
The Use of Bandit Algorithms in Intelligent Interactive Recommender Systems

Intern: Qing Wang
6/6/2018

Outline

- Introduction
- Research Problems
 - Online Context-aware Recommendation with Time-varying Multi-armed Bandit
 - Dynamical Context Drift Model
 - Online Context-based recommendation Using Hierarchical Multi-armed Bandit
 - Hierarchical Multi-armed Bandit Model
 - Online Interactive Collaborative Filtering Using Multi-armed Bandit with Dependent Arms
 - Interactive Collaborative Topic Regression Model

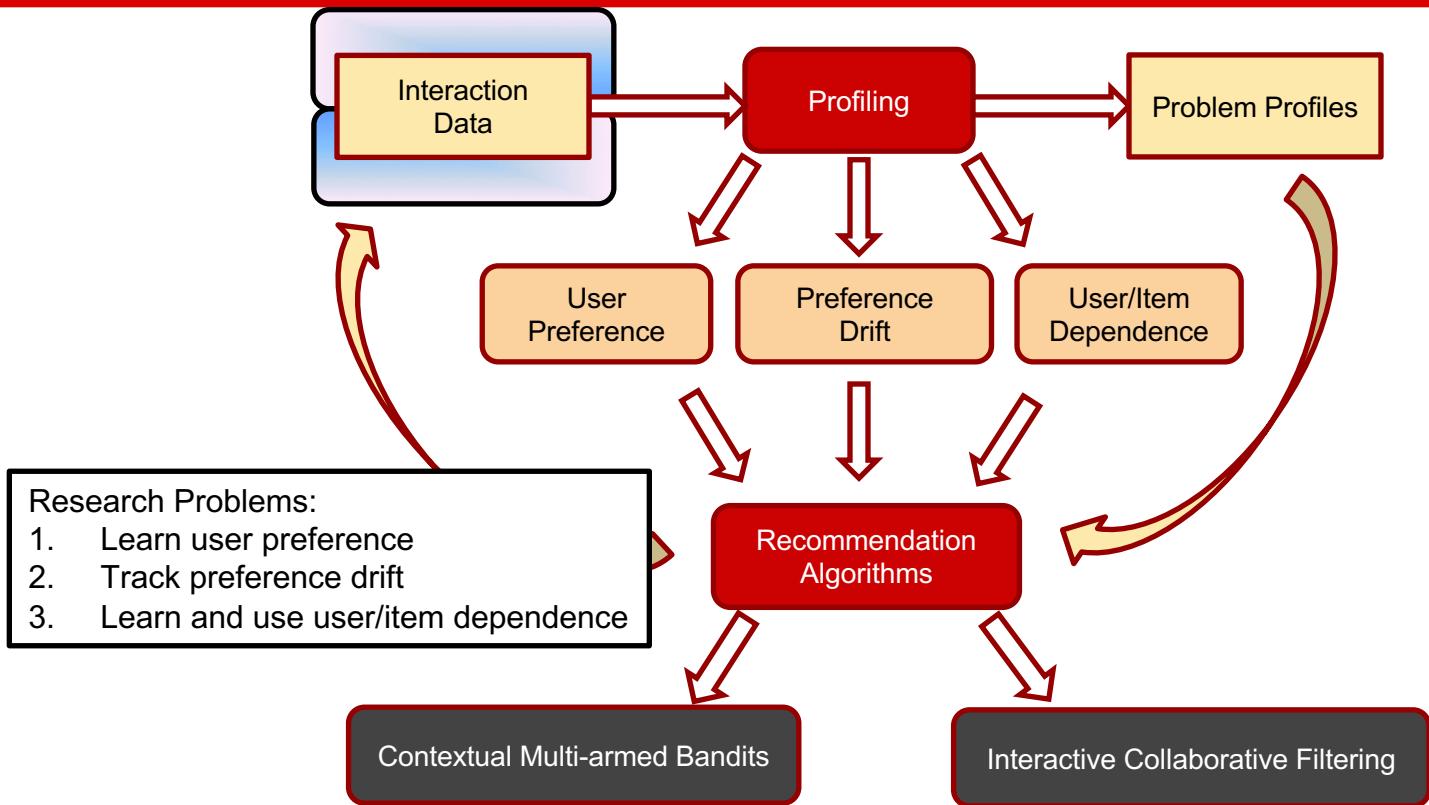
Introduction

Interactive recommender systems:

- 1) Promptly feed users the interesting items (e.g., news, movies);
- 2) Adaptively optimize the underlying model using the up-to-date feedback ( );
- 3) Ultimate Goal: continuously maximize user satisfaction in a long run.



