Linear Submodular Bandits with a Knapsack Constraint

Baosheng Yu[†*] and **Meng Fang**[‡*] and **Dacheng Tao**[†]
[†]Centre for Quantum Computation and Intelligent Systems, University of Technology, Sydney
[‡]Department of Computing and Information Systems, The University of Melbourne
baosheng.yu@student.uts.edu.au, meng.fang@unimelb.edu.au, dacheng.tao@uts.edu.au

Abstract

Linear submodular bandits has been proven to be effective in solving the diversification and feature-based exploration problems in retrieval systems. Concurrently, many web-based applications, such as news article recommendation and online ad placement, can be modeled as budget-limited problems. However, the diversification problem under a budget constraint has not been considered. In this paper, we first introduce the budget constraint to linear submodular bandits as a new problem called the linear submodular bandits with a knapsack constraint. We then define an α-approximation unit-cost regret considering that submodular function maximization is NP-hard. To solve this problem, we propose two greedy algorithms based on a modified UCB rule. We then prove these two algorithms with different regret bounds and computational costs. We also conduct a number of experiments and the experimental results confirm our theoretical analyses.

Introduction

Multi-armed bandit (MAB) problem is the simplest instance of the exploration versus exploitation dilemma (Auer, Cesa-Bianchi, and Fischer 2002), which is a trade-off between exploring the environment to find a better action (exploration) and adopting the current best action as often as possible (exploitation). The classical multi-armed bandit problem is formulated as a system of m arms. At each time step, we choose an arm to pull and obtain a reward from an unknown distribution. The goal is to maximize cumulative rewards by optimally balancing exploration and exploitation in finite time steps T. The most popular measure of an algorithm's success is the regret, which is the cumulative loss of failing to pull the optimal arm. A variant of the classical multi-armed bandit problem, which is usually called the combinatorial multi-armed bandit problem (Chen, Wang, and Yuan 2013; Gai, Krishnamachari, and Jain 2012), allows multiple arms chosen at each time step. With the rapid development of the Internet, many web-based applications can be modeled as a combinatorial multi-armed bandit problem, for example, the personalized recommendation of news articles (Li et al. 2010; Fang and Tao 2014), in which multiple news articles are recommended to a user.

Diversification is a key problem for information retrieval systems such as diverse rankings of documents (Radlinski, Kleinberg, and Joachims 2008), products recommendation (Ziegler et al. 2005) and news article recommendation (Li et al. 2010). Implications affecting user satisfaction have been observed in practice: recommendation requires the proposal of a diverse set of items and redundant items are helpless (Ziegler et al. 2005). Submodularity is an intuitive notion of diminishing returns, which states that adding a new item to a larger environment help less than adding the same item to a smaller environment. It has turned out that diversification can be well captured by a submodular function (Krause and Golovin 2012) and the linear submdoular bandits (Yue and Guestrin 2011) has thus been proposed to handle the diversification problem in bandit setting.

There is nevertheless always a budget constraint in real-world scenarios, where limited resources are consumed during whole process of actions. For example, in dynamic procurement (Badanidiyuru, Kleinberg, and Slivkins 2013), budget is limited for buying items. In clinical trials, experiments on alternative medical treatments are limited by the cost of materials. However, it is more reasonable for other applications to put the budget constraint on each time step, not the whole process. For example, in online advertising (Chakrabarti et al. 2009), the size of a webpage is limited while the ads are changing each time the user visits the webpage. In news article recommendation, several articles are recommended to a user and feedback is obtained each time, but users will only have limited time to read those articles (if we recommend three short news articles, the user may read all of them, but if we recommend three long articles, the user might be not patient enough to read all of them). In order to improve user satisfaction, we formulate this per-round budget constraint imposed on each time step, as follows: $\forall i \in E$, let c_i denote the cost of pulling arm i, where E is the set of all arms. The total costs of arm-pulling are limited by a budget B. At each time step t, we choose a subset of arms $A_t \subseteq E$ under the budget constraint $C(A_t) = \sum_{i \in A_t} c_i \leq B$, which is known as a knapsack constraint (Sviridenko 2004).

In order to improve user satisfaction by considering both diversification and budget constraint, we introduce the per-

[*] Equal contribution.

round budget constraint to linear submodular bandits as a new problem called the linear submodular bandits with a knapsack constraint. To solve this new problem, we use a modified upper confidence bounds (UCB) under the budget constraint, which is called the unit-cost upper confidence bounds, to control the trade-off between exploration and exploitation. Inspired by other knapsack solutions, we try to obtain the maximum rewards for each budget unit. Specifically, we greedily choose the arms, which give the maximum modified upper confidence bounds on utility gain, to construct a subset of arms in our algorithms.

