Contextual Combinatorial Cascading Bandits

Shuai Li[1], Baoxiang Wang[1], Shengyu Zhang[1], Wei Chen[2]

1 The Chinese University of Hong Kong

2 Microsoft Research

ICML 2016

Multi-armed Bandit Problem

- A special case of reinforcement learning

- There are m arms (machines)
- Arm i has an unknown reward distribution with unknown mean μ_i
  - best arm $\mu^* = \max \mu_i$

Multi-armed Bandit Problem

- In each round t, the learning agent selects one arm i_t to play and observes the reward $\boldsymbol{R}_t(i_t)$
- Regret after playing T rounds:

Always play the best arm

$$\text{Regret} = T\mu^* - \mathbb{E}[\textstyle\sum_{t=1}^{T} \boldsymbol{R}_t(i_t)\,]$$

- Objective: minimize regret in T rounds
- Balancing tradeoff between exploitation and exploration
  - Exploration: try options that have not been tried much before
  - Exploitation: try options that yield good results so far

Multi-armed Bandit Problem

- UCB (Upper Confidence Bound) [Auer, Cesa-Bianchi, Fischer 2002]
  - UCB policy: select

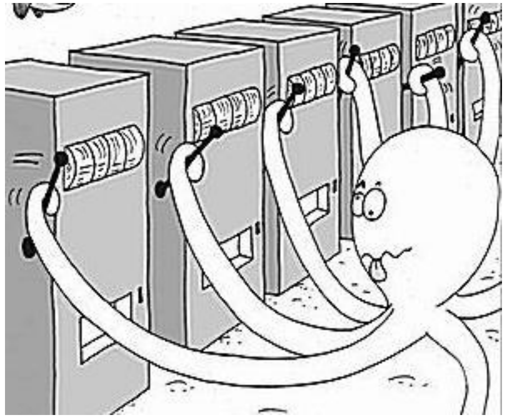
Exploitation

Exploration

$$i = \text{argmax}_{i\in[m]} \left(\hat{\mu}_i + \sqrt{\frac{2\ln t}{T_i}} \right)$$

