Bandits in Recommendation System

Mengyan Zhang

Australian National University; Data61, CSIRO

Currently @ MSRA Social Computing

Outline

- Background: bandits and categories
- Motivations and Applications
- Classical algorithms
- Bandits in recommendation system

Multi-armed Bandits:
Sequential decision making

In each round t ϵ {1, .., N},

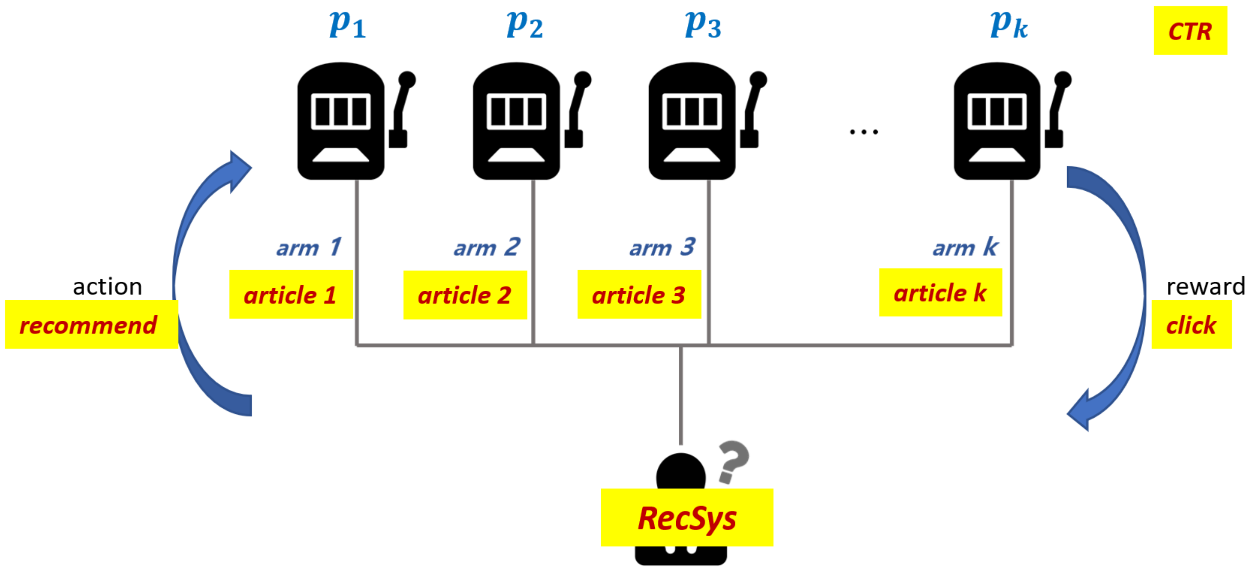
1. an agent selects an arm $A_t = i \; \epsilon \; \{1, .., K\}$ according policy π

2. then receive a reward $X_{A_t, T_{A_t}(t)}$ sampled from unknown reward distribution F_{A_t}

3. update estimations over reward distributions based on historical observations

Multi-armed Bandits
Sequential decision making

in recommendation system



In each round t ϵ {1, .., N},

For a given user

1. an <u>agent</u> selects an <u>arm</u> $A_t = i \; \epsilon \; \{1, .., K\}$ according <u>policy π</u>

   Recommender system — Item (e.g. news) — Recommendation strategy

2. then receive a <u>reward</u> $X_{A_t, T_{A_t}(t)}$ sampled from unknown reward distribution F_{A_t}

   CTR/click; non-click

3. update estimations over reward distributions based on historical observations

Multi-Armed Bandits

Simple regret $r_t = \mu^* - \mu_{A_t}$ where $\mu^* = \max_{k \in \{1,\ldots,K\}} \mu_k$

Best Arm Identification

to recommend best arm(s)
at the end of exploration stage

Fixed Budget:

the number of round for exploration phase is fixed and known

Fixed Confidence:

the confidence level of quality of returned arms is fixed

Regret minimization: maximize the cumulative reward
(i.e. minimize cumulative regret)

Cumulative regret $R_T = \sum_{t=1}^{T} r_t$

Multi-Armed Bandits

Simple regret $r_t = \mu^* - \mu_{A_t}$ where $\mu^* = \max_{k \in \{1,\ldots,K\}} \mu_k$

Best Arm Identification

to recommend best arm(s)
at the end of exploration stage

Fixed Budget:

the number of round for exploration phase is fixed and known

Fixed Confidence:

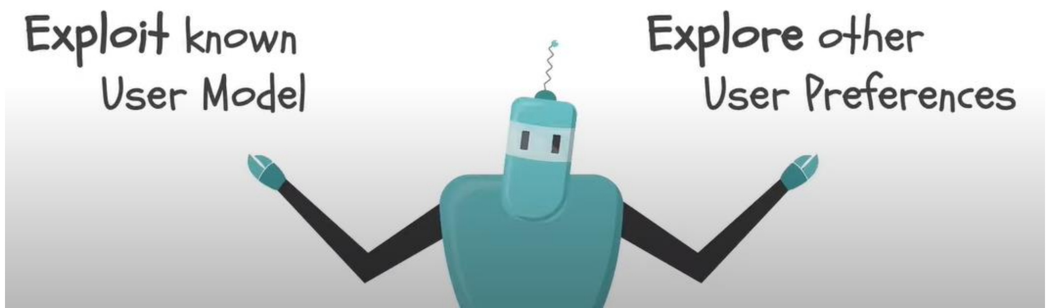
the confidence level of quality of returned arms is fixed

How to allocate samples adaptively?

Regret minimization: maximize the cumulative reward
(i.e. minimize cumulative regret)

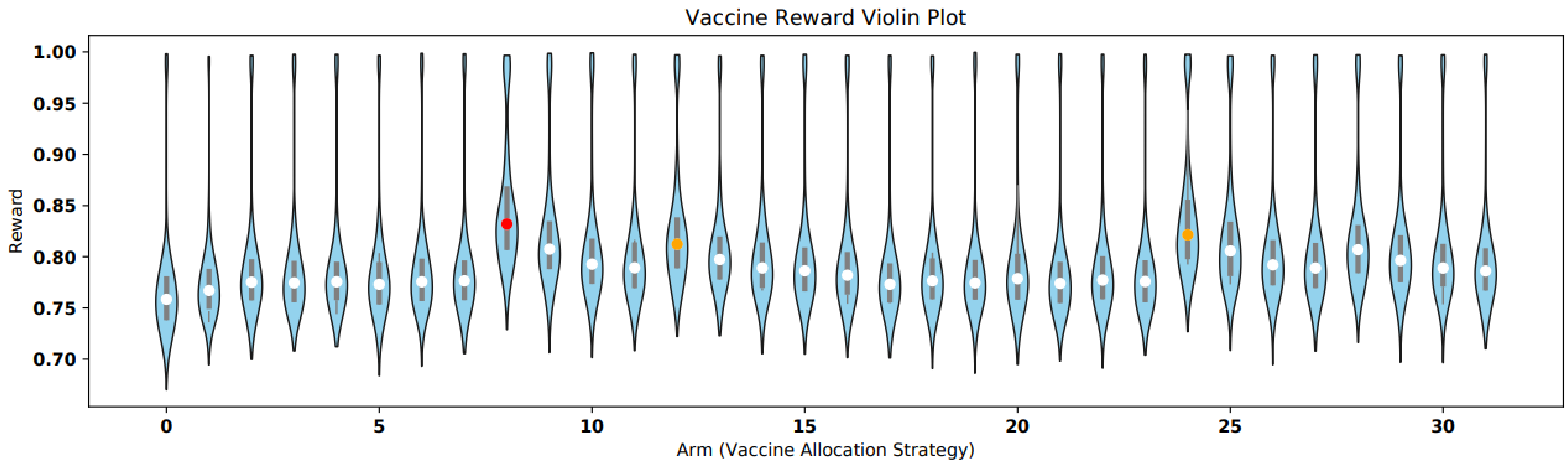
Cumulative regret $R_T = \sum_{t=1}^{T} r_t$

