The Minimum Conductance Dissimilarity Cut (MCDC) Algorithm to Increase Novelty and Diversity of Recommendations

Sanjay Krishnan [*], Ken Goldberg [*] [†]
[*] EECS Department UC Berkeley, [†] IEOR Department UC Berkeley
{sanjaykrishnan, goldberg}@berkeley.edu

ABSTRACT

As established by Herlocker et al. and Ekstrand et al., recommender systems that consider diversity and novelty as well as expected ratings when selecting sets of items to recommend can increase user engagement and exposure to a broader base of inventory addressing the Pigeonhole and Blockbuster problems. Novelty, diversity, and expected rating define a multi-objective optimization problem that is difficult to solve. Existing approaches rely on iterative optimization techniques like evolutionary algorithms or simulated annealing. In this paper, we present MCDC, an algorithm for approximately solving this problem as an eigenvector problem, solving a graph partition problem on a weighted dissimilarity graph. MCDC efficiently exploits the local-global structure of the problem (novelty is a global property and diversity is a local property) to approximate a Pareto-optimal balance between novelty, coverage (which is related to diversity), and the recommendation score. Results with several datasets, including Jester and the latest Movielens data, show that MCDC provides recommendations sets with up to 21% higher combined geometric mean values and that MCDC is up to an order of magnitude faster than previous methods. MCDC can be easily applied to any recommender system based on a user-item ratings matrix. Code is provided at: `https://github.com/sjyk/mcdc`

1. INTRODUCTION

The recommendation problem is traditionally defined as selecting the top-k items that maximize expected rating (i.e., most likely to be purchased by the user). Recent research argues that there are, in fact, numerous other important considerations. For example, the top-k formulation can lead to a lack of *diversity*, where items in the top-k are overly similar to each other [8, 22, 21, 6, 22, 7]. A related issue is recommending "novel items" [6, 7, 15, 8], ones in the long tail whose ratings do not correlate well with others in the system. In the seminal work of Herlocker et al. [8] on evaluating recommender systems, diversity and novelty were established as important metrics when evaluating the performance of recommendation algorithms.

Novelty and diversity have the potential to increase user engagement while exposing more items in the inventory to rating. However, the key challenge is balancing these metrics. Recommending a set of items that is very diverse or novel, but not relevant to the user, can have a negative effect. Recently, in Ekstrand et al. [7], these metrics have been evaluated in a controlled user study. Ekstrand et al. [7] found that an overzealous maximization of novelty actually has a negative effect on user satisfaction and trust in a system as it leads to irrelevant recommendations. While, there is a growing consensus that balancing metrics, when done sensibly, has the potential to improve user engagement and expose more inventory, the community still lacks a computational framework for increasing the both novelty and diversity of recommendations while selecting items that are still likely to be rated highly.

In this paper, we propose the Minimum Conductance Dissimilarity Cut (MCDC) algorithm that increases the novelty and diversity of recommendations from any recommendation algorithm given a partial set of similarities between items. To do this, we notice that the metrics, novelty and diversity, have a special structure, where novelty is a global property (each item has a novelty value) and diversity is a local property (each set of items has a diversity value). This structure can be represented as a graph connectivity property called conductance which is the ratio of a local property (edges across a cut) and global property (weighted in-degree of vertices in the cut). The problem of increasing novelty and diversity of a recommendation set can be formulated as finding a low-conductance cut on a dissimilarity graph. However, we must ensure that this cut still extracts highly recommended items from the recommendation algorithm, and we do this by weighting conductance by "scores" from the recommendation algorithm.

To illustrate MCDC, consider an example movie recommender system. The first problem that MCDC mitigates is what we call the *Pigeonhole* problem, where a recommender learns that you like comedies from a few initial ratings and then, subsequently, repeatedly recommends comedies. While the first few comedy recommendations may be welcome, eventually, you will tire of the lack of variety. The other problem that MCDC addresses is the *Blockbuster* problem, where a recommender may favor the most-popular, high-budget movies excluding the less-popular independent movies. To make this challenging, novelty and diversity, are potentially competing objectives. Consider a corpus of ten movies where eight are dramas and two are nearly identical comedies. In this example, the most novel recommendation set of size two is selecting the two comedies which is also the least diverse recommendation set. It is also easy to a construct the converse example, where two popular movies that are representative of their genres are recommended leading to high diversity but poor novelty. However, it is not enough to address both the Blockbuster and Pigeonhole problems, we also have to ensure that the recommendations are movies that the user will likely rate highly.

Simultaneously optimizing for these three objectives is challenging as they are limited by computational hardness results. Optimizing for diversity, even without novelty, is NP-Hard. We address this problem by relaxing diversity to a metric called *coverage*, which essentially constrains that not only is the recommendation set is diverse but also is “similar" to its complement. In other words, coverage is a measure of representativeness which, when maximized, naturally leads to higher diversity. This relaxation allows us to pose the problem as a spectral problem (solving for eigenvectors), which can be solved at scale with fixed-point iteration which is a common programming model for many large-scale graph analytics frameworks [20].

Existing solutions to this problem have been explored in a line of work called hybrid recommender systems [15], that study aggregating the results from multiple recommendation algorithms. For general recommendation criteria, this problem is hard and one state-of-the-art solution (Ribeiro et al.) relies on evolutionary algorithms to iteratively move towards a Pareto-optimal ranking by weighting each algorithm differently. In this work, we find that the global-local structure of the novelty-diversity problem allows us to approximate Pareto-optimal solutions as a linear algebra problem faster than hybrid recommenders which model the recommendation algorithms as black boxes. While we do not address the same generality of problems as in hybrid recommendation, in fact only a specific definition of novelty and diversity, we argue that this special case is an important one for many domains.