In this paper, we first briefly review the related works. We then describe the new problem called linear submodular bandits with a knapsack constraint and the definition of regret. After that, we propose two greedy algorithms based on modified UCB rule and prove that both two algorithms have theoretical regret bounds. Lastly, we use news article recommendation as a case study, which requires us to recommend multiple news articles under a per-round budget constraint. Experimental results demonstrate that our two algorithms outperform the baselines for the linear submodular bandits, such as LSBGreedy (Yue and Guestrin 2011) and Epsilon-Greedy.

Related Work

Diversification problem has been addressed in recommendation systems (Ziegler et al. 2005; Yue and Guestrin 2011) and information retrieval systems (Küçüktunç et al. 2013; Clarke et al. 2008). In recommendation systems, multi-armed bandits has been widely used to identify user interests (Li et al. 2010; Kohli, Salek, and Stoddard 2013; Fang and Tao 2014). Linear submodular bandits has been proposed by Yue and Guestrin (2011) as a typical combinatorial bandit model to solve the diversification problem in recommendation systems. However, it ignores the budget constraint, which nevertheless exists in real-world applications.

Budget constraint has been well studied in classical multi-armed bandit setting (Tran-Thanh et al. 2010; Badanidiyuru, Kleinberg, and Slivkins 2013). In budget-limited multi-armed bandit problem proposed by Tran-Thanh et al. (2010), the goal is to obtain the maximum cumulative rewards under a limited budget. The budget-limited multi-armed bandit problem was firstly solved by a simple budgeted ϵ-first algorithm (Tran-Thanh et al. 2010) and subsequently by an improved algorithm called KUBE (Tran-Thanh et al. 2012). The budget-limited multi-armed bandit problem is also known as the multi-armed bandit problem with budget constraint and fixed costs (MAB-BF). There is also a more complex problem called the multi-armed bandit problem with budget constraint and variable costs (Ding et al. 2013), where the cost of arm is not fixed. A more general budget-limited bandit model has been proposed by Badanidiyuru, Kleinberg, and Slivkins (2013) and is known as bandits with knapsacks (BwK). However, most of previous works focus on the budget constraint for the whole process. Unlike previous works, we impose a per-round budget constraint on each time step separately. Our work is based on the linear submodular bandits, to which we introduce the budget constraint as a new problem called the linear submodular bandits with a knapsack constraint.

Problem Definition

We formulate the linear submodular bandits with a knapsack constraint as follows: let $E = \{1, 2, \ldots, m\}$ be a set of m arms and $C = \{c_1, c_2, \ldots, c_m\}$ $(c_i > 0,\ \forall i = 1, 2, \ldots, m)$ be a set of costs for m arms. At each time step t, we sequentially choose each arm for $A_t \subseteq E$ under the budget constraint $C(A_t) = \sum_{i \in A_t} c_i \leq B$, then obtain the rewards $r_t(A_t)$, which is a random variable with the martingale assumption (Abbasi-Yadkori, Pál, and Szepesvári 2011b). The expected rewards of A_t is measured by a monotone submodular utility function $F_w(A_t)$ $(i.e., E\left[r_t(A_t)\right] = F_w(A_t))$, where w is a parameter vector. For all time steps $t = 1, 2, \ldots, T$, we choose a subset of arms A_t with respect to budget B and the goal is to obtain the maximum cumulative rewards.

Definition 1. *(submodularity). Let E be a nonempty finite set and 2^E be a collection of all subsets of E. Let $f\colon 2^E \to R$ be a submodular function, i.e.,*

$$\forall X \subseteq Y \in 2^E \text{ and } \forall a \in E \setminus Y, f(a|X) \geq f(a|Y), \quad (1)$$

where $f(a|X) = f(X \cup \{a\}) - f(X)$.

