OPT2023: 15th Annual Workshop on Optimization for Machine Learning

Fair Representation in Submodular Subset Selection: A Pareto Optimization Approach

Adriano Fazzone ADRIANO.FAZZONE@CENTAI.EU
CENTAI Institute, Turin, Italy

Yanhao Wang YHWANG@DASE.ECNU.EDU.CN
East China Normal University, Shanghai, China

Francesco Bonchi FRANCESCO.BONCHI@CENTAI.EU
CENTAI Institute, Turin, Italy
Eurecat, Barcelona, Spain

Abstract

In this paper, we study a novel multi-objective combinatorial optimization problem called *Submodular Maximization with Fair Representation* (SMFR), which selects subsets of bounded costs from a ground set such that a submodular (utility) function f is maximized while a set of d submodular (*representativeness*) functions $g_1, \ldots, g_d$ are also maximized. SMFR can find applications in machine learning problems where utility and representativeness objectives should be considered simultaneously, such as social advertising, recommendation, and feature selection. We show that the maximization of f and $g_1, \ldots, g_d$ might conflict with each other, so that no single solution can approximate all of them at the same time. Therefore, we propose a Pareto optimization approach to SMFR, which finds a set of solutions to approximate all Pareto optimal solutions with different trade-offs between these objectives. Specifically, it converts an instance of SMFR into several submodular cover instances by adjusting the weights of objective functions and provides approximate solutions by running the greedy algorithm on each submodular cover instance. In future work, we will consider how to apply SMFR in real-world problems and extend it to more general cases.

1. Introduction

The problem of subset selection is to select a subset S, under a budget constraint, from a ground set V of items, so as to maximize an objective function f measuring the utility of the subset. This problem arises in a wide range of machine learning applications, such as viral marketing on social media [9, 29], recommender systems [18, 32], data summarization [16, 19], and feature selection [2, 17], to name just a few. A common combinatorial structure in such problems is *submodularity* [11], which naturally captures the "diminishing returns" property that adding an item to a smaller set produces more marginal gains than adding it to a larger set. This property not only captures the desired *coverage* and *diversity* of subsets but also allows the design of efficient approximation algorithms.

Among the various combinatorial optimization problems for subset selection in the literature, maximizing a monotone submodular function subject to a knapsack constraint (SMK) has attracted a lot of attention [4, 5, 7, 8, 12, 15, 27, 28, 37], as it captures common scenarios in which different items have non-uniform costs and the total budget is limited. For a monotone submodular function $f : 2^V \to \mathbb{R}^+$ on a ground set V, a cost function $c : V \to \mathbb{R}^+$ that assigns each item $v \in V$ with a

cost $c(v)$, and a budget $k \in \mathbb{R}^+$, SMK is formally defined as:

$$S^* = \arg\max_{S \subseteq V} f(S) \text{ subject to } c(S) \leq k,$$

where $c(S) = \sum_{v \in S} c(v)$ is the cost of set S computed as the sum of the costs of all items in S.

In many real-world problems, in addition to the primary objective of maximizing the utility function f, it is often essential to take into account the representativeness of different groups of items. For instance, influence maximization (IM) [9] requires selecting a subset $S \subseteq V$ of nodes in a social network, with $c(S) \leq k$, that maximizes the submodular influence spread function. If the information propagated is related to education and employment opportunities, fairness in access to information between protected groups [3, 31] becomes a critical issue to consider. As another example, personalized recommendation requires choosing a set $S \subseteq V$ of items, with $c(S) \leq k$, so as to maximize a submodular utility function denoting its relevance to the user and coverage among all the items. In this setting, one or more advertising agencies may require that their products are well represented in the set S. The above two problems, as well as many subset selection problems with fairness or other representativeness considerations [13, 35], can be formulated as a multi-objective optimization problem of maximizing a monotone submodular *utility* function f and a set of d monotone submodular *representativeness* functions $g_1, \ldots, g_d$, all defined on the same ground set V, subject to a knapsack constraint k:

$$\arg\max_{S \subseteq V \,:\, c(S) \leq k} \big(f(S), g_1(S), \ldots, g_d(S)\big).$$

We call this problem *Submodular Maximization with Fair Representation* (SMFR) since it captures the case where the submodular utility function is maximized while all the submodular representativeness functions are also maximized to avoid under-representing any of them.

Our Contributions. To the best of our knowledge, SMFR is a novel optimization problem, never addressed before (see Appendix A for a detailed discussion of the problems relevant to SMFR and their differences from SMFR). It is easy to see that SMFR is at least as hard as SMK, which cannot be approximated within a factor better than $1 - 1/e$ unless $P = NP$ [10]. However, SMFR is much more challenging than SMK due to its multi-objective nature. By providing a simple counterexample, we show that there might not exist any single solution to an instance of SMFR that achieves an approximation factor greater than 0 to maximize f and $g_1, \ldots, g_d$ simultaneously, even for a special case of $d = 1$. As such, we consider approaching SMFR by *Pareto optimization*. Specifically, we call a set S an (α, β)-approximate solution for an instance of SMFR if $c(S) \leq k$, $f(S) \geq \alpha \texttt{OPT}_f$, where $\texttt{OPT}_f = \max_{S' \subseteq V : c(S') \leq k} f(S')$, and $g_i(S) \geq \beta \texttt{OPT}_{g_i}$ for all $i = 1, \ldots, d$, where $\texttt{OPT}_{g_i} = \max_{S' \subseteq V : c(S') \leq k} g_i(S')$. An (α, β)-approximate solution S is Pareto optimal if there does not exist any (α', β')-approximate solution for any $\alpha' \geq \alpha, \beta' \geq \beta$ (and at least one is strictly larger). Since computing any Pareto optimal solution to SMFR is still NP-hard, we propose an efficient algorithm to find a set of solutions to approximate the *Pareto frontier* consisting of all Pareto optimal solutions. Our algorithm first uses any existing algorithm for SMK [5, 15, 27, 28, 37] to approximate $\texttt{OPT}_f$ and each $\texttt{OPT}_{g_i}$. Based on the approximations, it transforms an instance of SMFR into multiple instances of the submodular cover problem with different weights on $\texttt{OPT}_f$ and each $\texttt{OPT}_{g_i}$ to represent the trade-offs between f and each g_i. Subsequently, it calls the celebrated greedy algorithm [36] to obtain an approximate solution for each submodular cover

