Submodular Bandits for Online Personalized Ranking

Author names withheld

Abstract

To be written

1 Introduction

In this paper, we study the problem of recommending the best items to users who are coming sequentially. The learner has access to very less prior information about the users (cold start) and it has to adapt quickly to the user preferences and suggest the best item to each user. Furthermore, we consider a latent variable model such that the user-item preferences are generated as a mixture of user features and item features. Note, that only noisy observations from the user-item preference matrix is only visible to the learner and not the latent user or item features. This results in the user-item preference matrix of being low-rank in nature which is very common in recommender systems (Koren, Bell, and Volinsky, 2009), (Ricci, 2011). Also, we assume that each user has a single best item preference.

This complex problem can be conceptualized as a low rank online learning problem where there are K users and L items. The reward matrix, denoted by $M \in [0,1]^{K\times L}$, generating the rewards for user, item pair has a low rank structure. The online learning game proceeds as follows, at every timestep t, nature reveals one user (or row) from M where user is denoted by i_t. The learner selects some items (or columns) from M, where an item is denoted by $j_t \in [L]$. Then the learner receives feedback $r_t(i_t, j_t)$ from one noisy realization of $(M(i_t, j_t))$, and $\mathbb{E}[r_t(i_t, j_t)] = M(i_t, j_t)$. Then the goal of the learner is to minimize the cumulative regret by quickly identifying the best item j^* for each $i \in [K]$ where $M(i, j^*) = \arg\max_{j\in[L]} M(i, j)$.

1.1 Related Works

Our work lies at the intersection of several exciting areas, which we survey below.

Latent Bandits: The existing algorithms in latent bandit literature can be broadly classified into two groups: the online matrix completion algorithms and the online contextual bandit algorithms. The online matrix completion algorithms tries to reconstruct the reward matrix M from a noisy realization combining different approaches of online learning algorithms and matrix factorization algorithms. The NMF-Bandit algorithm in Sen et al. (2016) is an online matrix completion algorithm which is an ϵ-greedy algorithm that tries to reconstruct the matrix M through non-negative matrix factorization. Note, that this approach requires that all the matrices satisfy a weak statistical Restricted Isometric Property, which is not always feasible in real life applications. Another approach is that of Gopalan, Maillard, and Zaki (2016) where the authors come up with an algorithm which uses the Robust Tensor Power (RTP) method of Anandkumar et al. (2014) to reconstruct the matrix M, and then use the OFUL procedure of Abbasi-Yadkori, Pál, and Szepesvári (2011) to behave greedily over the reconstructed matrix. But the RTP is a costly operation whereas the learner needs to construct a matrix of order $L \times L$ and $L \times L \times L$ to calculate the second and third order tensors for the reconstruction. A more simpler setting has also been studied in Maillard and Mannor (2014) where the users tend to come from only one class and hence this approach is also not quite realistic.

The second type of algorithms are the online contextual bandit algorithms where for each user a separate instance of a base-bandit algorithm is implemented to find the best item for the user. The base-bandits can be randomized algorithms suited for the adversarial settings like EXP3 (Auer et al., 2002) or UCB type algorithms suited for the stochastic setting llike UCB1 (Auer, Cesa-Bianchi, and Fischer, 2002), MOSS (Audibert and Bubeck, 2009), OCUCB (Lattimore, 2015), KL-UCB (Cappé et al., 2013), (Garivier and Cappé, 2011) or the Bayesian algorithms like TS (Thompson, 1933), (Thompson, 1935), (Agrawal and Goyal, 2012).

Ranked Bandits: Bandits have been used to rank items for online recommendations where the goal is is to present a list of d items out of L that maximizes the satisfaction of the user. A popular approach is to model each of the d rank positions as a MAB problem and use a base-bandit algorithm to solve it. This was first proposed in Radlinski, Kleinberg, and Joachims (2008) which showed that query abandonment by user can also be successfully used to learn rankings. Later works such as Slivkins, Radlinski, and Gollapudi (2010) and Slivkins, Radlinski, and Gollapudi (2013) uses additional assumption for ranking to handle exponentially large number of items such that items and user models lie within a metric space and satisfy Lipschitz condition.