To evaluate MCDC, we run a series of experiments on three datasets: the latest MovieLens dataset [1], Jester [13], and CAFE [14]. For each of these datasets, we have a base recommendation algorithm, and we improve the diversity and novelty of its recommendations. We show that MCDC provides recommendations sets with up to 20% higher combined geometric mean values (of novelty, diversity, and score) and that MCDC is up to an order of magnitude faster than previous methods.

To summarize our contributions:

1. We propose the MCDC algorithm. Given any base recommendation algorithm, it improves the novelty and the diversity of the recommendation set by solving a graph partition problem on the dissimilarity graph.
2. Recognizing that the problem is NP-Hard, we use a real-valued eigenvector relaxation to approximate a Pareto-optimal balance between novelty, coverage (which is related to diversity), and the score from the base algorithm.
3. We evaluate this algorithm on datasets from three recommender systems (Movie Lens, Jester [13] and CAFE [14]) and find that it out performs the hybridization approach [15], and a recently proposed submodular coverage approach [21].

2. RELATED WORK

The earliest formalizations of novelty and diversity were studied in the informational retrieval (IR) community [4, 23, 24, 22, 6]. In IR, the problem is studied from the perspective of search results. The most relevant search results may be highly redundant. In recommender systems, these metrics are now well established as desirable [8]. Presenting users diverse recommendations has been long argued to have many benefits including increased user satisfaction, and it exposes more products in the long-tail [1, 16].

Recently, the prevalence of news recommendation sites (e.g., Yahoo news, Google news) has been a significant motivating factor for the development recommendation algorithms that diversify [21]. News articles are often highly clustered with multiple articles relating to exactly the same topic/event. Yue et al. [21] looked at the problem of diversification in the Multi-arm Bandit Setting. In particular, they studied a problem called linear submodular contextual bandits. Each item in the recommender system corresponds to a Bandit arm, and there is a diminishing reward for pulling arms that are too similar. When you can pose an optimization criterion, such a diversification, as a submodular problem, they showed this could be applied in the MAB setting with guarantees. In this work, we do not consider the problem of “exploration" as in [21] and are eager to investigate this in future work. However, we do compare against the diversification algorithm used in [21] without the bandit reward evaluation.

There are a variety of emerging recommender system domains, such as online civic engagement, MOOCS, and discussion systems, where this research can be applied [14, 11]. In these applications, the goal of the recommender system is often to select comments from other users to present to a new user. Recent research suggest that novelty and diversity can catalyze group creativity [2, 12, 18, 19]. In this work, we actually evaluate against data in one such recommender system, CAFE, that adaptively presents comments to users to rate. We use MCDC to increase the novelty and diversity of these recommendations.

In terms of user studies, Ekstrand et al. [7] presents the most comprehensive results where they designed an experiment in which they presented users with different lists of movie recommendations. They had a “control" recommendation set using a standard Collaborative Filtering algorithm, and presented alternative sets with diverse ideas. They then asked users which list they prefer, their satisfaction, and their trust of the recommender system. They found diversity (defined in that work as lack of redundancy in recommendations) had a positive effect where users were more satisfied with diverse recommendations. Interestingly enough, Ekstrand et al. found that overly novel recommendations actually had a negative effect where the recommendations become increasingly irrelevant. This motivates the problems studied in this work and in Ribeiro et al. [15] about balancing criteria.

Ribeiro et al. [15] proposed a model for balancing potentially competing objectives from many recommendation algorithms. The meta-algorithm (that optimizes weighting between the various constituent recommendation algorithms) used in this work is the Strength Pareto Evolutionary method [25]. In this model, each recommendation algorithm independently scores an item, and then in each iteration the solver hybridizes and evaluates the “fitness" of the hybridization in terms of its distance to Pareto-optimality. This model is very general and treats each recommendation algorithm as a black box, and as a result is often very computationally intensive. In this work, we explore the case of specific definitions of novelty and diversity that allow us to exploit the structure of the problem. Our formulation allows us to solve this problem with linear algebra which is much more amenable to large scale implementation. We evaluate MCDC against this algorithm and find that it optimizes the problem in an order of magnitude less time, and with higher novelty/diversity in 5 out of the 6 datasets on which we experimented.

We posed the multi-objective optimization problem as a ratio objective. The seminal work of Shi and Malik on the NCUT problem [17] shows that hard combinatorial ratio objectives can be relaxed as a spectral problem when they have a local-global structure. The work by Kannan et al. [9] looked at this in detail. They found that the quality of a clustering could be measured by a property called conductance; the ratio of edges across the cut over the total

[1]Dataset Released April 01, 2015, `http://grouplens.org/datasets/movielens/`

connectivity within the cut. Kannan et al. proposed many different methodologies to find low-conductance clusterings/cuts including a variation of Shi and Malik's NCUT using a spectral decomposition of the graph.

3. PROBLEM DESCRIPTION

We first characterize the problem of novelty and diversity, and then describe the problem statement.

3.1 Definitions

Recommender System Model: Let I be the set of items, and U be the set of users. Each user $u \in U$ rates an item i from the set I on a scale from $[0, 1]$. There is a *base recommendation algorithm*, $\mathcal{R}$, such that it assigns a real-valued score to each $i \in I$ for every user $u \in U$. We denote this score as $\mathcal{R}(i, u)$ which we also model on a scale from $[0, 1]$. $\mathcal{R}$ abstracts the choice of recommendation algorithm from the subsequent definitions. At the minimum, if we have a rating matrix of users and items, and a recommendation algorithm, our methodology can apply.