instance. Finally, all the above-computed solutions that are not "dominated"[1] by any other solution are returned as the set $\mathcal{S}$ of at most $O(\frac{1}{\varepsilon})$ approximate solutions to SMFR for any $\varepsilon \in (0, 1)$. When using a δ-approximation algorithm, where $\delta \in (0, 1 - 1/e]$, for SMK, our algorithm provides a set $\mathcal{S}$ such that for any (α, β)-approximate Pareto optimal solution of SMFR, there must exist a corresponding $(\delta\alpha - \varepsilon, \delta\beta - \varepsilon)$-approximate solution of cost $O(k \log \frac{d}{\varepsilon})$ in $\mathcal{S}$.

Paper Organization. The rest of this paper is organized as follows. We introduce the basic concepts and problem formulation in Section 2. Then, our algorithmic framework is presented in Section 3. Finally, we conclude the paper and discuss future work in Section 4. Further discussions on related work and theoretical analysis of proposed algorithms are deferred to Appendices A and B due to space limitations.

2. Preliminaries

Given a positive integer n, we use $[n]$ to denote the set of integers $\{1, \ldots, n\}$. Let V be a ground set of n items indexed by $[n]$. We define a set function $f : 2^V \to \mathbb{R}$ to measure the *utility* $f(S)$ of any set $S \subseteq V$. We consider that f is normalized, i.e., $f(\emptyset) = 0$, monotone, i.e., $f(S) \leq f(T)$ for any $S \subseteq T \subseteq V$, and submodular, $f(S \cup \{v\}) - f(S) \geq f(T \cup \{v\}) - f(T)$ for any $S \subseteq T \subseteq V$ and $v \in V \setminus T$. We then define a set of d normalized, monotone, and submodular set functions $g_1, \ldots, g_d$ on the same ground set V, each measure the *representativeness* $g_i(S)$ of a set $S \subseteq V$ with respect to a given criterion depending on the specific application. We assume that the value of $f(S)$ or $g_i(S)$ for any $S \subseteq V$ is given by an oracle in $O(1)$ time.

In this work, we focus on maximizing a submodular function f and each g_i subject to a *knapsack constraint* (SMK). Let us define the cost function $c : V \to \mathbb{R}^+$ to assign each item $v \in V$ with a positive real number $c(v)$. The cost $c(S)$ of a set $S \subseteq V$ is then defined as the sum of costs for all individual items in S, i.e., $c(S) = \sum_{v \in S} c(v)$. For a given budget $k \in \mathbb{R}^+$, the set $\mathcal{I}_k$ of all feasible solutions that satisfy the knapsack constraint k contains all subsets of V with costs at most k, i.e.,

$$\mathcal{I}_k = \{S \subseteq V : c(S) \leq k\}.$$

Accordingly, the problem SMK in f is indicated as $S_f^* = \arg\max_{S \in \mathcal{I}_k} f(S)$ with the optimal function value $\texttt{OPT}_f = f(S_f^*)$. In addition to maximize f, we consider that each function g_i for $i \in [d]$ should also be maximized to ensure a fair representation, which is denoted as $S_{g_i}^* = \arg\max_{S \in \mathcal{I}_k} g_i(S)$ with $\texttt{OPT}_{g_i} = g_i(S_{g_i}^*)$. Based on the above notions, our main problem, referred to as *Submodular Maximization with Fair Representation* (SMFR), is formulated as the following multi-objective maximization problem:

$$\arg\max_{S \in \mathcal{I}_k} \big(f(S), g_1(S), \ldots, g_d(S)\big).$$

Since SMK is NP-hard and cannot be approximated within a factor $1 - 1/e + \varepsilon$ in polynomial time for any $\varepsilon > 0$ unless $P = NP$ [10], the problem of maximizing f or each g_i individually can only be solved approximately. Furthermore, we provide a trivial example to indicate that the objectives of maximizing f and each g_i might conflict with each other, and there might not exist any $S \in \mathcal{I}_k$ with approximation factors greater than 0 for both of them, even when $d = 1$.

1. A solution S will be dominated by another solution T if the approximation factors α, β of S are both no greater than those of T and at least one is strictly smaller.

Example 1 *Suppose that $d = 1$, $k = 1$, and $c(v) = 1$ for any $v \in V$. For the two functions f and g_1, we have $\texttt{OPT}_f = f(\{v_0\}) = 1$, $\texttt{OPT}_{g_1} = g_1(\{v_1\}) = 1$, and $f(\{v_j\}) = g_1(\{v_j\}) = 0$ for any $j > 1$. In the above* SMFR *instance, there is no set $S \in \mathcal{I}_k$ with $f(S) > 0$ and $g_1(S) > 0$.*

Thus, we introduce a well-known concept for multi-objective optimization, *Pareto optimization* [22, 26], which provides more than one solution with different (best possible) trade-offs between multiple objectives, in SMFR. We call a set $S \in \mathcal{I}_k$ an (α, β)-approximate solution for an instance of SMFR if $f(S) \geq \alpha\texttt{OPT}_f$ and $g_i(S) \geq \beta\texttt{OPT}_{g_i}$ for each $i \in [d]$. An (α, β)-approximate solution S is Pareto optimal if there does not exist any (α', β')-approximate solution for $\alpha' \geq \alpha$ and $\beta' \geq \beta$ (and at least one is strictly larger). Ideally, by enumerating all distinct Pareto optimal solutions (called the *Pareto frontier*), one can obtain all different optimal trade-offs between maximizing f and each g_i. However, computing any Pareto optimal solution is still NP-hard. To circumvent the barrier, a feasible approach to SMFR is to find a set $\mathcal{S}$ of approximate solutions, where for any Pareto optimal solution, at least one solution close to it is included, as shown in Section 3.