Exploration & Exploitation Balance



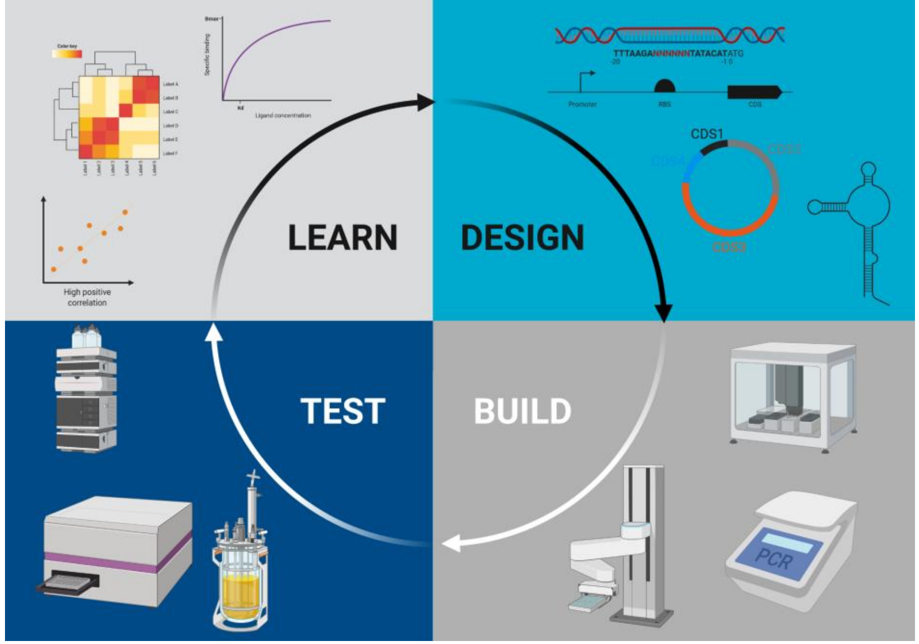
Applications: Vaccine testing

- Identify optimal strategies (highest mean/median reward) for allocation vaccines
- Arm: vaccine allocation strategy
- Reward: proportion of individuals that did not experience symptomatic infection



Zhang, M. ; Ong, C.S.. Quantile Bandits for Best Arms Identification. ICML2021.

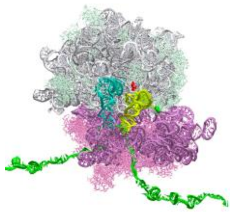
Applications: Biological design



With fixed budget, design Ribosome Binding Site (RBS) sequences

↓

Optimize the protein expression level
Identify the DNA sequences with highest possible protein expression level



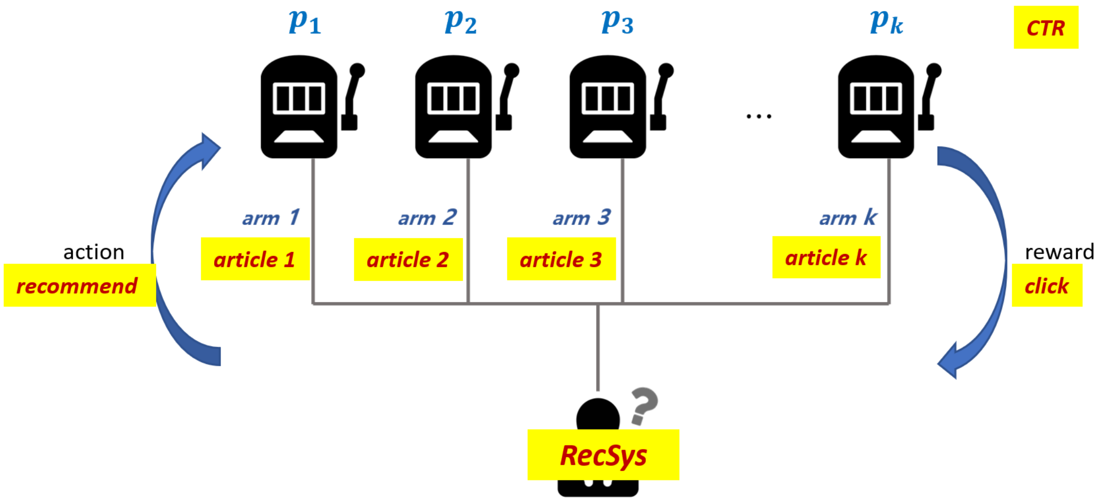
Arm: RBS sequence	Reward: Normalized* Protein Expression Level
TTTAAGAGTTATATATACAT	1.58
TTTAAGAATATGCTATACAT	1.42
TTTAAGACTCGGATATACAT	0.14
TTTAAGAGTTTTTTATACAT	2.88

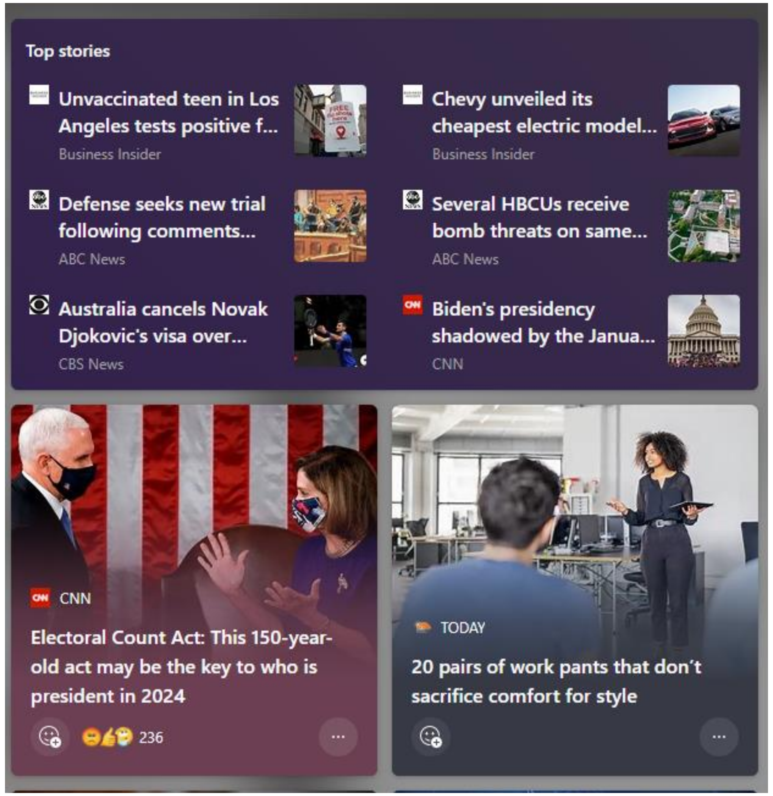
* zero mean and unit variance normalization $z = \frac{x-\mu}{\sigma}$

Zhang, M., and Ong, C. S. Opportunities and Challenges in Designing Genomic Sequences. ICML Workshop on Computational Biology 2021.

Applications: Recommendation System

- BAI: identify the most popular items (with potential high CTR) above some level of confidence using fewest possible samples/ with fixed budget
- Regret minimization: recommend items sequentially to users with the goal of minimize cumulative regret
- Arm: item (e.g. news)
- Reward: click/ preference





Nan Li, Peng Hu, Wenming Xiao, Liangjie Zhang Exploration & Exploitation in Recommender System 2021 Slides

Why *Bandits* in Recommendation System?

