Optimal Greedy Diversity for Recommendation

Azin Ashkan
Technicolor Research, USA
azin.ashkan@technicolor.com

Branislav Kveton
Adobe Research, USA
kveton@adobe.com

Shlomo Berkovsky
CSIRO, Australia
shlomo.berkovsky@csiro.au

Zheng Wen
Yahoo! Labs, USA
zhengwen@yahoo-inc.com

Abstract

The need for diversification manifests in various recommendation use cases. In this work, we propose a novel approach to diversifying a list of recommended items, which maximizes the utility of the items subject to the increase in their diversity. From a technical perspective, the problem can be viewed as maximization of a modular function on the polytope of a submodular function, which can be solved optimally by a greedy method. We evaluate our approach in an offline analysis, which incorporates a number of baselines and metrics, and in two online user studies. In all the experiments, our method outperforms the baseline methods.

1 Introduction

Recommender systems are widely used in social networks, entertainment, and eCommerce [Ricci *et al.*, 2011]. Recommenders typically score items according to their match to the user's preferences and interests, and then recommend a list of top-scoring items. A naive selection of top items may yield a suboptimal recommendation list. For instance, collaborative filtering may recommend popular items that are known to the user [Koren and Bell, 2011]. Likewise, content-based filtering may target user's favorite topics and produce recommendations that overlook other topics [Lops *et al.*, 2011].

This has brought to the fore the problem of diversity in recommender systems, which can be addressed by constructing recommendation lists that cover a range of user interests [Castells *et al.*, 2011; Halvey *et al.*, 2009; McNee *et al.*, 2006; Vargas and Castells, 2011; Ziegler *et al.*, 2005]. The problem is particularly acute for users with eclectic interests, the recommendations for whom should include a variety of items, to increase the chance of answering ephemeral user needs. Repercussions of the diversity problem manifest in other recommendation use cases. Consider recommendations to heterogeneous user groups or sequential music recommendations. In both cases, the recommendation list should incorporate diverse items that either appeal to a number of group members or represent a number of music genres.

In all of the above use cases, it is important to maintain the trade-off between the diversity and utility of the results [Carbonell and Goldstein, 1998; Zhou *et al.*, 2010]. One common solution is to strike the balance between the two objectives by maximizing a weighted sum of a modular utility function and a submodular diversity function [Qin and Zhu, 2013; Santos *et al.*, 2010]. Another common solution is to introduce a submodular objective function that accounts for the diversity based on the utility of the recommended items in individual topics [Agrawal *et al.*, 2009]. In both cases, the optimized function is submodular. Therefore, a $(1 - 1/\mathbf{e})$-approximate solution to the problems can be computed greedily [Nemhauser *et al.*, 1978]. Despite being computationally efficient, the solution is suboptimal and it is well-known that the optimal solution is NP-hard to compute.

In our work, we propose a new approach to recommending diverse items, which we call *diversity-weighted utility maximization* (DUM). The intuition behind this method is to maximize the utility of the items recommended to users with respect to the diversity of their tastes. In other words, the utility of items remains the primary concern, but it is subjected to increasing the diversity of the recommendation list. We cast this problem as maximizing a modular utility function on the polytope of a submodular diversity function. The key difference from existing work on diversification and submodularity is that we do not maximize a submodular function; we maximize a modular function subject to a submodular constraint. This problem can be solved optimally by a greedy method [Edmonds, 1970].

We conduct an extensive evaluation of the proposed approach. We present an offline evaluation that compares DUM to three baseline methods, in terms of the trade-off between diversity and utility. We also present two online studies that compare the lists generated by DUM to baselines maximizing a convex combination of utility and diversity. All experiments show the superiority of the DUM lists over the baseline methods. Overall, we demonstrate that DUM can deliver recommendations with high degree of utility and diversity, while not requiring a-priori parameter tuning. Hence, the contribution of this work is two-fold. First, we propose a parameter-free, efficient method based on a new objective function that improves the diversity of the recommendation lists, while maintaining

their utility. Second, we present a solid empirical evidence supporting the validity of the proposed approach.

Notation: Let A and B be sets, and e be an element of a set. We use $A + e$ for $A \cup \{e\}$, $A + B$ for $A \cup B$, $A - e$ for $A \setminus \{e\}$, and $A - B$ for $A \setminus B$. We represent *ordered sets* by vectors and also refer to them as *lists*.

2 Related Work

A common approximation to diversified ranking is based on the notion of *maximal marginal relevance (*MMR*)* [Carbonell and Goldstein, 1998]. There, utility (relevance) and diversity are represented by independent metrics. Marginal relevance is defined as a convex combination of the two metrics. Let E be the ground set of L recommendable items and $\mathbf{w}(e)$ be the utility of item $e \in E$. As illustrated in Algorithm 1, MMR creates a diversified ranking of the items in E by choosing an item $e^* \in E - S$ in each iteration such that it maximizes the marginal relevance:

$$e^* = \arg\max_{e \in E-S} (1-\lambda)\mathbf{w}(e) + \lambda(f(S+e) - f(S)) \quad (1)$$

where S is the list of recommended items, $f : 2^E \rightarrow \mathbb{R}^+$ is the diversity function, and the parameter λ controls the tradeoff between utility and diversity. Typically, the utility $\mathbf{w}$ is a modular function of S, whereas the diversity f is a submodular function of S, and $f(S+e) - f(S)$ is the gain in diversity after e is added to S.