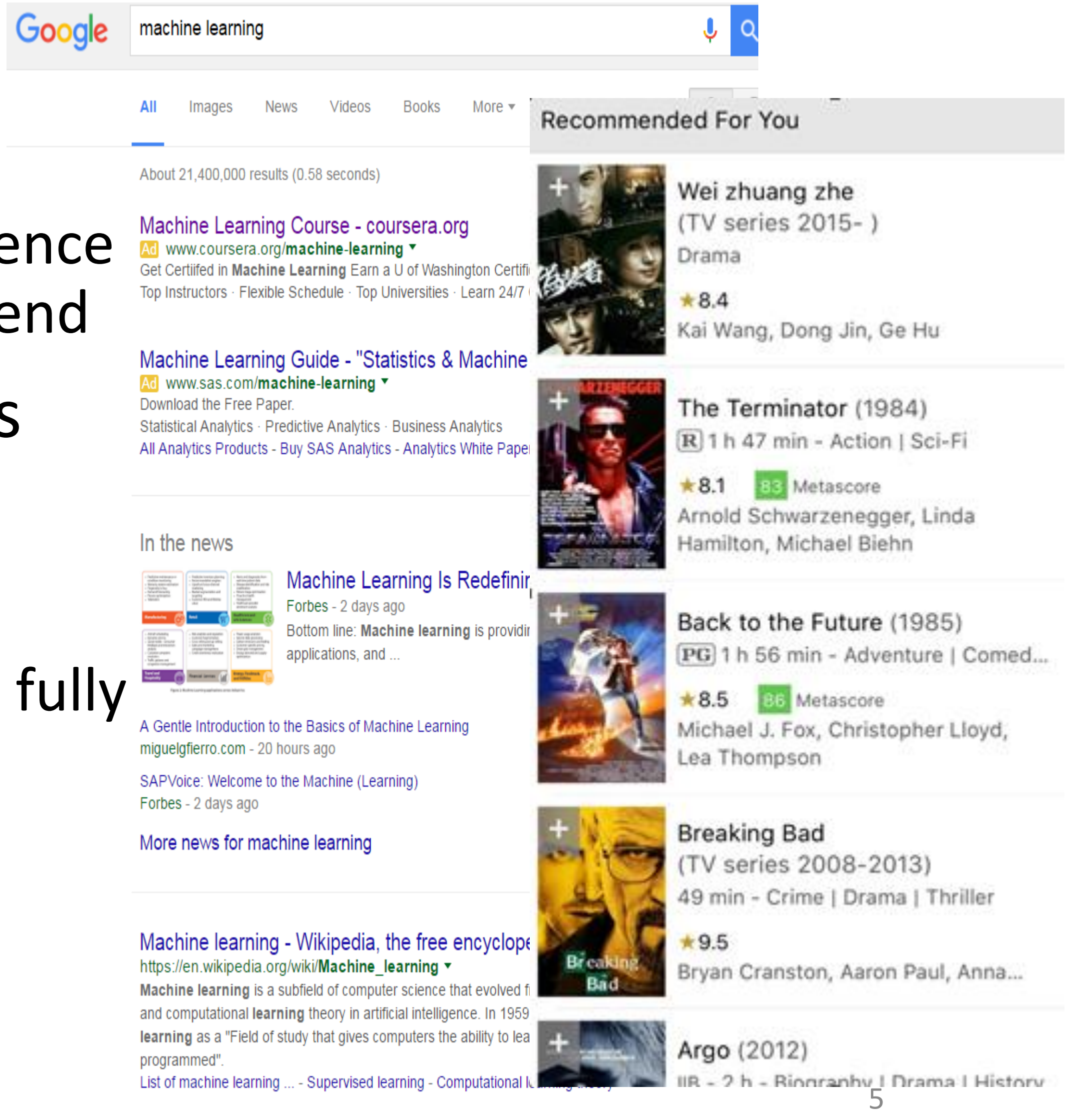
where T_i is the played times of arm i.

  - Gap-dependent bound $O(\log T \sum_{i:\Delta_i>0} 1/\Delta_i)$, $\Delta_i = \mu^* - \mu_i$, match lower bound
  - Gap-free bound $O(\sqrt{mT\log T})$, tight up to a factor of $\sqrt{\log T}$