Item-Item (Dis)similarity: For the set of items I, we define a function

$$S : I \times I \mapsto [0, 1]$$

S is called a similarity metric, if it defines a semi-metric space over I. Since S is bounded between $[0, 1]$, we define the dissimilarity between two items as $1 - S(i, j)$. In many settings, we may not have the similarity between all pairs of items. We say that our similarity metric S is partially observed if it is only defined for a subset $M \subset (I \times I)$. For example, if our similarity metric measures the correlation between ratings given by users who rated both item i and j, we may not have corresponding ratings for all pairs.

Weighted Dissimilarity Graph: The partially observed semi-metric defines a graph where items are vertices and edges are similarity relations. We weight the edges on this graph by the product of the dissimilarity between two items and the sum of their base algorithm recommendation scores. Therefore, this graph, which we denote as G_u is personalized for a given user u.

We can model $I \times I$ as a graph, and in particular, we will look at one such graph called the dissimilarity graph for the user u, G_u. Each item in I is a vertex of the graph. For each $(i, j) \in M$, we add an edge with weight:

$$W_{ij} = (1 - S(i, j))(\mathcal{R}(u, i) + \mathcal{R}(u, j))$$

3.2 Recommendation Set Properties

Let L be a recommendation set, that is, a subset of I that is recommended to a user. Let N be the number of items and k be the number of items in the recommendation set. We define the following properties of the recommendation set L.

Novelty: Novelty for an item is defined as total dissimilarity of an item with all other items in I. For an item j in L:

$$\text{novelty}(j) = \sum_{\forall i \in I} (1 - S(i, j))$$

Therefore, the mean novelty of L is defined as:

$$\frac{1}{k} \sum_{\forall l \in L} \text{novelty(l)}$$

We will use the notation novelty(L) to denote the mean novelty of set L.

Diversity: The diversity the set L is defined as the average dissimilarity of all pairs of items within L.

$$\text{diversity}(L) = \frac{1}{k(k-1)} \sum_{\forall i,j \in L, i \neq j} (1 - S(i, j))$$

Compared to novelty, diversity is a local property of the recommended set that does not require information about the corpus. Given this definition, finding the most diverse set is equivalent to maximal independent set [10], i.e., we can construct a partially observed similarity metric that implies a solution to the most-diverse problem implies a solution to the independent set problem. This problem is not only known to be NP-Hard, but is also known to be APX-Hard which means there does not exist a polynomial time bounded approximation. In part due to this hardness result, many current algorithms such as [21] relax diversity to a quantity called coverage, which instead measures representativeness (See Section 5.1.1).

Coverage: We define "coverage" in terms of our partially observed similarity metric. This definition is inspired by the definitions using $\mathbf{R}^n$ in [21], but are instead, defined on a graph. Let L be a subset of I, and $I - L$ be the complement. The coverage of the subset L is defined as the amount of similarity in $I - L$ accounted for by the set L. In other words, on average, how similar is the set L to its complement:

$$\text{coverage}(L) = \frac{\sum_{r \in L, j \in (I-L)} S(i, j)}{N(N-k)}$$

Due to the normalization the scale of coverage is independent of the size of the recommendation set. We can also define the inverse coverage, which measures the amount of similarity not accounted for by L:

$$\text{coverage}^{-1}(L) = 1 - \text{coverage}(L)$$

The inverse coverage is related to diversity in the following way:

$$\text{coverage}^{-1}(L) \propto N^2 \text{diversity}(I) -$$
$$(k^2 \text{diversity}(L) + (N-k)^2 \text{diversity}(I - L))$$

The inverse coverage measures the average diversity of both sides of the partition L and $I - L$ weighted by their sizes and how this compares to the diversity of the base set. Coverage is a measure of representativeness and intuitively a subset that represents a set of items well will naturally have low-redundancy and be diverse.

Score: Finally, for the subset L, we can also define the score. We define this as the mean value of the scores given by the base recommendation algorithm $\mathcal{R}$. For a user u,

$$\text{score}(L, u) = \frac{1}{k} \sum_{\forall l \in L} \mathcal{R}(l, u)$$

3.3 Problem Statement

Suppose, a user is given a recommendation score for each item $\mathcal{R}(i, u)$ using the base algorithm. We want to find a set of items L such that it simultaneously maximizes novelty, coverage, and score. However, while we can optimize each criterion individually, these objectives are potentially competing. We pose it as the following optimization problem. Let I be a set of items. We have a partially observed similarity metric defined on a subset of the pairs of items M.

We select a subset $L \subset I$ of size k such that it approximates a Pareto-optimal [5] balance between novelty, coverage, and score. The Pareto-optimal solution is a set L of size k such that there does not exist another L' of size k with a higher value on any one of the three criterion, without at least one of the other criterion being lower.

4. GRAPH CUT FORMULATION

In this section, we address the problem from the previous section. We first describe how we can model the items as a graph. Then, we describe a property called conductance which is proportional to a ratio of novelty and inverse coverage. We then describe a spectral algorithm to find a low-conductance cut.

4.1 Graph Conductance

The property that we will leverage is called graph conductance that has been well studied in the context of clusterings and random walks [9]. Conductance describes the convergence rate of a random walk on a graph, where random walks on graphs with higher conductance converge faster. Intuitively, graphs that have large cliques connected by only a few edges will have low conductance; that is random walks will get "stuck" in the cliques. In our setting, these are sets of mutually dissimilar items that are very similar to the other items (few edges in the dissimilarity graph).