Algorithm 1 MMR: Maximal Marginal Relevance

Input: Ground set of items E
$S \leftarrow ()$, $L = |E|$
while $|S| < L$ **do**
  $e^* \leftarrow \arg\max_{e \in E-S} (1-\lambda)\mathbf{w}(e) + \lambda(f(S+e) - f(S))$
  Append item e^* to list S
Output: List of recommended items S

Implicit approaches assume that items covering similar topics should be penalized. For instance, [Yu *et al.*, 2014] computes $f(S + e) - f(S) = -\max_{e' \in S} \mathrm{sim}(e, e')$ to account for the redundancy of user intent e with respect to a set of intents S. Similarly, [Gollapudi and Sharma, 2009] targets diversification using distance functions, which are based on implicit metrics of pairwise similarity between documents. On the other hand, explicit approaches model the topics, and promote diversity by maximizing the coverage with respect to these topics. For instance, [Santos *et al.*, 2010] defines $f(S + e) - f(S) = \Sigma_{t \in \mathcal{T}_q} P(t|q)P(e, \bar{S}|t)$, where $P(e, \bar{S}|t)$ is the probability of e satisfying topic t while the ones in S failed to do so, and $P(t|q)$ is the popularity of t among all possible topics $\mathcal{T}_q$ that may satisfy a query q.

Another group of related works learns a diverse ranking by maximizing a submodular objective function. Among these, [Radlinski *et al.*, 2008] and [Yue and Guestrin, 2011] propose online learning algorithms for optimizing submodular objective functions for diversified retrieval and recommendation, respectively. The work by [Agrawal *et al.*, 2009] addresses search diversification in an offline setting, and targets the maximization of a submodular objective function following the definition of marginal relevance. They approximate the objective function and show that an optimal solution is found when each document belongs to exactly one topic. [Vallet and Castells, 2012] studies personalization in combination with diversity, such that the objectives complement each other and satisfy user needs derived from the available user preferences.

One of the initial works in recommendation diversification is by [Ziegler *et al.*, 2005] that proposes a metric computing the average pairwise similarity of items in a recommendation list. This metric is used to control the balance between the accuracy and diversity. [Zhang and Hurley, 2008] formulate the diversification problem as finding the best subset of items to be recommended. They address this as the maximization of the diversity of a recommendation list, subject to maintaining the accuracy of the items. [Zhou *et al.*, 2010] proposes a hybrid method that maximizes a weighted combination of utility- and diversity-based approaches but requires parameter tuning to control the tradeoff between the two.

Most of the existing diversification approaches are based on the marginal relevance, maximizing the submodular objective function (1). Thus, a $(1 - 1/\mathbf{e})$-approximation to the optimal solution can be computed greedily [Nemhauser *et al.*, 1978] while the exact solution to the problem is computationally intractable. The current paper is an extension to prior work in [Ashkan *et al.*, 2014], where we introduce a new objective function for diversification, the optimal solution of which can be found greedily. This function targets the utility as the primary concern, and maximizes it with respect to the diversity of user's tastes. In this paper, we show that this method is computationally efficient and parameter-free, and it guarantees that high-utility items appear at the top of the recommendation list, as long as they contribute to the diversity of the list. We elaborate on the details of the greedy algorithm, and provide extensive online and offline evaluations on its performance.

3 Motivating Examples

Our formulation of recommending diverse items is motivated by the following problem. Suppose that a user wants to watch a movie from genre t, which is chosen randomly from a set of movie genres $\mathcal{T}$. The recommender does not know t and recommends a list of movies. The user examines the list, from the first recommended item to the last, and chooses the first movie e from t. Then the user watches e and is satisfied with probability $\mathbf{w}(e)$. The recommender knows the satisfaction probability $\mathbf{w}(e)$ for each e. The goal of the recommender is to generate a list of movies that maximizes the probability that the user is satisfied for any choice of t.

We illustrate our optimization problem with two examples, where $\mathcal{T} = \{t_1, t_2\}$ are two movie genres. In Figure 1a, the optimal list is $S = (m_1, m_3)$. If the user prefers genre t_1, then user chooses m_1 and is satisfied with probability 0.8. If the user prefers t_2, the user chooses m_3 and is satisfied with probability 0.5. Now suppose that movie m_4 is replaced with m_5, which satisfies the user with probability 0.9 and belongs to both genres (Figure 1b). Then the optimal list is $S = (m_5)$, as m_5 is the most satisfactory movie in both genres in $\mathcal{T}$.

movie e	$\mathbf{w}(e)$	t_1	t_2
m_1	0.8	X	
m_2	0.7	X	
m_3	0.5		X
m_4	0.2		X

(a)

movie e	$\mathbf{w}(e)$	t_1	t_2
m_1	0.8	X	
m_2	0.7	X	
m_3	0.5		X
m_5	0.9	X	X

(b)

Figure 1: Illustrative examples in Section 3.