Definition 2. *(monotonity). Let E be a nonempty finite set and 2^E be a collection of all subsets of E. Let $f\colon 2^E \to R$ be a monotone function, i.e.,*

$$\forall X \subseteq Y \in 2^E, f(Y) \geq f(X). \quad (2)$$

However, in bandit setting, the parameter vector w is unknown to us, we use w_* to represent the real value of the parameter vector and assume that $\|w_*\| \leq S$, where S is a positive constant. The linear submodular bandits is a feature-based bandit model known as the contextual bandits (Li et al. 2010), in which we can acquire the side information before making a decision. The utility function of linear submodular bandits is a linear combination of d submodular functions, i.e.,

$$F_{w_*}(A_t) = w_*^\top * \left[F_1(A_t), F_2(A_t), \ldots, F_d(A_t)\right], \quad (3)$$

where $w_* \in R_+^d$ and $F_i(A_t)$ is a submodular function with $F_i(\emptyset) = 0$ (for all $i = 1, 2, \ldots, d$). The submodular function $F_i(A_t)$ can be constructed by the probabilistic coverage model in news article recommendation (Yue and Guestrin 2011; El-Arini et al. 2009).

The goal of linear submodular bandits with a knapsack constraint is to obtain the maximum expected cumulative rewards under a per-round budget constraint, i.e.,

$$\max_{A_t : A_t \in 2^E, C(A_t) \leq B} \sum_{t=1}^{T} E\left[r_t(A_t)\right], \quad (4)$$

where T is the total number of time steps.

α-Regret

Regret, which is the loss of not always pulling the optimal arm, has been widely used in bandit problem as a measure of an algorithm's success. Considering that submodular function maximization is NP-hard, we can only find approximated solutions ($\alpha \in (0,1)$) in polynomial time (Sviridenko 2004; Leskovec et al. 2007). As a result, we can only guarantee an α-approximation solution for the linear submodular bandits with a knapsack constraint, even if we know the parameter w_*. However, w_* is unknown in bandit setting. We use w_t (an estimate of w_*, which is constructed according to the last $t-1$ time steps' feedback), to help us make decision.

For linear submodular bandits with a knapsack constraint, we therefore define an α-approximation unit-cost regret (for simplicity, called α-Regret) as follows: let A^* denote the optimal subset of arms, i.e.,

$$A^* \in \underset{A:A\subseteq E, C(A)\leq B}{\arg\max} F_{w_*}(A). \tag{5}$$

Let A_t denote our chosen subset at time step t, then the α-approximation unit-cost regret or α-Regret is

$$Reg_B^{\alpha}(T) = \alpha * \sum_{t=1}^{T} E\left[\frac{r_t(A^*)}{B}\right] - \sum_{t=1}^{T} E\left[\frac{r_t(A_t)}{B}\right]. \tag{6}$$

Algorithms

In this section, we first introduce the evaluation of w_t and the modified upper confidence bounds. We then propose two greedy algorithms based on the modified UCB rule to solve the linear submodular bandits with a knapsack constraint. Both of these algorithms can be seen as extensions of the greedy algorithms (Sviridenko 2004; Leskovec et al. 2007) to bandit setting. There is a regret vs computational cost trade-off between our two algorithms.

Evaluation of w_t

Let $A_t = \{a_1, a_2, \dots, a_k\}$[1] denote the subset of arms chosen at time step t. For simplicity, we define $A_t(1\colon j) = \{a_1, a_2, \dots, a_j\}$, then the feature vector of the arm a_j (for all $j = 1, 2, \dots, k$) is

$$\Delta(a_j|A_t) = \left[F_1(a_j|A_t), \dots, F_d(a_j|A_t)\right], \tag{7}$$

where

$$F_i(a_j|A_t) = F_i(A_t(1\colon j)) - F_i(A_t(1\colon j-1)) \tag{8}$$

and $F_i(\emptyset) = 0$. Let $r_t(a_j)$ denote the rewards of a_j (for all $j = 1, 2, \dots, k$) and the expected rewards of a_j are

$$E\left[r_t(a_j)\right] = F_{w_*}(a_j|A_t) = w_*^{\top} * \Delta(a_j|A_t). \tag{9}$$

At each time step t, we greedily choose each arm to construct a subset of arms A_t and then acquire all rewards $r_t(a_1), r_t(a_2), \dots, r_t(a_k)$. However, the parameter vector w_* is unknown in bandit setting, we first need to estimate w_*.