3. Our Algorithm: SMFR-SATURATE

To find approximate solutions to an instance of SMFR, we propose to transform it into a series of instances of its corresponding decision problems, that is, to determine whether there exists any (α, β)-approximate solution for it, and then introduce the SATURATE framework first proposed in [13] to approximately solve each instance of the decision problem as *Budgeted Submodular Cover* (BSC), that is, the problem of finding a set S_c^* with the minimum cost such that $f(S_c^*) \geq l$ for some $l \in \mathbb{R}^+$. We now formally define the decision problem and analyze why the transformation follows.

Definition 1 (SMFR-DEC) *Given an instance of* SMFR *and two approximation factors $\alpha, \beta \in [0, 1]$, find a set $S \in \mathcal{I}_k$ such that $f(S) \geq \alpha\texttt{OPT}_f$ and $g_i(S) \geq \beta\texttt{OPT}_{g_i}$ for each $i \in [d]$, or decide that such a set does not exist.*

Assuming that $\texttt{OPT}_f$ and each $\texttt{OPT}_{g_i}$ are known, the above conditions can be equivalently expressed as $\frac{f(S)}{\alpha\texttt{OPT}_f} \geq 1$ and $\frac{g_i(S)}{\beta\texttt{OPT}_{g_i}} \geq 1$. Then, using the truncation technique in [13], SMFR-DEC is converted to decide whether the objective value of the following problem is $d + 1$:

$$\max_{S \in \mathcal{I}_k} F_{\alpha,\beta}(S) := \min\left\{1, \frac{f(S)}{\alpha\texttt{OPT}_f}\right\} + \sum_{i=1}^{d} \min\left\{1, \frac{g_i(S)}{\beta\texttt{OPT}_{g_i}}\right\}. \tag{1}$$

This conversion holds because $F_\alpha(S) = d + 1$ if and only if $f(S) \geq \alpha\texttt{OPT}_f$ and $g_i(S) \geq \beta\texttt{OPT}_{g_i}$, $\forall i \in [d]$. In addition, $F_{\alpha,\beta}$ is a normalized, monotone, and submodular function because the minimum of a positive real number and a monotone submodular function is monotone and submodular [13], and the nonnegative linear combination of monotone submodular functions is monotone and submodular [11]. As such, SMFR-DEC is transformed to BSC on $F_{\alpha,\beta}$.

Since computing $\texttt{OPT}_f$ and $\texttt{OPT}_{g_i}$ is NP-hard, we should use any existing algorithm for SMK [5, 15, 27, 28, 37] to compute their approximations. Suppose that any δ-approximation algorithm for SMK, where $\delta \in (0, 1-1/e]$, is used and $\texttt{OPT}'_f \in [\delta\texttt{OPT}_f, \texttt{OPT}_f]$ and $\texttt{OPT}'_{g_i} \in [\delta\texttt{OPT}_{g_i}, \texttt{OPT}_{g_i}]$, $\forall i \in [d]$ are obtained accordingly. The problem in Eq. 1 is thus relaxed as follows:

$$\max_{S \in \mathcal{I}_k} F'_{\alpha,\beta}(S) := \min\left\{1, \frac{f(S)}{\alpha\texttt{OPT}'_f}\right\} + \sum_{i=1}^{d} \min\left\{1, \frac{g_i(S)}{\beta\texttt{OPT}'_{g_i}}\right\}.^{2} \tag{2}$$

2. When $\alpha = 0$, the first term of $F'_{\alpha,\beta}$ is replaced with 1; when $\beta = 0$, the second term of $F'_{\alpha,\beta}$ is replaced with d.

Algorithm 1: SMFR-SATURATE

Input: (Normalized, monotone, and submodular) set functions $f, g_1, \ldots, g_d : 2^V \to \mathbb{R}$, cost function $c : V \to \mathbb{R}^+$, budget $k \in \mathbb{R}^+$, error parameter $\varepsilon \in (0, 1)$

Result: A set $\mathcal{S}$ of approximate solutions to SMFR

Initialize $\mathcal{S} \leftarrow \varnothing$ and run an SMK algorithm on $f, g_1, \ldots, g_d$ and k to compute $\texttt{OPT}'_f, \texttt{OPT}'_{g_1}, \ldots, \texttt{OPT}'_{g_d}$;

for $\beta \leftarrow 0$; $\beta \leq 1$; $\beta \leftarrow \beta + \frac{\varepsilon}{2}$ **do**

  Initialize $\alpha_{max} \leftarrow 1$, $\alpha_{min} \leftarrow 0$;

  while $\alpha_{max} - \alpha_{min} > \frac{\varepsilon}{2}$ **do**

    Set $\alpha \leftarrow (\alpha_{max} + \alpha_{min})/2$ and define $F'_{\alpha,\beta}(S)$ according to Eq. 2;

    Initialize $S \leftarrow \emptyset$;

    while $\exists v \in V \setminus S$ *such that* $c(S \cup \{v\}) \leq k(1 + \ln \frac{2d+2}{\varepsilon})$ **do**

      $I \leftarrow \{v \in V : c(S \cup \{v\}) \leq k(1 + \ln \frac{2d+2}{\varepsilon})\}$;

      $v^* \leftarrow \arg\max_{v \in I} \big(F'_{\alpha,\beta}(S \cup \{v\}) - F'_{\alpha,\beta}(S)\big)/c(v)$ and $S \leftarrow S \cup \{v^*\}$;

    end

    if $F'_{\alpha,\beta}(S) \geq d + 1 - \frac{\varepsilon}{2}$ **then**

      $\alpha_{min} \leftarrow \alpha$ and $S_{\alpha,\beta} \leftarrow S$;

    else

      $\alpha_{max} \leftarrow \alpha$;

    end

  end

  Add $S_{\alpha_{min},\beta}$ to $\mathcal{S}$ and remove all $S_{\alpha',\beta'}$ with $\alpha' \leq \alpha_{min}$ and $\beta' < \beta$ from $\mathcal{S}$;

end

return $\mathcal{S}$;

Next, the following lemma indicates that SMFR-DEC can still be answered approximately by solving the relaxed problem in Eq. 2.

