

Improved Algorithm on Online Clustering of Bandits

Shuai Li

Shanghai Jiao Tong University

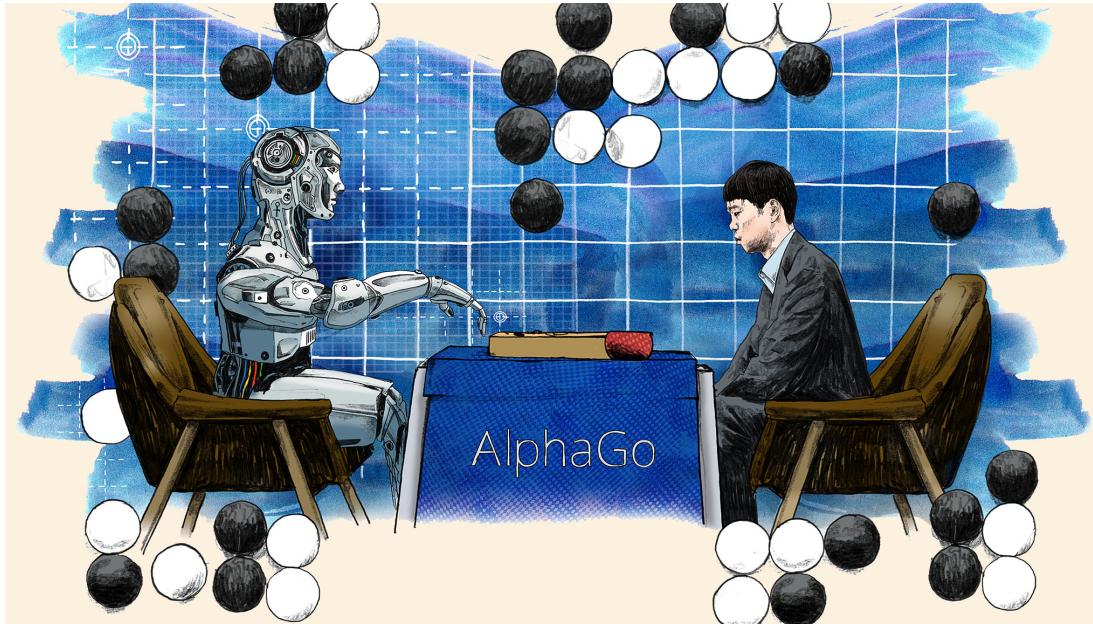
was in

The Chinese University of Hong Kong

joint work with

Wei Chen, S Li, Kwong-Sak Leung

Motivation – Reinforcement Learning



AlphaGo, AlphaStar



Motivation -- Multi-armed Bandits

- A special case of Reinforcement Learning



Multi-armed Bandits

- There are L arms
 - Each arm a has an unknown reward distribution with unknown mean $\alpha(a)$
 - The best arm is $a^* = \operatorname{argmax}_a \alpha(a)$



- At each time t
 - The learning agent selects an arm a_t
 - Observes the reward $X_{a_t,t}$

Multi-armed Bandits (Continued)

- The objective is to minimize the **regret** in T rounds

$$R(T) = T \cdot \alpha(a^*) - \mathbb{E} \left[\sum_{t=1}^T \alpha(a_t) \right]$$

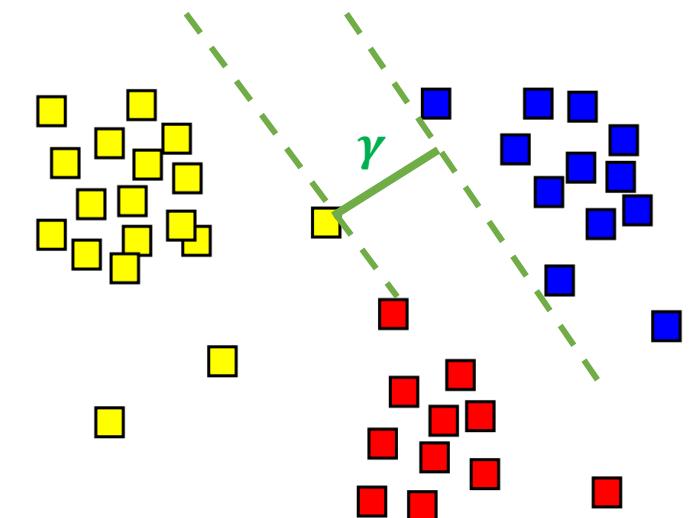
- Balance the trade-off between exploitation and exploration
 - Exploitation**: select arms that yield good results so far
 - Exploration**: select arms that have not been tried much before