DEFINITION 1. *Let $C \subset V$ be a cut of the graph. The conductance of C, $\phi(C)$ is defined as follows:*

$$\phi(C) = \frac{\|\partial C\|}{\|C\|}$$

where $\|\partial C\|$ is the sum of edge weights across the cut:

$$\|\partial C\| = \sum_{i \in C, j \in V \setminus C} W(e_{ij})$$

and $\|C\|$ is total in-degree of all of the vertices in c:

$$\|C\| = \sum_{i \in C} \sum_{j \in V} W(e_{ij})$$

The lowest conductance cut (see [9]) is defined as the cut C.

$$\arg \min_{C \in G} \phi(C)$$

There have been numerous algorithmic techniques to find these cuts. What is relevant to us is that conductance is a ratio that balances the weight of the cut with the connectivity of the vertices inside the cut. In other words, the numerator measures the relationship with the complement, and the denominator measures the relationship with the vetrex set. These are exactly the local-global relationships that novelty and coverage measure.

4.2 Novelty

We can show that optimizing novelty for a fixed coverage and expected score:

$$\arg \max_{L \in 2^I, |L|=k} novelty(L)$$

is equivalent to optimizing the denominator of conductance, if L is a cut on the dissimilarity graph:

$$\arg \max_{L \in 2^I, |L|=k} \|L\|$$

PROPOSITION 1. *Let $L \subset I$ be a cut on the graph G_u. $\|L\|$ is proportional to the novelty(L).*

PROOF SKETCH. $\|L\|$ is the sum of all the weighted edges that connect to the vertices in L. On the dissimilarity graph, this is $\sum_{i \in L, j \in I} 1 - S(i,j)$. Up to the normalization by the size of L this is proportional to novelty(L). □

4.3 Coverage

Next, if we fix novelty and expected score, we can minimize the inverse coverage:

$$\arg \min_{L \in 2^I, |L|=k} \text{coverage}^{-1}(L)$$

This is equivalent to minimizing the numerator term of conductance if L is a cut on the dissimilarity graph:

$$\arg \min_{L \in 2^I, |L|=k} \|\partial L\|$$

PROPOSITION 2. *Let $L \subset I$ be a cut on the graph G_u. $\|\partial L\|$ is proportional to the coverage$^{-1}(L)$.*

PROOF SKETCH. $\|\partial L\|$ is the sum of all the weighted edges that go across the cut. coverage$^{-1}(L)$ is defined as $1-$coverage(L). If we scale this by size of the cut:

$$\lambda = (N-k)k$$

Then we can show that the inverse coverage scaled by λ is:

$$\lambda \cdot \text{coverage}^{-1}(L) = \lambda - \lambda * \text{coverage}(L)$$

$$\lambda \cdot \text{coverage}^{-1}(L) = \sum_{(i,j) \subset M : i \in L, j \in I-L} 1 - S(i,j)$$

Up to the normalization by the size of λ, the inverse coverage is proportional to:

$$\lambda \cdot \text{coverage}^{-1}(L) = \mid \partial L \mid$$

□

4.4 Score Weighting

Next, we analyze conductance with the score weighting when the other two criteria are fixed. Each edge in the dissimilarity graph has weight:

$$W_{ij} = (1 - S(i,j))(\mathcal{R}(u,i) + \mathcal{R}(u,j))$$

Therefore, this weighting increases the expense for cutting edges between highly scored items ($\|\partial L\|$). Also, when L contains higher scored items $\|L\|$ is larger. As a result, with this weighting $\phi(L)$ is larger when the cut has higher scored items.

4.5 The Pareto-optimal Solution

What we have shown is that finding the lowest conductance cut in the graph:

$$\arg \min_{C \in G} \phi(C)$$

Is equivalent to solving this problem, that optimizes the ratio of novelty and diversity, scaled by the score.

$$\arg \max_{L \in 2^I, |L|=k} score(L,u) \frac{novelty(L)}{\text{coverage}^{-1}(L)}$$

When solved exactly, this problem yields a Pareto-optimal solution. From this formulation, we can clearly see that the naive solution is to try all of the $\binom{N}{k}$ subsets. However, we will see that a relaxation of this problem can give us an approximate solution in $O(\mid M \mid)$ time.

5. GRAPH CUT RELAXATION

In this section, we discuss how to solve a relaxation of the formulation in the previous section.

5.1 Finding Low-Conductance Cuts

Kannan et al. showed that the problem of finding low conductance cuts could be posed as Minimum Normalized Cut proposed by Shi and Malik. Let N be the total number of vertices. Let p be an binary indicator vector in $\{-1,1\}^N$ that indicates whether which side of the partition a vertex is on. We represent the edge weights W as an $N \times N$ matrix, where component (i,j) is the weight of the edge between i and j. We normalize W such that each row sums to 1. It follows that the NCUT can be solved by minimizing[2]:

$$\min_p \frac{p^T(I-W)p}{p^Tp}$$
$$\text{subject to: } p^T 1 = 0$$
$$p \in \{-1, 1\}$$

This problem was proven to be NP-Hard in its integral form [17]. However, it can be solved in PTIME if we relax p to be real valued. This relaxation leads to a spectral formulation, i.e the optimal value is an eigenvector. If we relax p to be real-valued, it becomes a eigenvector system:

$$(I-W)p = \lambda p$$

The eigenvector corresponding to the second smallest eigenvalue (p^*), is optimal solution for the relaxed problem [3]. The implication is that if we rank the comments by p_i^* that indicates a stronger chance that the comment is included in the un-relaxed normalized cut. The top components of this ranking gives us the dissimilar subset L.

5.1.1 Complexity

There are many interesting complexity theoretical properties of novelty, diversity, and coverage. We highlight some of the important results that motivate why we need the proposed relaxations proposed.