A General Process of Interactive Recommendation



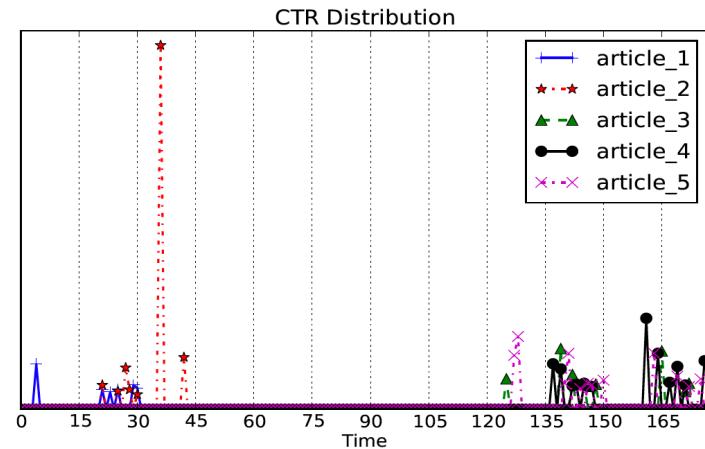
Outline

- Introduction
- Research Problems
 - Online Context-aware Recommendation with Time-varying Multi-armed Bandit
 - Dynamical Context Drift Model
 - Online Context-based recommendation Using Hierarchical Multi-armed Bandit
 - Hierarchical Multi-armed Bandit Model
 - Online Interactive Collaborative Filtering Using Multi-armed Bandit with Dependent Arms
 - Interactive Collaborative Topic Regression Model

Online Context-aware Recommendation with Time-varying Multi-armed Bandit

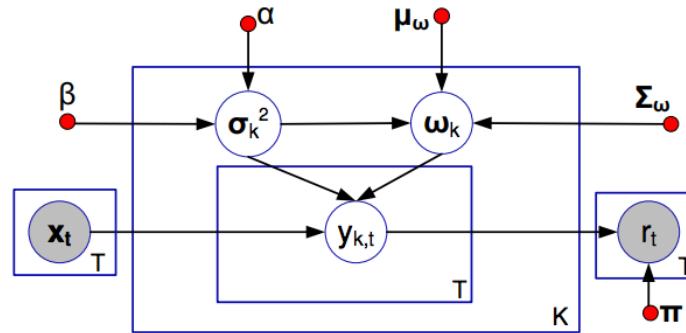
Example:

Given the same contextual information for each article, the average CTR distribution of five news articles from Yahoo! news repository is displayed. The CTR is aggregated by every hour.



Challenge: How do we promptly capture both of the varying popularity of item content and the evolving customer preferences over time, and further utilize them for recommendation improvement?

Contextual Multi-armed Bandit Algorithm



(a) Multi-armed bandit problem.

The reward $r_{k,t}$ is typically modeled as a linear combination of the feature vector x_t with coefficient vector w_k , given at time $t = [1, \dots, T]$ as follows:

$$r_{k,t} \sim N(x_t^T w_k, \sigma_k^2)$$

The optimal policy π^* is defined as the one with maximum accumulated expected reward after T iterations:

$$\pi^* = \operatorname{argmax}_{\pi} \sum_{t=1}^T E_{w_{\pi(x_t)}}(x_t^T w_{\pi(x_t)} | t)$$

Dynamic Context Drift Modeling

The aforementioned model is based on the assumption that coefficient vector w_k is unknown but fixed.

$$y_{k,t} \sim \mathcal{N}(\mathbf{x}_t^\top (\mathbf{c}_{w_k} + \theta_k \odot \eta_{k,t}), \sigma_k^2)$$

The drift component
 $\delta_{w_{k,t}} = \theta_k \odot \eta_{k,t}$

$$\mathbf{w}_{k,t} = \mathbf{c}_{w_k} + \delta_{w_{k,t}},$$

The stationary component

$$\mathbf{c}_{w_k} \sim \mathcal{N}(\mu_c, \sigma_k^2 \Sigma_c)$$

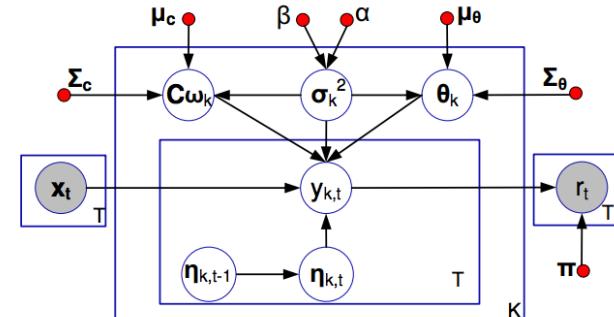
a standard Gaussian random walk $\eta_{k,t} \in \mathbb{R}^d$