- Learn more about the whole distribution
  - reduce model uncertainty in regions of sparse user interaction/feedback
  - Feedback loop debias [1]
  - Might cost user experience in the short term, while the indirect benefit of better model quality arrives at a later time
- Discover new user interests [2]
  - Diversity, novelty, and serendipity, …
  - Good for long-term user experience: e.g. user stickness, conversion of casual users to core users,…
- Interactive methods for diversified recommendation [3]
- Cold start problem
- …….

[1] Jiang R, Chiappa S, Lattimore T, György A, Kohli P. Degenerate Feedback Loops in Recommender Systems. *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society*.
[2] Chen M, Wang Y, Xu C, et al. Values of User Exploration in Recommender Systems. In: *Fifteenth ACM Conference on Recommender Systems*. ACM; 2021
[3] Wu Q, Liu Y, Miao C, Zhao Y, Guan L, Tang H. Recent Advances in Diversified Recommendation. *arXiv:190506589 [cs]*. 2019

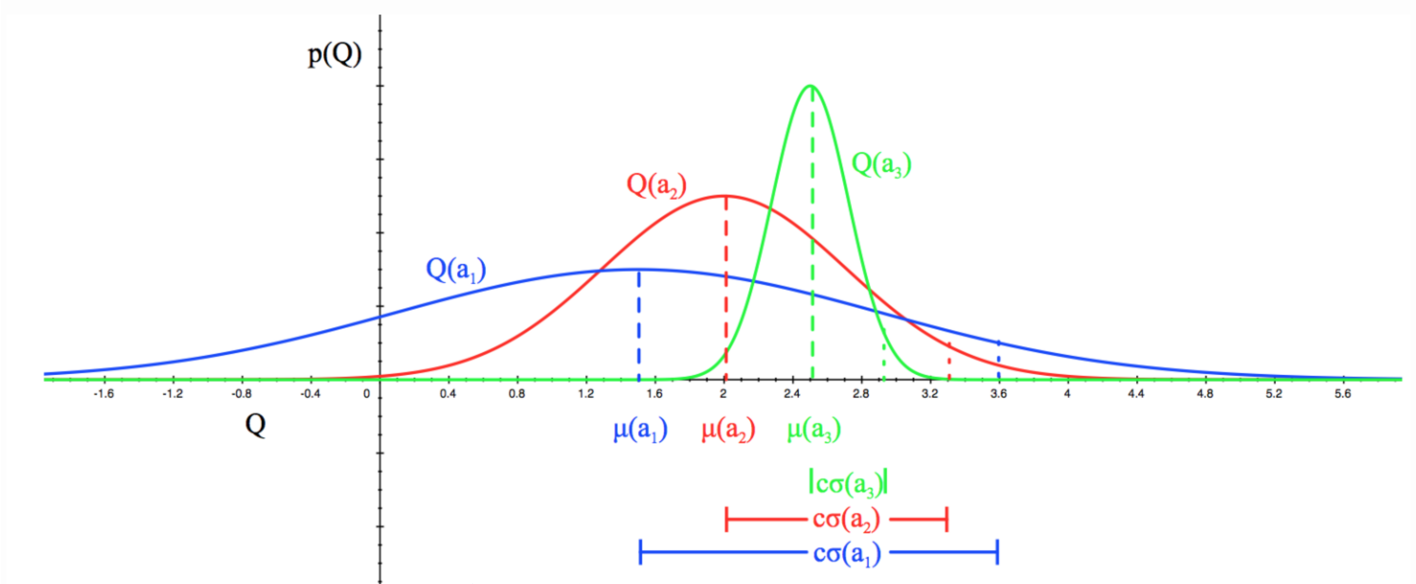
MAB Regret minimization: maximize the cumulative reward (i.e. minimize cumulative regret)

Cumulative regret $R_T = \sum_{t=1}^{T} r_t$

A good policy should have sublinear cumulative regret $\lim_{T \to \infty} \frac{R_T}{T} = 0$

Classical algorithms:

- **Explore-Then-Commit (ETC)**: select each arm a fixed number of times and then exploit by committing to the predicted best arm
- **Epsilon-Greedy**: select a random arm with probability ϵ and select the predicted best arm with probability 1 - ϵ
- **Upper Confidence Bound (UCB)**: select arm with highest UCB score



$$UCB_t(a) = \hat{\mu}_t(a) + \sqrt{\frac{2 \log t}{T_t(a)}}$$

image source: UCL RL course lecture 9's slides)

A Contextual-Bandit Approach to Personalized News Article Recommendation

Lihong Li[†], Wei Chu[†],
[†]Yahoo! Labs
lihong,chuwei@yahoo-inc.com

John Langford[‡]
[‡]Yahoo! Labs
jl@yahoo-inc.com

Robert E. Schapire[+][*]
[+]Dept of Computer Science
Princeton University
schapire@cs.princeton.edu

WWW2010

Contextual Bandits - LinUCB

- MAB with contextual information $\mathbf{x}_{t,a}$
- Assumption: linear reward $\mathbf{E}[r_{t,a}|\mathbf{x}_{t,a}] = \mathbf{x}_{t,a}^\top \boldsymbol{\theta}_a^*.$
- Estimate the coefficient by ridge regression $\hat{\boldsymbol{\theta}}_a = (\mathbf{D}_a^\top \mathbf{D}_a + \mathbf{I}_d)^{-1} \mathbf{D}_a^\top \mathbf{c}_a$
- With probability at least 1 - δ,

$$\left| \mathbf{x}_{t,a}^\top \hat{\boldsymbol{\theta}}_a - \mathbf{E}[r_{t,a}|\mathbf{x}_{t,a}] \right| \leq \alpha \sqrt{\mathbf{x}_{t,a}^\top (\mathbf{D}_a^\top \mathbf{D}_a + \mathbf{I}_d)^{-1} \mathbf{x}_{t,a}}$$

- Policy: at trial t, select arm

$$a_t \stackrel{\text{def}}{=} \arg\max_{a \in \mathcal{A}_t} \left(\mathbf{x}_{t,a}^\top \hat{\boldsymbol{\theta}}_a + \alpha \sqrt{\mathbf{x}_{t,a}^\top \mathbf{A}_a^{-1} \mathbf{x}_{t,a}} \right) \qquad \tilde{O}(\sqrt{T})$$

$$\mathbf{A}_a \stackrel{\text{def}}{=} \mathbf{D}_a^\top \mathbf{D}_a + \mathbf{I}_d$$

Contextual Bandits - LinUCB

Hybrid Linear Models

$$\mathbf{E}[r_{t,a}|\mathbf{x}_{t,a}] \quad = \quad \mathbf{z}_{t,a}^{\top}\boldsymbol{\beta}^* + \mathbf{x}_{t,a}^{\top}\boldsymbol{\theta}_a^*,$$

- MAB with contextual information $\mathbf{x}_{t,a}$
- Assumption: linear reward $\mathbf{E}[r_{t,a}|\mathbf{x}_{t,a}] \quad = \quad \mathbf{x}_{t,a}^{\top}\boldsymbol{\theta}_a^*.$
- Estimate the coefficient by ridge regression $\hat{\boldsymbol{\theta}}_a = (\mathbf{D}_a^{\top}\mathbf{D}_a + \mathbf{I}_d)^{-1}\mathbf{D}_a^{\top}\mathbf{c}_a$
- With probability at least 1 - δ,

$$\left|\mathbf{x}_{t,a}^{\top}\hat{\boldsymbol{\theta}}_a - \mathbf{E}[r_{t,a}|\mathbf{x}_{t,a}]\right| \leq \alpha\sqrt{\mathbf{x}_{t,a}^{\top}(\mathbf{D}_a^{\top}\mathbf{D}_a + \mathbf{I}_d)^{-1}\mathbf{x}_{t,a}}$$

- Policy: at trial t, select arm

$$a_t \stackrel{\text{def}}{=} \arg\max_{a\in\mathcal{A}_t}\left(\mathbf{x}_{t,a}^{\top}\hat{\boldsymbol{\theta}}_a + \alpha\sqrt{\mathbf{x}_{t,a}^{\top}\mathbf{A}_a^{-1}\mathbf{x}_{t,a}}\right)$$

$$\tilde{O}(\sqrt{T})$$

$$\mathbf{A}_a \stackrel{\text{def}}{=} \mathbf{D}_a^{\top}\mathbf{D}_a + \mathbf{I}_d$$

Off-policy evaluation

- Off-policy evaluation: use logged data to evaluate a bandit algorithm π: $\mathcal{H} \times \chi \rightarrow \mathcal{A}$
- Interactive nature of the problem: ideally run algorithm on live data!
- Unbiased simulator

Algorithm 3 Policy_Evaluator.