Diversity: Given a dissimilarity graph, finding the most diverse set for any size k is APX-Hard, that is, there does not exist a polynomial time algorithm that can even approximate this problem within a constant factor. To prove this, we can show that solving this problem implies solving the APX-Hard maximal independent set problem; where we find the maximum set of vertices that do not have an edge between them. We can see that on an unweighted *similarity graph*, independent set is the most diverse set. Therefore, if we construct a graph with a maximal independent set of size k (which by the definition of "NP" we can do in PTIME), the most diverse set of size k is the maximal independent set.

Coverage: Coverage can be interpreted as a relaxation of diversity, avoiding the APX-Hard result. For example, in Yue et al. [21], the submodular optimization algorithm lends itself to a provable $1 - \frac{1}{e} \approx 63\%$ bound which is independent of the data. Similarly, our definition of coverage lends itself to Minimum K-Cardinality Cut reduction that gives a bound that is essentially $\frac{k}{k-1}$, which is independent of the data and dataset size.

Novelty: Since novelty is a global property it can be optimized in PTIME. In fact, at worst, novelty requires $O(N^2)$ time where you compare all pairs.

MCDC Time Complexity: Modern large scale analytics frameworks are well suited to solving eigenvector problems at scale, the most notable of which is PageRank. Power-iteration can be used to solve this problem iteratively. Since the graph is sparse, each iteration will require $\mid M \mid$ multiplications. The convergence rate of power-iteration in this problem is governed by the ratio of the 2nd and 3rd smallest eigenvalues $c = \frac{1-\lambda}{1-\lambda_{i-1}}$. Thus, the complexity of the problem is $O(c \mid M \mid)$. $\mid M \mid$ is at most N^2 when the similarity metric is fully observed. What is fascinating about MCDC is that we can optimize, albeit approximately, for both novelty *and* coverage in at most N^2 time.

On the other hand, the existing hybrid recommender system alternatives are much less efficient. If we analyze the SPEA-2 algorithm used in [15], for the partially observed similarity metrics the time complexity for one iteration is $O(MN^2)$, which is quadratically more than our method. Our approach exploits sparsity in the problem, and exploits the local-global nature of the objective to make these improvements.

[2]Here I refers to the identity matrix not the set of items

[3]The smallest eigenvalue corresponds to the trivial cut (taking the entire graph) which has cost 0.

5.2 Pruning Low-Score Items

While more efficient than the general hybrid recommender system alternatives [15], the number of edges in the graph $\mid M \mid$ might be very large (quadratic in the items), which makes running this optimization for each user relatively expensive. To avoid this cost, we can prune low score items. More precisely, if $\mathcal{R}(i,u)$ is the score that the base recommendation algorithm gives to each item, let T be the top-k' scored items. We can run our optimization on T instead of I. Of course, this will add additional approximation error to the result and we evaluate sensitivity to k' in our experiments.

5.3 Summary and Algorithm

Algorithm 1: MCDC

Data: I := set of items
Data: R(u,I) := scores for each item for user u
Data: M := set of pairs of items with similarity weights
Data: S(i,j) := similarity metric
Data: k' a low-score pruning threshold
Result: A subset L of size k, that has high coverage, novelty, and score

1. Create graph G, where each vertex is one of the top k' scored items and add an edge for each $m \in M$ with weight:

$$(1 - S(i,j))(R(u,i) + R(u,j))$$

2. Represent this graph as an adjacency matrix W with each weighted edge as a component (i,j).
3. Transform the adjacency matrix W by normalizing each of its rows to sum to 1.
4. Find the second smallest eigenvector of $I - W$.
5. Rank each element i by the component of the eigenvector.
6. Return the top k items.

To summarize the last two section, we show that the Pareto-optimal solution to balancing novelty, coverage, and score is the lowest conductance cut on a weighted dissimilarity graph. Finding the lowest conductance cut is NP-Hard, however, there is a polynomial time approximation algorithm based on the Normalized Cut that can give a solution. This solution is a spectral approximation that can be solved at scale and in distributed environment. However, if we wish to run this algorithm online, even this solution might be too slow. Therefore, we can prune low-score items first and optimize on the subset. This algorithm, Minimum Conductance Dissimilarity Cut (MCDC), is listed in Algorithm 1.

6. RESULTS

We evaluate the efficacy of MCDC to increase the novelty and diversity of a recommendation set. The organization of the results is as follows: (1) we describe the experimental setup including the datasets and the similarity metrics used, (2) we evaluate our algorithm against alternatives in terms of optimizing for novelty, diversity, and score, (3) we illustrate the benefits of this optimization in an end-to-end experiment, and (4) we describe the sensitivity of the algorithm to the choice of parameters.

Algorithm	CRCV1	MCAFEV1	MCAFEV2	CRCV2	OSL	JESTER	MoveLens
Base	0.693	0.695	0.687	0.672	0.715	0.680	0.583
TopNovelty	0.809	0.803	0.746	0.802	0.767	0.736	0.444
Submodular	0.707	0.708	0.689	0.726	0.676	0.695	0.571
Hybrid	0.796	0.801	**0.818**	0.780	0.770	0.792	0.599
MCDC	**0.895**	**0.858**	0.792	**0.884**	**0.820**	**0.855**	**0.725**

Table 1: For each of the algorithms, we list the geometric mean of the novelty, diversity, and score of the recommendation sets. We find in all but one dataset, MCDC gives improved performance compared to the alternatives.