$$\eta_{k,t} \sim \mathcal{N}(\eta_{k,t-1}, \mathcal{I}_d)$$

a scale variable θ_k

$$\theta_k \sim \mathcal{N}(\mu_\theta, \sigma_k^2 \Sigma_\theta)$$

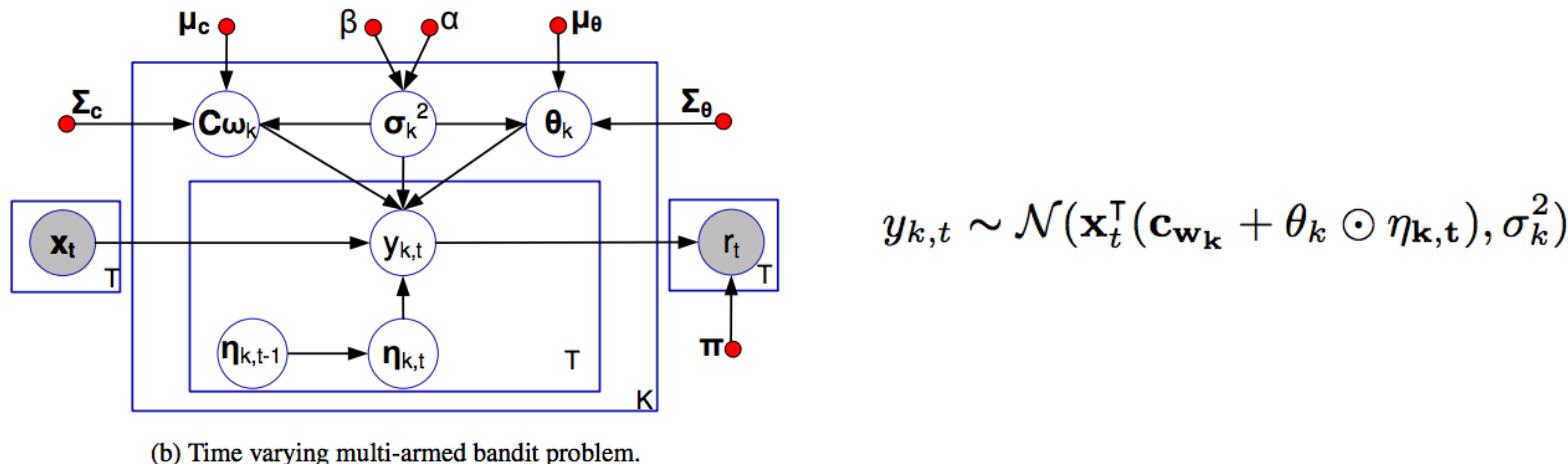
a $d \times d$ dimension identity matrix



(b) Time varying multi-armed bandit problem.

Dynamic Context Drift Modeling

The reward $y_{k,t}$ is modeled to be drawn from the following Gaussian distribution.



(b) Time varying multi-armed bandit problem.

Experiments

- Context Change Tracking, CTR (Click-Through Rate) Optimization
 - a) Dataset: KDD Cup 2012 Online Advertising, Yahoo! Today News.
 - b) Evaluation Method: simulation and replayer [1].

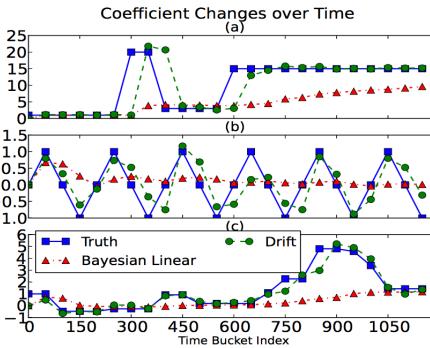


Figure 3: A segment of data originated from the whole data set is provided. The reward is simulated by choosing one dimension of the coefficient vector, which is assumed to vary over time in three different ways. Each time bucket contains 100 time units.

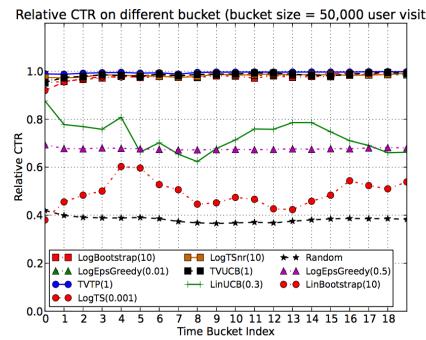


Figure 4: The CTR of KDD CUP 2012 online ads data is given for each time bucket. LogBootstrap, LogTS, LogTsnr, and LogEpsGreedy are bandit algorithms with logistic regression model. LinUCB, LinBootstrap, TVTP, and TVUCB are based on linear regression model.

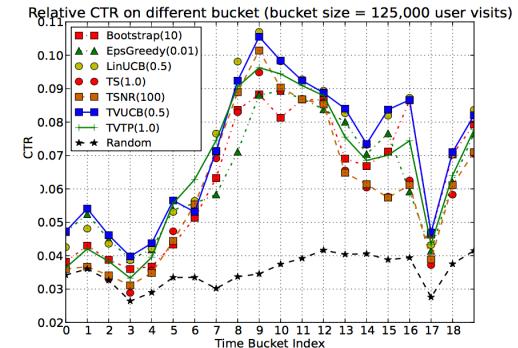


Figure 5: The CTR of Yahoo! News data is given for each time bucket. Those baseline algorithms are configured with their best parameters settings.