Inputs: $T > 0$; policy π; stream of events
$h_0 \leftarrow \emptyset$ {An initially empty history}
$R_0 \leftarrow 0$ {An initially zero total payoff}
for $t = 1, 2, 3, \ldots, T$ **do**
  repeat
    Get next event $(\mathbf{x}_1, ..., \mathbf{x}_K, a, r_a)$
  until $\pi(h_{t-1}, (\mathbf{x}_1, ..., \mathbf{x}_K)) = a$
  $h_t \leftarrow \text{CONCATENATE}(h_{t-1}, (\mathbf{x}_1, ..., \mathbf{x}_K, a, r_a))$
  $R_t \leftarrow R_{t-1} + r_a$
end for
Output: R_T/T

Dataset: Yahoo! Today Module

4.7 million events in random bucket

Each user's interaction event:

- Random article chosen to serve the user
- user/article information
- Whether the user clicks on the article at the story position

Figure 1: A snapshot of the "Featured" tab in the Today Module on Yahoo! Front Page. By default, the article at F1 position is highlighted at the story position.

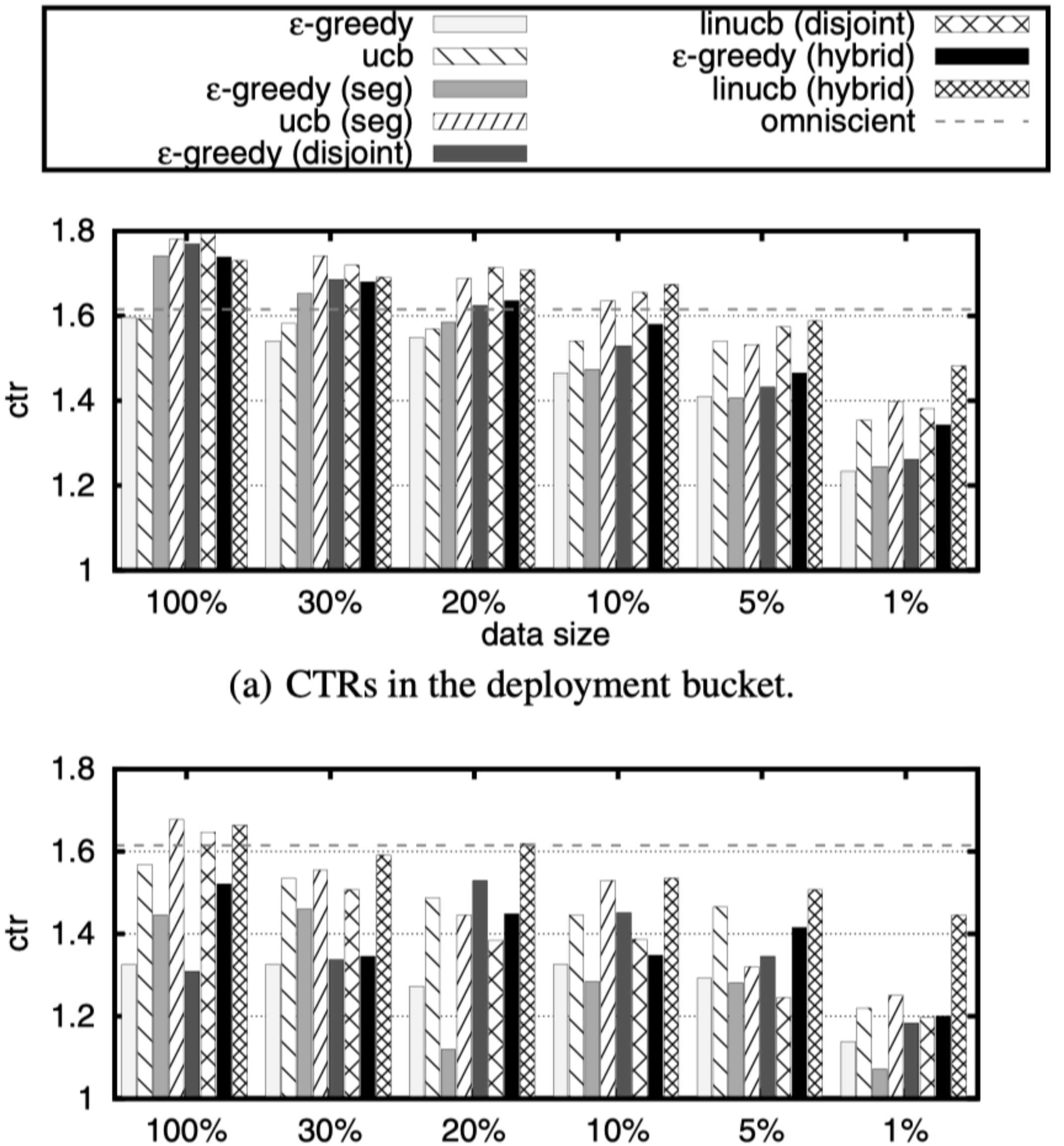
Experiments

- Metric:
  - CTR = $\frac{\#\text{ clicks}}{\#\, recommendations}$;   Relative CTR = $\frac{CTR\ (policy)}{CTR\ (random\ policy)}$
- Randomly split all traffic into two buckets
  - Learning bucket: a small fraction of traffic – various bandits algorithms are run to learn/estimate articles CTRs
  - Deployment bucket: greedily serves users using CTR estimates obtained from the learning bucket.

Experiments

- Omniscient: computes each article's empirical CTR from logged events, and then always chooses the article with highest empirical CTR when it is evaluated using the *same* logged events.
- (seg): all users are partitioned into five groups (a.k.a. user segments), in each of which a separate algorithm was run.

- Observations
  - Features are useful
  - UCB methods outperform epsilon-greedy
  - linucb (hybrid) showed significant benefits when data size was small



(a) CTRs in the deployment bucket.

(b) CTRs in the learning bucket.

Figure 4: CTRs in evaluation data with varying data sizes.