6.1 Experimental Setup

We evaluate our method with three recommender system datasets, MovieLens, Jester [13] and CAFE [14]. Our goal is to show that MCDC increases the novelty and diversity of recommendations on these datasets given a base algorithm. Jester is a joke recommender system which uses the Eigentaste algorithm to recommend jokes, for this dataset we use the Eigentaste algorithm as a base algorithm. We use the latest MovieLens (small) dataset, in which we also apply the Eigentaste algorithm as the base recommendation algorithm. CAFE is an online discussion tool that suggests comments from other users current using a VarianceSampling algorithm [14]. We evaluate our method on one dataset from MovieLens, one dataset from Jester, and five CAFE datasets. Each CAFE dataset is from a different instance of CAFE with a different main discussion topic. Furthermore, each dataset has a different sparsity of similarities.

	Domain	No. Items	Sim Pairs
CAFE: CRC-V1	Politics	656	100790
CAFE: MCAFE-V1	Education	74	830
CAFE: MCAFE-V2	Education	53	104
CAFE: CRC-V2	Politics	286	23044
CAFE: OSL	Education	231	11234
Jester	Jokes	100	8650
MovieLens	Movies	8570	1446210

6.1.1 *Similarity Metrics*

In our evaluation, the similarity metric S is defined as the squared correlation of two item's ratings. We list ratings for each item i and j for which the same user rated both. Then, we take the correlation of these ratings. In this work, we handle positive and negative correlations symmetrically, and since the correlation is between $[-1, 1]$, the square correlation is between $[0, 1]$ satisfying our assumptions on the similarity metric. Since we will not always have ratings for all pairs, this metric is partially observed.

6.1.2 *Alternative Algorithms*

For comparison, we look at the following alternative techniques:
Base Algorithm: We select the top-k items using the base recommendation algorithm either Eigentaste or VarianceSampling.
Top Novelty: We directly compute the novelty for each idea based on the metric discussed in Section 3. Then, we select the items with the highest novelty.
Submodular Coverage: We apply a submodular representative subset selection as proposed in Yue et al [21]. The objective function we used is a coverage objective, ie. select a subset that maximizes the number of items in the corpus that are within β similarity with at least one member of the subset.
Hybrid Recommender: We apply the methodology of Ribeiro et al. [15] to balance the three optimization criteria using their parameter choice but our metrics. The hybrid algorithm uses a hybridization of the base algorithm, top novelty, and submodular coverage.

To evaluate the novelty, diversity, and base algorithm score, we use each of these techniques to recommend items to all of the users in the dataset and then average their novelty, diversity, and score.

6.2 Experiment 1. Novelty, Diversity, Score

In the first experiment, we compare the performance of each of the algorithms. For each dataset, we recommend a set of 10 items and evaluate the metrics: novelty, diversity, and score. Our ideal result is that an algorithm should have a high value for each of the metrics. In Table 1, we list the geometric mean of each of the algorithms on the metrics. For multi-objective optimization, the geometric mean is preferred to the arithmetic mean since it measures the area spanned by the frontier [3]. We find that MCDC has the highest geometric mean in all but one dataset (MCAFEV2). MCAFEV2 is the smallest dataset and in this case the evolutionary algorithm used in the hybrid recommender was able to converge quickly to a better optimum. On the other hand, if we look at the larger datasets CRCV1 and CRCV2, we see that our technique gives a 12% improvement and 13% improvement respectively. For the MovieLens dataset, there was a 17% improvement. The key difference is that the hybrid strategy treats each of the recommendation algorithms as a black box, which can give good results when the dataset is small. However, when the dataset is larger, thus increasing the degrees of freedom for the recommender, our results suggest that MCDC is able to exploit the structure of the problem.

6.2.1 *Novelty*

To better understand where MCDC makes its gains, we can compare the algorithms on each of the criteria. We first look at novelty in Figure 1. Clearly, the TopNovelty algorithm will give us the largest possible novelty for recommendation set. However, we find that novelty of the recommendations given by MCDC are 87% of this maximum value. In fact, for all but one dataset MCDC gives the second highest novelty value. As before, MCAFEV2 is the exception. On average MCDC gives a novelty value that is 6% higher than that given by the hybrid recommender system.

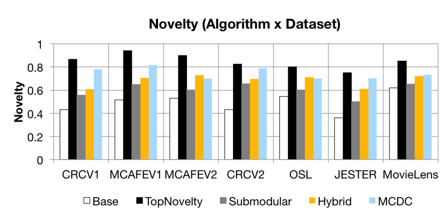

Figure 1: While directly optimizing for novelty gives the highest novelty value for the recommendation set, MCDC gives the second highest value in all but one of the datasets.

6.2.2 *Diversity*

The next criterion that we will explore in more detail is diversity (Figure 2). The submodular coverage technique proposed in Yue et al [21], greedily optimizes for diversity. As with novelty, we compare MCDC to the technique that individually optimizes this metric. We find that as before in all but one of the datasets, MCDC gives the second highest value (92%) of the submodular coverage.

However, with diversity, our gains w.r.t the hybrid recommender are not as significant. We find that MCDC gives recommendations that are only 3% more diverse than the hybrid recommender. We believe this is due to the definition of coverage optimized in MCDC which is the amount of similarity accounted for in the complement. In a datasets where there are conflicting transitive relationships (i.e., a is similar to b, b is similar to c, c is dissimilar to a), this definition may lead to cuts that encourage conflicts on one side of the partition. However, this problem seems to be mitigated when the graph is sparse, as in the MovieLens dataset with a larger gain of 10%. We hope to explore this problem in more detail in future work.

