo! Research, New York, NY, USA*
{congyu, sihem}@yahoo-inc.com
**University of British Columbia, Vancouver, Canada*
laks@cs.ubc.ca

***Abstract*— We introduce the novel notion of *explanation-based diversification* to address the well-known problem of *over-specialization* in item recommendations. Over-specialization in recommender systems leads to result sets with items that are too similar to one another, thus reducing the diversity of results and limiting user choices. Traditionally, the problem is addressed through *attribute-based diversification*—grouping items in the result set that share many common attributes (e.g., genre for movies) and selecting only a limited number of items from each group. It is, however, not always applicable, especially for social content recommendations. For example, attributes may not be available as in the case of recommending URLs for users of del.icio.us. Explanation-based diversification provides a novel and complementary alternative—it leverages the *reason for which a particular item is being recommended* (i.e., explanation)—for diversifying the results, without the need to access the attributes of the items. In this paper, we formally define the problem of *explanation-based diversification* and, without going into the details of the actual diversification process, demonstrate its effectiveness on a real world data set, Yahoo! Movies.**

I. Introduction

Recommendation is becoming a popular mechanism to help users find relevant information on the Web. The successes of Netflix and Amazon are good testaments to this trend. However, while users enjoy receiving relevant content items, they also tend to loose interest quickly if the recommended items are too similar to each other. As an isolated example, during the heat of the US 2008 Democratic Primary Election, the authors of this paper visited the topic “election” on del.icio.us and were shown, on the first page, URLs all about “Barack Obama,” and none about “Hillary Clinton” or “John McCain.” Such a homogeneous set of “recommendations” could easily turn off the user and lower her interest in the site over time [2]. This problem is commonly referred to as *over-specialization*, which is addressed by *recommendation diversification*.

Intuitively, the goal of *recommendation diversification* is to identify a list of items that are dissimilar, but nonetheless relevant to the user’s interests. To the best of our knowledge, all currently proposed methods for recommendation diversification are *attribute-based* [7], i.e., the diversity of the recommended list is defined by how much each item in the list differs from the others in terms of their attribute values. For example, in [13], diversity is measured based on specific attributes of each item, which is defined by the taxonomy in which the item is classified. In Yahoo! Movies[1], recommended movies are post-processed using a distance measure that is defined on the attributes of each movie, including genre, actor and director.

However, attribute-based diversification has two main drawbacks when applied to social content sites. The first drawback is the *lack of attributes for items*. Unlike their counterparts in traditional recommender systems, items in social content sites often do not have a comprehensive set of attributes associated with them. For example, URLs in del.icio.us are only described by the set of tags associated with them, as are photos in Flickr and videos in YouTube. While they reveal information about the item, tags are provided by individual users and are fundamentally different from the intrinsic descriptions of the items. For example, two URLs with the same tag, “vacation”, could be entirely different in terms of what they are about. Meanwhile, trying to obtain intrinsic properties of these items is computationally prohibitive or impossible with current technologies. For example, for URLs in del.icio.us, one would have to crawl and analyze the actual web page, which is too costly and beyond the purpose and capability of del.icio.us. Similarly, for photos and videos, the technologies for extracting their content features are simply not there yet.

The second issue is the *computational overhead of attribute retrieval for the sole purpose of diversification*. Recommendation algorithms follow two main approaches: content-based and collaborative filtering. The former leverages item attributes to recommend items similar to what the target user likes currently, thus attribute retrieval comes for free when diversification needs to be performed. The latter, however, leverages user activities to recommend items liked by other users similar to the target user. Often, attributes of recommended items are never retrieved, which means there will be extra costs associated with retrieving them when diversification is performed.