In the next section, we state the problem of recommending diverse items slightly more formally.

4 Optimal Greedy Diversification

Let $E = \{1, \ldots, L\}$ be a set of L *ground items* and $\mathbf{w} \in (\mathbb{R}^+)^L$ be a vector of *item utilities*, where $\mathbf{w}(e)$ is the utility of item e. Let $f : 2^E \to \mathbb{R}^+$ be a *diversity function*, which maps any subset of E to a non-negative real number. Then the problem of *diversity-weighted utility maximization* is:

$$A^* = \arg\max_{A \in \Theta} \sum_{k=1}^{L} [f(A_k) - f(A_{k-1})]\mathbf{w}(a_k), \qquad (2)$$

where $A = (a_1, \ldots, a_L)$ is a list of items from E, Θ is the set of all permutations of E, $A_k = \{a_1, \ldots, a_k\}$ is a set of the first k items in A, and $f(A_k) - f(A_{k-1})$ is the gain in diversity after a_k is added to A_{k-1}. The solution to (2) is a list $A^* = (a_1^*, \ldots, a_L^*)$ that maximizes the utility of the recommended items weighted by the increase in their diversity.

The problem in Section 3 can be formulated in our framework as follows. The ground set E are all recommendable movies; the utility of movie e is the satisfaction probability of that movie; and the diversity function is defined as:

$$f(X) = \sum_{t \in \mathcal{T}} \mathbb{1}\{\exists e \in X : \text{item } e \text{ covers topic } t\}, \qquad (3)$$

where $\mathcal{T} = \{1, \ldots, M\}$ is a finite set of topics. We state this result more formally in Proposition 1.

4.1 Greedy Solution

For a general function f, the optimization problem in (2) is NP-hard. However, when f is *submodular* and *monotone*, the problem can be cast as finding a maximum-weight basis of a polymatroid [Edmonds, 1970], which can be solved greedily and optimally. The pseudocode of our greedy algorithm is in Algorithm 2. We call it *diversity-weighted utility maximization (*DUM*)*. DUM works as follows. Let $A^* = (a_1^*, \ldots, a_L^*)$ be a list of items that are ordered in decreasing order of utility, $\mathbf{w}(a_1^*) \geq \ldots \geq \mathbf{w}(a_L^*)$. DUM examines the list A^* from the first item to the last. When $f(A_k^*) - f(A_{k-1}^*) > 0$, item a_k^* is added to the recommended list S. When $f(A_k^*) - f(A_{k-1}^*) = 0$, item a_k^* is not added to S because it does not contribute to the diversity of S. Finally, the algorithm returns S.

DUM has several notable properties. First, DUM does not have any *tunable parameters* and therefore we expect it to be robust in practice. Second, DUM is a greedy method. Therefore, it is *computationally efficient*. In particular, suppose that the diversity function f is an oracle that can be queried in $O(1)$ time. Then the time complexity of DUM is $O(L \log L)$, similar to the time complexity of sorting L numbers. Finally, DUM computes the *optimal solution* to our problem (2).

Algorithm 2 DUM: Diversity-Weighted Utility Maximization

Input: Ground set E and item utilities $\mathbf{w}$

// Compute the maximum-weight basis of a polymatroid
Let $a_1^*, \ldots, a_L^*$ be an ordering of items E such that:
$\mathbf{w}(a_1^*) \geq \ldots \geq \mathbf{w}(a_L^*)$
$A^* \leftarrow (a_1^*, \ldots, a_L^*)$

// Generate the list of recommended items S
$S \leftarrow ()$
for $k = 1, \ldots, L$ **do**
  if $(f(A_k^*) - f(A_{k-1}^*) > 0)$ **then**
    Append item a_k^* to list S

Output: List of recommended items S

The optimality of DUM can be proved as follows. The optimization problem (2) is equivalent to maximizing a modular function on a polymatroid [Edmonds, 1970], a well-known combinatorial optimization problem that can be solved greedily. Let $M = (E, f)$ be a polymatroid, where E is its ground set and f is a monotone submodular function. Let:

$$P_M = \left\{ \mathbf{x} : \mathbf{x} \in \mathbb{R}^L,\ \mathbf{x} \geq 0,\ \forall X \subseteq E : \sum_{e \in X} \mathbf{x}(e) \leq f(X) \right\}$$

be the independence polyhedron associated with function f. Then the *maximum-weight basis* of M is defined as:

$$\mathbf{x}^* = \arg\max_{\mathbf{x} \in P_M} \langle \mathbf{w}, \mathbf{x} \rangle, \qquad (4)$$

where $\mathbf{w} \in (\mathbb{R}^+)^L$ is a vector of non-negative weights. Since P_M is a submodular polytope and $\mathbf{w}(e) \geq 0$ for all $e \in E$, the problem in (4) is equivalent to finding the order of dimensions A in which $\langle \mathbf{w}, \mathbf{x} \rangle$ is maximized [Edmonds, 1970]. That is, the problem can be written as (2) and has the same greedy solution. Let $A^* = (a_1^*, \ldots, a_L^*)$ be a list of items that are ordered in decreasing order of their weights, $\mathbf{w}(a_1^*) \geq \ldots \geq \mathbf{w}(a_L^*)$. Then $\mathbf{x}^*(a_k^*) = f(A_k^*) - f(A_{k-1}^*)$ for all $1 \leq k \leq L$.