Ranking in Click Models: Several algorithms have been proposed to solve the ranking problem in specific click models. For a survey of existing click models a reader may look into Chuklin, Markov, and Rijke (2015). While Katariya et al. (2017), Katariya et al. (2016) works in position based model, Zoghi et al. (2017) works in position based and cascade click model. Finally, Kveton et al. (2017) can be viewed as a generalization of rank-1 bandits of Katariya et al. (2016) to a higher rank. Note, that the theoretical guarantees of these algorithms does not hold beyond the specific click models.

Online Sub-modular maximization: Maximization of submodular functions has wide applications in machine learning, artificial intelligence and in recommender systems (Nemhauser, Wolsey, and Fisher, 1978), (Krause and Golovin, 2014). A submodular function $f : 2^V \to \mathbb{R}$ for a finite ground set V is a set function that assign each subset $S \subseteq V$ a value $f(S)$. We define the gain of the function f as $G_f(e|S) = f(S \cup \{e\}) - f(S)$ where the element $\{e\} \in V \setminus S$ and $S \subseteq V$. Also, it satisfies the following two criteria:-

1. Monotonicity: A set function $f : 2^V \to \mathbb{R}$ is monotone if for every $A \subseteq B \subseteq V, f(A) \leq f(B)$.
2. Submodularity: A set function $f : 2^V \to \mathbb{R}$ is submodular if for every $A \subseteq B \subseteq V$ and $\{e\} \in V \setminus B$ it holds that $G_f(e|A) \geq G_f(e|B)$.

Intuitively, a submodular function states that after performing a set A of actions, the marginal gain of another action e does not increase the gain for performing other actions in $B \setminus A$. Online submodular function maximization has been studied in Streeter and Golovin (2009) where the authors propose a general algorithm (Radlinski, Kleinberg, and Joachims (2008) is a special case of this) and in the contextual feature based setup by Yue and Guestrin (2011).

1.2 Notations, Problem Formulation and Assumptions

We define $[n] = \{1, 2, \ldots, n\}$ and for any two sets A and B, A^B denotes the set of all vectors who take values from A and are indexed by B. Let, $M \in [0,1]^{K \times L}$ denote any matrix, then $M(I, :)$ denote any submatrix of k rows such that $I \in [K]^k$ and similarly $M(:, J)$ denote any submatrix of j columns such that $J \in [L]^j$.

Let M be reward matrix of dimension $K \times L$ where K is the number of user or rows and L is the number of arms or columns. Also, let us assume that this matrix M has a low rank structure of rank $d << \min\{L, K\}$. Let U and V denote the latent matrices for the users and items, which are not visible to the learner such that,

$$M = UV^\intercal \quad \textbf{s.t.} \quad U \in [\mathbb{R}^+]^{K \times d}, V \in [0,1]^{L \times d}$$

Furthermore, we put a constraint on V such that, $\forall j \in [L]$, $\|V(j, :)\|_1 \leq 1$.

Assumption 1. ***(Hott-Topics)*** *We assume that there exists d-column base factors, denoted by $V(J^*, :)$, such that all rows of V can be written as a convex combination of $V(J^*, :)$ and the zero vector and $J^* = [d]$. We denote the column factors by $V^* = V(J*, :)$. Therefore, for any $i \in [L]$, it can be represented by*

$$V(i, :) = a_i V(J^*, :),$$

where $\exists a_i \in [0,1]^d$ and $\|a_i\|_1 \leq 1$.