To address these two issues, we propose to formalize a notion of diversity based on *recommendation explanations* [2], [8]. Intuitively, explanations are reasons behind why a particular item is recommended to a particular user. For content-based strategies, the explanation for a recommended item is the set of similar items that the user has liked in the past. For collaborative filtering strategies, the explanation for a recommended item is simply a set of users who like this item and who are in the target user’s network (i.e., are friends of the target user or share similar behavior with the target user). The idea for explanation-based diversification is that, given two different

[1] *http://movies.yahoo.com/*

recommended items, the more overlap they have in their set of explanations, the more similar they are with each other, and diversification can be accomplished by identifying items that are ranked highly by the recommendation algorithms but share few common explanations. We emphasize here that this novel diversity notion is *independent of the item attributes*. In particular, for collaborative filtering strategies, no attribute of any item is required for either the recommendation or diversification—a characteristic that is very important in social content recommendation.

We make the following main contributions in this paper. First, we formally define the notion of item explanation for various recommendation strategies (Section III). Second, we formalize the notion of explanation-based diversity for a set of recommended items (Section IV). Third, we conduct an experiment on a real data set (Yahoo! Movies) which shows that explanation-based techniques achieve similar diversification as attribute-based ones (Section V). Finally, we discuss related work and conclude in Sections VI and VII, respectively. We begin with an overview of existing recommendation strategies.

II. Overview of Recommendation Strategies

In this study, we focus on social content sites where users share items online and express their endorsements of items by tagging (e.g., del.icio.us) and rating (e.g., Yahoo! Movies). Users also establish ties with other users through explicit links (e.g., friendship network) or implicit ones (e.g., similar ratings of similar movies). To unify tagging and rating activities, we consider each tagging activity (which may be more than one tag per user and item) as a rating activity. A thorough tag analysis would indicate the nature of the tag and allow mapping of a set of tags to a positive or a negative rating. This study is beyond the scope of this work. For the purpose of this paper, we choose to map tagging to a positive rating. We assume that each social content site consists of a set of users $\mathcal{U}$ and a set of items $\mathcal{I}$.

The goal of a recommendation strategy is to estimate a user's rating for items, and return those items with highest estimated ratings. We explore the two most popular families of recommendation strategies and briefly review them below. These strategies are also referred to as rating-based since they predict ratings as opposed to preference-based, which aim to predict the relative ordering of items. Rating-based strategies rely on either finding items similar to the user's previously highly rated items (content-based or item-based strategies), or finding items highly rated by people who share the user's interests (user-based or collaborative filtering strategies) [2].

A. Item-Based Strategies

These are the oldest recommendation strategies. They aim to recommend items similar to the ones the target user preferred in the past. While different strategies use different approaches to predict ratings, we present one common formulation. The rating of an item $i \in \mathcal{I}$ by the target user $u \in \mathcal{U}$ is estimated as follows:

$$\texttt{relevance}(u, i) = \Sigma_{i' \in \mathcal{I}} \texttt{ItemSim}(i, i') \times \texttt{rating}(u, i')$$

Here, $\texttt{ItemSim}(i, i')$ returns a measure of similarity between two items i and i', and $\texttt{rating}(u, i')$ indicates the rating of item i' by user u (it is 0 if u has not rated i'). Item-based strategies are very effective when the given user has a long history of rating activities. However, it does have two drawbacks. First, items recommended by item-based strategies are often very similar to what the user already knows [7] and therefore other interesting items that the user might like have little chance of being recommended. Second, item-based strategy does not work well when a user first joins the system. To partially address those drawbacks, collaborative filtering strategies have been proposed.

B. Collaborative Filtering Strategies

These strategies aim to recommend to the target user items which are highly rated by users who share similar interests with or have declared relationship with the target user. The key is to find other users connected to the target user. The rating of an item i by the target user u is estimated as follows:

$$\texttt{relevance}(u, i) = \Sigma_{u' \in \mathcal{U}} \texttt{UserSim}(u, u') \times \texttt{rating}(u', i)$$

Here, $\texttt{UserSim}(u, u')$ returns a measure of similarity or connectivity between two users u and u' (it is 0 if u and u' are not connected). Collaborative filtering strategies broaden the scope of items being recommended to the user and have become increasingly popular. Note that in both item-based and collaborative filtering strategies, we use $relevance(u, i)$ to denote the estimated rating of i by u. *In the rest of the paper, the term relevance refers to this estimated rating.*

We note that there are also so-called fusion strategies which combine ideas from item-based and collaborative filtering strategies. While we do not consider them in this paper, it should be straightforward to extend our methods for diversification for those strategies. Another set of recommendation strategies are so-called model-based [2], where machine learning techniques are employed. How to provide explanation and diversification for those strategies is an interesting future topic of study, but beyond the scope of this paper.