Contextual Multi-armed Bandits

- Contexts
 - User profiles, search key words
 - Important for search, recommendations
- Usually suppose each arm a has a feature representation $x_{a,t} \in \mathbb{R}^d$
 - Contexts could change over time
- The reward mean is $\alpha_t(a) = \theta^\top x_{a,t}$
 - for some fixed but unknown weight vector θ

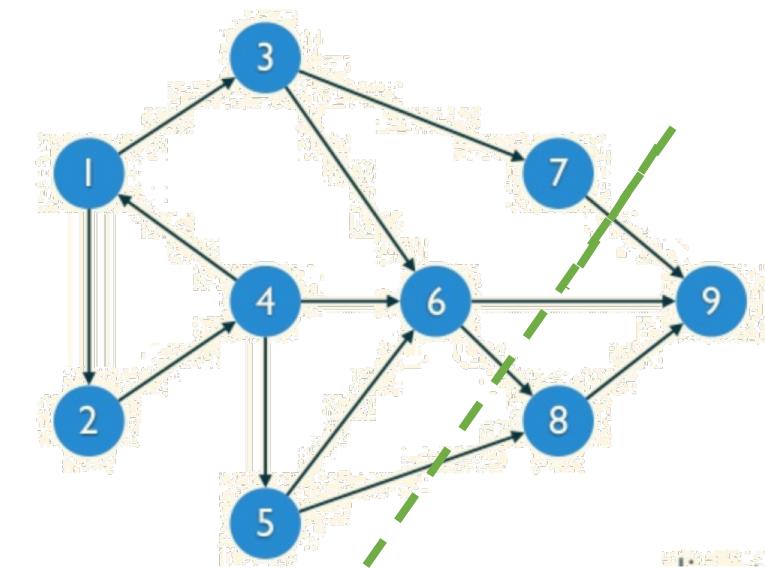
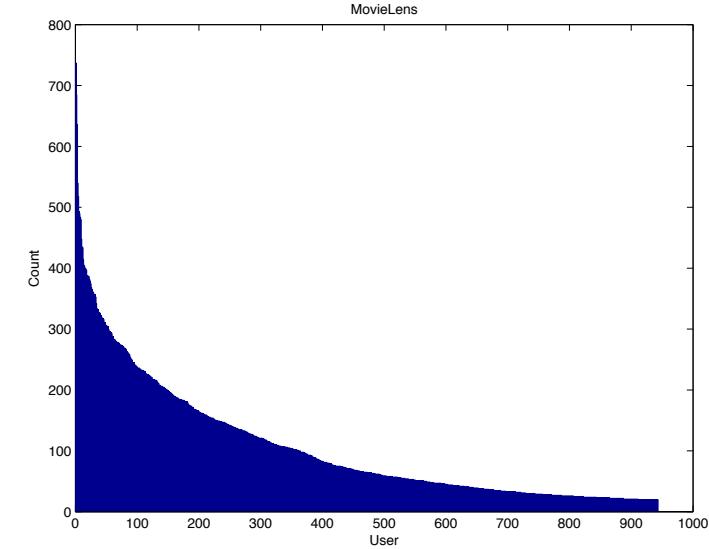
Online Clustering of Bandits

- Drawbacks of simple contextual bandits
 - They assume the weight vector θ is the **same** for all users
- Online clustering of bandits
 - Users with strong ties (like friendship) usually have similar interests
 - Assume
 - Users within the **same cluster** have the **same** θ
 - Users of **different clusters** have weight gap $\|\theta_i - \theta_j\| \geq \gamma$
 - Find clustering adaptively as well as recommending