Show Me the Whole World: Towards Entire Item Space Exploration for Interactive Personalized Recommendations

Yu Song
yusonghust@gmail.com
Huazhong University of Science and Technology
Wuhan, China

Jianxun Lian
jianxun.lian@outlook.com
Microsoft Research Asia
Beijing, China

Shuai Sun
540507710@qq.com
Huazhong University of Science and Technology
Wuhan, China

Hong Huang
honghuang@hust.edu.cn
Huazhong University of Science and Technology
Wuhan, China

Yu Li
liyu65@meituan.com
Meituan Group
Beijing, China

Hai Jin
hjin@hust.edu.cn
Huazhong University of Science and Technology
Wuhan, China

Xing Xie
xingx@microsoft.com
Microsoft Research Asia
Beijing, China

WSDM2022

Hierarchical Contextual Bandits

- N items are clustered into k_l subsets based on similarity of item embeddings on level l

$$\theta_u^{(l)} = \left(D^{(l)^T} D^{(l)} + I\right)^{-1} D^{(l)^T} r^l$$

$$n^{(l+1)}(t) = \underset{n \in Ch(n^{(l)}(t))}{\arg\max} \left(\theta_u^{(l)^T} X_n + \alpha \sqrt{X_n^T A^{(l)^{-1}} X_n}\right)$$

- Each node on Path(root -> $n^{(L)}(t)$) receives the same rewards $r_\pi(t)$, then $\{\theta_u^{(0)}, \theta_u^{(1)}, \theta_u^{(2)}, \cdots, \theta_u^{(L)}\}$ are updated

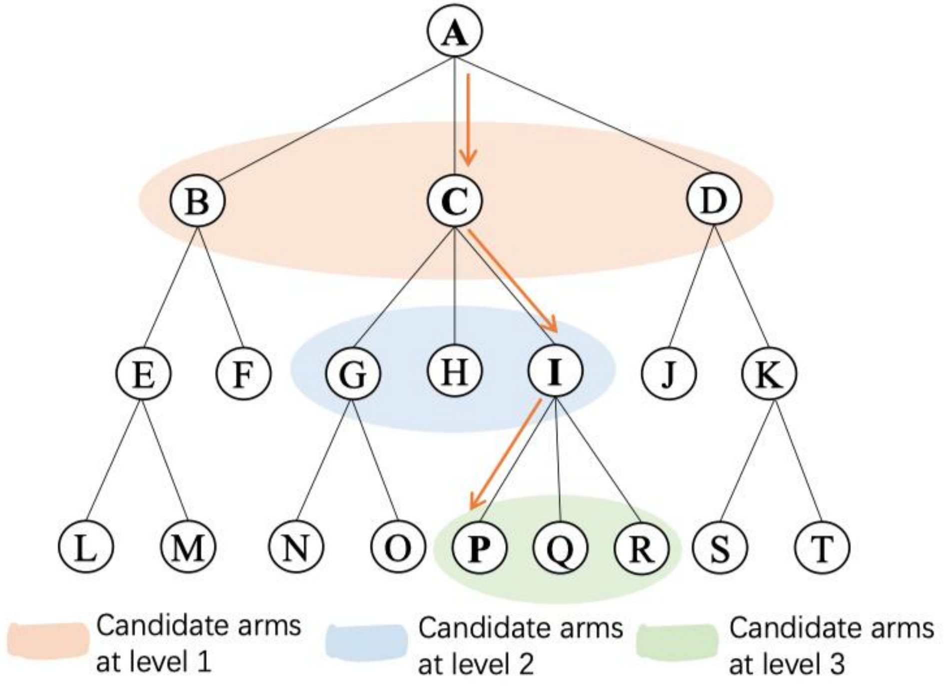


Figure 1: An illustration of HCB. The policy selects a path { A, C, I, P } from root to a certain leaf node.

Potential problems:

- Error propagation: Once the policy makes a bad decision at a certain level, the rest selections are all sub-optimal.
- Users may be interested in more than one child node

Hierarchical Contextual Bandits

- N items are clustered into k_l subsets based on similarity of item embeddings on level l

$$\theta_u^{(l)} = \left(\boldsymbol{D}^{(l)^T}\boldsymbol{D}^{(l)} + \boldsymbol{I}\right)^{-1}\boldsymbol{D}^{(l)^T}\boldsymbol{r}^l$$

$$n^{(l+1)}(t) = \underset{n \in Ch(n^{(l)}(t))}{\arg\max} \left(\theta_u^{(l)^T}\boldsymbol{X}_n + \alpha\sqrt{\boldsymbol{X}_n^T \boldsymbol{A}^{(l)^{-1}}\boldsymbol{X}_n}\right)$$

- Each node on Path(root -> $n^{(L)}(t)$) receives the same rewards $r_\pi(t)$, then $\{\theta_u^{(0)}, \theta_u^{(1)}, \theta_u^{(2)}, \cdots, \theta_u^{(L)}\}$ are updated

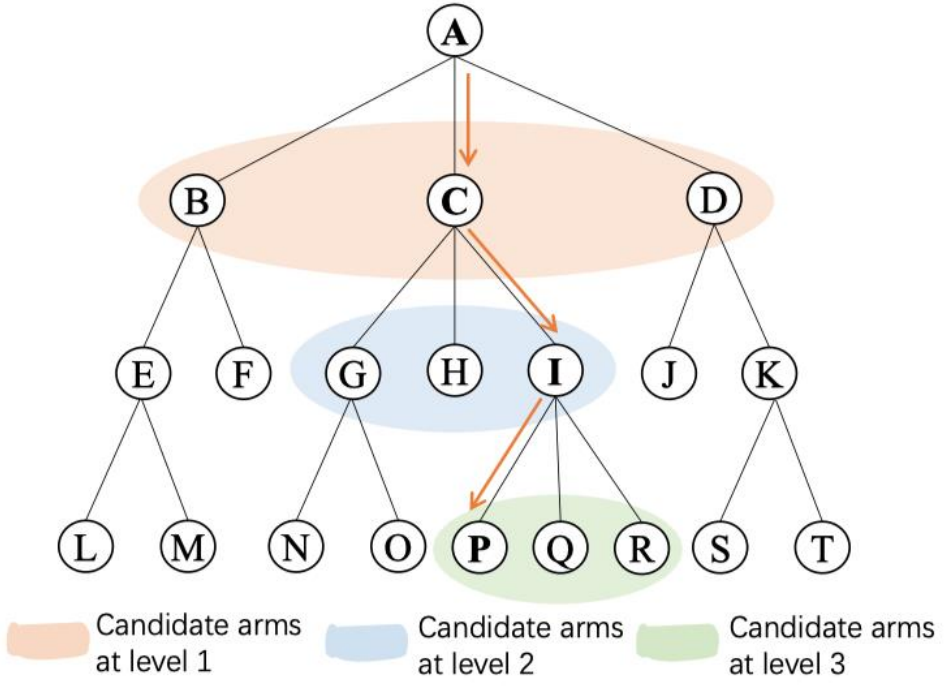


Figure 1: An illustration of HCB. The policy selects a path { A, C, I, P } from root to a certain leaf node.

Progressive Hierarchical Contextual Bandits

- Main idea: the policy continuously expands the (personalized) receptive field from top to bottom according to the feedback obtained from historical exploration.
- Expansion conditions:
  - # selections >= $\lfloor q \log l \rfloor$
  - Average reward > $p \log l$

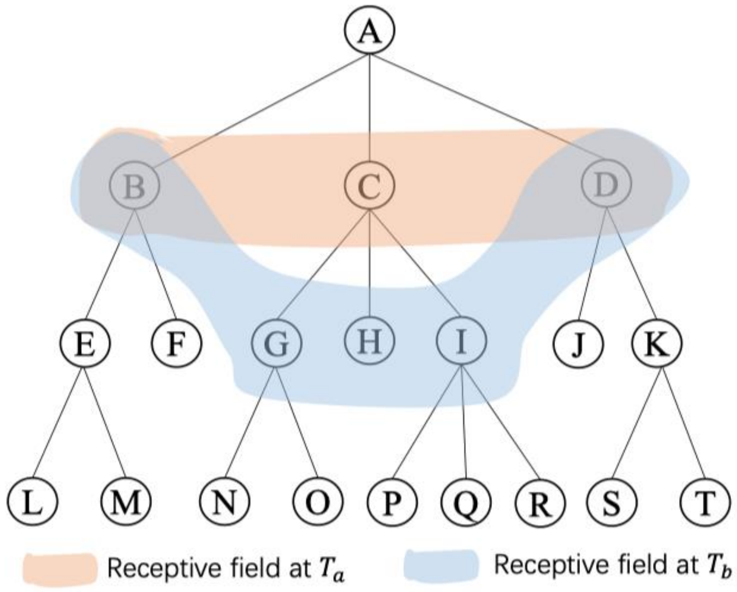


Figure 2: An illustration of pHCB. At round T_a, the receptive field consists of nodes B, C and D; After several trials, at round T_b, node C meets the conditions of expansion, so the receptive field changes to nodes B, D, G, H and I