Lemma 2 *If $F'_{\alpha,\beta}(S) \geq d + 1 - \frac{\varepsilon}{2}$ for any set $S \in \mathcal{I}_k$, then S is a $(\delta\alpha - \frac{\varepsilon}{2}, \delta\beta - \frac{\varepsilon}{2})$-approximate solution to* SMFR. *If there is no set $S \in \mathcal{I}_k$ with $F'_{\alpha,\beta}(S) = d + 1$, then there is no (α, β)-approximate solution to* SMFR.

Based on Lemma 2, we propose SMFR-SATURATE in Algorithm 1. We first run an SMK algorithm on each objective function individually with the same knapsack constraint k to compute $\texttt{OPT}'_f, \texttt{OPT}'_{g_1}, \ldots, \texttt{OPT}'_{g_d}$. Then, we iterates over each value of β from 0 to 1 with a step of $\frac{\varepsilon}{2}$. For each value of β, we perform a bisection search on α between 0 and 1. Given a pair of α and β, we formulate an instance of BSC on $F'_{\alpha,\beta}$ in Eq. 2 and run the cost-effective greedy algorithm, which starts from $S = \varnothing$ and adds the most "cost-effective" item v^* with the largest ratio between its marginal gain w.r.t. S and cost until no more item can be added with the knapsack constraint $k(1 + \ln \frac{2d+2}{\varepsilon})$, to find a candidate solution S. Next, if $F'_{\alpha,\beta}(S) \geq d + 1 - \frac{\varepsilon}{2}$, that is, S reaches the "saturation level" w.r.t. α, β according to Lemma 2, we set S as the current solution $S_{\alpha,\beta}$ and search in the upper half for a better solution with a higher value of α; otherwise, we search in the lower half for a feasible solution. When $\alpha_{max} - \alpha_{min} \leq \frac{\varepsilon}{2}$, we add the solution $S_{\alpha_{min},\beta}$ to $\mathcal{S}$, remove all solutions dominated by $S_{\alpha_{min},\beta}$, and move on to the next value of β. Finally, all non-dominated solutions in $\mathcal{S}$ are returned for SMFR.

Theorem 3 SMFR-SATURATE *runs in $O(dt(\mathcal{A}) + \frac{n^2}{\varepsilon} \log \frac{1}{\varepsilon})$ time, where $t(\mathcal{A})$ is the time complexity of the* SMK *algorithm, and provides a set $\mathcal{S}$ of solutions with the following properties:*

(1) $|\mathcal{S}| = O(\frac{1}{\varepsilon})$*, (2)* $c(S) = O(k \log \frac{d}{\varepsilon})$ *for each* $S \in \mathcal{S}$*, (3) for each* (α^*, β^*)*-approximate Pareto optimal solution* S^* *to* SMFR*, there must exist its corresponding solution* $S \in \mathcal{S}$ *such that* $f(S) \geq (\delta\alpha^* - \varepsilon)\texttt{OPT}_f$ *and* $g_i(S) \geq (\delta\beta^* - \varepsilon)\texttt{OPT}_{g_i}, \forall i \in [d]$.

We defer the proofs to Appendix B due to space limitations.

4. Conclusion and Discussion

In this paper, we study a novel multi-objective combinatorial optimization problem called *Submodular Maximization with Fair Representation* (SMFR), which aims to select subsets of bounded costs from a ground set such that a submodular (utility) function f is maximized while d submodular (representativeness) functions $g_1, \ldots, g_d$ are also maximized. We show the hardness of finding optimal solutions to SMFR and propose a Pareto optimization approach that enumerated a set of approximate solutions to all Pareto optimal solutions with different trade-offs between multiple objectives for SMFR. In future work, we will showcase the applications of SMFR in real-world problems. In addition, we would like to extend SMFR to other classes of constraints, e.g., matroid and p-system constraints. Finally, it would also be interesting to consider non-monotone submodular functions and weakly submodular functions to capture more general cases of subset selection.

Acknowledgments

Yanhao Wang was supported by the National Natural Science Foundation of China under grant number 62202169.

References

[1] Nima Anari, Nika Haghtalab, Seffi Naor, Sebastian Pokutta, Mohit Singh, and Alfredo Torrico. Structured robust submodular maximization: Offline and online algorithms. In *Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 3128–3137. PMLR, 2019.

[2] Wei-Xuan Bao, Jun-Yi Hang, and Min-Ling Zhang. Submodular feature selection for partial label learning. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '22)*, pages 26–34. Association for Computing Machinery, 2022.

[3] Ruben Becker, Federico Corò, Gianlorenzo D'Angelo, and Hugo Gilbert. Balancing spreads of influence in a social network. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(01):3–10, 2020.

[4] Alina Ene and Huy L. Nguyen. A nearly-linear time algorithm for submodular maximization with a knapsack constraint. In *46th International Colloquium on Automata, Languages, and Programming (ICALP 2019)*, pages 53:1–53:12. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2019.

[5] Moran Feldman, Zeev Nutov, and Elad Shoham. Practical budgeted submodular maximization. *Algorithmica*, 85(5):1332–1371, 2022.

[6] Chao Feng and Chao Qian. Multi-objective submodular maximization by regret ratio minimization with theoretical guarantee. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(14):12302–12310, 2021.

[7] Kai Han, Shuang Cui, Tianshuai Zhu, Enpei Zhang, Benwei Wu, Zhizhuo Yin, Tong Xu, Shaojie Tang, and He Huang. Approximation algorithms for submodular data summarization with a knapsack constraint. *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, 5(1):05:1–05:31, 2021.

