A Learning-Augmented Framework for Knapsack-Constrained Submodular Maximization Problems

Yanhui Zhu
Montclair State University
Montclair, NJ, USA

ABSTRACT

Many machine learning tasks involve maximizing a submodular function with constraints. However, constrained submodular maximization problems are known to be NP-Hard. Although there exist efficient approximation algorithms, approximation guarantees cannot be improved without auxiliary information, unless $\mathsf{P} = \mathsf{NP}$. In this work, we propose a framework for the general knapsack-constrained submodular maximization problems that incorporates advice from learning models or domain experts. Our framework is flexible and can use various existing offline ρ-approximation algorithms as a subroutine. For both monotone and non-monotone cases, we prove that when the advice is arbitrarily bad, our framework produces solutions with guarantees no worse than the subroutine algorithms (without advice); when the advice is optimal, the framework is optimal. In other words, the proposed framework is ρ-robust and 1-consistent, where ρ is the provable worst-case guarantee.

1. INTRODUCTION

A function f defined over a ground set V is submodular if it exhibits diminishing return property, i.e., for any subsets S and T of V where $S \subseteq T$, for any $x \in V \setminus T$, $f(S \cup \{x\}) - f(S) \geq f(T \cup \{x\}) - f(T)$. The function f is monotone if $f(T) \geq f(S)$ when S is a subset of T. Submodular function optimization is a fundamental problem in combinatorial optimization, applied extensively across diverse fields such as feature compression, deep learning, sensor placement, and information diffusion, among others. Over the last decade, various versions of submodular optimization problems have garnered substantial attention. For example, the classical greedy algorithm for the cardinality-constrained problem provides a tight $(1 - 1/e)$-approximation [8]. It is known that the approximation ratio is optimal if $\mathsf{P} \neq \mathsf{NP}$.

Most approximation algorithms for submodular maximization are designed for adversarial worst-case instances and can thus be conservative in structured data settings. In many modern pipelines, however, decisions are not made in a vacuum. Similar instances repeat, objectives evolve gradually, and learned systems can provide predictive signals. For example, there may be forecasts of future arrivals or deletions, estimates of marginal gains, or a suggested near-optimal solution produced by a model trained on historical data. The central theme is to convert such predictions into *provable* improvements in theoretical performance or efficiency, while retaining classical worst-case guarantees when predictions are arbitrarily bad.

The consideration of machine learning predictions led to a new paradigm of algorithm design known as *learning-augmented algorithms* [4, 3, 2, 7]. In this framework, traditional algorithms are equipped with machine learning predictions. The performance of such learning-augmented algorithms is analyzed in terms of how prediction quality affects performance, while still retaining worst-case guarantees. This line of work has led to new results in online algorithms, caching, scheduling, and clustering, where predictions can provably improve performance without sacrificing robustness.

Despite this progress, the role of a learning-augmented framework in submodular maximization remains largely unexplored. Existing analyses of the traditional submodular maximization assume no auxiliary information, while heuristic methods that incorporate predictions lack theoretical guarantees. Additionally, these results are largely problem-specific (e.g., facility allocation [2], clustering [6], graph problems [1]). Hence, a fundamental question is raised:

Can we design a framework for a series of submodular maximization problems that systematically exploit predictions, and can we quantify how prediction quality improves approximation guarantees while preserving worst-case bounds?

In this work, we affirmatively answer this question by formalizing a principled way to integrate predictions into knapsack-constrained submodular optimization and highlight a flexible framework that extends to other constraints.

Our Contributions. This work designs a learning augmented framework that considers a feasible prediction set as advice. Instead of either fully trusting or discarding the advice, the algorithm evaluates prefixes of the predicted set and completes each prefix using a scalable residual submodular knapsack routine. This design yields a clean reduction from the learning-augmented problem to standard residual maximization, making the framework broadly compatible with existing approximation subroutines.

Technically, we prove that the objective of the residual maximization subroutine preserves monotonicity and submodularity. Thereafter, we prove that, for both monotone and non-monotone cases, the approximation guarantees inherit robustness from the underlying knapsack algorithm while exploiting accurate advice when predictions align well with the optimum.

Learning-Augmented Algorithms: Theory and Applications (LATA) 2026
Ann Arbor, MI, USA

2. PRELIMINARY RESULTS

Inspired by [5], we propose a framework (Algorithm 1) that solves the Submodular Maximization with Knapsack constraint problem (SMK), formally defined as follows.

Problem 1 (SMK). *Given a submodular function $f : 2^V \to \mathbb{R}$ and a modular cost function $c : 2^V \to \mathbb{R}$, a budget B, find a subset $S \in \operatorname{arg\,max}_{T \subset V, c(T) \leq B} f(T)$.*

If the modular costs are uniform, Problem 1 reduces to the cardinality-constrained problem.