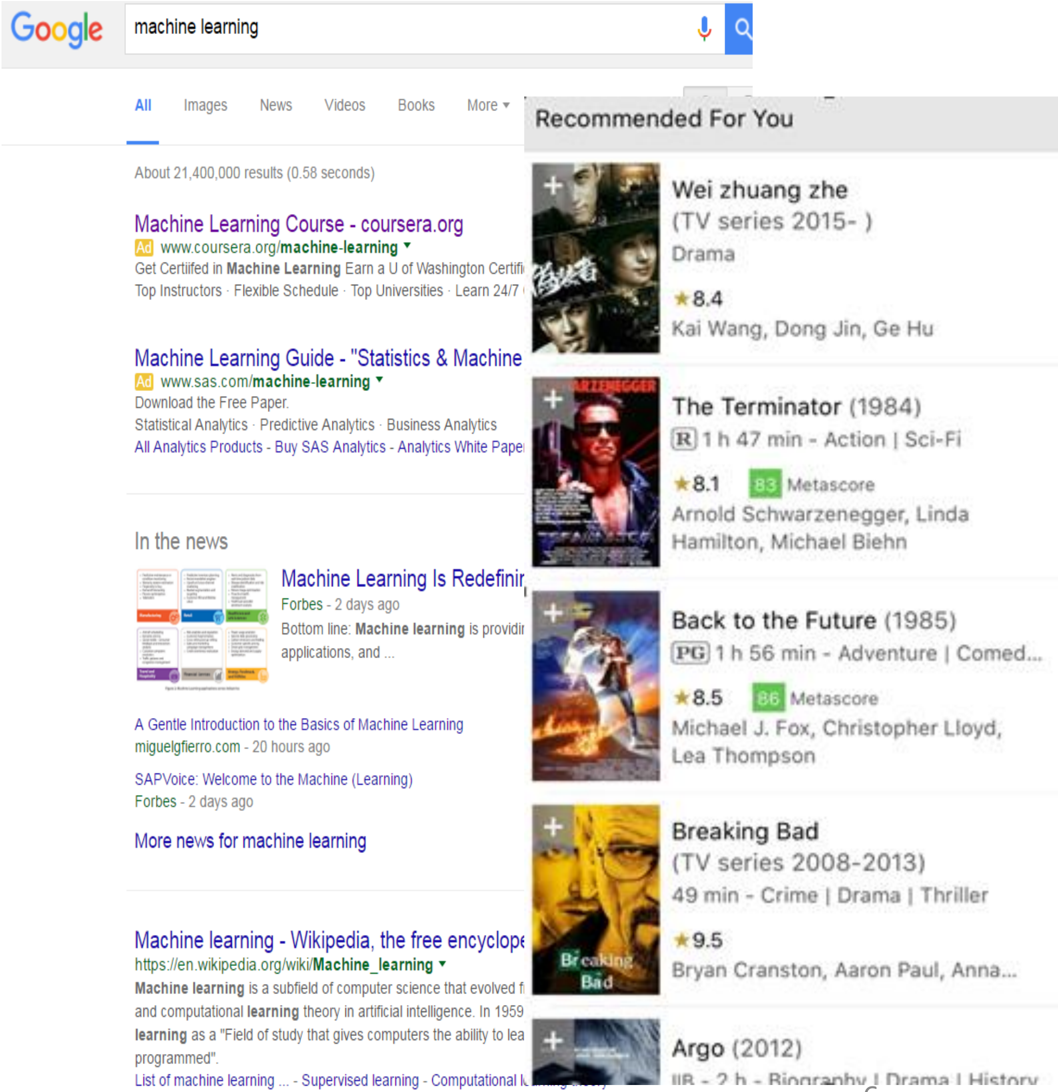
Combinatorial Multi-Armed Bandit

- Action is combinatorial
  - Selecting a matching, a routing path, a sequence of ads to display, a list of movies to recommend
- May observe some feedback on elements involved (e.g. semi-bandit feedback)
- Challenges
  - Exponential number of actions --- cannot be fully explored
  - Offline optimization may already be hard