The l^2-regularized least-squares estimation has been widely used in the linear bandit problem (Abbasi-Yadkori, Pál, and Szepesvári 2011b; Filippi et al. 2010; Dani, Hayes, and Kakade 2008). Considering the utility function $F_{w_*}(A_t)$ is a linear function, let w_t be the l^2-regularized least-squares estimate of w_* with regularization parameter $\lambda > 0$, i.e.,

$$w_t = \left(X_t^{\top} X_t + \lambda I_d\right)^{-1} X_t^{\top} Y_t. \tag{10}$$

where X_t indicates the features of all chosen arms through last $t-1$ time steps and Y_t forms all corresponding rewards. The row of matrix X_t is the arm feature, i.e.,

$$\forall t, j, X_t(\cdot, :) = \Delta(a_j|A_t) \in R^d. \tag{11}$$

Modified UCB Rule

The confidence bounds can be used elegantly for the trade-off between exploration and exploitation in the multi-armed bandit problem (Auer 2003), especially in the linear bandit problem (Yue and Guestrin 2011; Filippi et al. 2010). Inspired by this, we use the confidence bounds in our algorithms. Let $x = \Delta(a_j|A_t)$ and $V_t = X_t^{\top} X_t + \lambda * I_d$. From the martingale assumption and results of the linear bandits (Abbasi-Yadkori, Pál, and Szepesvári 2011b) we have, with probability $1-\delta$,

$$|w_*^{\top} x - w_t^{\top} x| \leq \beta_t * \|x\|_{V_t^{-1}}, \tag{12}$$

where

$$\beta_t = R\sqrt{2\log\left(\frac{\det(V_t)^{-1/2}\det(\lambda I)^{1/2}}{\delta}\right)} + \lambda^{-1/2} S, \tag{13}$$

$$\|x\|_{V_t^{-1}} = \sqrt{x^{\top} V_t^{-1} x} \tag{14}$$

and R is a positive constant. The confidence interval is

$$\mu(a_j) = \beta_t * \|x\|_{V_t^{-1}}. \tag{15}$$

Considering the budget constraint, we denote the unit-cost confidence interval by the confidence interval for each budget unit, i.e.,

$$\frac{\mu(a_j)}{c_{a_j}} = \frac{\beta_t * \|x\|_{V_t^{-1}}}{c_{a_j}}. \tag{16}$$

Following the optimism in the face of uncertainty principle (Abbasi-Yadkori, Pál, and Szepesvári 2011a), we construct the unit-cost upper confidence bounds as

$$\frac{w_t^{\top} x + \mu(a_j)}{c_{a_j}} = \frac{w_t^{\top} x + \beta_t * \|x\|_{V_t^{-1}}}{c_{a_j}}. \tag{17}$$

We use this modified UCB rule to deal with the trade-off between exploration and exploitation in linear submodular bandits with a knapsack constraint.

Algorithm I: MCSGreedy

In our first algorithm, we greedily choose each arm a with the maximum unit-cost upper confidence bounds, i.e.,

$$a \in \underset{a:a\in E\setminus A_t, C(A_t\cup\{a\})\leq B}{\arg\max} \frac{w_t^{\top} * \Delta(a|A_t) + \mu(a)}{c_a}. \tag{18}$$

However, we also need a partial enumeration ($|A_t| \geq 3$) to achieve the $(1-1/e)$-approximation guarantee and it has

[1] k may be different for different t.

been proven in submodular function maximization problem (Sviridenko 2004). As a result, at each time step t, we first choose the initial best set of arms for cardinality equal to three through a partial enumeration, i.e.,

$$A_t \in \underset{A:A\subseteq E,|A|=3,C(A)\le B}{\arg\max} \left\{F_{w_t}(A)+\mu(A)\right\}. \quad (19)$$

Then we greedily choose each arm as described in Eq. (18) until the budget is exhausted. Considering the partial enumeration, this algorithm is called Modified Cost-Sensitive UCB-based Greedy Algorithm or MCSGreedy, and we give details of MCSGreedy in Algorithm 1.

Algorithm 1 MCSGreedy Algorithm

```
Input: $E = \{1,2,\dots,m\}, C, B$.
$w_t = 0, V_t = \lambda * I_d, X_t = [\,], Y_t = [\,]$.
for $t = 1,2,\dots,T$ do
    $A_t \in \underset{A:A\subseteq E,|A|=3,C(A)\le B}{\arg\max} \left\{F_{w_t}(A)+\mu(A)\right\}$.
    $S_t = \{a \mid a \in E \setminus A_t, C(A_t \cup \{a\}) \le B\}$.
    while $S_t \ne \emptyset$ do
        $a \in \underset{a\in S_t}{\arg\max} \left(\frac{w_t^\top * \Delta(a|A_t)+\mu(a)}{c_a}\right)$.
        $X_t = [X_t; \Delta(a|A_t)]$.
        $A_t = A_t \cup \{a\}$.
        $S_t = \{a \mid a \in S_t, C(A_t \cup \{a\}) \le B\}$.
    end while
    Recommend $A_t$ and obtain rewards $r_t(a_j)$ for all $j = 1,2,\dots,k$.
    $Y_t = [Y_t; r_t(a_j)]$ for all $j = 1,2,\dots,k$.
    $V_t = X_t^\top X_t + \lambda * I_d$.
    $w_t = V_t^{-1} X_t^\top Y_t$. //obtain $w_t$.
end for
```

What's more, if $|A| \le 3$, then we can find the estimated optimal set A_t^* through a enumeration process.