Conclusion

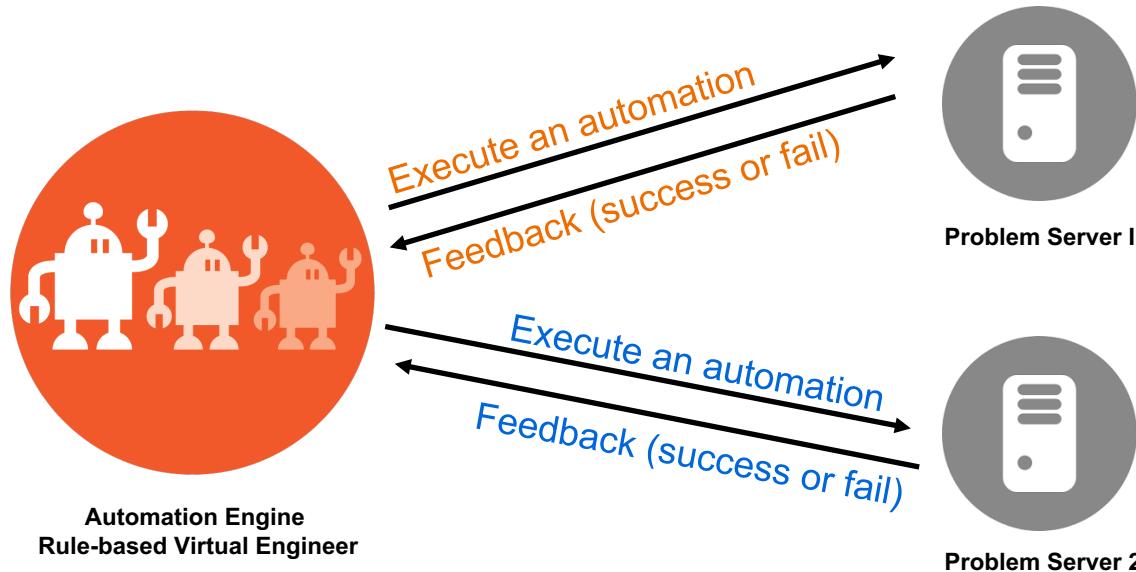
- Take the dynamic behavior of reward into account and model it as a random walk.
- A dynamic context drift model is proposed to track the contextual dynamics and consequently improve the performance of personalized recommendation in terms of CTRs

Outline

- Introduction
- Research Problems
 - Online Context-aware Recommendation with Time-varying Multi-armed Bandit
 - Dynamical Context Drift Model
 - **Online Context-based recommendation Using Hierarchical Multi-armed Bandit**
 - Hierarchical Multi-armed Bandit Model
 - Online Interactive Collaborative Filtering Using Multi-armed Bandit with Dependent Arms
 - Interactive Collaborative Topic Regression Model

IT automation recommendation modeling

IT Automation Services (ITAS) is introduced into IT service management. An automation is a scripted resolution.



IT Automation Recommendation Modeling

ALERT_KEY	cpc_cpoutil_gntw_win_v3		AUTOMATON_NAME	CPC:WIN:GEN:R:W:System Load Handler			
OPEN_DTTM	CLIENT_ID	HOSTNAME	ORIGINAL SEVERITY	OSTYPE	COMPONENT	SUBCOMPONENT	AUTO RESOLVED
2016-04-30 12:43:07	136	LEXSBWS01 VH	2	WIN	WINDOWS	CPU	1
TICKET SUMMARY	CPU Workload High. CPU 1, busy 99% time.		TICKET RESOLUTION	The CPU Utilization was quite reduced, hence closing the ticket.			

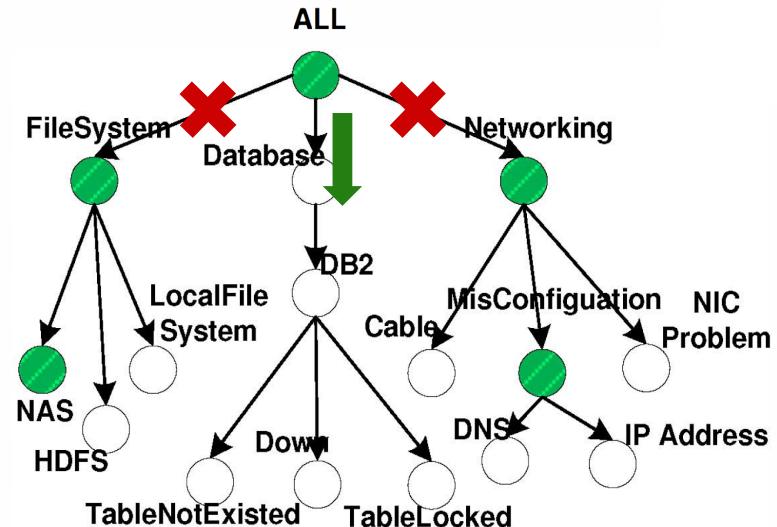
feedback

A sample ticket in ITSM and its corresponding automaton.