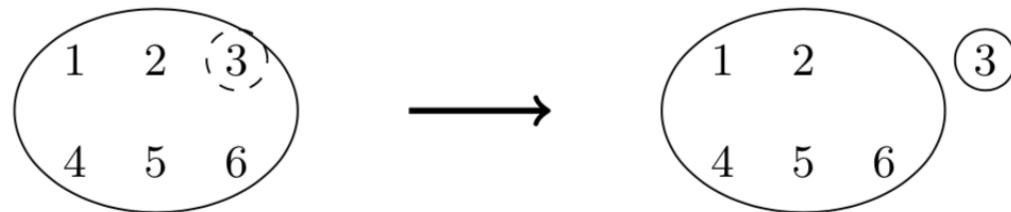
Existing Problems

- They assume the user distribution is uniform
- If generalizing the algorithm to arbitrary distribution over users
 - Their algorithm is much inefficient
 - The regret will depend on the minimal user frequency
 - $R(T) = O\left(d\sqrt{mT} \ln T + \frac{1}{p_{\min}\gamma^2\lambda_x^3} \ln T\right)$

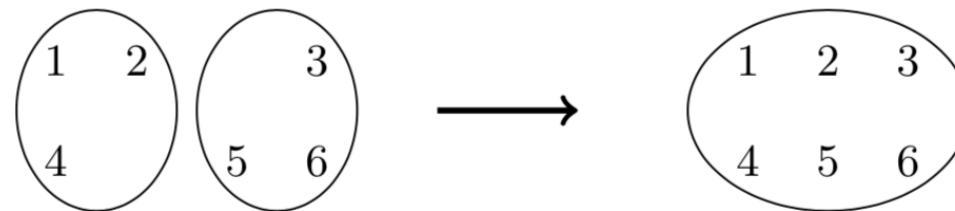


Our Work – SCLUB (set-based clustering of bandits)

- Generalize the setting to allow arbitrary distribution over users
- Split a user out of the current cluster if we finds *inconsistency*



- Merge two *good* clusters together



Results

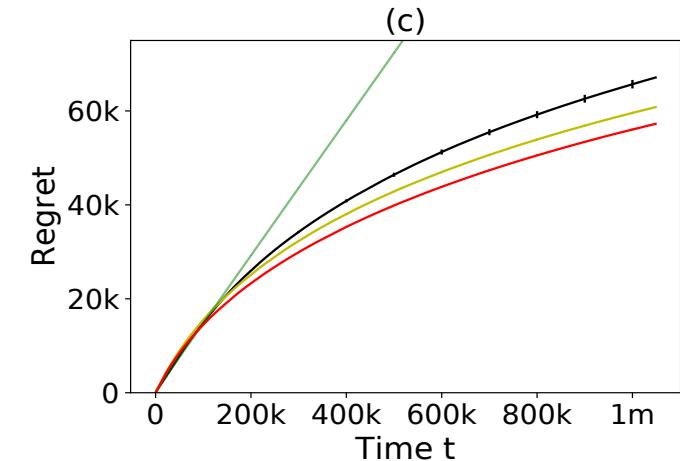
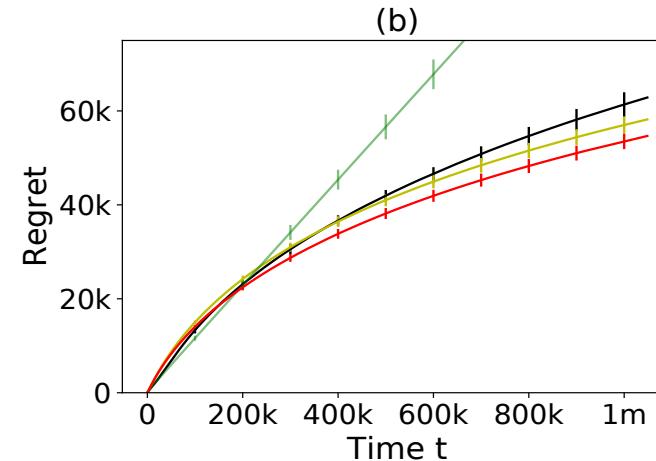
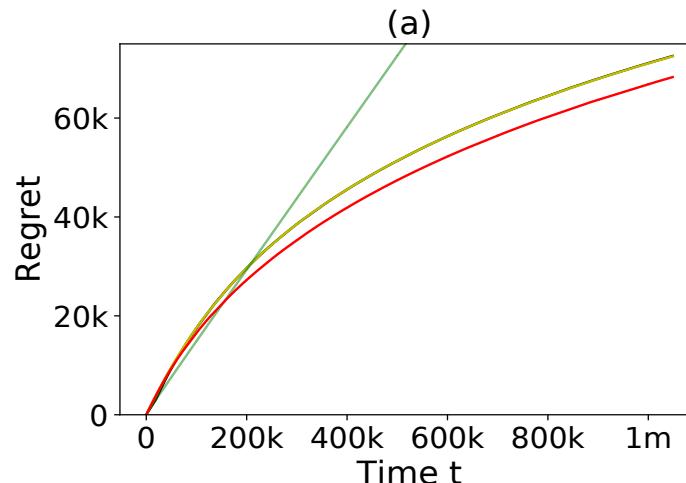
- Regret