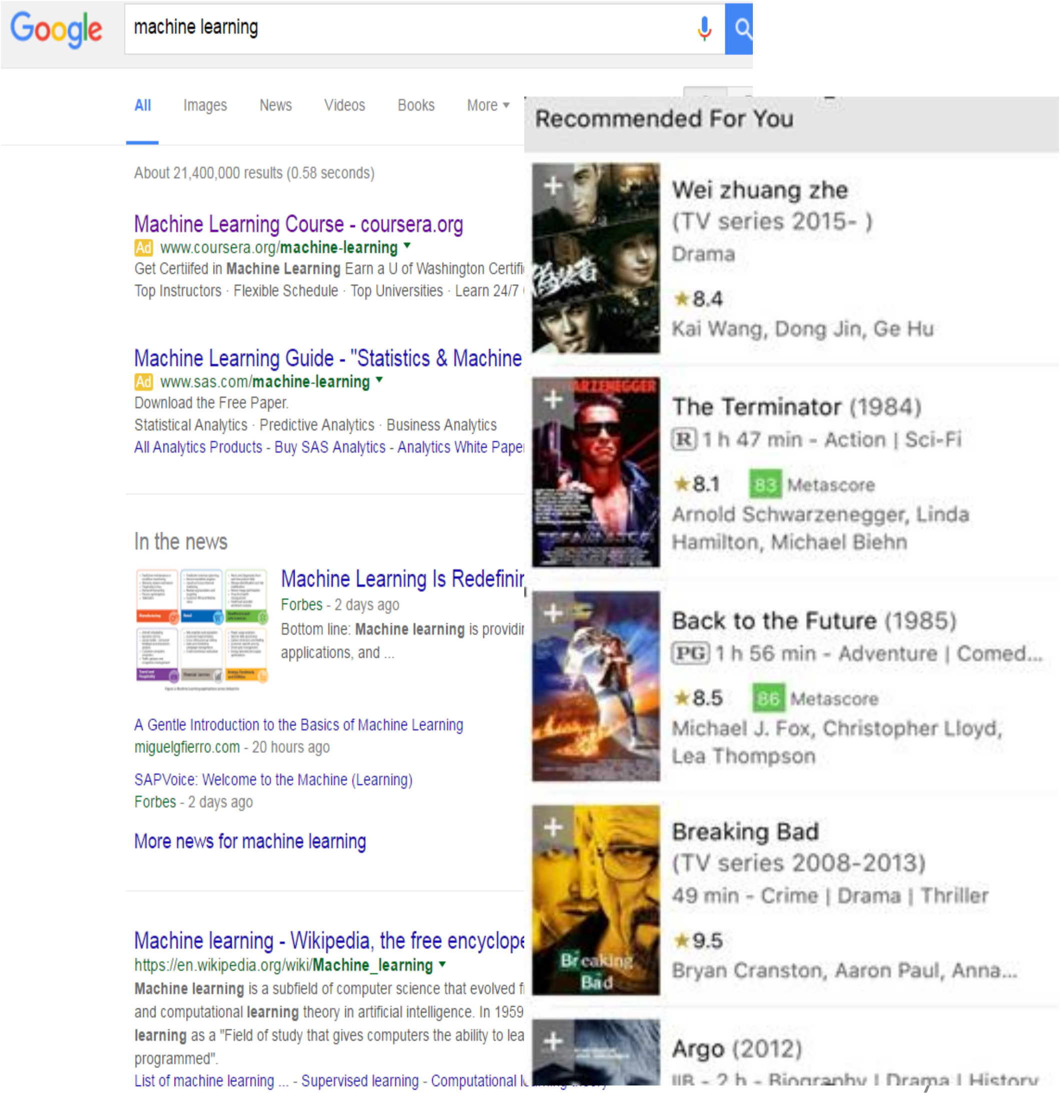
Motivation of Cascading Bandit

- Websites search results
- Recommended movies
- Etc.
- All are sequential lists
  - Users are likely to go through the list from top down
  - Stop at the first satisfactory item
  - Click as the feedback
  - Online feedback helps improving list quality



Contextual Combinatorial Cascading Bandit

- Contexts
  - User profiles, search keywords
  - Important for search, recommendations
- Combinatorial
  - Action is selection of a sequence
  - May have other combinatorial constraints (children movies)



Our Contribution

- Formulate the Contextual Combinatorial Cascading Bandits problem
- Proposed C^3-UCB algorithm, handles
  - contextual information
  - cascading feedback
  - position discount
    (top positions may be more important)
  - general reward function

	context	cascading	Position discount	General reward
Combinatorial UCB[1]	No	Yes	No	Yes
Contextual Combinatorial UCB[2]	Yes	No	No	Yes
Comb-Cascade[3]	No	Yes	No	No
C^3-UCB(ours)	Yes	Yes	Yes	Yes

- Theoretical analysis and empirical evaluation

1 Chen et al. 2013

2 Qin et al. 2014

3 Kveton et al. 2015

Setting & Algorithms

Setting of C^{3}B

- $E = \{1, \dots, L\}$: set of base arms
- Action $A = (a_1, \dots, a_k)$: a sequence of base arms
  - There is a feasible action set $\boldsymbol{\mathcal{S}}$.
- At each time $t \geq 1$
  - set of contexts $\{x_{t,a}\}_{a \in E}$ are given (e.g. user/keyword features)
  - learning agent selects a feasible action $\boldsymbol{A}_t = (\boldsymbol{a}_1^t, \dots, \boldsymbol{a}_{|\boldsymbol{A}_t|}^t)$
  - The user checks from the first item and stops at $\boldsymbol{O}_t$-th item.
  - Feedback: observe weights of first $\boldsymbol{O}_t$ items, $\boldsymbol{R}_t(\boldsymbol{a}_k^t), k \leq \boldsymbol{O}_t$.