Algorithm II: CGreedy

Algorithm 1 needs a partial enumeration, which is a time-consuming procedure (the partial enumeration needs to compute the utility function F_{w_t} for $O(N^3)$ times at each time step, where $N = |E|$ is the total number of arms). We therefore propose another greedy algorithm, which is able to provide $\frac{1-1/e}{2}$-approximation guarantee without the partial enumeration. We construct our second algorithm as an extension of the Cost-Effective Forward Selection Algorithm (Leskovec et al. 2007) to bandit setting.

In our second algorithm, we first choose two subsets of arms at each time step: A_1 is greedily selected according to UCB rule, i.e.,

$$a \in \underset{a:a\in E\setminus A_t, C(A_t\cup\{a\})\le B}{\arg\max} \left(w_t^\top * \Delta(a|A_t)+\mu(a)\right). \quad (20)$$

The other one A_2 is greedily selected by the modified UCB rule (see Eq. (18)) without the partial enumeration. We then choose the best subset from those two subsets as our final choice. This algorithm is called Competitive UCB-based Greedy Algorithm or CGreedy, and we give details of CGreedy in Algorithm 2.

Algorithm 2 CGreedy Algorithm

```
Input: $E = \{1,2,\dots,m\}, C, B$
$w_t = 0, V_t = \lambda * I_d, X_t = [\,], Y_t = [\,]$.
for $t = 1,2,\dots,T$ do
    $[A_1, X_1]$ = Greedy-Choose($w_t$, $V_t$, Option = 1).
    $[A_2, X_2]$ = Greedy-Choose($w_t$, $V_t$, Option = 2).
    if $F_{w_t}(A_1) \ge F_{w_t}(A_2)$ then
        $A_t = A_1, X_t = X_1$.
    else
        $A_t = A_2, X_t = X_2$.
    end if
    Recommend $A_t$ and obtain rewards $r_t(a_j)$ for all $j = 1,2,\dots,k$.
    $Y_t = [Y_t; r_t(a_j)]$ for all $j = 1,2,\dots,k$.
    $V_t = X_t^\top X_t + \lambda * I_d$.
    $w_t = V_t^{-1} X_t^\top Y_t$. //obtain $w_t$.
end for
```

Theoretical Analysis

In this section, we use α-Regret for the analysis of the linear submodular bandits with a knapsack constraint. We prove Algorithm 1 and Algorithm 2 with different regret bounds and computational costs.

α-Regret Bounds

α-Regret is the difference between the reward derived from our algorithm and the α-approximation of the optimal reward. In Algorithm 1, we have $\alpha = 1 - 1/e \approx 0.632$, which means that Algorithm 1 is at least a $(1-1/e)$-approximation of the optimal solution. We prove the regret bound for Algorithm 1 in Theorem 1.

Theorem 1. *For $\alpha = 1 - 1/e$ and $\lambda \ge K$ ($K = \max |A|, s.t.\ A \in 2^E, C(A) \le B$), with probability at least $1 - \delta$, the α-Regret of Algorithm 1 is bounded by*

$$Reg_B^\alpha(T) \le \frac{2e-1}{Be}\beta_T \left(R\sqrt{d\log\left(\frac{1+TK/\lambda}{1-(1-\delta)^{1/2}}\right)}\right),$$

where $\beta_T = \sqrt{2dTK\log\left(K\left(1+\frac{T}{d}\right)\right)}$.

In Algorithm 2, the α-Regret is a $\frac{1-1/e}{2}$-approximation regret. We prove the regret bound for Algorithm 2 in Theorem 2.

Theorem 2. *For $\alpha = \frac{1-1/e}{2}$ and $\lambda \ge K$ ($K = \max |A|, s.t.\ A \in 2^E, C(A) \le B$), with probability at least $1 - \delta$, the α-Regret of Algorithm 2 is bounded by*

$$Reg_B^\alpha(T) \le \frac{5e-1}{2Be}\beta_T \left(R\sqrt{d\log\left(\frac{1+TK/\lambda}{1-(1-\delta)^{1/3}}\right)}\right),$$

where $\beta_T = \sqrt{2dTK\log\left(K\left(1+\frac{T}{d}\right)\right)}$.