Challenge

- Challenge: How do we efficiently improve the performance of recommendation using the explicit **automation hierarchies** of IT automation services?

For example, a ticket is generated due to a failure of the DB2 database. The root cause may be database deadlock, high usage or other issues.



We model it as a multi-armed problem with dependent arms, where arms are in the form of hierarchies.

Hierarchical IT Automation Recommendation Modeling

In hierarchical IT automation recommendation, x_t indicates ticket problem, a^i represents an automation. \mathcal{H} denotes the hierarchy.

Our objective function is:

$$\pi^* = \arg \max_{\pi} \sum_{t=1}^T \left(\sum_{\substack{a^{(i)} \in \pi_{\mathcal{H}}(x_t | t), \\ ch(a^{(i)}) \neq \emptyset}} E_{\theta_{\pi(x_t | ch(a^{(i)}))}} (x_t^T \theta_{\pi(x_t | ch(a^{(i)}))} | t) \right)$$

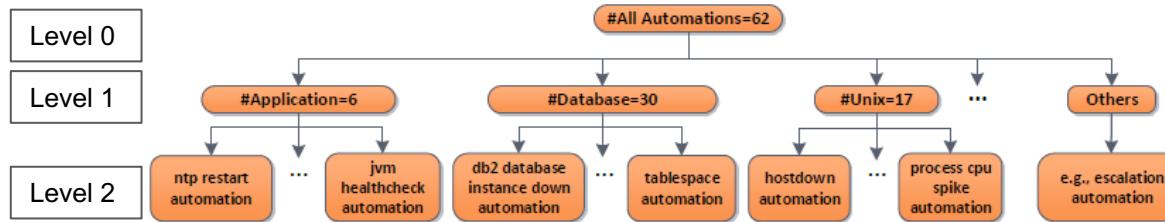
Let \mathcal{H} denote the taxonomy, which contains a set of nodes (i.e., arms) organized in a tree-structured hierarchy. Given a node $a^{(i)} \in \mathcal{H}$, $pa(a^{(i)})$ and $ch(a^{(i)})$ are used to represent the parent and children sets, respectively. Accordingly, we have Property 3.1.

PROPERTY 3.1. *If $pa(a^{(i)}) = \emptyset$, node $a^{(i)}$ is assumed to be the root node. If $ch(a^{(i)}) = \emptyset$, then $a^{(i)}$ is a leaf node, which represents an automation. Otherwise, $a^{(i)}$ is a category node when $ch(a^{(i)}) \neq \emptyset$.*

PROPERTY 3.2. *Given the contextual information x_t at time t , if a policy π selects a node $a^{(i)}$ in the hierarchy \mathcal{H} and receives positive feedback (i.e., success), the policy π receives positive feedback as well by selecting the nodes in $pth(a^{(i)})$.*

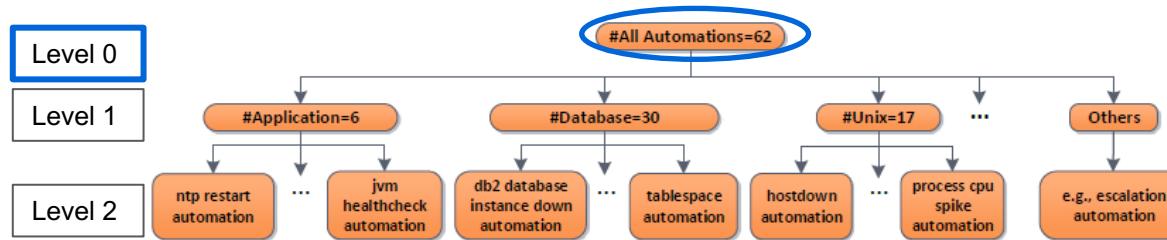
Hierarchial Multi-armed Bandit Algorithm

At time $t = [1, \dots, T]$:



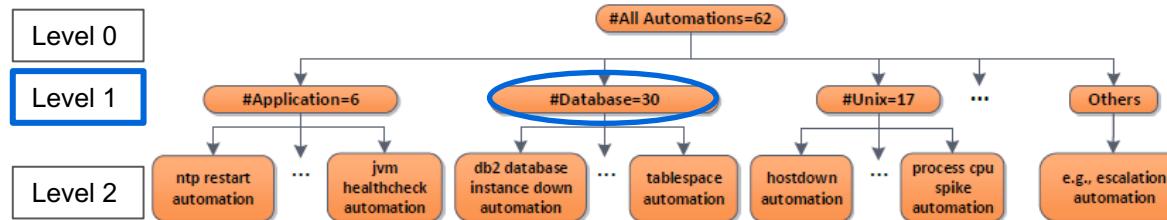
Hierarchial Multi-armed Bandit Algorithm