$$\mathbb{E}[\boldsymbol{R}_t(a)] = \theta_*^\top \cdot x_{t,a} = w_{t,a}$$

Fixed but unknown

Setting of C³B

- Assume the expected reward of an action A is a function of $w_t = \{w_{t,a}\}_{a \in E}$ of each base arm, $f(A, w_t)$.
- Regret in T rounds

Best cumulative reward

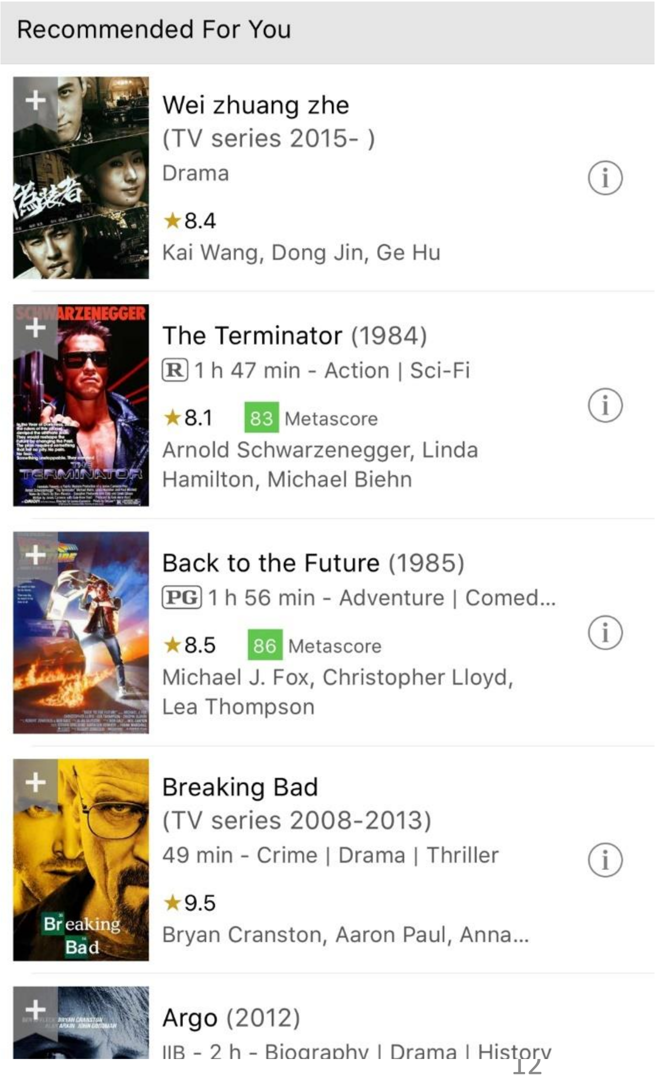
$$Regret = \sum_{t=1}^{T} f_t^* - \mathbb{E}\left[\sum_{t=1}^{T} f(\boldsymbol{A}_t, w_t)\right]$$

  - f_t^*: max expected reward in round t

Example – movie recommendation

- Each movie i has a feature vector m_i
- At time t,
  - A random user comes with feature vector u_t
  - Use $x_{i,t} = g(m_i, u_t)$, a function of m_i and u_t, (e.g. direct sum, outer-product) as context
  - The learning agent recommends a list of movies $\boldsymbol{A}_t$
  - The user checks from the first movie and stops at the attractive one.
  - The learning agent receives reward γ_k if the user stops at position k.