4.2 Diversity Functions

Our approach is practical when the list of recommended items S is manageably short. Therefore, not all diversity functions f are suitable for our approach. In this section, we discuss two classes of diversity functions on topics $\mathcal{T} = \{1, \ldots, M\}$ that allow us to control the length of the recommended list.

Proposition 1. *Let the diversity function f be defined as in* (3)*. Then* DUM *returns a list S such that each topic $t \in \mathcal{T}$ is covered by the highest-utility item in that topic. The length of S is at most $|\mathcal{T}|$.*

Proof. We prove the first claim by contradiction. Let e be the highest-utility item in topic t. Suppose that item e is not chosen by DUM. Then DUM must choose another item that covers topic t. However, this contradicts to the definition of DUM. In particular, since item e is the highest-utility item in topic t,

DUM must choose it before any other item in topic t. The second claim follows from the fact that $f(A_k^*) - f(A_{k-1}^*) > 0$ implies $f(A_k^*) - f(A_{k-1}^*) \geq 1$. By definition, $f(E) \leq |\mathcal{T}|$. So the maximum number of items added to S is $|\mathcal{T}|$. □

Another suitable diversity function is:

$$f(X) = \sum_{t \in \mathcal{T}} \min \Big\{ \sum_{e \in X} \mathbb{1}\{\text{item } e \text{ covers topic } t\}, N_t \Big\}, \quad (5)$$

where N_t is the number of items from topic t that is required to be in the recommended list. This function is motivated by a similar model of user behavior as in Section 3. The difference is that the user demands N_t recommended items from topic t and that the quality of the recommended list is measured by the lowest-utility item among these items, for each topic t. Under this assumption, the optimal recommended list is the list returned by DUM for the diversity function f in (5). We characterize the output of DUM for this diversity function below.

Proposition 2. *Let the diversity function f be defined as in* (5)*. Then* DUM *returns a list S such that each topic $t \in \mathcal{T}$ is covered by at least N_t highest-utility items in that topic. The length of S is at most $\sum_{t \in \mathcal{T}} N_t$.*

Proof. The proof is similar to that of Proposition 1. □

When the number of topics $\mathcal{T}$ is huge, neither of the proposed diversity functions in (3) and (5) may be practical. In this case, we suggest reducing $\mathcal{T}$ by any existing dimensionality reduction method in machine learning, such as topic modeling [Blei *et al.*, 2003]. In this particular approach, our topics $\mathcal{T}$ would be words, the items would be sentences, and the topics $\mathcal{T}'$ generated by topic modeling would be the new topics in (3) and (5). We leave the analysis and evaluation of this approach for future work, and focus on the case when the number of topics $\mathcal{T}$ is small.

5 Experiments

The proposed method is evaluated in an offline setting as well as in two online user studies. In the offline evaluation, we compare DUM to a group of existing works and under various conditions, such as recommendations across multiple users with their interest profiles defined based on different combinations of genres. In the online user studies, we choose a simpler evaluation setting in order to maintain the studies at a fair level of complexity. We choose MMR as our baseline for online studies as existing diversification approaches are mainly based on the objective function of MMR.

5.1 Offline Evaluation

We use the *1M MovieLens* dataset[1], which consists of movie ratings given on a 1-to-5 stars scale. We exclude users with less than 300 ratings, and end up with 1000 users and a total of 515k ratings.

Movies rated by each user are split randomly into the training and test set with the $2:1$ ratio; on average, 343 movies in the training set and 171 in the test set. The split is performed three times, and the reported results are based on the average of three experiments. The training set is used for creating the user's interest profile, whereas the test set contains the recommendable movies. We use matrix factorization [Thurau *et al.*, 2011] to predict the ratings in the test set and feed these as the utility scores into DUM and the baseline methods.

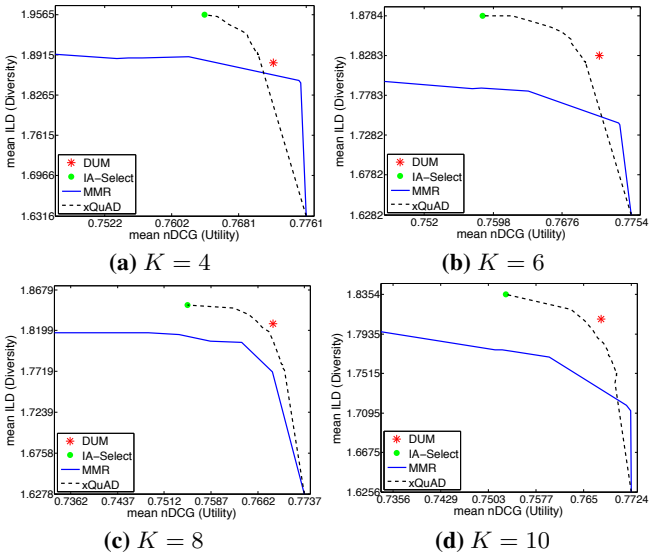
There are 18 genres in the dataset, and each movie belongs to one genre or more. For each user, we create a multinomial distribution over the popularity of genres of the movies rated by the user in the training set, assuming that users rate movies that they watched. We sample 10 times from this distribution, to create the user's preference profile over genres, and normalize it so that the sum of the scores is 1. For each user, we set $N_t = \lfloor r_t \times K \rfloor$ in (5), where K is the length of the recommendation list and r_t is the user's preference score for genre t . That is, the coverage of a genre in the list is proportional to the degree of user preference for the genre.