The algorithm LA-SMK treats the prediction set A as advice that may be partially reliable. It orders the elements of A into prefixes $A_0 \subseteq A_1 \subseteq \cdots \subseteq A_m = A$, where each A_i is feasible under the knapsack budget. For every prefix A_i, the algorithm assumes that these i predicted items are trusted, then computes an augmentation R_i with the remaining budget $B - c(A_i)$ using a `Subroutine` algorithm. The `Subroutine` solves the residual knapsack-constrained problem anchored at A_i, over sets $S \subseteq V \setminus A_i$ with $c(S) \leq B - c(A_i)$ with some worst-case approximation ρ.

Every prefix yields a candidate solution $U_i = A_i \cup R_i$. Finally, it returns the candidate with the maximum value $f(U_i)$. Thus, the framework balances ρ-robustness and 1-consistency by searching over all trust levels in the prediction rather than committing fully to the advice.

Algorithm 1: LA-SMK(f, c, B, A, V)

Input : submodular function $f : 2^V \to \mathbb{R}_{\geq 0}$, modular cost function $c : V \to \mathbb{R}_{\geq 0}$, total budget B, a feasible prediction set A, ground set V

Output: learning-augmented selection U

1 Form sets $A_0, A_1, \cdots, A_m$ from A, where $|A_i| = i$ and $A = A_m$

2 **for** $i = 0$ *to* m **do**

3 $\quad R_i \leftarrow$ `Subroutine`$(f, c, B, A_i, V \setminus A_i)$

4 $\quad U_i \leftarrow A_i \cup R_i$

5 **end**

6 Return U_{i^*} where $i^* = \operatorname{arg\,max}_{i \in \{1, \cdots, m\}} f(U_i)$

Before presenting our results, we introduce some notations used in this work. Denote the optimal solution for Problem 1 (regardless of monotonicity of f) as $\mathsf{OPT} \in \operatorname{arg\,max}_{T \subseteq V, c(T) \leq B - c(A_i)} f(T)$. For any partial advice A_i, define $\mathsf{OPT}_i \in \operatorname{arg\,max}_{\substack{T \subseteq \mathsf{OPT} \setminus A_i, \\ c(T) \leq B - c(A_i)}} f(T \cup A_i)$.

Monotone Case. We first consider the case when the objective function is monotonically non-decreasing. For this case, we design the `Subroutine` to maximize an anchored function $g_A : 2^{V \setminus A} \to \mathbb{R}_{\geq 0}$ with residual budget $B - c(A)$, which remains monotone submodular (see Lemma 1). This is a key property that makes the framework flexible and extendable, thereby guaranteeing ρ-approximation (see Lemma 2) w.r.t. g_A.

Lemma 1. *Let $f : 2^V \to \mathbb{R}_{\geq 0}$ be a monotone submodular set function, and let $A \subseteq V$ be fixed. Define the residual function $g_A : 2^{V \setminus A} \to \mathbb{R}_{\geq 0}$ defined by*

$$g_A(S) := f(S \cup A) - f(A), \quad \forall S \subseteq V \setminus A.$$

Then g_A is monotone and submodular on the ground set $V \setminus A$ and $g_A(\emptyset) = 0$.

Lemma 2. *Let $A_i \subseteq V$ be fixed with $c(A_i) \leq B$. If the subroutine (line 3 of Algorithm 1) is a ρ-approximation knapsack-constrained submodular maximization algorithm with respect to the monotone submodular function f, suppose R_i is the selection of the subroutine applied to the residual instance A_i with residual budget $B - c(A_i)$, then*

$$g_{A_i}(R_i) \geq \rho \cdot g_{A_i}(\Gamma_i),$$

where $\Gamma_i \in \operatorname{arg\,max}_{\substack{T \subseteq V \setminus A_i, \\ c(T) \leq B - c(A_i)}} g_{A_i}(T)$.

With Lemma 1 and Lemma 2, we derive the final theorem of Algorithm 1 for the monotone submodular maximization problems, as follows.