$$1 = \gamma_k \geq \cdots \geq \gamma_k \geq 0$$



C^3-UCB Algorithm

- For round $t = 1,2,\dots,T$
  - obtain context: $\{x_{t,a}\}_{a\in E}$
  - From $\mathbb{E}[\boldsymbol{R}(a)] = \theta_*^\top x_a = w_a$, we get an estimate $\widehat{\boldsymbol{\theta}}_{t-1}$ of θ_*.
    (use linear regression, details omitted.)
    With high probability
    $w_{t,a} \in (\widehat{\boldsymbol{\theta}}_{t-1}^\top x_{t,a} - \boldsymbol{\beta}_{t-1}\|x_{t,a}\|_{\boldsymbol{V}_{t-1}^{-1}}, \quad \widehat{\boldsymbol{\theta}}_{t-1}^\top x_{t,a} + \boldsymbol{\beta}_{t-1}\|x_{t,a}\|_{\boldsymbol{V}_{t-1}^{-1}})$
  - The upper confidence bound (UCB) of base arms:
    $$\boldsymbol{U}_t(a) = \min\left\{\widehat{\boldsymbol{\theta}}_{t-1}^\top x_{t,a} + \boldsymbol{\beta}_{t-1}\|x_{t,a}\|_{\boldsymbol{V}_{t-1}^{-1}}, 1\right\}$$
  - use offline oracle to find the best action for UCB: $\boldsymbol{A}_t = \boldsymbol{\mathcal{O}}_S(\boldsymbol{U}_t)$
  - play action $\boldsymbol{A}_t$, observe prefix feedback $\boldsymbol{R}_t(\boldsymbol{a}_k^t), j \le \boldsymbol{O}_t$
  - update observations (details omitted)

Result

- Regret bound in T rounds:

$$Regret = O\left(\frac{d}{p^*}\sqrt{TK}\ln(T)\right)$$

  - d: dimension of latent and feature vectors
  - p^*: minimum probability of triggering all arms in a sequence
  - K: largest length of the sequence

- Regret bound of disjunctive objective in T rounds:

$$Regret = O\left(\frac{d}{1 - f^*}\sqrt{TK}\ln(T)\right)$$

  - $f^* = \max f_t^*$: the maximal expected reward in T rounds.

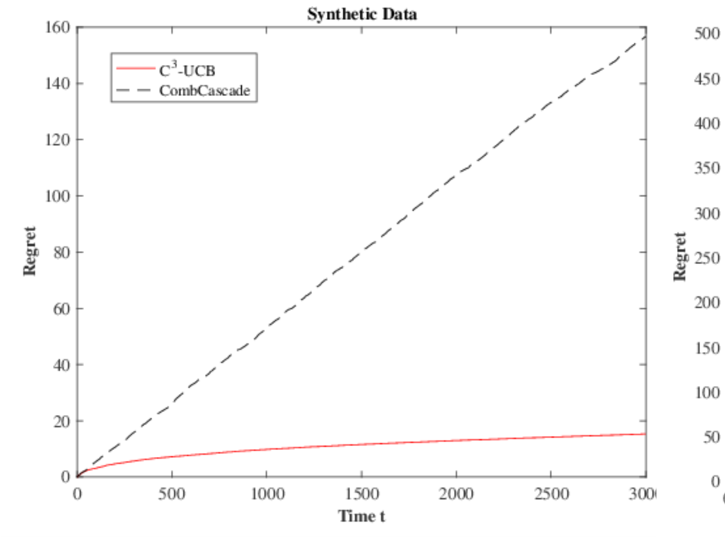
Result

	context	cascading	Position discount	General reward	Regret bound
Combinatorial UCB[1]	No	Yes	No	Yes	$O(m\sqrt{mT\log T})$
Contextual Combinatorial UCB[2]	Yes	No	No	Yes	$O(d\sqrt{T}\log T)$
Comb-Cascade[3]	No	Yes	No	No	$O(\sqrt{\frac{KLT\log T}{f^*}})$
C[3]-UCB (ours)	yes	Yes	Yes	Yes	$O\left(\frac{dB}{p^*}\sqrt{TK}\ln(T)\right)$