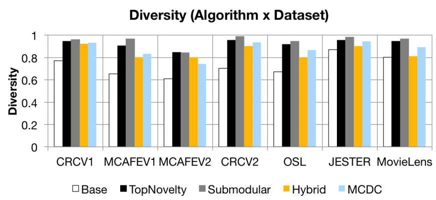

Figure 2: Submodular coverage greedily optimizes for diversity. In all but one of the datasets, MCDC gives a recommendation set that has 92% of the diversity of the direct optimization

6.2.3 *Score*

Finally, we evaluate how well do the algorithms respect the scoring of the base algorithm. In Figure 3, we evaluate the scores of each of the algorithms. We find that compared to the base algorithm, MCDC gives a recommendation with 90% of the mean score. On this metric, we find that MCDC makes its largest gains compared to the hybrid recommender of 10%.

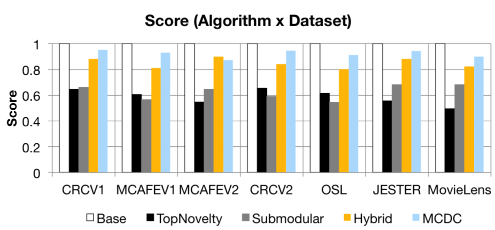

Figure 3: We normalize the highest score in the algorithm to 1, and plot the scores of the recommendation sets for the alternative algorithms. MCDC gives recommendations that are only 10% lower than the base algorithm

6.2.4 *Exposure*

We also explore how MCDC increases the exposure of less popular items in the corpus. Recall, that for each user in the dataset, we recommend 10 items. We explore the recommendation frequency of different items in the Jester dataset. To measure this frequency, we look at the number of times an item is recommended per 1000 users. Over all the items, there is a distribution of recommendation frequencies. We list the 90th percentile (Most Exposed), the median, and the 10th percentile frequency (Least Exposed). We find that when MCDC is applied with Eigentaste it distributes exposure of items for ratings more evently than Eigentaste alone.

	Most Rec/1000u	Median Rec/1000u	Least Rec/1000u
Eigentaste	360	140	72
MCDC	280	220	120

6.3 Experiment 2. Efficiency and Scalabilty

In the next experiment, we compare three algorithms from related work (Yue et al., Ribeiro et al., and ours) on run time. In Figure 4, we plot the running times of the algorithms on a logarithmic scale. MCDC gives a running time between submodular coverage and the hybrid recommender. We find that hybrid recommender requires an order of magnitude more time to achieve its results (14.2x).

On the other hand, the submodular optimization is 2.5x faster than MCDC on average. These gains increase for larger datasets such as MovieLens, where it is 6.4x faster. This is understandable from our complexity result in the previous section. Multicriteria optimization is markedly harder than optimizing for a single criterion. Submodular coverage only optimizes for coverage thus is faster.

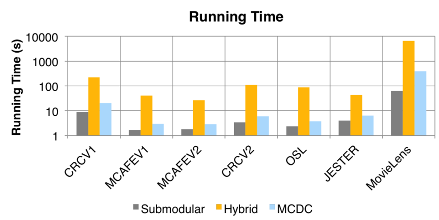

Figure 4: We compare the running time of our algorithm against submodular coverage and the hybrid recommender system. We find that our algorithm is an order of magnitude faster than the hybrid recommender, yet also gives improved recommendations.

6.4 Experiment 3. Pruning

In the previous section, we described an optimization to MCDC by pruning the recommendation set to a fraction p of highest scored items first and then running the optimization in the subset. We evaluate the performance degradation of MCDC when we apply this optimization as a function of p in Figure 5. We plot the geometric mean of the three criteria: novelty, diversity, and score for the CRCV1 dataset and vary p from the full dataset to 10% to the full dataset 100%. On this dataset, for all of the values that we experimented with, we found that that this technique gives a result that is never worse than the base algorithm (though theoretically possible). In fact, we find that we can run the algorithm on only 22% of the data and retain 90% of its benefits.

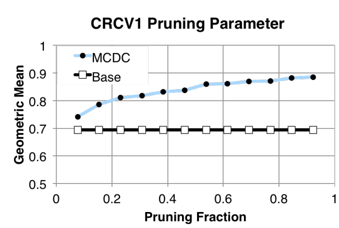

Figure 5: The novelty of the initial two ideas affects the number of ratings a user will subsequently give and the quality of an idea that the will subsequently submit.

6.5 Discussion

These results are promising and an open question is how increasing novelty and diversity will ultimately affect participation and engagement. We present some preliminary results from an online experiment by applying the recommendation technique in January

	Before	After	P-Value
Ratings/User	3.35 (N=12482)	4.59 (N=112)	**<0.001**

2015 on the California Report Card [4]. The California Report Card is part of the CAFE family (datasets we used in this paper) and is a similar online comment recommendation platform. We evaluated this on 112 participants from January 2015 to March 2015. We compare the statistics of these participants to those who visited the application prior to January. We found that indeed there was a statistically significant increase in the number of ratings/user (3.35 to 4.59, $p < 0.001$ Wilcoxon Rank-Sum). These results suggest further investigation with an A/B test to confirm these effects.

7. CONCLUSION

A number of recent works have argued that recommendations that are more novel and diverse lead to increased user participation, engagement, and satisfaction. While, there is a growing consensus that these metrics have a potential to improve recommendation quality and user engagement, the community still lacks a computational framework for increasing the novelty and diversity of a recommendation set from a general algorithm. In this paper, we propose MCDC an algorithm that given a base recommendation algorithm improves the novelty and the diversity of the recommendation set by solving a graph partition problem on the dissimilarity graph. We show that problem approximates a Pareto-optimal balance between novelty, coverage (which is related to diversity), and the score from the base algorithm. MCDC applies to any base recommendation algorithm and addresses the Pigeonhole and Blockbuster problems.