[8] Chien-Chung Huang, Naonori Kakimura, and Yuichi Yoshida. Streaming algorithms for maximizing monotone submodular functions under a knapsack constraint. *Algorithmica*, 82(4): 1006–1032, 2020.

[9] David Kempe, Jon Kleinberg, and Éva Tardos. Maximizing the spread of influence through a social network. In *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '03)*, pages 137–146. Association for Computing Machinery, 2003.

[10] Samir Khuller, Anna Moss, and Joseph (Seffi) Naor. The budgeted maximum coverage problem. *Information Processing Letters*, 70(1):39–45, 1999.

[11] Andreas Krause and Daniel Golovin. Submodular function maximization. In *Tractability: Practical Approaches to Hard Problems*, pages 71–104. Cambridge University Press, Cambridge, UK, 2014.

[12] Andreas Krause and Carlos Guestrin. A note on the budgeted maximization of submodular functions. Technical Report CMU-CALD-05-103, Carnegie Mellon University, 2005.

[13] Andreas Krause, H. Brendan McMahan, Carlos Guestrin, and Anupam Gupta. Robust submodular observation selection. *Journal of Machine Learning Research*, 9(93):2761–2801, 2008.

[14] Ariel Kulik, Roy Schwartz, and Hadas Shachnai. A refined analysis of submodular greedy. *Operations Research Letters*, 49(4):507–514, 2021.

[15] Wenxin Li, Moran Feldman, Ehsan Kazemi, and Amin Karbasi. Submodular maximization in clean linear time. *Advances in Neural Information Processing Systems*, 35:17473–17487, 2022.

[16] Hui Lin and Jeff Bilmes. Multi-document summarization via budgeted maximization of submodular functions. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 912–920. Association for Computational Linguistics, 2010.

[17] Yuzong Liu, Kai Wei, Katrin Kirchhoff, Yisong Song, and Jeff A. Bilmes. Submodular feature selection for high-dimensional acoustic score spaces. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7184–7188. IEEE, 2013.

[18] Anay Mehrotra and Nisheeth K. Vishnoi. Maximizing submodular functions for recommendation in the presence of biases. In *Proceedings of the ACM Web Conference 2023 (WWW '23)*, pages 3625–3636. Association for Computing Machinery, 2023.

[19] Baharan Mirzasoleiman, Ashwinkumar Badanidiyuru, and Amin Karbasi. Fast constrained submodular maximization: Personalized data summarization. In *Proceedings of The 33rd International Conference on Machine Learning (ICML)*, pages 1358–1367. PMLR, 2016.

[20] George L. Nemhauser, Laurence A. Wolsey, and Marshall L. Fisher. An analysis of approximations for maximizing submodular set functions–I. *Mathematical Programming*, 14:265–294, 1978.

[21] Naoto Ohsaka and Tatsuya Matsuoka. Approximation algorithm for submodular maximization under submodular cover. In *Proceedings of the Thirty-Seventh Conference on Uncertainty in Artificial Intelligence (UAI)*, pages 792–801. PMLR, 2021.

[22] Chao Qian, Yang Yu, and Zhi-Hua Zhou. Subset selection by pareto optimization. *Advances in Neural Information Processing Systems*, 28:1774–1782, 2015.

[23] Chao Qian, Jing-Cheng Shi, Yang Yu, Ke Tang, and Zhi-Hua Zhou. Subset selection under noise. *Advances in Neural Information Processing Systems*, 30:3560–3570, 2017.

[24] Chao Qian, Chao Bian, and Chao Feng. Subset selection by pareto optimization with recombination. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(03):2408–2415, 2020.

[25] Vahid Roostapour, Aneta Neumann, Frank Neumann, and Tobias Friedrich. Pareto optimization for subset selection with dynamic cost constraints. *Artificial Intelligence*, 302:103597, 2022.

[26] Tasuku Soma and Yuichi Yoshida. Regret ratio minimization in multi-objective submodular function maximization. *Proceedings of the AAAI Conference on Artificial Intelligence*, 31(1): 905–911, 2017.

[27] Maxim Sviridenko. A note on maximizing a submodular set function subject to a knapsack constraint. *Operations Research Letters*, 32(1):41–43, 2004.

[28] Jing Tang, Xueyan Tang, Andrew Lim, Kai Han, Chongshou Li, and Junsong Yuan. Revisiting modified greedy algorithm for monotone submodular maximization with a knapsack constraint. *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, 5 (1):08:1–08:22, 2021.

[29] Shaojie Tang. When social advertising meets viral marketing: Sequencing social advertisements for influence maximization. *Proceedings of the AAAI Conference on Artificial Intelligence*, 32(1):176–183, 2018.

[30] Alfredo Torrico, Mohit Singh, Sebastian Pokutta, Nika Haghtalab, Joseph (Seffi) Naor, and Nima Anari. Structured robust submodular maximization: Offline and online algorithms. *INFORMS Journal on Computing*, 33(4):1590–1607, 2021.

[31] Alan Tsang, Bryan Wilder, Eric Rice, Milind Tambe, and Yair Zick. Group-fairness in influence maximization. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*, pages 5997–6005. International Joint Conferences on Artificial Intelligence Organization, 2019.

[32] Sebastian Tschiatschek, Adish Singla, and Andreas Krause. Selecting sequences of items via submodular maximization. *Proceedings of the AAAI Conference on Artificial Intelligence*, 31 (1):2667–2673, 2017.

[33] Rajan Udwani. Multi-objective maximization of monotone submodular functions with cardinality constraint. *Advances in Neural Information Processing Systems*, 31:9513–9524, 2018.

[34] Yanhao Wang, Jiping Zheng, and Fanxu Meng. Improved algorithm for regret ratio minimization in multi-objective submodular maximization. *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(10):12500–12508, 2023.

[35] Yanhao Wang, Yuchen Li, Francesco Bonchi, and Ying Wang. Balancing utility and fairness in submodular maximization. In *Proceedings of the 27th International Conference on Extending Database Technology, EDBT 2024, Paestum, Italy, March 25 - March 28*, pages 1–14. OpenProceedings.org, 2024.