The proof of Theorem 1 and Theorem 2 are based on the results of linear bandits and submodular function maximization (see proofs in long version of the paper).

We simplify the regret bounds by ignoring the constants and some trivial items as follows: let $c_{min} = \min_{c_i \in C} \; c_i$ and A_t denote the subset of arms chosen at time step t, we have

$$\forall t, \sum_{a \in A_t} c_{min} \leq \sum_{a \in A_t} c_a = C(A_t) \leq B. \tag{21}$$

That is, $K = \max_t |A_t| \leq \frac{B}{c_{min}}$. We then have

$$\frac{\beta_T}{B} \leq \sqrt{\frac{dT \log\left(1 + \frac{T}{d}\right)}{c_{min} B}}. \tag{22}$$

Finally, the regret bounds for both Algorithm 1 and Algorithm 2 can be simplified as

$$Reg_B^{\alpha}(T) \leq O\left(d\sqrt{T} \log T\right), \tag{23}$$

which means that the cumulative loss of rewards is increased sublinearly with the total time steps T. That is, the average loss of rewards for each time step is decreased at a rate of $O\left(\frac{d \log T}{\sqrt{T}}\right)$.

Regret vs Computational Cost

Both our two algorithms have sublinear regret bounds on α-Regret. Algorithm 1 is a $(1 - 1/e)$-approximation algorithm while Algorithm 2 is a $\frac{1-1/e}{2}$-approximation algorithm. Therefore, Algorithm 1 has a better α-Regret bound. For computational cost, Algorithm 1 needs a partial enumeration procedure of $O(N^3)$ time and the whole algorithm needs $O(TN^5)$ time, where N is the number of arms. Considering Algorithm 2 only needs $O(TN^2)$ time, it is obvious that Algorithm 2 is more computationally efficient.

Overall, Algorithm 1 achieves a better regret bound and Algorithm 2 is more computationally efficient.

Experiment

In this section, following the previous work (Yue and Guestrin 2011), we empirically evaluate our algorithms on simulation dataset by using news article recommendation (Li et al. 2010) as a case study. We first formulate news article recommendation into the linear submodular bandits. We then introduce a knapsack constraint to the news article recommendation by considering reading-time of news article to be the cost of arm. Lastly, we compare the performance of our algorithms with the baselines through simulation experiments.

News Article Recommendation

Let $E = \{a_1, a_2, \ldots, a_{|E|}\}$ be a set of news articles and a news article $a \in E$ is represented by a feature vector

$$[P_1(a), P_2(a), \ldots, P_d(a)] \in R^d \tag{24}$$

as the information coverage on d different topics. For a subset of arms A_t, the information coverage on d different topics is a feature vector

$$[F_1(A_t), F_2(A_t), \ldots, F_d(A_t)] \tag{25}$$

where

$$F_i(A_t) = 1 - \prod_{a \in A_t} (1 - P_i(a)) \tag{26}$$

and $P_i(a) \in (0, 1)$ for all $i = 1, 2, \ldots, d$.

In this simulation experiment, we randomly generate a d-dimensional vector $(x_1, x_2, \ldots, x_d)$ to represent a news article a, where $x_i \in (0, 1)$(for all $i = 1, 2, \ldots, d$) represents the information coverage of a news article a on the topic i. For each news article a, we assume that it only has a limited number of main topics ($x_i > 0.5$) and noisy topics ($x_i \leq 0.5$). The number of main topics is $N_{main} = 2$ and the number of noise topics is $N_{noise} = 1$. The topics' coverage of each article are sampled from a uniform distribution.

We randomly generate the $w_* \in [0, 1]^d$, which is unknown to our algorithms, to represent a user's interest level in each topic. We also assume that a user will like some topics very much ($w_*(i) \geq 0.8$) and will dislike other topics ($w_*(i) \leq 0.1$). The costs set $C = \{c_1, c_2, \ldots, c_m\}$ are sampled from the uniform distribution and normal distribution. We assume the cost of each news article to be the reading-time and users will have limited time to read all the news articles (El-Arini et al. 2009).