$$R(T) = O\left(d\sqrt{mT} \ln T + \left(\frac{1}{\gamma_p^2} + \frac{n_u}{\gamma^2 \lambda_x^3}\right) \ln T\right)$$

- compared to $R(T) = O\left(d\sqrt{mT} \ln T + \frac{1}{p_{\min} \gamma^2 \lambda_x^3} \ln T\right)$

Experiments – Synthetic

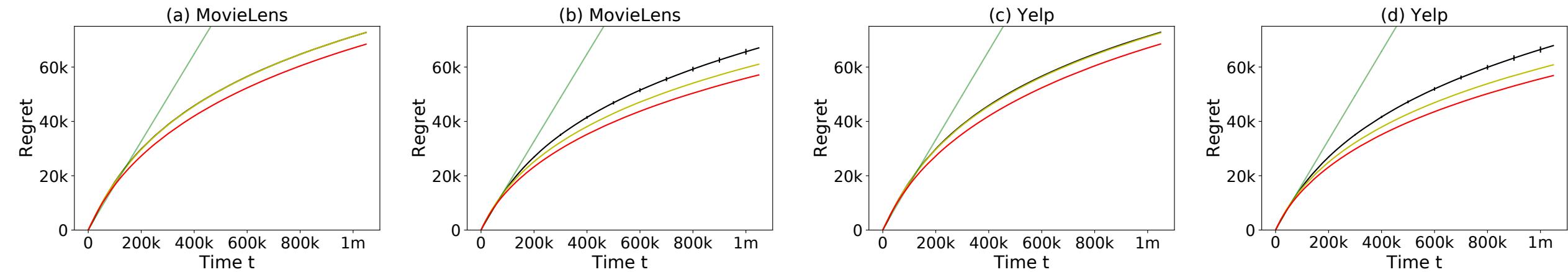
- 1000 users, 10 clusters, randomly generated weight vectors, $d = 20$
 - (a) uniform distribution over users
 - (b) arbitrary distribution over clusters
 - (c) arbitrary distribution over users
- —Ours ---CLUB ---LinUCB-One ---LinUCB-Ind



Experiments – Real Datasets

- 1000 users, $d = 20$
- (a)(c) uniform distribution over users
- (b)(d) arbitrary distribution over users

• ● ---Ours ---CLUB ---LinUCB-One ---LinUCB-Ind



Future Work

- Asymmetric relationships between users
 - Recommendations for low-frequency users can use information (or feedback) from high-frequency users, but not vice versa
 - Nested clusters
- Use the same idea to improve the collaborative filtering bandits
- Generalize the collaborative filtering bandits to the setting of changing item set

Thanks!

&

Questions?