III. Recommendation Explanations

We denote by $\texttt{RecItems}(u)$, the set of candidate recommended items generated by one of the recommendation strategies described above. The size of this set is typically larger than the final desired number of recommendations. We let $\texttt{Items(u)}$ denote the set of all items tagged/rated by user u.

The explanation for a recommended item depends on the underlying recommendation strategy used. On one hand, if an item i is recommended to user u by an item-based strategy, then the *explanation* for recommendation i is defined as:

$$\texttt{Expl}(u, i) = \{i' \in \mathcal{I} \mid \texttt{ItemSim}(i, i') > 0 \ \& \\ i' \in \texttt{Items}(u)\}$$

i.e., the set of items similar to i' that user u has rated in the past. We can also augment each item i' with the weight

$\texttt{ItemSim}(i, i') \times \texttt{rating}(u, i')$. One the other hand, if an item i is recommended to user u by a collaborative filtering strategy, then the *explanation* for recommendation i is:

$$\texttt{Expl}(u, i) = \{u' \in \mathcal{U} \mid \texttt{UserSim}(u, u') > 0 \;\&\; i \in \texttt{Items}(u')\}$$

i.e., the set of users similar to u who have rated item i. Similarly, we can augment each user u' with the weight $\texttt{UserSim}(u, u') \times \texttt{rating}(u', i)$.

Note that in both cases, the explanation of a recommendation is either a set of items or a set of users, with an optional weight associated with each member of the set.

IV. Explanation-Based Diversity

Let i, i' be a pair of items (recommended to target user u) with associated explanation sets $\texttt{Expl}(u, i)$ and $\texttt{Expl}(u, i')$. We can now define the *diversity distance* between the two sets of explanations as a similarity measure based on standard metrics such as Jaccard similarity coefficient or cosine similarity. More precisely, the *Jaccard diversity distance* between recommendations i and i' is:

$$DD_u^J(i, i') = 1 - \frac{|\texttt{Expl}(u,i) \cap \texttt{Expl}(u,i')|}{|\texttt{Expl}(u,i) \cup \texttt{Expl}(u,i')|}$$

Note that, as a distance measure, it is defined as the complement of the standard Jaccard coefficient.

When weights are incorporated, the *cosine diversity distance* between recommendations i and i' is defined by treating the explanations $\texttt{Expl}(u, i)$ and $\texttt{Expl}(u, i')$ as vectors in a multi-dimensional space and defining $DD_u^C(i, i')$ as the complement of standard cosine similarity between the vectors. We omit the details and refer the readers to the Vector Space Model [1] for more details. Furthermore, we use the notation $DD_u(i, i')$ as a type-neutral diversity distance measure. Depending on the context, it shall be interpreted as $DD_u^J(i, i')$ or as $DD_u^C(i, i')$.

This distance measure between two recommendations provides the basics for measuring the diversity of a set of recommended items. Formally, for a set of items $S \subseteq \texttt{RecItems}(u)$, we have:

$$DD_u(S) = avg\{DD_u(i, i') \mid i, i' \in S\}$$

i.e., $DD_u(S)$ is the average diversity distance between pairs of items in the set S. We note here that alternative aggregation functions like maximum or minimum could be used in place of, or in addition to, average.

Given this diversity notion, the general diversification problem in top-k recommendation can be defined as:

> Given a user u, find a subset $S \subseteq \texttt{RecItems}(u)$ such that $|S| = k$ and the choice of S strikes a good balance between relevance and diversity.