Competing Methods

We compare our two algorithms with the following baselines:

- The LSBGreedy Algorithm is proposed in (Yue and Guestrin 2011) and solves the linear submodular bandits without considering the different costs of arms.
- The Epsilon-Greedy Algorithm, randomly choose the arm with maximum unit-cost rewards according to the ϵ-greedy rule (Auer, Cesa-Bianchi, and Fischer 2002), where $\epsilon \in (0, 1)$.

Results

In Figure 1a, we demonstrate what the learned w_t look compared with w_* and find that w_t achieved by MCSGreedy has the smallest differences with w_*. Also, w_t achieved by CGreedy has smaller differences with w_* than LSBGreedy and Epsilon-Greedy. In Figure 1b, we find that MCSGreedy and CGreedy obtain more average rewards than LSBGreedy and Epsilon-Greedy. We compare the average rewards of different algorithms under different budget in Figure 1c, and we find that MCSGreedy and CGreedy obtain more average rewards with limited budget, while all the algorithms acquire the same average rewards with sufficient budget (which is impossible in real-world applications). It means that our algorithms work well with the budget constraint. We compare the average rewards under different cost intervals in Figure 1d, which are the maximum differences in costs between two arms. It is clear that MCSGreedy and CGreedy greatly outperform LSBGreedy and Epsilon-Greedy although the cost intervals are large (we assume the larger cost intervals to be the representative of more complex setting).

In other settings, a Gaussian distribution is more reasonable for the costs of arms. For example, most of news articles are in medium length while extremely long news articles and extremely short news articles are rare. We also demonstrate

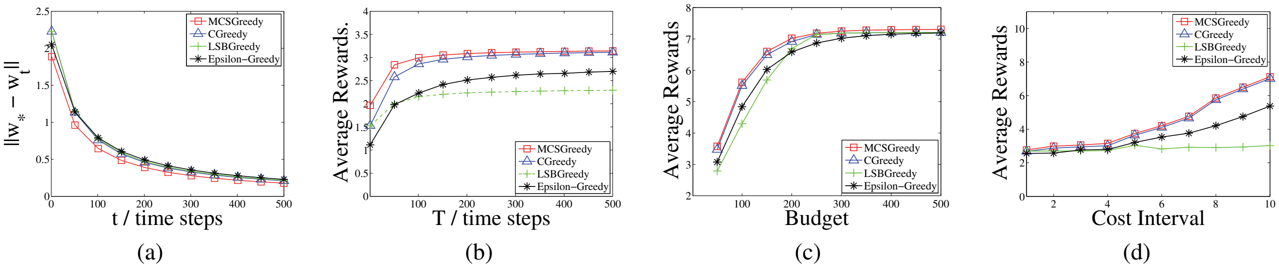

Figure 1: Results comparing MCSGreedy(red), CGreedy(blue), LSBGreedy(green) and Epsilon-Greedy(black).

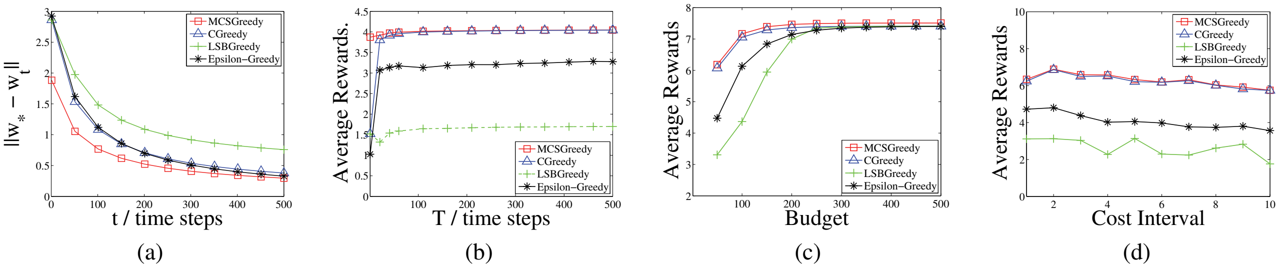

Figure 2: Results comparing MCSGreedy(red), CGreedy(blue), LSBGreedy(green) and Epsilon-Greedy(black).

almost the same results under a Gaussian distribution in Figure 2.

Conclusion and Future Work

In this paper, we introduce a new problem called the linear submodular bandits with a knapsack constraint. To solve this problem, we define an unit-cost upper confidence bounds to control the trade-off between exploration and exploitation. We also propose two algorithms with different regret bounds and computational costs to solve the new problem. To analysis our algorithms, we define an α-Regret and prove that both our algorithms have the sublinear regret bounds. We also demonstrate that our two algorithms outperform the baselines in simulation experiments.