Movies in the test set are used as the ground set E of recommendable movies, from which each method selects K movies. The predicted utility of the movies is used for the recommendation. The reason for using the predicted utility instead of the readily available movie ratings is to keep the evaluation close to real-world recommendation scenarios, where the utility of items is unknown. For the performance evaluation we use the actual ratings assigned by the users.

We compare DUM with three baselines. The first baseline is MMR [Carbonell and Goldstein, 1998], which is the basis of many diversification methods. The second baseline is IASelect [Agrawal *et al.*, 2009] that targets the maximization of a submodular diversity function computed based on the utility of the recommended items for each genre. The third baseline is xQuAD [Santos *et al.*, 2010] that considers the submodular objective function of IASelect as the diversity function and combines it linearly with the utility function in a setting similar to that of MMR. For MMR and xQuAD, we experiment with values of $\lambda \in [0, 1]$ to account for the trade-off between diversity and utility in these methods.

The performance of DUM is compared to these baselines with respect to diversity and utility individually, as well as in combination. We use the intra-list distance (ILD) metric [Zhang and Hurley, 2008] commonly used to measure the diversity of a recommendation list as the average distance between pairs of recommended items. We compute ILD based on the Euclidean distance between the genre vectors of movies. The second metric is the discounted cumulative gain (DCG) [Järvelin and Kekäläinen, 2002] that measures the accumulated utility gain of items in the recommendation list, with the gain of each item discounted by its position. We compute the normalized DCG (nDCG) as $\text{nDCG} = \text{DCG}/\text{IDCG}$, where IDCG is the ideal gain achievable when all the items have the highest utility. The final metric is the expected intra-list distance (EILD) which was proposed by [Vargas and Castells, 2011] as a compound metric of utility and diversity. It measures the average intra-list distance with respect to rank-sensitivity and utility. We compute all the metrics for every recommendation list provided to a user. Then, we average them across the three splits, to compute user-based mean of the metric, and the mean of each metric and method is computed across all the users.

[1] http://www.grouplens.org/node/12



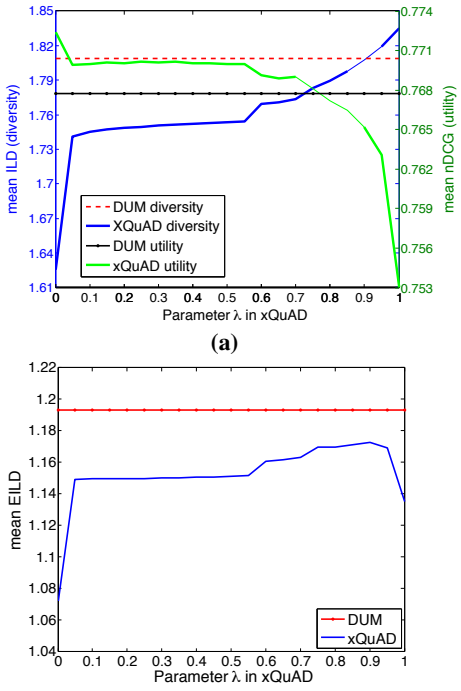
(a) $K = 4$ **(b)** $K = 6$ **(c)** $K = 8$ **(d)** $K = 10$

Figure 2: Tradeoff between diversity and utility of DUM versus three baselines for varying values of K.

Figure 2 shows the performance of DUM and the baseline methods in terms of diversity and utility metrics for recommendation lists with $K \in \{4, 6, 8, 10\}$. It can be seen that both MMR and xQuAD exhibit a trade-off between the values of ILD (diversity metric) and nDCG (utility metric), for $\lambda \in [0, 1]$. For low values of λ the utility is prioritized, such that the diversity of the lists is low and the utility is high. An opposite is observed for high λ, when the diversity is prioritized. The performance of IASelect is the same as of xQuAD at the point of $\lambda = 1$. This is expected, as IASelect is a special case of xQuAD that accounts only for the maximization of diversity. In general, xQuAD and IASelect achieve higher utility than MMR, since in both xQuAD and IASelect the diversity function is computed based on the utility estimate of items [Santos *et al.*, 2010].

The performance of DUM with respect to both metrics is superior to that of MMR, IASelect, and xQuAD. In particular, as shown in all four plots of Figure 2, the utility and diversity cannot be simultaneously optimized by MMR, while DUM achieves both of them, at the same time being a parameter-free method. The difference between nDCG and ILD of DUM and the corresponding values of MMR and IASelect is statistically significant at $p < 0.01$ (based on paired t-tests) and across the results of all users. There are small ranges of λ, where the difference between DUM and xQuAD is not significant and we further examine these ranges next.