Assumption 2. ***(Click Model)*** *For each user i_t revealed by the nature at round t, the learner is allowed to suggest at-most d-items, where d is the rank of the matrix $\bar{R}$. The user can click one, or all, or none of the recommendations.*

Discussion 1. *The above Assumption 2 is another instance of the document click model (DCM) discussed in section 1.1. It can be conceptualized in the real-world scenario where the learner has to suggest movies to users and each movie belongs to a different genre (say thriller, romance, comedy, etc). So, the learner can suggest d movies belonging to different genres to each user in a webpage, and the user can click one, or all, or none of the recommended movies (query abandonement).*

Noise Model: Our noise model is quite different from the existing stochastic noise model assumptions of various click models. Independent Bernoulli rewards on the entries of the user-item preference matrix M is not feasible because the hott-topics assumption is required for every realization of the matrix M. Hence, at every timestep t, we generate a noisy matrix $M_t = UD_tV^\intercal$, where D_t is a diagonal matrix such that $D_t(i, i) \in [0, 1]$. Thus, for every such realization of $M_t, \forall t \in [n]$ the hott-topics structure of M is preserved.

The main goal of the learning agent is to minimize the cumulative regret until the end of horizon n. We define the cumulative regret, denoted by $\mathcal{R}_n$ as,

$$\mathcal{R}_n = \sum_{t=1}^{n} \left\{ \sum_{z=1}^{d} \left(r_t\left(i_t, j^*\right) - r_t\left(i_t, j_{t,z}\right) \right) \right\}$$

where, $j^* = \arg\max_{j \in [L]} \{M(i_t, j)\}$ and $j_{t,z}$ be the suggestion of the learner for the i_t -th user for $z = 1, 2, \ldots, d$. Note that $r_t\left(i_t, j^*\right) \sim Ber(M\left(i_t, j^*\right))$ and $r_t\left(i_t, j_{t,z}\right) \sim Ber(M\left(i_t, j_{t,z}\right))$. Taking expectation over both sides, we can show that,

$$\begin{aligned}
\mathbb{E}[\mathcal{R}_n] &= \mathbb{E}\left[\sum_{t=1}^{n} \left\{ \sum_{z=1}^{d} \left(r_{z,t}\left(i_t, j^*\right) - r_{z,t}\left(i_t, j_{z,t}\right) \right) \right\} \right] \\
&= \mathbb{E}\left[\sum_{t=1}^{T} \sum_{z=1}^{d} \left(N_{i_t, j_{z,t}} \right) \right] \Delta_{i_t, j_{z,t}}
\end{aligned}$$

where, $\Delta_{i_t, j_{z,t}} = M(i_t, j^*) - M(i_t, j_{z,t})$ and $N_{i_t, j_{z,t}}$ is the number of times the learner has observed the $j_{z,t}$-th item for the i_t-th user. Let, $\Delta = \min_{i \in [K], j \in [L]} \{\Delta_{i,j}\}$ be the minimum gap over all the user, item pair in M.

2 Contributions

To be written.

3 Proposed Algorithms

We propose the algorithm Latent Ranked Bandit, abbreviated as LRB (see Algorithm 1) for solving the personalized ranking problem. This algorithm is motivated by the Ranked Bandit Algorithm (RBA) from Radlinski, Kleinberg, and Joachims (2008) which is suited for finding a global ranking amongst all the users in the cascade based model. LRB is divided into two main components, the d column MABs denoted by $\text{MAB}_1(n), \text{MAB}_2(n), \ldots, \text{MAB}_d(n)$ and the K row Weighted Majority Algorithms (WMA) for each user $[K]$. The WMA is motivated from Littlestone and Warmuth (1994) which is suited for the total information setting. Note, that in our DCM setting all the clicks by the user is seen by the system and so it is also a specific variation of total information setting. Each WMA consist of d arms for each of the ranks $1, 2 \ldots, d$ and its main goal is to suggest the best item for the user in rank 1. LRB proceeds as follows, at every timestep as user i_t is revealed by nature, the d column MABs suggests columns $\ell_1, \ell_2, \ldots, \ell_d$ which it deems to be the d best columns and by virtue of our setting, the d hott-topics. If, there is any overlap in the suggestion, an arbitrary column is suggested which has not been selected before. Then, the row WMA for the i_t-th user selects a permutation $\Pi_{i_t}(\ell_1, \ell_2, \ldots, \ell_d)$ by sampling through its distribution over the d-ranks and suggests $\tilde{\ell}_1, \tilde{\ell}_2, \ldots, \tilde{\ell}_d$ such that the item in rank 1 is the best item for the i_t-th user.