Here, the term "balance" is intentionally left undefined. It is subject to multiple possible interpretations, and thus implementations. One possible interpretation is to optimize diversity while maintaining relevance above a fixed threshold:

> Given a user u, a threshold θ on the aggregate score of a set of recommended items, find a set $S \subseteq RecItems(u)$ such that $|S| = k$, the aggregate score of S is no less than θ and the diversity distance $DD_u(S)$ is maximized.

How to algorithmically strike a good balance between relevance and diversity for the target user in the recommendation results is a non-trivial subject and beyond the scope of this paper. Further described in the full version of the paper [12]. In the next section, without going into details of the actual diversification algorithms, we experimentally verify the fact that explanation-based diversification is comparable with attribute-based diversification.

V. Evaluation

The goal of the experimental evaluation is two-fold. First, verify the effectiveness of explanation-based diversification by comparing it against attribute-based diversification. Second, evaluate the performance advantage that explanation-based diversification has over the attribute-based diversification.

We use a snapshot of Yahoo! Movies database from 2005 for our experiments. There, users rate movies on a scale of $0-100$ and each movie has a set of associated attributes, including genre, actor, director and writer. We adopt the collaborative filtering strategy and use two diversification algorithms described in [12]: *Algorithm Swap*, which starts with the top-k relevant items and progressively swaps out items that are too close to other higher ranked items, and *Algorithm Greedy*, which starts with the single top relevant item and iteratively constructs the diversified recommendation list by adding items that are of sufficient relevance and distant enough from existing items.

For explanation-based diversification, we adopt $DD_u^C(i, j)$ as the distance measure. For attribute-based diversification, we define the distance between two movies i and j as:

$$\sum_k \frac{1}{|A|} \texttt{Jaccard-Distance}(i_k, j_k)$$

where A is the set of attributes and k iterates over all the attributes. For a missing attribute, we assume the Jaccard distance is the average of the remaining attributes. We ignore movies with no attribute at all.

A. Effectiveness of Explanation-Based Diversity

To verify its effectiveness, we analyze the correlation between results obtained using explanation-based diversification and those obtained from attribute-based diversification. To compare the two generated recommendation lists, we measure both the Jaccard Similarity (i.e., intersection) and K_{min}, a modified Kendall's tau distance (that works with two lists of different sizes) described in [5].

As shown in Table I, our approach returns similar results as attribute-based diversification as measured by both Jaccard similarity and K_{min}. This indicates that explanation-based diversification can indeed serve as an effective alternative when attribute-based diversification is not applicable.

Metric	Algorithm Swap	Algorithm Greedy
Result Jaccard	0.95	0.98
Result $(1 - K_{min})$	0.91	0.95

TABLE I
COMPARISON BETWEEN EXPLANATION-BASED AND ATTRIBUTE-BASED DIVERSIFICATION RESULTS.

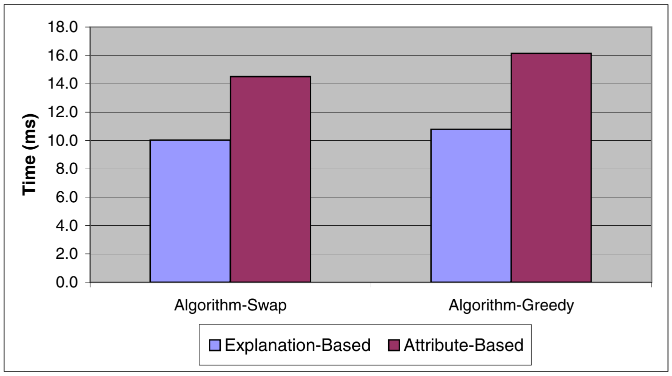

Fig. 1. Average Total (Recommendation + Diversification) Cost for Explanation and Attribute Based Diversifications.

B. *Efficiency of Explanation-Based Diversification*

As shown in Figure 1, explanation-based diversification outperforms attribute-based diversification significantly. While the former incurs overhead during the recommendation stage (of generating explanations), it offsets those costs during the diversification stage because it has all the information it needs to compute the diversification. For attribute-based diversification, while it saves time during the recommendation stage, retrieving attributes during the diversification stage can often be more costly.