The performance of xQuAD in terms of diversity (ILD) and utility (nDCG) for various values of λ is illustrated in Figure 3-a, along with the steady performance of DUM. For ranges of λ in which the differences between DUM and xQuAD are significant at $p < 0.01$, the curves of xQuAD are thicker. It can be seen that $\lambda = 0.74$ is the operating point of xQuAD where the utility and the diversity curves intersect. There, xQuAD slightly outperforms DUM in utility, but this is not statistically significant. At the same point, DUM outperforms xQuAD in terms of diversity, and this observation is significant. This confirms that the utility and diversity of DUM are either on par or superior to those of xQuAD at its operating point.



(a) **(b)**

Figure 3: Performance of DUM versus xQuAD for varying values of λ, with respect to: **(a)** individual metrics of diversity and utility, and **(b)** the combined metric. Both belong to results with $K = 10$.

Another argument in favor of DUM is obtained through the EILD metric that combines diversity and utility. A comparison of DUM and xQuAD with respect to EILD is shown in Figure 3-b. It can be seen that DUM substantially outperforms xQuAD for all the values of λ, confirming the superiority of DUM in balancing the utility and diversity goals.

5.2 Online User Studies

We conduct two online studies using Amazon's Mechanical Turk (MT)[2]. In the first study, we *evaluate* the lists generated by DUM and MMR, by asking MT workers to identify in the lists a movie that matches their genre of interest and to indicate the relevance of this movie. In the second study, we *compare* the lists generated by DUM and MMR, by asking MT workers to judge the coverage of two movie genres by the lists.

Study 1

The ground set E are 10k most frequently rated IMDb[3] movies. The utility $\mathbf{w}(e)$ of movie e is the number of ratings assigned to the movie. The values of $\mathbf{w}(e)$ are normalized such that $\max_{e \in E} \mathbf{w}(e) = 1$. The diversity function f is defined as in (3). We also normalize f such that $\max_{e \in E} f(e) = 1$. The topics $\mathcal{T}$ are 8 most popular movie

[2]http://www.mturk.com
[3]http://www.imdb.com

		DUM	MMR		
	K		$\lambda = \frac{1}{3}$	$\lambda = \frac{2}{3}$	$\lambda = 0.99$
Matching movie is among top K items	2	35.7%	27.1%	26.6%	31.2%
	4	61.3%	51.8%	**48.7%**	**49.2%**
	6	82.9%	**69.8%**	**65.3%**	**66.3%**
	8	84.4%	**70.9%**	**67.3%**	**66.8%**
Matching movie is considered as good	2	30.7%	23.1%	25.6%	28.6%
	4	55.8%	45.7%	45.2%	45.7%
	6	75.9%	**63.8%**	**60.8%**	**62.3%**
	8	77.4%	**64.8%**	**62.8%**	**62.8%**

Table 1: Comparison of DUM and MMR in study 1. The bold values are significantly inferior to DUM at $p < 0.01$.

genres in our dataset, $\mathcal{T}$ = {*Drama*, *Comedy*, *Thriller*, *Romance*, *Action*, *Crime*, *Adventure*, *Horror*}. For this $\mathcal{T}$, DUM generates a list of up to 8 movies (see Proposition 1). We compare DUM to three variants of MMR, which are parameterized by $\lambda \in \left\{\frac{1}{3}, \frac{2}{3}, 0.99\right\}$.

All methods are evaluated in 200 MT tasks (HITs). In each HIT, we ask the worker to choose a genre of interest. Then, we generate four recommendation lists: one by DUM and three by MMR for different values of λ. We ask the worker to evaluate each list with two questions. First, we ask the worker to identify a movie that matches the chosen genre. This question assesses whether the chosen genre is covered by the list (the worker can answer "none"). If a matching movie is identified, we ask if this movie is a good recommendation for the chosen genre. This question assesses whether the chosen genre is covered by a good movie.

In each HIT, the four recommendation lists are presented in a random order. This eliminates the *position bias*. Moreover, in each HIT, the recommendable items are 3.3k randomly chosen movies from the ground set E of 10k movies. Thus, the recommendation lists differ across the HITs, which eliminates the *item bias*. Finally, all the lists are of the same length – that of the list produced by DUM. This eliminates any potential bias due to a different length of the lists.

The 200 HITs are completed by 34 *master* workers, who were assigned with this status based on the quality of their prior HITs. Each worker is allowed to complete at most 8 HITs. On average, each HIT is completed in 72 seconds, so each recommendation list is evaluated in 19 seconds on average. The results of the study are presented in Table 1. For each method, we report the percentage of times when the matching movie is among top K movies in the list and is considered a good recommendation. We report $K \in \{2, 4, 6, 8\}$. We observe that the percentage of times that the worker finds a matching movie in the DUM list is significantly higher than in the list of the best performing baseline, MMR with $\lambda = \frac{1}{3}$. This result is statistically significant for larger values of K.