[36] Laurence A. Wolsey. An analysis of the greedy algorithm for the submodular set covering problem. *Combinatorica*, 2(4):385–393, 1982.

[37] Grigory Yaroslavtsev, Samson Zhou, and Dmitrii Avdiukhin. "Bring your own greedy"+max: Near-optimal 1/2-approximations for submodular knapsack. In *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 3263–3274. PMLR, 2020.

Appendix A. Related Work

There have been extensive studies on submodular maximization with knapsack constraints (SMK). For cardinality constraints, a special case of knapsack constraints with uniform costs, Nemhauser et al. [20] proposed a simple greedy algorithm that runs in $O(kn)$ time and yields the best possible approximation factor $1 - 1/e$ unless $P = NP$. However, the greedy algorithm can be arbitrarily bad for general knapsack constraints. Sviridenko [27] first proposed a greedy algorithm with partial enumerations that achieves the best possible approximation $1 - 1/e$ for SMK in $O(n^5)$ time. Kulik et al. [14] and Feldman et al. [5] improved the time complexity to $O(n^4)$ while keeping the same approximation factor. Krause and Guestrin [12] proposed an $O(n^2)$-time $\frac{1}{2}(1 - \frac{1}{e}) \approx 0.316$-approximation cost-effective greedy algorithm for SMK. Tang et al. [28], Kulik et al. [14], and Feldman et al. [5] improved the approximation factor of cost-effective greedy to 0.405, $[0.427, 0.4295]$, and $[0.427, 0.462]$ independently. Ene and Nguyen [4] proposed a $(1-1/e-\varepsilon)$-approximation algorithm for SMK running in $O(n \log^2 n(1/\varepsilon)^{1/\varepsilon^4})$ time based on multilinear relaxation. Yaroslavtsev et al. [37] proposed a $\frac{1}{2}$-approximation Greedy+Max algorithm for SMK in $O(n^2)$ time. Feldman et al. [5] further provided an approximation factor of 0.6174 in $O(n^3)$ time by enumerating each single item as a partial solution and running Greedy+Max on each partial solution. Li et al. [15] recently proposed a $(\frac{1}{2} - \varepsilon)$-approximation algorithm for SMK in $O(\frac{n}{\varepsilon} \log \frac{1}{\varepsilon})$ time. Although the

above algorithms cannot be directly applied to SMFR, any of them can serve as a subroutine in our algorithmic framework for SMFR.

There also exist several variants of submodular maximization problems to deal with more than one objective. The problem of maximizing the minimum of $d > 1$ submodular functions $g_1, \ldots, g_d$ was considered in [1, 13, 30, 33]. This problem differs from SMFR because it does not consider maximizing f and returns only a single solution. Nevertheless, we draw inspiration from the SATURATE framework used in these methods to solve SMFR. Another two relevant problems to SMFR are *Submodular Maximization under Submodular Cover* (SMSC) [21], which maximizes one submodular function subject to the value of the other submodular function not being below a threshold, and *Balancing utility and fairness in Submodular Maximization* (BSM) [35], which maximizes a submodular utility function subject to that a fairness function in form of the minimum of $d > 1$ submodular functions is approximately maximized. SMSC and BSM differ from SMFR in three aspects: (*i*) they still return a single solution to optimize a user-specified trade-off between multiple objectives; (*ii*) they are specific to cardinality constraints; (*iii*) SMSC is limited to $d = 1$, while BSM requires that all objective functions are decomposable. Thus, SMFR can work in more general scenarios than SMSC and BSM. Due to the above differences, the algorithms for SMSC and BSM cannot be used for SMFR. The problem of regret-ratio minimization [6, 26, 34] for multi-objective submodular maximization is similar to SMFR in the sense that they also aim to find a set of approximate solutions for different trade-offs between multiple objectives. However, they consider denoting the trade-offs as different non-negative linear combinations of multiple submodular functions, but cannot guarantee any approximation for each objective individually. Several subset selection problems, e.g., [22–25], utilize a Pareto optimization method by transforming a single-objective problem into a bi-objective problem and then solving the bi-objective problem to obtain a solution to the original problem. Those problems are interesting but orthogonal to our work.

Appendix B. Proofs of Lemmas and Theorems

B.1. Proof of Lemma 2

Lemma 2 *If $F'_{\alpha,\beta}(S) \geq d + 1 - \frac{\varepsilon}{2}$ for any set $S \in \mathcal{I}_k$, then S is a $(\delta\alpha - \frac{\varepsilon}{2}, \delta\beta - \frac{\varepsilon}{2})$-approximate solution to* SMFR. *If there is no set $S \in \mathcal{I}_k$ with $F'_{\alpha,\beta}(S) = d + 1$, then there is no (α, β)-approximate solution to* SMFR.

Proof For the proof of the first statement, we first consider the two special cases of $\alpha = 0$ and $\beta = 0$. When $\alpha = 0$ or $\beta = 0$, if $F'_{\alpha,\beta}(S) > d + 1 - \frac{\varepsilon}{2}$, we will have $\frac{g_i(S)}{\beta \mathtt{OPT}'_{g_i}} > 1 - \frac{\varepsilon}{2}$ for every $i \in [d]$ or $\frac{f(S)}{\alpha \mathtt{OPT}'_f} > 1 - \frac{\varepsilon}{2}$. In the general case of $\alpha, \beta > 0$, if $F'_{\alpha,\beta}(S) > d + 1 - \frac{\varepsilon}{2}$, we will have $\frac{f(S)}{\alpha \mathtt{OPT}'_f} > 1 - \frac{\varepsilon}{2}$ and $\frac{g_i(S)}{\beta \mathtt{OPT}'_{g_i}} > 1 - \frac{\varepsilon}{2}$ for every $i \in [d]$ at the same time. Thus, it holds that

$$f(S) \geq (1 - \frac{\varepsilon}{2})\alpha \mathtt{OPT}'_f \geq \delta\alpha(1 - \frac{\varepsilon}{2})\mathtt{OPT}_f \geq (\delta\alpha - \frac{\varepsilon}{2})\mathtt{OPT}_f$$

and

$$g_i(S) \geq (1 - \frac{\varepsilon}{2})\beta \mathtt{OPT}'_{g_i} \geq \delta\beta(1 - \frac{\varepsilon}{2})\mathtt{OPT}_{g_i} \geq (\delta\beta - \frac{\varepsilon}{2})\mathtt{OPT}_{g_i}, \forall i \in [d].$$

Therefore, S is a $(\delta\alpha - \frac{\varepsilon}{2}, \delta\beta - \frac{\varepsilon}{2})$-approximate solution to SMFR.