The existing approach to this problem is to use a hybrid recommender system that aggregate multiple recommendation criteria. The problem is that these techniques model these recommendations as black-boxes and often rely on iterative (and very expensive) optimization techniques like evolutionary algorithms or simulated annealing. MCDC outperforms an existing hybrid recommender approach to this problem in both computational efficiency and in the novelty, diversity, and score of the recommendations; which we argue is due to how it exploits the global-local structure of the novelty-diversity trade-off. Another important contribution of this work was to evaluate the benefits, in terms of user participation and engagement, of increasing novelty and diversity in a real system. These results are promising and suggest numerous avenues for future work. We are also interested in exploring other similarity metrics, i.e., using tags to compare the similarity between items, and also other datasets. We also will explore the question of timeliness and making recommender systems robust to "fad" problems, and weighting our graph conductance terms accordingly. Our code is available at: `github.com/sjyk/mcdc`.

Acknowledgements: This work was supported in part by the Blum Center for Developing Economies and the Development Impact Lab (USAID Cooperative Agreement AID-OAA-A-12-00011) as part of the USAID Higher Education Solutions Network, UC Berkeley's Algorithms, Machines, and People Lab, and the UC CITRIS Data and Democracracy Initiative.

8. REFERENCES

[1] C. Anderson. *The long tail: Why the future of business is selling less of more*. Hachette Digital, Inc., 2006.

[2] C. R. Aragon and A. Williams. Collaborative creativity: a complex systems model with distributed affect. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1875–1884. ACM, 2011.

[3] J. Borwein. The geometry of pareto efficiency over cones. *Optimization*, 11(2):235–248, 1980.

[4] J. Carbonell and J. Goldstein. The use of mmr, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 335–336. ACM, 1998.

[5] Y. Censor. Pareto optimality in multiobjective problems. *Applied Mathematics and Optimization*, 4(1):41–59, 1977.

[6] C. L. Clarke, M. Kolla, G. V. Cormack, O. Vechtomova, A. Ashkan, S. Büttcher, and I. MacKinnon. Novelty and diversity in information retrieval evaluation. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 659–666. ACM, 2008.

[7] M. D. Ekstrand, F. M. Harper, M. C. Willemsen, and J. A. Konstan. User perception of differences in recommender algorithms. In *Proceedings of the 8th ACM Conference on Recommender systems*, pages 161–168. ACM, 2014.

[8] J. L. Herlocker, J. A. Konstan, L. G. Terveen, and J. T. Riedl. Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems (TOIS)*, 22(1):5–53, 2004.

[9] R. Kannan, S. Vempala, and A. Vetta. On clusterings: Good, bad and spectral. *Journal of the ACM (JACM)*, 51(3):497–515, 2004.

[10] M. W. Krentel. The complexity of optimization problems. In *Proceedings of the eighteenth annual ACM symposium on Theory of computing*, pages 69–76. ACM, 1986.

[11] S. Krishnan, J. Patel, M. J. Franklin, and K. Goldberg. A methodology for learning, analyzing, and mitigating social influence bias in recommender systems. In *ACM Conference on Recommender Systems*, 2014.

[12] P. J. Ludford, D. Cosley, D. Frankowski, and L. Terveen. Think different: increasing online community participation using uniqueness and group dissimilarity. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 631–638. ACM, 2004.

[13] T. Nathanson, E. Bitton, and K. Goldberg. Eigentaste 5.0: constant-time adaptability in a recommender system using item clustering. In *Proceedings of the 2007 ACM conference on Recommender systems*, pages 149–152. ACM, 2007.

[14] M. Nelimarkka, B. Nonnecke, S. Krishnan, T. Aitamurto, D. Catterson, C. Crittenden, C. Garland, C. Gregory, C.-C. A. Huang, G. Newsom, et al. Comparing three online civic engagement platforms using the spectrum of public participation framework.

[15] M. T. Ribeiro, A. Lacerda, A. Veloso, and N. Ziviani. Pareto-efficient hybridization for multi-objective recommender systems. In *Proceedings of the sixth ACM conference on Recommender systems*, pages 19–26. ACM, 2012.

[16] S. Sandoval. Novelty and diversity enhancement and evaluation in recommender systems. *PhD Thesis*, 2012.

[17] J. Shi and J. Malik. Normalized cuts and image segmentation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22(8):888–905, 2000.

[18] L. Thompson. Improving the creativity of organizational work groups. *The Academy of Management Executive*, 17(1):96–109, 2003.

[19] A. Topper. Facilitating student interactions through discursive moves. *The Quarterly Review of Distance Education*, 6(1):55–67, 2005.

[20] R. S. Xin, J. E. Gonzalez, M. J. Franklin, and I. Stoica. Graphx: A resilient distributed graph system on spark. In *First International Workshop on Graph Data Management Experiences and Systems*, page 2. ACM, 2013.

[21] Y. Yue and C. Guestrin. Linear submodular bandits and their application to diversified retrieval. In *Advances in Neural Information Processing Systems*, pages 2483–2491, 2011.

[22] M. Zhang and N. Hurley. Avoiding monotony: improving the diversity of recommendation lists. In *Proceedings of the 2008 ACM conference on Recommender systems*, pages 123–130. ACM, 2008.

[23] Y. Zhang, J. Callan, and T. Minka. Novelty and redundancy detection in adaptive filtering. In *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 81–88. ACM, 2002.

[24] C.-N. Ziegler, S. M. McNee, J. A. Konstan, and G. Lausen. Improving recommendation lists through topic diversification. In *Proceedings of the 14th international conference on World Wide Web*, pages 22–32. ACM, 2005.

[25] E. Zitzler, M. Laumanns, and L. Thiele. Spea2: Improving the strength pareto evolutionary algorithm, 2001.

[4] http://californiareportcard.org