At time $t = [1, \dots, T]$:



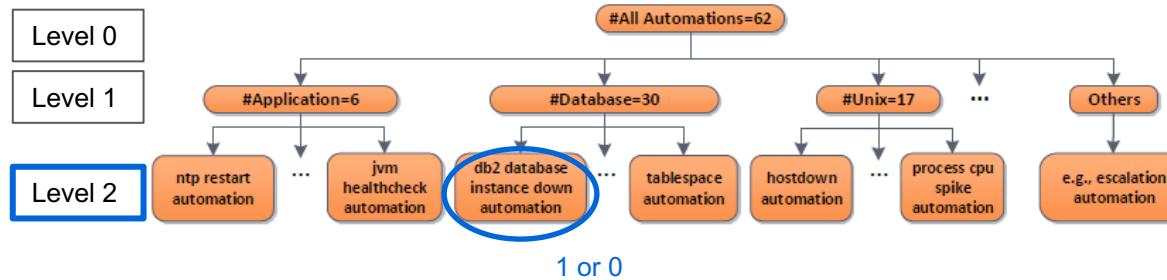
Hierarchial Multi-armed Bandit Algorithm

At time $t = [1, \dots, T]$:



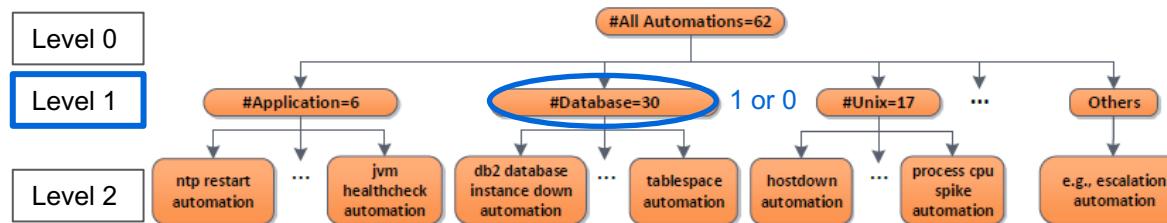
Hierarchial Multi-armed Bandit Algorithm

At time $t = [1, \dots, T]$:



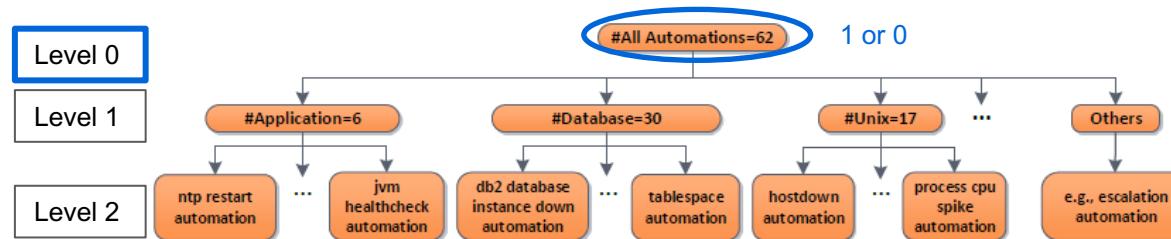
Hierarchial Multi-armed Bandit Algorithm

At time $t = [1, \dots, T]$:



Hierarchial Multi-armed Bandit Algorithm

At time $t = [1, \dots, T]$:



Experiment

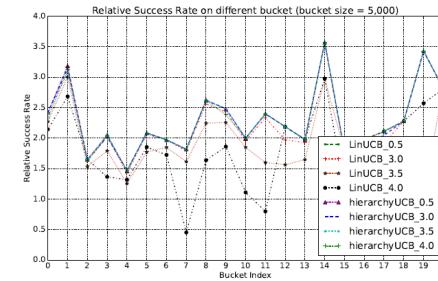
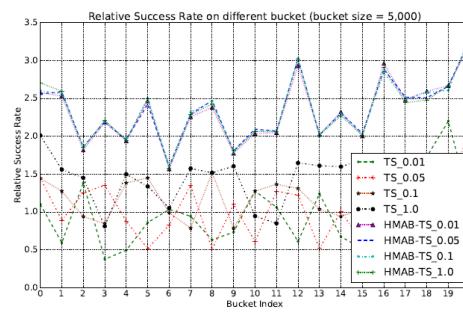
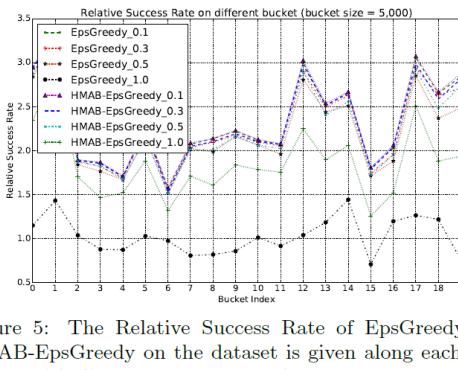
➤ Data Set