Finally, after all the user clicks are recorded by the system both the column MABs and row WMA (for the i_t-th user only) is updated. The column bandits are updated with feedback $f_{k,t}$ such the monotonicity and submodularity properties discussed in section 1.1 are maintained. Note that RBA is also a special case of submodular bandits such that $f_{k,t} \in \{0, 1\}$ (Streeter and Golovin, 2009). The row WMA update for the i_t-th user is quite straightforward as all the rewards for the d clicks are observed. An illustrative diagram of the entire process is shown in Figure 1.

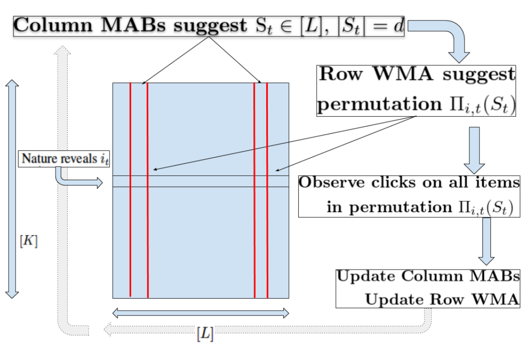


Figure 1: Latent Ranked Bandit in rank $d = 2$ scenario.

Algorithm 1 Latent Ranked Bandit

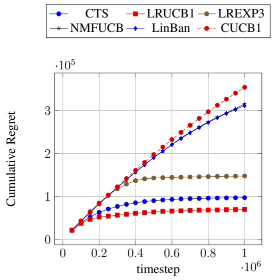
Input: Rank d, horizon n.
Initialize $\text{MAB}_1(n), \text{MAB}_2(n), \ldots, \text{MAB}_d(n)$
Initialize $\text{WMA}_1(n), \text{WMA}_2(n), \ldots, \text{WMA}_K(n)$
for $t = 1, \ldots, n$ **do**
  User i_t comes to the system
  for $k = 1, \ldots, d$ **do**
    $\ell_{k,t} \leftarrow$ suggest item $MAB_k(n)$
    if $\ell_{k,t} \in \ell_{1,t}, \ldots, \ell_{k-1,t}$ **then**
      $\ell_{k,t} \leftarrow$ Select arbitrary unselected item from $[L] \setminus \ell_{1,t}, \ldots, \ell_{k-1,t}$
  $\tilde{\ell}_{1,t}, \tilde{\ell}_{2,t}, \ldots, \tilde{\ell}_{d,t} \leftarrow$ Permutation by $\text{WMA}_{i_t}(\ell_{1,t}, \ell_{2,t}, \ldots, \ell_{d,t})$ by sorting descendingly according to $w_{i_t,1}, w_{i_t,2}, \ldots, w_{i_t,d}$.
  Present $\tilde{\ell}_{1,t}, \tilde{\ell}_{2,t}, \ldots, \tilde{\ell}_{d,t}$ to user i_t and record feedback $r_t(\tilde{\ell}_{1,t}), r_t(\tilde{\ell}_{2,t}), \ldots, r_t(\tilde{\ell}_{d,t})$.
  Call Procedure UpdateColumnMAB
  Call Procedure UpdateRowWMA(i_t)
procedure UPDATECOLUMNMAB
  for $k = 1, \ldots, d$ **do**
    Update $\text{MAB}_k(n)$ with feedback $f_{k,t} = \max_{j \in [k]} r_t(i_t, \ell_{j,t}) - \max_{j \in [k-1]} r_t(i_t, \ell_{j,t})$
procedure UPDATEROWWMA(i_t)
  for $j = 1, \ldots, d$ **do**
    $w_{i_t,j} = w_{i_t,j} + \frac{1}{j} r_t(\tilde{\ell}_{j,t})$