For the proof of the second statement, if $F'_{\alpha,\beta}(S) < d+1$, then we will have $f(S) < \alpha\texttt{OPT}'_f \leq \alpha\texttt{OPT}_f$ or there is some $i \in [d]$ with $g_i(S) < \beta\texttt{OPT}'_{g_i} \leq \beta\texttt{OPT}_{g_i}$. Therefore, if $F'_{\alpha,\beta}(S) < d+1$, S will not be an (α,β)-approximate solution to SMFR. Accordingly, if there is no set $S \in \mathcal{I}_k$ with $F'_{\alpha,\beta}(S) = d+1$, then there does not exist any (α,β)-approximate solution to SMFR. ■

B.2. Proof of Theorem 3

Theorem 3 SMFR-SATURATE *runs in* $O(dt(\mathcal{A}) + \frac{n^2}{\varepsilon}\log\frac{1}{\varepsilon})$ *time, where* $t(\mathcal{A})$ *is the time complexity of the* SMK *algorithm, and provides a set* $\mathcal{S}$ *of solutions with the following properties: (1)* $|\mathcal{S}| = O(\frac{1}{\varepsilon})$*, (2)* $c(S) = O(k\log\frac{d}{\varepsilon})$ *for each* $S \in \mathcal{S}$*, (3) for each* (α^*,β^*)*-approximate Pareto optimal solution* S^* *to* SMFR*, there must exist its corresponding solution* $S \in \mathcal{S}$ *such that* $f(S) \geq (\delta\alpha^* - \varepsilon)\texttt{OPT}_f$ *and* $g_i(S) \geq (\delta\beta^* - \varepsilon)\texttt{OPT}_{g_i}, \forall i \in [d]$.

Proof Let us first analyze the time complexity of SMFR-SATURATE. First, it runs the SMK algorithm $d+1$ times to compute $\texttt{OPT}'_f$ and $\texttt{OPT}'_{g_i}$ for every $i \in [d]$. Then, it iterates over $\lceil\frac{2}{\varepsilon}\rceil$ values of β in the `for` loop. For each value of β, it attempts to use $O(\log\frac{1}{\varepsilon})$ different values of α in the bisection search. Finally, the cost-effective greedy algorithm takes $O(n^2)$ time for BSC on each $F'_{\alpha,\beta}$. In summary, the time complexity of SMFR-SATURATE is $O(dt(\mathcal{A}) + \frac{n^2}{\varepsilon}\log\frac{1}{\varepsilon})$ time, where $t(\mathcal{A})$ is the time complexity of the SMK algorithm.

For the solution $\mathcal{S}$ of SMFR-SATURATE, it is easy to see that $|\mathcal{S}| \leq \lceil\frac{2}{\varepsilon}\rceil$ and thus $|\mathcal{S}| = O(\frac{1}{\varepsilon})$ because SMFR-SATURATE adds at most one set to $\mathcal{S}$ for each value of β. Then, due to the condition in the inner `while` loop of Algorithm 1, it must hold that $c(S) \leq k(1+\ln\frac{2d+2}{\varepsilon})$ and thus $c(S) = O(k\log\frac{d}{\varepsilon})$ for each $S \in \mathcal{S}$. Finally, given an (α^*,β^*)-approximate Pareto optimal solution S^*, there must exist a value of β in the `for` loop such that $0 \leq \beta^* - \beta \leq \frac{\varepsilon}{2}$. Let $S_{\alpha_{min},\beta}$ be the solution of SMFR-SATURATE w.r.t. such β and its corresponding α_{min}. Since $F'_{\alpha_{min},\beta}(S_{\alpha_{min},\beta}) \geq d+1-\frac{\varepsilon}{2}$, $S_{\alpha_{min},\beta}$ is a $(\delta\alpha_{min} - \frac{\varepsilon}{2}, \delta\beta - \frac{\varepsilon}{2})$-approximate solution according to Lemma 2. Furthermore, we have $F'_{\alpha_{max},\beta}(S_{gr}) < d+1-\frac{\varepsilon}{2}$, where S_{gr} is the solution w.r.t. $F'_{\alpha_{max},\beta}$ with knapsack constraint $k(1+\ln\frac{2d+2}{\varepsilon})$ returned by the cost-effective greedy procedure in Algorithm 1, and $\alpha_{max} - \alpha_{min} < \frac{\varepsilon}{2}$. Suppose that S'_{gr} is the first intermediate subset of S_{gr} with $c(S'_{gr}) \geq k\ln\frac{2d+2}{\varepsilon}$ constructed using the cost-effective greedy procedure. Let $S^*_k = \arg\max_{S\in\mathcal{I}_k} F'_{\alpha_{max},\beta}(S)$ and $\texttt{OPT}_{F'_{\alpha_{max},\beta}} = F'_{\alpha_{max},\beta}(S^*_k)$. According to the monotonicity and submodularity of $F'_{\alpha_{max},\beta}$,

$$F'_{\alpha_{max},\beta}(S^*_k) \leq F'_{\alpha_{max},\beta}(S^{(i)}_{gr}) + \sum_{v\in S^*_k\setminus S^{(i)}_{gr}} \Delta(v|S^{(i)}_{gr}) = F'_{\alpha_{max},\beta}(S^{(i)}_{gr}) + \sum_{v\in S^*_k\setminus S^{(i)}_{gr}} \frac{c(v)\cdot\Delta(v|S^{(i)}_{gr})}{c(v)},$$