Off-policy evaluation

- Inverse Propensity Score (IPS) simulator: re-weigh the training samples by the propensity score to learn an unbiased simulator.

$$P_{u,i} = P(o_{u,i} = 1 | \boldsymbol{x}_{u,i}, \phi) = \sigma(\boldsymbol{w}^T \boldsymbol{x}_{u,i} + \boldsymbol{\beta}_i + \boldsymbol{\gamma}_u);$$

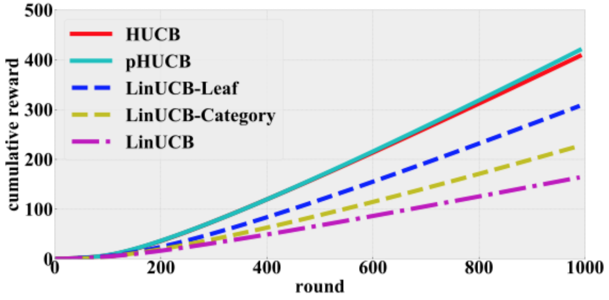
$$\mathcal{L}_{IPS} = \frac{1}{U \cdot I} \sum_{(u,i):o_{u,i}=1} \frac{\delta_{u,i}(Y, \hat{Y})}{P_{u,i}},$$

- Trained on the whole data
- Metric: cumulative rewards
- Score-computing per recommendation: 50
- randomly select 10000 users for testing and 1000 users for validation

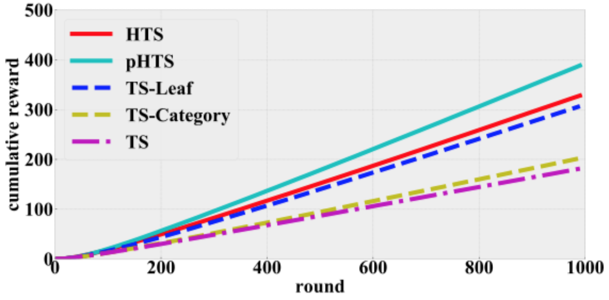
Experiments

Table 2: Overview of Datasets

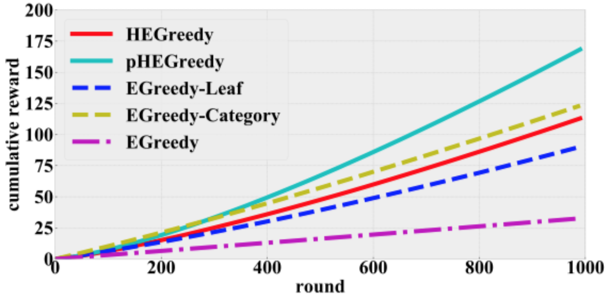
Dataset	#users	#items	# categories	# interactions
MIND	1,000,000	161,013	285	24,155,470
Taobao	987,994	4,162,024	9,439	100,150,807



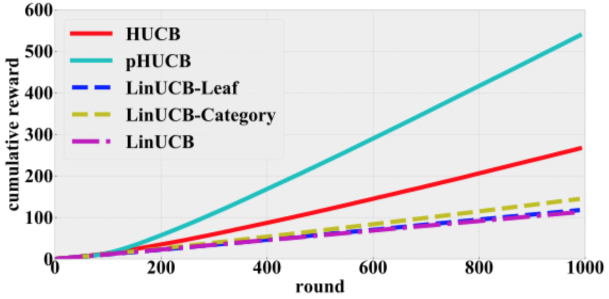
(a) MIND, LinUCB



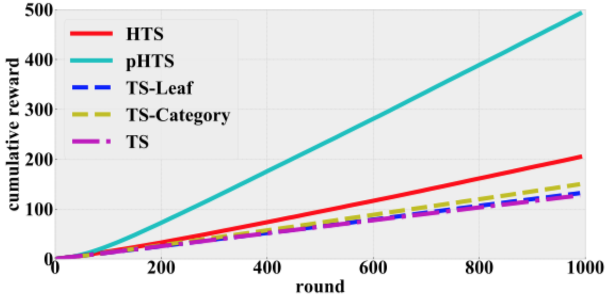
(b) MIND, Thompson Sampling



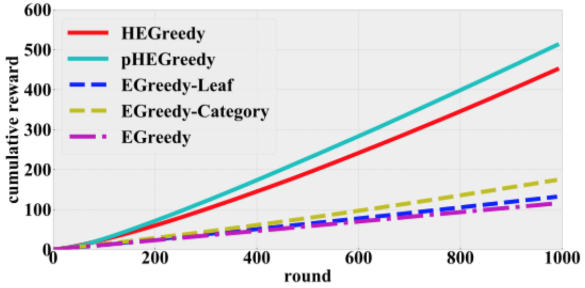
(c) MIND, ϵ-greedy



(d) Taobao, LinUCB



(e) Taobao, Thompson Sampling



(f) Taobao, ϵ-greedy

Figure 3: Cumulative rewards of our algorithms and variants based on LinUCB, Thompson Sampling and ϵ-greedy, on the MIND dataset and Taobao dataset, respectively.

Round: one pass of all users receiving one recommended item

Experiment: Alleviate Closed-Loop Effects

- **Pre-train** exploitation models (Linear, GRU, Transformer) by the historical logs of existing users
- **Deploy**: recommend 200 items to each user and collect feedback according to exploitation models and bandits models
- **Evaluate** the quality of impression logs produced by the deployed models: train matrix factorization model on collected data, evaluate the model performance on 200 users with diversified interests

Table 4: Test LogLoss and AUC of different algorithms

Dataset	**MIND**		**TaoBao**	
Method	**LogLoss**	**AUC**	**LogLoss**	**AUC**
Linear	$1.679_{\pm 0.005}$	$0.703_{\pm 0.005}$	$0.693_{\pm 0.001}$	$0.530_{\pm 0.001}$
GRU	$1.759_{\pm 0.004}$	$0.686_{\pm 0.003}$	$0.688_{\pm 0.001}$	$0.535_{\pm 0.002}$
Transformer	$1.377_{\pm 0.008}$	$0.695_{\pm 0.006}$	$0.683_{\pm 0.001}$	$0.546_{\pm 0.001}$
HUCB	$\mathbf{0.681}_{\pm 0.004}$	$\mathbf{0.720}_{\pm 0.003}$	$\mathbf{0.660}_{\pm 0.001}$	$\mathbf{0.649}_{\pm 0.002}$
pHUCB	$\mathbf{0.680}_{\pm 0.005}$	$\mathbf{0.723}_{\pm 0.002}$	$\mathbf{0.661}_{\pm 0.002}$	$\mathbf{0.647}_{\pm 0.003}$

Can we go deeper? – Deep CB

- Bayesian
  - BNN-based
  - Bayes-layers [5]
  - Dropout-based [1]
- Bootstrapped
  - Bootstrapped sampling [5]
  - GIRO: with pseudo rewards [3]
- Frequentist
  - NeuralUCB-related [2]
  - SAU [4]
- RL based
  - Policy gradient with entropy reg [6]
  - ...

[1] Guo D, Ktena SI, Myana PK, et al. Deep Bayesian Bandits: Exploring in Online Personalized Recommendations. In: *Fourteenth ACM Conference on Recommender Systems*. ACM; 2020

[2] Zhou, Dongruo, Lihong Li, and Quanquan Gu. "Neural Contextual Bandits with UCB-Based Exploration." *ArXiv:1911.04462 [Cs, Stat]*, July 2, 2020.