- Experimental tickets are collected by IBM Tivoli Monitoring system covering from July 2016 to March 2017 with the size of $|D| = 116,429$.
- The dataset contains 1,091 alert keys (e.g., cpusum_xuxc_aix, prccpu_rlzc_std) and 62 automations (e.g., NFS automation, process CPU spike automation) in total.
- A three-layer hierarchy H.

➤ Evaluate Method

- Replayer method.

Experiment



Conclusion

- Take the hierarchical information into account and model it as a multi-armed bandit problem with dependent arms.
- Propose hierarchical multi-armed bandit (HMAB) algorithms.

Outline

- Introduction
- Research Problems
 - Online Context-aware Recommendation with Time-varying Multi-armed Bandit
 - Dynamical Context Drift Model
 - Online Context-based recommendation Using Hierarchical Multi-armed Bandit
 - Hierarchical Multi-armed Bandit Model
 - **Online Interactive Collaborative Filtering Using Multi-armed Bandit with Dependent Arms**
 - Interactive Collaborative Topic Regression Model

Challenge

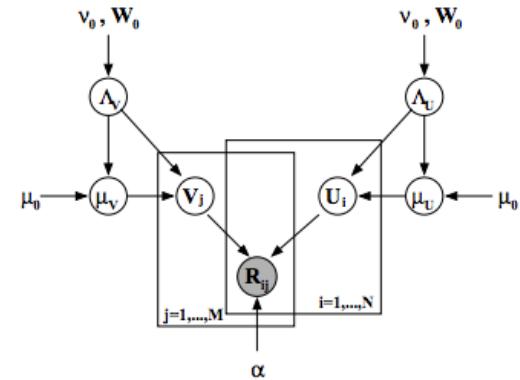
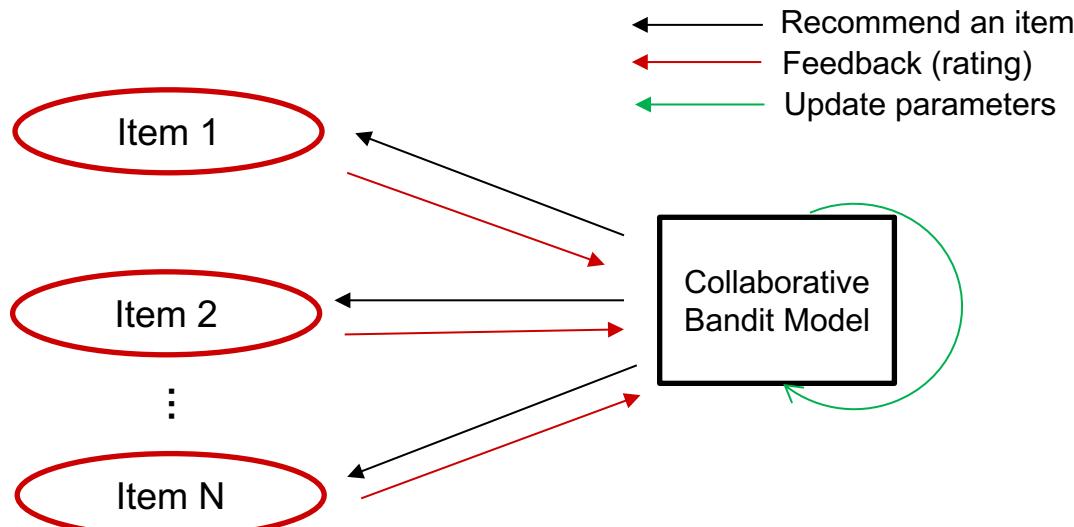
- **Challenge 1:** How do we effectively recommend a proper item to the target user with no contextual information of user/item, but only the users' interaction data on items can be utilized?

This can be naturally modeled as an [interactive collaborative filtering problem](#), which has been first introduced in [2].

	Image	Book	Video	Game
User 1	Like	Dislike	Like	Like
User 2		Like	Dislike	Dislike
User 3	Like	Like	Dislike	
User 4	Dislike		Like	
User 5	Like	Like	?	Dislike

Interactive Collaborative Filtering Problem

No context information can be observed.