4 Experiments

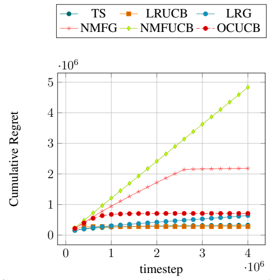
In this section, we show

5 Conclusions and Future Direction

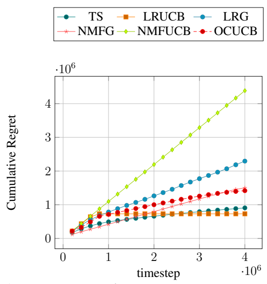
To be written.



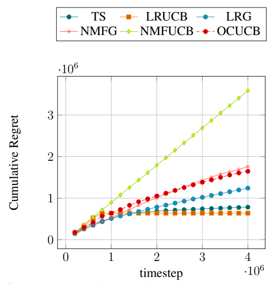
(a) Expt-1: 1000 Users, 50 Bernoulli-distributed arms, Round-Robin, Noisy Setting, Rank 2, equal sized clusters



(b) Expt-2: 2048 Users, 64 Bernoulli-distributed arms, Round-Robin, Noisy Setting, Rank 2, un-equal sized clusters, 70:30 split

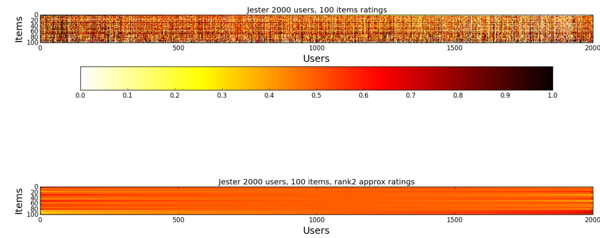


(c) Expt-3: 4096 Users, 128 Bernoulli-distributed arms, Round-Robin, Noisy Setting, Rank 2, equal sized clusters

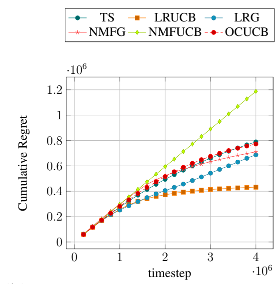


(d) Expt-4: 4096 Users, 128 Bernoulli-distributed arms, Round-Robin, Noisy Setting, Rank 2, un-equal sized clusters, 80:20 split

Figure 2: A comparison of the cumulative regret incurred by the various bandit algorithms.



(a) Expt-1: 2000 Users, 100 arms, Round-Robin, Noisy Setting, Rank 2, Jester Dataset



(b) Expt-1: 2000 Users, 100 arms, Round-Robin, Noisy Setting, Rank 2, Jester Dataset

Figure 3: A comparison of the cumulative regret in Jester Dataset

References

Abbasi-Yadkori, Y.; Pál, D.; and Szepesvári, C. 2011. Improved algorithms for linear stochastic bandits. *Advances in Neural Information Processing Systems 24: 25th Annual Conference on Neural Information Processing Systems 2011. Proceedings of a meeting held 12-14 December 2011, Granada, Spain.* 2312–2320.

Agrawal, S., and Goyal, N. 2012. Analysis of thompson sampling for the multi-armed bandit problem. *COLT 2012 - The 25th Annual Conference on Learning Theory, June 25-27, 2012, Edinburgh, Scotland* 39.1–39.26.

Anandkumar, A.; Ge, R.; Hsu, D.; Kakade, S. M.; and Telgarsky, M. 2014. Tensor decompositions for learning latent variable models. *The Journal of Machine Learning Research* 15(1):2773–2832.

Audibert, J., and Bubeck, S. 2009. Minimax policies for adversarial and stochastic bandits. *COLT 2009 - The 22nd Conference on Learning Theory, Montreal, Quebec, Canada, June 18-21, 2009* 217–226.