Theorem 1. *Consider the learning-augmented algorithm for the monotone submodular maximization under a knapsack constraint. For each subset $A_i \subseteq A$, the algorithm completes A_i using a ρ-approximation algorithm for the residual knapsack objective g_{A_i}, then the final output U satisfies*

$$f(U) \geq \max_i \left\{\rho \cdot f(\mathsf{OPT}_i \cup A_i) + (1 - \rho) f(A_i)\right\}.$$

Corollary 1. *For the monotone submodular maximization with knapsack constraint problem, Algorithm 1 is ρ-robust and 1-consistent.*

Proof Sketch. When the advice A is arbitrarily bad, by Theorem 1, for the round when A_0 is extended, $f(U_0) \geq \rho \cdot f(\mathsf{OPT})$ is a feasible output. When the advice is optimal, i.e., $A = \mathsf{OPT}$, for some round, $f(U_m) \geq f(\mathsf{OPT})$ for $|A| = m$ since f is monotone and $f(\mathsf{OPT}_i \cup A_i) - f(A_i) \geq 0$. □

Corollary 1 indicates that the theoretical guarantee of Algorithm 1 depends on the implementation of the `Subroutine` at Line 3. Thus, we present in Corollary 2 some choices of `Subroutine` with approximation-time tradeoffs.

Corollary 2. *For Algorithm 1,*

- *if the `Subroutine` is implemented with methods in [9], Algorithm 1 obtains the tight $(1 - 1/e)$-robustness and 1-consistency within $O(mn^5)$ oracle evaluations.*
- *if the `Subroutine` is implemented with methods in [10, 11], Algorithm 1 obtains 1/2-robustness and 1-consistency within $O(mn^2)$ oracle evaluations.*

Non-monotone Case. We also generalize the framework to the scenario that the objective function is non-monotone (adding an element could potentially decrease the function value). This setting is more difficult because maximizing a possibly negative anchored residual function $g_A(S)$ is intractable and just deciding whether the approximation is positive itself is NP-Hard. Nonetheless, we prove that leveraging the framework (Algorithm 1) and using an alternative anchored residual function h_A for the `Subroutine` can still produce a (ρ-robust, 1-consistent) solution.

Theorem 2. *Consider the learning-augmented algorithm for the non-monotone submodular maximization under a knapsack constraint. For each subset $A_i \subseteq A$, the algorithm completes A_i using a ρ-approximation algorithm for the residual knapsack objective h_{A_i}, then the final output U satisfies*

$$\begin{aligned} f(U) &\geq \max_i \left\{f(A_i) + \max\left\{0, \rho \cdot f(\mathsf{OPT}_i \cup A_i) - f(A_i)\right\}\right\} \\ &= \max_i \left\{\rho \cdot f(\mathsf{OPT}_i \cup A_i), f(A_i)\right\}. \end{aligned}$$

3. REFERENCES

[1] A. Aamand, J. Y. Chen, S. Gollapudi, S. Silwal, and H. WU. Improved approximations for hard graph problems using predictions. In *Forty-second International Conference on Machine Learning*.

[2] P. Agrawal, E. Balkanski, V. Gkatzelis, T. Ou, and X. Tan. Learning-augmented mechanism design: Leveraging predictions for facility location. In *Proceedings of the 23rd ACM Conference on Economics and Computation*, pages 497–528, 2022.

[3] E. Balkanski, V. Gkatzelis, and X. Tan. Strategyproof scheduling with predictions. In *14th Innovations in Theoretical Computer Science Conference (ITCS 2023)*, volume 251, page 11, 2023.

[4] E. Balkanski, V. Gkatzelis, X. Tan, and C. Zhu. Online mechanism design with predictions. In *Proceedings of the 25th ACM Conference on Economics and Computation*, pages 1184–1184, 2024.

[5] D. Choo, Y. Trabelsi, F. Getnet, S. W. Lamma, W. Nigatu, K. Sime, L. Matay, M. Tambe, and S. Verguet. Optimizing health coverage in ethiopia: A learning-augmented approach and persistent proportionality under an online budget. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 38313–38320, 2026.

[6] X. Deng, D. Huang, D.-H. Chen, C.-D. Wang, and J.-H. Lai. Strongly augmented contrastive clustering. *Pattern Recognition*, 139:109470, 2023.

[7] J. Huang, Q. Feng, Z. Huang, Z. Zhang, J. Xu, and J. Wang. New algorithms for the learning-augmented k-means problem. In *The Thirteenth International Conference on Learning Representations*, 2025.

[8] G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher. An analysis of approximations for maximizing submodular set functions—i. *Mathematical programming*, 14(1):265–294, 1978.

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

[10] G. Yaroslavtsev, S. Zhou, and D. Avdiukhin. "bring your own greedy"+ max: near-optimal 1/2-approximations for submodular knapsack. In *International Conference on Artificial Intelligence and Statistics*, pages 3263–3274. PMLR, 2020.

[11] Y. Zhu, S. Basu, and A. Pavan. Improved evolutionary algorithms for submodular maximization with cost constraints. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*, pages 7082–7090, 2024.