[3] Kveton, Branislav, Csaba Szepesvari, Sharan Vaswani, Zheng Wen, Mohammad Ghavamzadeh, and Tor Lattimore. "Garbage In, Reward Out: Bootstrapping Exploration in Multi-Armed Bandits." 2019.

[4] Zhu, Rong, and Mattia Rigotti. "Deep Bandits Show-Off: Simple and Efficient Exploration with Deep Networks," 2021, 25.

[5] Riquelme, Carlos, George Tucker, and Jasper Snoek. "DEEP BAYESIAN BANDITS SHOWDOWN," 2018, 27.

[6] Chen M, Wang Y, Xu C, et al. Values of User Exploration in Recommender Systems. In: *Fifteenth ACM Conference on Recommender Systems*. ACM; 2021

Can we go deeper? – Deep CB

- Bayesian
  - BNN-based
  - Bayes-layers [5]
  - Dropout-based [1]
- Bootstrapped
  - Bootstrapped sampling [5]
  - GIRO: with pseudo rewards [3]
- Frequentist
  - NeuralUCB-related [2]
  - SAU [4]
- RL based
  - Policy gradient with entropy reg [6]
  - …

neural network with arbitrary depth and non-linearities, with dropout applied before every weight layer, is mathematically equivalent to an approximation to the probabilistic deep Gaussian process [7]

$$U_{t,a} = f(\mathbf{x}_{t,a};\boldsymbol{\theta}_{t-1}) + \gamma_{t-1}\sqrt{\mathbf{g}(\mathbf{x}_{t,a};\boldsymbol{\theta}_{t-1})^\top \mathbf{Z}_{t-1}^{-1}\mathbf{g}(\mathbf{x}_{t,a};\boldsymbol{\theta}_{t-1})/m}$$

$$\mathbf{g}(\mathbf{x};\boldsymbol{\theta}) = \nabla_{\boldsymbol{\theta}} f(\mathbf{x};\boldsymbol{\theta}) \in \mathbb{R}^p$$

[1] Guo D, Ktena SI, Myana PK, et al. Deep Bayesian Bandits: Exploring in Online Personalized Recommendations. In: *Fourteenth ACM Conference on Recommender Systems*. ACM; 2020
[2] Zhou, Dongruo, Lihong Li, and Quanquan Gu. "Neural Contextual Bandits with UCB-Based Exploration." *ArXiv:1911.04462 [Cs, Stat]*, July 2, 2020.
[3] Kveton, Branislav, Csaba Szepesvari, Sharan Vaswani, Zheng Wen, Mohammad Ghavamzadeh, and Tor Lattimore. "Garbage In, Reward Out: Bootstrapping Exploration in Multi-Armed Bandits." 2019.
[4] Zhu, Rong, and Mattia Rigotti. "Deep Bandits Show-Off: Simple and Efficient Exploration with Deep Networks," 2021, 25.
[5] Riquelme, Carlos, George Tucker, and Jasper Snoek. "DEEP BAYESIAN BANDITS SHOWDOWN," 2018, 27.
[6] Chen M, Wang Y, Xu C, et al. Values of User Exploration in Recommender Systems. In: *Fifteenth ACM Conference on Recommender Systems*. ACM; 2021
[7] Gal, Yarin, and Zoubin Ghahramani. "Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning." *International Conference on Machine Learning*, 2016.

Can we go deeper? – Deep CB

- Bayesian
  - BNN-based
  - Bayes-layers [5]
  - Dropout-based [1]
- Bootstrapped
  - Bootstrapped sampling [5]
  - GIRO: with pseudo rewards [3]
- Frequentist
  - NeuralUCB-related [2]
  - SAU [4]
- RL based
  - Policy gradient with entropy reg [6]
  - …

Still an active and open research area!

Some interesting questions:

- How can we generate uncertainty (confidence interval) for DNN?
- How do we understand the deviation between predictions and true rewards wrt the uncertainty (i.e. concentration inequality)?
- How can we evaluate uncertainty empirically?
- Apply on recommendation system?

[1] Guo D, Ktena SI, Myana PK, et al. Deep Bayesian Bandits: Exploring in Online Personalized Recommendations. In: *Fourteenth ACM Conference on Recommender Systems*. ACM; 2020
[2] Zhou, Dongruo, Lihong Li, and Quanquan Gu. "Neural Contextual Bandits with UCB-Based Exploration." *ArXiv:1911.04462 [Cs, Stat]*, July 2, 2020.
[3] Kveton, Branislav, Csaba Szepesvari, Sharan Vaswani, Zheng Wen, Mohammad Ghavamzadeh, and Tor Lattimore. "Garbage In, Reward Out: Bootstrapping Exploration in Multi-Armed Bandits." 2019.
[4] Zhu, Rong, and Mattia Rigotti. "Deep Bandits Show-Off: Simple and Efficient Exploration with Deep Networks," 2021, 25.
[5] Riquelme, Carlos, George Tucker, and Jasper Snoek. "DEEP BAYESIAN BANDITS SHOWDOWN," 2018, 27.
[6] Chen M, Wang Y, Xu C, et al. Values of User Exploration in Recommender Systems. In: *Fifteenth ACM Conference on Recommender Systems*. ACM; 2021

Context Uncertainty in Contextual Bandits with Applications to Recommender Systems

Hao Wang[1], Yifei Ma[2], Hao Ding[2], Yuyang Wang[2]

[1]Department of Computer Science, Rutgers University [2]AWS AI Lab
hw488@cs.rutgers.edu, {yifeim,haodin,yuyawang}@amazon.com

AAAI2022

(the next few slides are taken from the AAAI presentation)

Chicken-and-Egg Problem

Recommendations rely on learned latent representations

Need

Effective Exploration

Good Latent Representations

Need

Exploration happens in the latent space

Reward Uncertainty versus Context Uncertainty

Reward Uncertainty in Typical Contextual Bandits: $r_k = \boxed{\boldsymbol{x}_k^T}\boldsymbol{\theta} + \boxed{\epsilon}$

Deterministic context: problematic if x_k is latent

Reward uncertainty: independent of $\boldsymbol{x}_k$

Context Uncertainty in Our REN: $r_k = \boxed{\boldsymbol{x}_k^T}\boldsymbol{\theta} + \epsilon$

Probabilistic context:

$\boldsymbol{x}_k \sim N(\boldsymbol{\mu}_k, \boldsymbol{\sigma}_k^2)$

$\boldsymbol{\sigma}_k^2$ represents **context uncertainty**

Relevance + Diversity: Not Good Enough

For one user at time t, the score for item k is:

$$p_{k,t} = \underbrace{\mathbf{x}_k^\top \boldsymbol{\theta}_t}_{\text{Relevance Term}} + \underbrace{\lambda_d \sqrt{\mathbf{x}_k^\top (\mathbf{I}_d + \mathbf{X}_t^\top \mathbf{X}_t)^{-1} \mathbf{x}_k}}_{\text{Diversity Term}}$$

To capture uncertainty, consider both **mean** and **variance** of item representation $\boldsymbol{x_k}$

$$\boldsymbol{x_k} \Rightarrow (\boldsymbol{\mu_k}, \boldsymbol{\sigma_k^2})$$

$$\boldsymbol{X_t} = [\boldsymbol{x_{k_\tau}}]_{\boldsymbol{\tau=1}}^{\boldsymbol{t-1}} \Rightarrow \boldsymbol{D_t} = [\boldsymbol{\mu_{k_\tau}}]_{\boldsymbol{\tau=1}}^{\boldsymbol{t-1}}$$