Auer, P.; Cesa-Bianchi, N.; Freund, Y.; and Schapire, R. E. 2002. The nonstochastic multiarmed bandit problem. *SIAM Journal on Computing* 32(1):48–77.

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

Cappé, O.; Garivier, A.; Maillard, O.-A.; Munos, R.; Stoltz, G.; et al. 2013. Kullback–leibler upper confidence bounds for optimal sequential allocation. *The Annals of Statistics* 41(3):1516–1541.

Chuklin, A.; Markov, I.; and Rijke, M. d. 2015. Click models for web search. *Synthesis Lectures on Information Concepts, Retrieval, and Services* 7(3):1–115.

Garivier, A., and Cappé, O. 2011. The KL-UCB algorithm for bounded stochastic bandits and beyond. *COLT 2011 - The 24th Annual Conference on Learning Theory, June 9-11, 2011, Budapest, Hungary* 19:359–376.

Gopalan, A.; Maillard, O.-A.; and Zaki, M. 2016. Low-rank bandits with latent mixtures. *arXiv preprint arXiv:1609.01508*.

Katariya, S.; Kveton, B.; Szepesvari, C.; Vernade, C.; and Wen, Z. 2016. Stochastic rank-1 bandits. *arXiv preprint arXiv:1608.03023*.

Katariya, S.; Kveton, B.; Szepesvári, C.; Vernade, C.; and Wen, Z. 2017. Bernoulli rank-1 bandits for click feedback. *arXiv preprint arXiv:1703.06513*.

Koren, Y.; Bell, R.; and Volinsky, C. 2009. Matrix factorization techniques for recommender systems. *Computer* (8):30–37.

Krause, A., and Golovin, D. 2014. Submodular function maximization.

Kveton, B.; Szepesvari, C.; Rao, A.; Wen, Z.; Abbasi-Yadkori, Y.; and Muthukrishnan, S. 2017. Stochastic low-rank bandits. *arXiv preprint arXiv:1712.04644*.

Lattimore, T. 2015. Optimally confident UCB : Improved regret for finite-armed bandits. *CoRR* abs/1507.07880.

Littlestone, N., and Warmuth, M. K. 1994. The weighted majority algorithm. *Inf. Comput.* 108(2):212–261.

Maillard, O.-A., and Mannor, S. 2014. Latent bandits. In *International Conference on Machine Learning*, 136–144.

Nemhauser, G. L.; Wolsey, L. A.; and Fisher, M. L. 1978. An analysis of approximations for maximizing submodular set functionsi. *Mathematical programming* 14(1):265–294.

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.

Ricci, F. 2011. Liorrokach, and brachashapira." introduction to recommender systems handbook.

Sen, R.; Shanmugam, K.; Kocaoglu, M.; Dimakis, A. G.; and Shakkottai, S. 2016. Contextual bandits with latent confounders: An nmf approach. *arXiv preprint arXiv:1606.00119*.

Slivkins, A.; Radlinski, F.; and Gollapudi, S. 2010. Ranked bandits in metric spaces: learning optimally diverse rankings over large document collections. *arXiv preprint arXiv:1005.5197*.

Slivkins, A.; Radlinski, F.; and Gollapudi, S. 2013. Ranked bandits in metric spaces: learning diverse rankings over large document collections. *Journal of Machine Learning Research* 14(Feb):399–436.

Streeter, M., and Golovin, D. 2009. An online algorithm for maximizing submodular functions. In *Advances in Neural Information Processing Systems*, 1577–1584.

Thompson, W. R. 1933. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika* 285–294.

Thompson, W. R. 1935. On the theory of apportionment. *American Journal of Mathematics* 57(2):450–456.

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

Zoghi, M.; Tunys, T.; Ghavamzadeh, M.; Kveton, B.; Szepesvari, C.; and Wen, Z. 2017. Online learning to rank in stochastic click models. *arXiv preprint arXiv:1703.02527*.