for any $S^{(i)}_{gr} \subset S'_{gr}$ after i iterations and $\Delta(v|S^{(i)}_{gr}) = F'_{\alpha_{max},\beta}(S^{(i)}_{gr}\cup\{v\}) - F'_{\alpha_{max},\beta}(S^{(i)}_{gr})$. Let u^*_i be the i-th item added to S'_{gr} for any $i = 1,\ldots,|S'_{gr}|$. Based on the cost-effective greedy selection in Algorithm 1,

$$\frac{\Delta(u^*_{i+1}|S^{(i)}_{gr})}{c(u^*_{i+1})} \geq \frac{\Delta(v|S^{(i)}_{gr})}{c(v)}$$

for any $v \in S^*_k \setminus S^{(i)}_{gr}$ and $i \in [0,\ldots,|S'_{gr}-1|]$ because $c(v) \leq k$ for any $v \in S^*_k$ and thus no item from S^*_k is excluded from consideration due to budget violation when u^*_{i+1} is added to $S^{(i)}_{gr}$.

Therefore, we further obtain

$$F'_{\alpha_{max},\beta}(S^*_k) \leq F'_{\alpha_{max},\beta}(S^{(i)}_{gr}) + \frac{\Delta(u^*_{i+1}|S^{(i)}_{gr})}{c(u^*_{i+1})} \sum_{v \in S^*_k \setminus S^{(i)}_{gr}} c(v) \leq F'_{\alpha_{max},\beta}(S^{(i)}_{gr}) + \frac{\Delta(u^*_{i+1}|S^{(i)}_{gr})}{c(u^*_{i+1})} \cdot k,$$

After rearranging the inequality above, we have

$$F'_{\alpha_{max},\beta}(S^*_k) - F'_{\alpha_{max},\beta}(S^{(i+1)}_{gr}) \leq \big(1 - \frac{c(u^*_{i+1})}{k}\big)\big(F'_{\alpha_{max},\beta}(S^*_k) - F'_{\alpha_{max},\beta}(S^{(i)}_{gr})\big).$$

Moreover, since $1 - x \leq e^{-x}$ for any $x > 0$, it holds that $1 - \frac{c(u^*_{i+1})}{k} \leq \exp(-\frac{c(u^*_{i+1})}{k})$. Therefore,

$$F'_{\alpha_{max},\beta}(S^*_k) - F'_{\alpha_{max},\beta}(S^{(i+1)}_{gr}) \leq \exp(-\frac{c(u^*_{i+1})}{k}) \cdot \big(F'_{\alpha_{max},\beta}(S^*_k) - F'_{\alpha_{max},\beta}(S^{(i)}_{gr})\big). \quad (3)$$

By applying Eq. 3 recursively for $i = 0, \ldots |S'_{gr}| - 1$, we have

$$\begin{aligned}
F'_{\alpha_{max},\beta}(S^*_k) - F'_{\alpha_{max},\beta}(S'_{gr}) &\leq \exp(-\frac{c(u^*_{i+1})}{k}) \cdot \big(F'_{\alpha_{max},\beta}(S^*_k) - F'_{\alpha_{max},\beta}(S^{(i)}_{gr})\big) \\
&\leq \exp(-\frac{c(u^*_{i+1})}{k}) \exp(-\frac{c(u^*_i)}{k})\big(F'_{\alpha_{max},\beta}(S^*_k) - F'_{\alpha_{max},\beta}(S^{(i-1)}_{gr})\big) \\
&\leq \ldots\ldots \leq \exp(-\frac{\sum_{i=0}^{|S'_{gr}|-1} c(u^*_{i+1})}{k}) F'_{\alpha_{max},\beta}(S^*_k) \\
&= \exp(-\frac{c(S'_{gr})}{k}) F'_{\alpha_{max},\beta}(S^*_k) = \exp(-\frac{c(S'_{gr})}{k})\texttt{OPT}_{F'_{\alpha_{max},\beta}}.
\end{aligned}$$

Since $c(S'_{gr}) \geq k \ln \frac{2d+2}{\varepsilon}$, it holds that

$$F'_{\alpha_{max},\beta}(S'_{gr}) \geq (1 - \exp\big(-\frac{c(S'_{gr})}{k}\big))\texttt{OPT}_{F'_{\alpha_{max},\beta}} \geq (1 - \frac{\varepsilon}{2d+2})\texttt{OPT}_{F'_{\alpha_{max},\beta}}.$$

In addition, $F'_{\alpha_{max},\beta}(S_{gr}) \geq F'_{\alpha_{max},\beta}(S'_{gr})$ since $S'_{gr} \subseteq S_{gr}$. Therefore, we have $\texttt{OPT}_{F'_{\alpha_{max},\beta}} < d + 1$ and, according to Lemma 2, there does not exist any (α_{max}, β)-approximate solution of cost at most k. Since S^* is an (α^*, β^*)-approximate Pareto optimal solution and $\beta \leq \beta^*$, S^* must be an (α^*, β)-approximate solution of cost at most k. As such, we obtain $\alpha_{max} > \alpha^*$ and $\alpha_{min} > \alpha^* - \frac{\varepsilon}{2}$. Because we have shown that $S_{\alpha_{min},\beta}$ is a $(\delta\alpha_{min} - \frac{\varepsilon}{2}, \delta\beta - \frac{\varepsilon}{2})$-approximate solution, $S_{\alpha_{min},\beta}$ is guaranteed to be a $(\delta\alpha^* - \varepsilon, \delta\beta^* - \varepsilon)$-approximate solution. If $S_{\alpha_{min},\beta}$ is included in $\mathcal{S}$, we will conclude the proof directly; otherwise, the solution in $\mathcal{S}$ dominating $S_{\alpha_{min},\beta}$ can confirm our conclusion. ■