In summary, compared with attribute-based diversification, explanation-based diversification can achieve a similar level of diversification with better performance. Furthermore, the former can also apply in scenarios where the latter is not applicable, for example, recommending bookmarks in del.icio.us.

VI. RELATED WORK

Recommender Systems: The need for explaining recommendations is discussed in [8] and [10]. There has been a push toward going beyond improving recommendation relevance [7], [6]. In [13], the authors introduced an order-independent intra-list similarity metric to assess the *topical diversity* of recommendation lists and a topic diversification approach for decreasing intra-list similarity.

Web Search: Most Web search engines, except [3] and [9], enforce diversity as a post-processing step. These methods are quite different from ours since they were developed in the context of search. The method in [3] relies on sampling term posting in Web documents using taxonomies, in order to reduce homogeneity. The method in [9] is based on query reformulation to re-rank the top-N results such that documents likely to be preferred by the user are presented higher. Indeed, large numbers of users modify their search queries in order to detect the kinds of results which tend to be missing from top results. Different query reformulations are selected and evaluated for pre-determined user interests.

Database Queries: Chen and Li [4] propose a notion of diversity over structured results which are post-processed and organized in a decision tree to help users navigate them. In [11], the authors introduce a hierarchical notion of diversity in databases and develop efficient top-k processing algorithms. Our formal definition of diversity is more general by allowing a flexible combination of relevance scores and of diversity.

VII. CONCLUSIONS

Web 2.0 has ushered in the era of social content sites and one popular mechanism for content delivery is recommendation. Traditional recommenders focus on bringing forth recommendations that maximize users' estimated ratings. However, this can come at the price of diversity and lead to homogeneous results that fail to engage the users [7]. To mitigate this, we studied the problem of generating diversified recommendations in a principled manner. Previous work has mainly relied on using item attributes to achieve diversity. We identified limitations of such approaches and proposed a first-class notion of explanation for recommendations both for content-based and for collaborative filtering strategies. We then defined explanation-based diversification and experimentally verified that such diversification is as effective as the attribute-based one. We are currently conducting a detailed user study to gauge the effect of diversity on user acceptance of the recommendations.

REFERENCES

[1] http://en.wikipedia.org/wiki/vector_space_model.
[2] G. Adomavicius and A. Tuzhilin. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Trans. Knowl. Data Eng.*, 17(6), 2005.
[3] A. Anagnostopoulos, A. Z. Broder, and D. Carmel. Sampling Search-Engine Results. In *WWW*, 2006.
[4] Z. Chen and T. Li. Addressing Diverse User Preferences in SQL-Query-Result Navigation. In *SIGMOD*, 2007.
[5] R. Fagin, R. Kumar, and D. Sivakumar. Comparing top k lists. *SIAM J. Discrete Math*, 17(1):134–60, 2003.
[6] J. A. Konstan. Introduction to recommender systems. In *SIGIR*, 2007.
[7] S. McNee, J. Riedl, and J. A. Konstan. Being Accurate Is Not Enough: How Accuracy Metrics Have Hurt Recommender Systems. In *CHI*, 2006.
[8] P. Pu and L. Chen. Trust Building with Explanation Interfaces. In *IUI*, 2006.
[9] F. Radlinski and S. T. Dumais. Improving Personalized Web Search using Result Diversification. In *SIGIR*, 2006.
[10] N. Tintarev and J. Masthoff. Effective Explanations of Recommendations: User-Centered Design. In *RecSys*. ACM SIGCHI, 2007.
[11] E. Vee, U. Srivastava, J. Shanmugasundaram, P. Bhat, and S. Amer-Yahia. Efficient Online Computation of Diverse Query Results. In *ICDE*, 2008.
[12] C. Yu, L. Lakshmanan, and S. Amer-Yahia. It takes variety to make a world: Diversification in recommendation systems. Submitted for publication, 2008.
[13] C.-N. Ziegler, S. M. McNee, J. A. Konstan, and G. Lausen. Improving recommendation lists through topic diversification. In *WWW*, 2005.