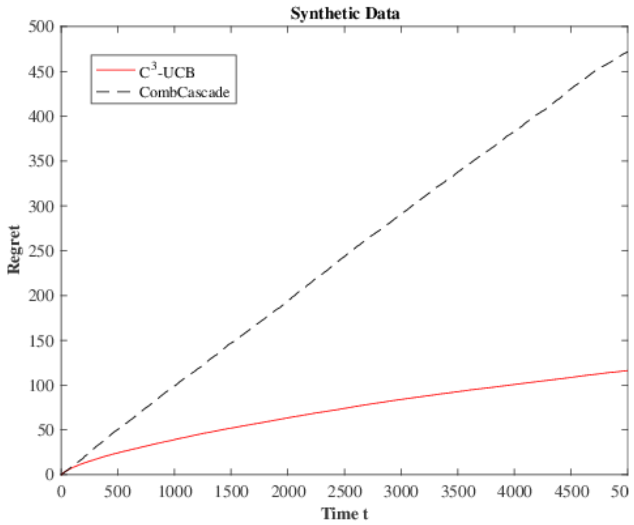
1 Chen et al. 2013
2 Qin et al. 2014
3 Kveton et al. 2015

Experimental Results

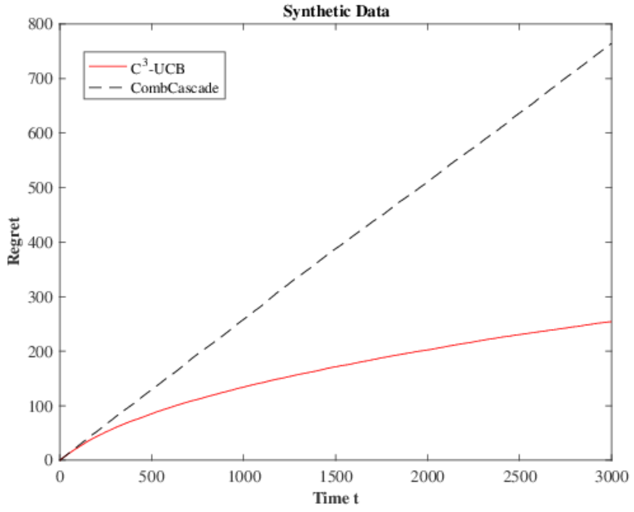
Regret comparisons in Synthetic Data



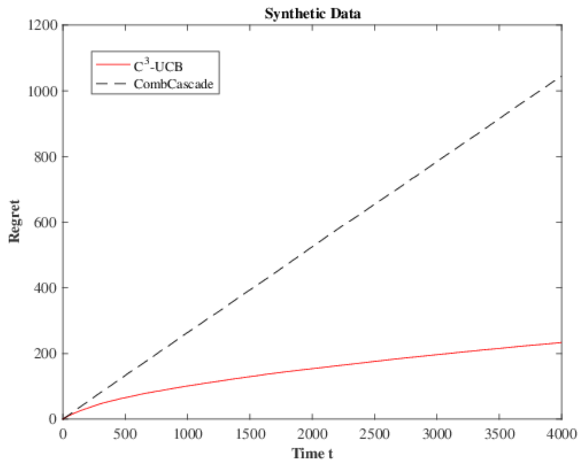
Disjunctive, $\gamma_k = 1$
9.77%



Disjunctive, $\gamma_k = 0.9^{\mathrm{k}-1}$
24.6%



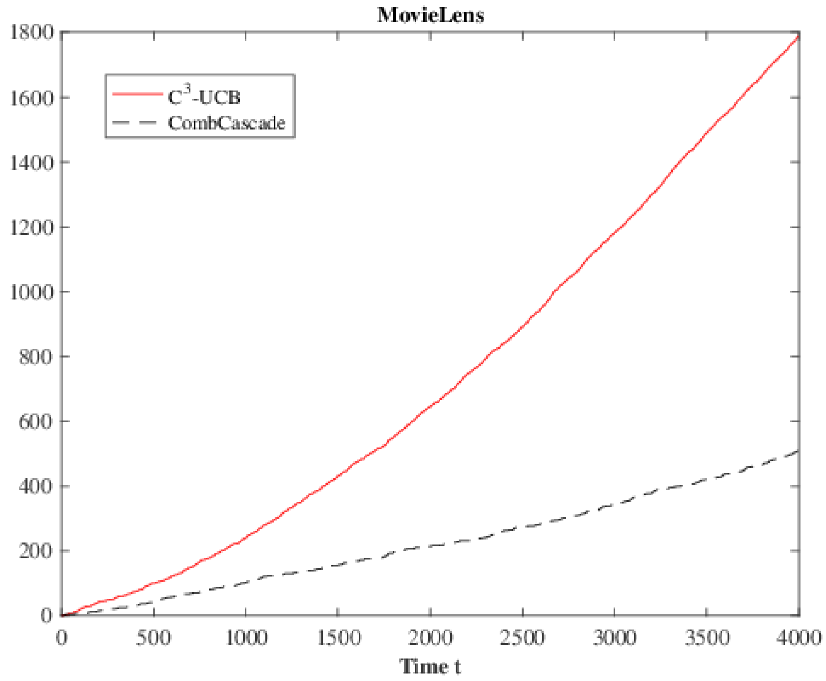
Conjunctive, $\gamma_k = 1$
33.3%



Conjunctive, $\gamma_k = 0.9^{\mathrm{k}-1}$
22.4%

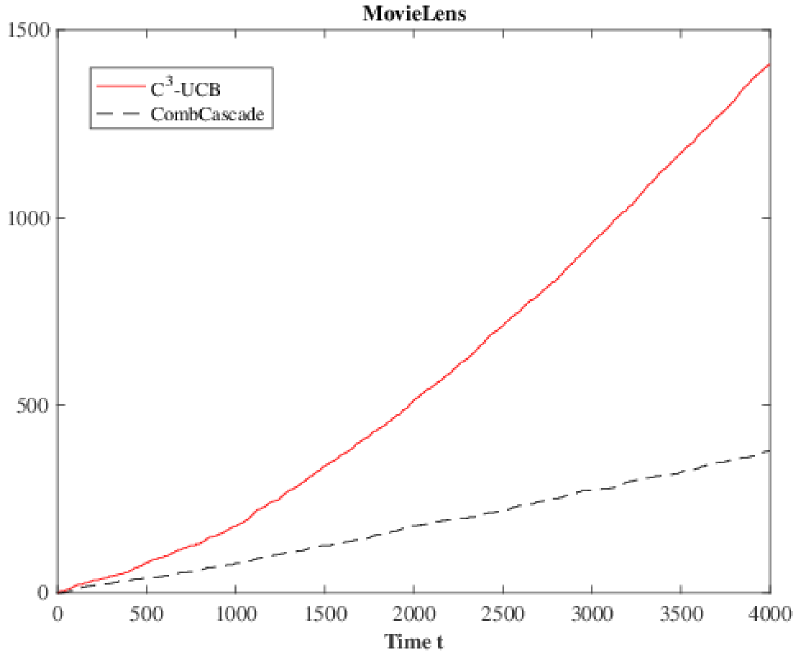
100 items, select 4 items
latent and feature vector dimension = 4

Reward comparisons in MovieLens



$\gamma_k = 1$

3.52 times



$\gamma_k = 0.9^{k-1}$

3.74 times

MovieLens dataset, 200 movies, select 4 items
d= 400 (By SVD decomposition)

Conclusions

- Incorporating contextual information to cascading bandit
- Advancing the research in combinatorial online learning
- Application potential
  - Any sequential list recommendation (search, ads, mobile recommendations)
    - Need online (real-time) feedback
- Future work
  - Theoretical lower bounds
  - Other non-sequential click models

Thank you!

Q & A