Note that for all the methods in Table 1, the ratio between the percentage of times that the movie is a good recommendation and that the matching movie is found is between 0.92 and 0.94. This implies that if a matching movie is found, it is very likely to be a good recommendation. We conjecture that this is due to the popularity of movies in E, which practically guarantees the high utility of the movies and erodes the differences between the compared methods.

Suitable for	DUM	MMR		
		$\lambda = \frac{1}{3}$	$\lambda = \frac{2}{3}$	$\lambda = 0.99$
Alice and Bob	74.51%	64.92%	58.39%	28.98%
Alice or Bob	23.53%	32.68%	39.43%	66.67%
Neither	1.96%	2.40%	2.18%	4.36%

Table 2: Comparison of DUM and MMR in user study 2.

Study 2

In the second study, we evaluate DUM on a specific use case of recommending a diverse set of movies that covers exactly two genres. In each HIT, we ask the MT worker to consider a situation where Alice and Bob, who prefer two different genres, go for a vacation and can take with them several movies. We compare DUM to three variants of MMR parameterized by $\lambda \in \left\{\frac{1}{3}, \frac{2}{3}, 0.99\right\}$, and generate four lists: one by DUM and three by MMR for these values of λ. For each list, we ask the worker to indicate whether the list is appropriate for both Alice and Bob, only for one of them, or for neither. The lists are presented in a random order to eliminate the *position bias*.

Each HIT is associated with two genres preferred by Alice and Bob, t_1 and t_2. We generate three HITs for each pair of the 18 most frequent IMDb genres. So the recommendation lists are evaluated $3 \times \frac{18 \times 17}{2} = 459$ times. Like in the first study, the ground set E are 10K most frequently rated IMDb movies. The utility $\mathbf{w}(e)$ is the number of ratings assigned to e. The diversity function f is defined as in (5). We set $N_{t_1} = N_{t_2} = 4$. In this case, DUM generates a list of at most 8 movies, at least 4 from each genre. The length of the other lists is set to that of DUM. The utility and diversity are normalized as in the first study.

The 459 HITs are completed by 57 *master* workers. Each worker is allowed to complete at most 10 HITs. On average, each HIT is completed in 57 seconds, so that each list is evaluated in 14 seconds. The results of the study are presented in Table 2. For each method, we report the percentage of times that the worker considers the recommendation list as suitable for both Alice and Bob, only for one of them, or for neither. We observe that the workers consider the DUM list to be suitable for both Alice and Bob in 74.51% of cases. This is 9.6% higher than the best performing baseline, MMR with $\lambda = \frac{1}{3}$. This difference is statistically significant at $p < 0.01$. Hence, DUM is perceived superior to MMR in generating diverse lists that cover exactly two movie genres.

6 Conclusion

In this work, we propose a new approach to diversifying a list of recommended items, DUM, which maximizes the utility of the items subject to the increase in their diversity. We show that the problem can be solved optimally by a greedy method, because it is an instance of maximizing a modular function on the polytope of a submodular function. We evaluate DUM on a variety of problems. In the offline experiments, we compare DUM to three popular baselines in terms of the utility and diversity of recommended lists. Our results show that DUM effectively balances the utility and diversity of the lists, despite the fact that it has no tunable parameters. We also present two online user studies and show that DUM outperforms a baseline

that maximizes a linear combination of utility and diversity.

A future direction for this work is to account for the novelty of the recommended items [Castells *et al.*, 2011; Clarke *et al.*, 2008] with respect to prior consumption history of the user. This may be incorporated into the diversity function by considering, apart from the diversity contribution, also the novelty contribution of the items. Moreover, similar to [Radlinski *et al.*, 2008; Yue and Guestrin, 2011], the utilities of items can be learned in an online fashion using the learning variants of maximizing a modular function on a polymatroid [Kveton *et al.*, 2014a; 2014b]. Another issue that deserves investigation is the changes in the diversity function needed to reflect the tolerance for redundancy in different domains. For instance, a diversity metric for news filtering may differ from the metric we derived here for the movie recommendation task. We intend to address these questions in the future

References

[Agrawal *et al.*, 2009] Rakesh Agrawal, Sreenivas Gollapudi, Alan Halverson, and Samuel Ieong. Diversifying search results. In *WSDM*, pages 5–14, 2009.

[Ashkan *et al.*, 2014] Azin Ashkan, Branislav Kveton, Shlomo Berkovsky, and Zheng Wen. Diversified utility maximization for recommendations. In *RecSys Poster Proceedings*, 2014.

[Blei *et al.*, 2003] David M Blei, Andrew Y Ng, and Michael I Jordan. Latent Dirichlet allocation. *Journal of machine Learning research*, 3:993–1022, 2003.

[Carbonell and Goldstein, 1998] Jaime Carbonell and Jade Goldstein. The use of MMR, diversity-based reranking for reordering documents and producing summaries. In *SIGIR*, pages 335–336, 1998.

[Castells *et al.*, 2011] Pablo Castells, Saúl Vargas, and Jun Wang. Novelty and diversity metrics for recommender systems: choice, discovery and relevance. In *International Workshop on Diversity in Document Retrieval*, pages 29–36, 2011.