Determinantal Point Processes for Diversity and Exploration

- Diversity is achieved by picking a subset of items to cover the maximum volume spanned by the items, measured by the log-determinant of the corresponding kernel matrix,

$$\ker(\mathbf{X}_t) = \log\det(\mathbf{I}_K + \mathbf{X}_t\mathbf{X}_t^\top)$$

- penalizes colinearity, which is an indicator that the topics of one item are already covered by the other topics in the full set

$$\begin{aligned}
&\operatorname{argmax}_k \; \log\det(\mathbf{I}_d + \mathbf{X}_t^\top\mathbf{X}_t + \mathbf{x}_k\mathbf{x}_k^\top) && (1)\\
&\qquad - \log\det(\mathbf{I}_d + \mathbf{X}_t^\top\mathbf{X}_t) \\
&= \operatorname{argmax}_k \; \log(1 + \mathbf{x}_k^\top(\mathbf{I}_d + \mathbf{X}_t^\top\mathbf{X}_t)^{-1}\mathbf{x}_k) && (2)\\
&= \operatorname{argmax}_k \; \sqrt{\mathbf{x}_k^\top(\mathbf{I}_d + \mathbf{X}_t^\top\mathbf{X}_t)^{-1}\mathbf{x}_k}. && (3)
\end{aligned}$$

same form as the confidence interval in LinUCB!

Relevance + Diversity + Uncertainty

For one user at time t, the score for item k is:

Uncertainty score for item k:
how uncertain about item k's representation

$$p_{k,t} = \boldsymbol{\mu}_k^\top \boldsymbol{\theta}_t + \lambda_d \sqrt{\boldsymbol{\mu}_k^\top (\mathbf{I}_d + \mathbf{D}_t^\top \mathbf{D}_t)^{-1} \boldsymbol{\mu}_k} + \boxed{\lambda_u \|\boldsymbol{\sigma}_k\|_\infty}$$

At the beginning, item k's representation will have high uncertainty, i.e., large $||\boldsymbol{\sigma}_{\boldsymbol{k}}||_\infty$

System will tend to recommend Item k more frequently

$||\boldsymbol{\sigma}_{\boldsymbol{k}}||_\infty$ gets smaller as we see more data

$$\mathbf{diag}(\boldsymbol{\sigma}_k) = 1/\sqrt{n_k}\ \mathbf{I}_d$$

Experiments

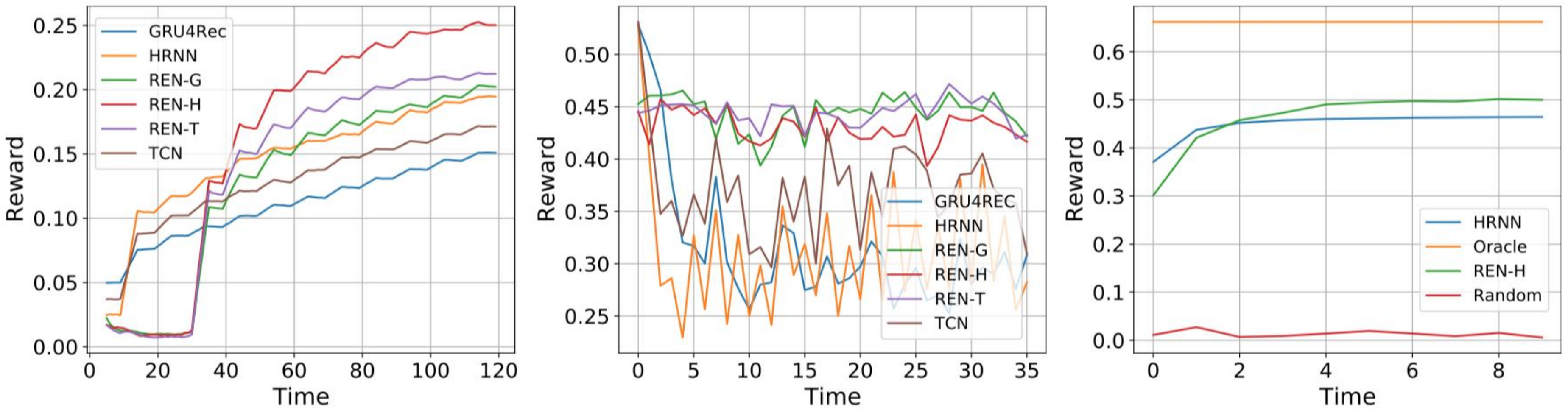


Figure 3: Rewards (precision@10, MRR, and recall@100, respectively) over time on *MovieLens-1M* (**left**), *Trivago* (**middle**), and *Netflix* (**right**). One time step represents 10 recommendations to a user, one hour of data, and 100 recommendations to a user for *MovieLens-1M*, *Trivago*, and *Netflix*, respectively.

three REN variants in the experiments: REN-G, REN-T, and REN-H, which use GRU4Rec, TCN, and HRNN as base models, respectively.

Ablation study

$$p_{k,t} = \boldsymbol{\mu}_k^\top \boldsymbol{\theta}_t + \lambda_d \sqrt{\boldsymbol{\mu}_k^\top (\mathbf{I}_d + \mathbf{D}_t^\top \mathbf{D}_t)^{-1} \boldsymbol{\mu}_k} + \lambda_u \|\boldsymbol{\sigma}_k\|_\infty$$

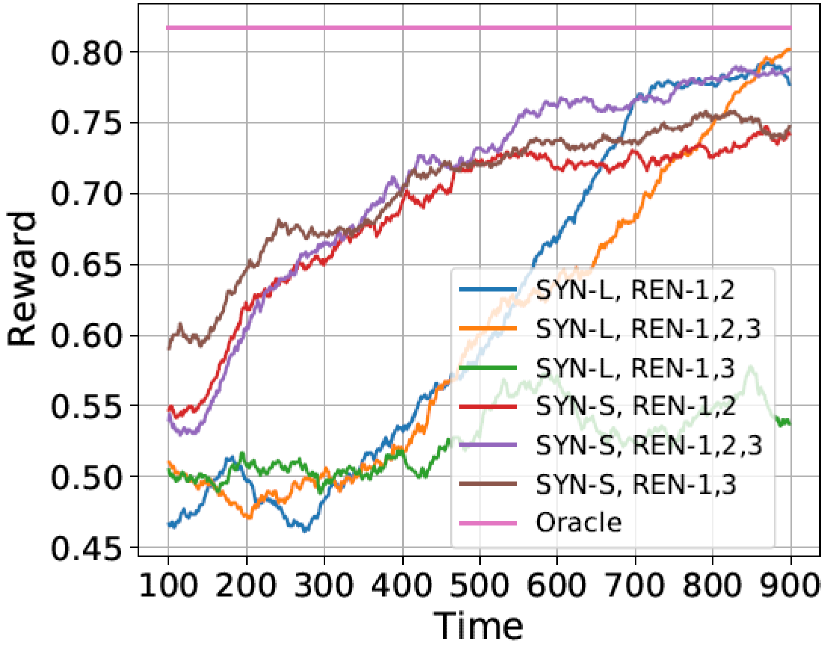


Figure 2: Ablation study on different terms of REN. 'REN-1,2,3' refers to the full 'REN-G' model.

SYN-S: synthetic large dataset (28 items)
SYN-L: synthetic large dataset (28*50 items)

Conclusion

- Background: What is bandits and categories
  - Best Arm Identification
  - Regret minimisation
- Motivations and Applications
  - Feedback loop debias
  - Discover new user interests
  - diversified recommendation
  - Cold start problem
- Classical algorithms
  - Explore-Then-Commit (ETC)
  - Epsilon-Greedy
  - Upper Confidence Bound (UCB)
- Bandits in recommendation system
  - Contextual bandits: LinUCB
  - What if arm space is large: HCB
  - Can we go deeper?
  - Context uncertainty: REN