Considering that there are still other constraints in real-world applications, the linear submodular bandits with more complex constraint, such as the multiple knapsack constraints or a matroid constraint, will be the subject of future study.

Acknowledgments

This research is supported by Australian Research Council Projects: DP-140102164 and FT-130101457.

References

Abbasi-Yadkori, Y.; Pál, D.; and Szepesvári, C. 2011a. Improved algorithms for linear stochastic bandits. In *Advances in Neural Information Processing Systems*, 2312–2320.

Abbasi-Yadkori, Y.; Pál, D.; and Szepesvári, C. 2011b. Online least squares estimation with self-normalized processes: An application to bandit problems. *arXiv preprint arXiv:1102.2670*.

Auer, P.; Cesa-Bianchi, N.; and Fischer, P. 2002. Finite-time analysis of the multiarmed bandit problem. *Machine learning* 47(2-3):235–256.

Auer, P. 2003. Using confidence bounds for exploitation-exploration trade-offs. *The Journal of Machine Learning Research* 3:397–422.

Badanidiyuru, A.; Kleinberg, R.; and Slivkins, A. 2013. Bandits with knapsacks. In *Foundations of Computer Science (FOCS), 2013 IEEE 54th Annual Symposium on*, 207–216. IEEE.

Chakrabarti, D.; Kumar, R.; Radlinski, F.; and Upfal, E. 2009. Mortal multi-armed bandits. In *Advances in Neural Information Processing Systems*, 273–280.

Chen, W.; Wang, Y.; and Yuan, Y. 2013. Combinatorial multi-armed bandit: General framework and applications. In *Proceedings of the 30th International Conference on Machine Learning*, 151–159.

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

Dani, V.; Hayes, T. P.; and Kakade, S. M. 2008. Stochastic linear optimization under bandit feedback. In *COLT*, 355–366.

Ding, W.; Qin, T.; Zhang, X.-D.; and Liu, T.-Y. 2013. Multi-armed bandit with budget constraint and variable costs. In *AAAI*.

El-Arini, K.; Veda, G.; Shahaf, D.; and Guestrin, C. 2009. Turning down the noise in the blogosphere. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, 289–298. ACM.

Fang, M., and Tao, D. 2014. Networked bandits with disjoint linear payoffs. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, 1106–1115. ACM.

Filippi, S.; Cappe, O.; Garivier, A.; and Szepesvári, C. 2010. Parametric bandits: The generalized linear case. In *Advances in Neural Information Processing Systems*, 586–594.

Gai, Y.; Krishnamachari, B.; and Jain, R. 2012. Combinatorial network optimization with unknown variables: Multi-armed bandits with linear rewards and individual observations. *IEEE/ACM Transactions on Networking (TON)* 20(5):1466–1478.

Kohli, P.; Salek, M.; and Stoddard, G. 2013. A fast bandit algorithm for recommendation to users with heterogenous tastes. In *AAAI*.

Krause, A., and Golovin, D. 2012. Submodular function maximization. *Tractability: Practical Approaches to Hard Problems* 3:19.

Küçüktunç, O.; Saule, E.; Kaya, K.; and Çatalyürek, Ü. V. 2013. Diversified recommendation on graphs: pitfalls, measures, and algorithms. In *Proceedings of the 22nd international conference on World Wide Web*, 715–726. International World Wide Web Conferences Steering Committee.

Leskovec, J.; Krause, A.; Guestrin, C.; Faloutsos, C.; VanBriesen, J.; and Glance, N. 2007. Cost-effective outbreak detection in networks. In *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, 420–429. ACM.

Li, L.; Chu, W.; Langford, J.; and Schapire, R. E. 2010. A contextual-bandit approach to personalized news article recommendation. In *Proceedings of the 19th international conference on World wide web*, 661–670. ACM.

Radlinski, F.; Kleinberg, R.; and Joachims, T. 2008. Learning diverse rankings with multi-armed bandits. In *Proceedings of the 25th international conference on Machine learning*, 784–791. ACM.

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

Tran-Thanh, L.; Chapman, A.; Munoz De Cote Flores Luna, J. E.; Rogers, A.; and Jennings, N. R. 2010. Epsilon–first policies for budget–limited multi-armed bandits.

Tran-Thanh, L.; Chapman, A.; Rogers, A.; and Jennings, N. R. 2012. Knapsack based optimal policies for budget-limited multi-armed bandits. *arXiv preprint arXiv:1204.1909*.

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

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