[Clarke *et al.*, 2008] Charles LA Clarke, Maheedhar Kolla, Gordon V Cormack, Olga Vechtomova, Azin Ashkan, Stefan Büttcher, and Ian MacKinnon. Novelty and diversity in information retrieval evaluation. In *SIGIR*, pages 659–666, 2008.

[Edmonds, 1970] Jack Edmonds. Submodular functions, matroids, and certain polyhedra. In *International Conference on Combinatorial Structures and their Applications*, pages 69–87. 1970.

[Gollapudi and Sharma, 2009] Sreenivas Gollapudi and Aneesh Sharma. An axiomatic approach for result diversification. In *WWW*, pages 381–390, 2009.

[Halvey *et al.*, 2009] Martin Halvey, P Punitha, David Hannah, Robert Villa, Frank Hopfgartner, Anuj Goyal, and Joemon M Jose. Diversity, assortment, dissimilarity, variety: A study of diversity measures using low level features for video retrieval. In *Advances in Information Retrieval*, pages 126–137. 2009.

[Järvelin and Kekäläinen, 2002] Kalervo Järvelin and Jaana Kekäläinen. Cumulated gain-based evaluation of IR techniques. *ACM Transactions on Information Systems*, 20(4):422–446, 2002.

[Koren and Bell, 2011] Yehuda Koren and Robert M. Bell. Advances in collaborative filtering. In *Recommender Systems Handbook*, pages 145–186. 2011.

[Kveton *et al.*, 2014a] Branislav Kveton, Zheng Wen, Azin Ashkan, Hoda Eydgahi, and Brian Eriksson. Matroid bandits: Fast combinatorial optimization with learning. In *UAI*, pages 420–429, 2014.

[Kveton *et al.*, 2014b] Branislav Kveton, Zheng Wen, Azin Ashkan, and Michal Valko. Learning to act greedily: Polymatroid semi-bandits. *CoRR*, abs/1405.7752, 2014.

[Lops *et al.*, 2011] Pasquale Lops, Marco de Gemmis, and Giovanni Semeraro. Content-based recommender systems: State of the art and trends. In *Recommender Systems Handbook*, pages 73–105. 2011.

[McNee *et al.*, 2006] Sean M McNee, John Riedl, and Joseph A Konstan. Being accurate is not enough: how accuracy metrics have hurt recommender systems. In *Proceedings of International Conference on Human Factors in Computing Systems*, pages 1097–1101, 2006.

[Nemhauser *et al.*, 1978] G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher. An analysis of approximations for maximizing submodular set functions - I. *Mathematical Programming*, 14(1):265–294, 1978.

[Qin and Zhu, 2013] Lijing Qin and Xiaoyan Zhu. Promoting diversity in recommendation by entropy regularizer. In *IJCAI*, pages 2698–2704, 2013.

[Radlinski *et al.*, 2008] Filip Radlinski, Robert Kleinberg, and Thorsten Joachims. Learning diverse rankings with multi-armed bandits. In *ICML*, pages 784–791, 2008.

[Ricci *et al.*, 2011] Francesco Ricci, Lior Rokach, Bracha Shapira, and Paul B. Kantor, editors. *Recommender Systems Handbook*. Springer, 2011.

[Santos *et al.*, 2010] Rodrygo LT Santos, Craig Macdonald, and Iadh Ounis. Exploiting query reformulations for Web search result diversification. In *WWW*, pages 881–890, 2010.

[Thurau *et al.*, 2011] Christian Thurau, Kristian Kersting, Mirwaes Wahabzada, and Christian Bauckhage. Convex non-negative matrix factorization for massive datasets. *Knowledge and Information Systems*, 29(2):457–478, 2011.

[Vallet and Castells, 2012] David Vallet and Pablo Castells. Personalized diversification of search results. In *SIGIR*, pages 841–850, 2012.

[Vargas and Castells, 2011] Saúl Vargas and Pablo Castells. Rank and relevance in novelty and diversity metrics for recommender systems. In *RecSys*, pages 109–116, 2011.

[Yu *et al.*, 2014] Jun Yu, Sunil Mohan, Duangmanee Pew Putthividhya, and Weng-Keen Wong. Latent Dirichlet allocation based diversified retrieval for e-commerce search. In *WSDM*, pages 463–472, 2014.

[Yue and Guestrin, 2011] Yisong Yue and Carlos Guestrin. Linear submodular bandits and their application to diversified retrieval. In *NIPS*, pages 2483–2491, 2011.

[Zhang and Hurley, 2008] Mi Zhang and Neil Hurley. Avoiding monotony: improving the diversity of recommendation lists. In *RecSys*, pages 123–130, 2008.

[Zhou *et al.*, 2010] Tao Zhou, Zoltán Kuscsik, Jian-Guo Liu, Matúš Medo, Joseph Rushton Wakeling, and Yi-Cheng Zhang. Solving the apparent diversity-accuracy dilemma of recommender systems. *National Academy of Sciences*, 107(10):4511–4515, 2010.

[Ziegler *et al.*, 2005] Cai-Nicolas Ziegler, Sean M McNee, Joseph A Konstan, and Georg Lausen. Improving recommendation lists through topic diversification. In *WWW*, pages 22–32, 2005.