Bayesian Probabilistic Matrix Factorization

Interactive Collaborative Filtering Problem

There are M users and N items. The partially observed matrix R is the preference of the users for the items. In the collaborative bandit model, the rating is estimated by a product of user and item feature vectors p_m and q_n .

$$r_{m,n} \sim N(p_m^T q_n, \sigma^2)$$

The objective function can be written as follows:

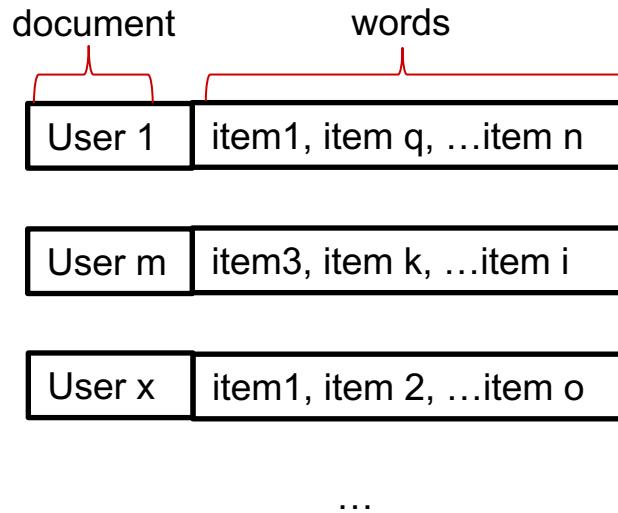
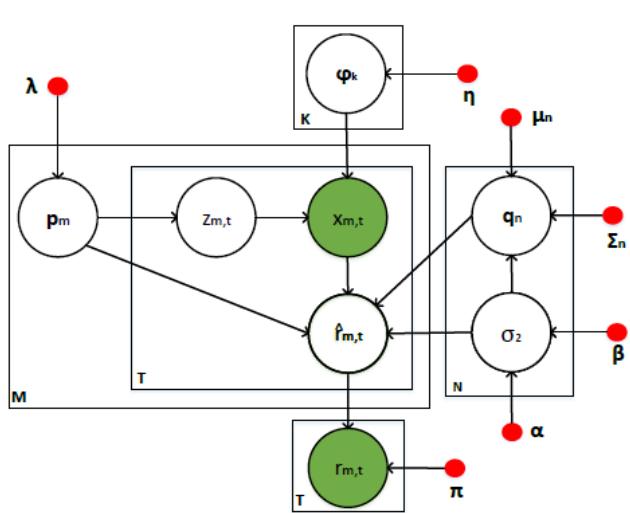
$$\pi^* = \arg \max_{\pi} \sum_{t=1} \mathbb{E}_{p_m, q_{\pi(\mathbb{S}(t))}} (p_m^T q_{\pi(\mathbb{S}(t))} | t).$$

Where $\mathbb{S}(t) = \{(n(1), r_{m,n(1)}), \dots, (n(t-1), r_{m,n(t-1)})\}$. $\mathbb{S}(t)$ is available information observed at time t .

However, these models assume the arms (i.e.) are independent.

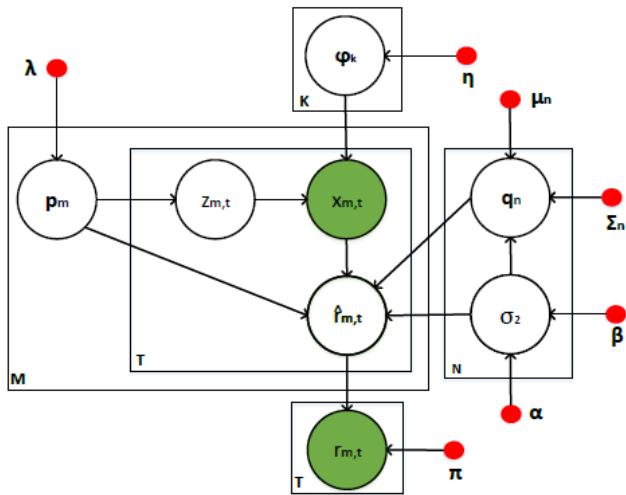
Interactive Collaborative Topic Regression Model

In light of the topic modeling techniques, we formulate the item dependencies as the clusters on arms and come up with a generative model to generate the items from their underlying topics.



The graphical representation for ICTR model

Interactive Collaborative Topic Regression Model



The graphical representation for ICTR model

$$p_m | \lambda \sim Dir(\lambda) \quad p(\sigma_n^2 | \alpha, \beta) = \mathcal{IG}(\alpha, \beta)$$

$$z_{m,t} | p_m \sim Mult(p_m), \quad x_{m,t} | \varphi_k \sim Mult(\varphi_k)$$

$$q_n | \mu_q, \Sigma_q, \sigma_n^2 \sim \mathcal{N}(\mu_q, \sigma_n^2 \Sigma_q), \quad \varphi_k | \eta \sim Dir(\eta)$$

$$n = x_{m,t}$$

The predicted rating $\hat{r}_{m,t}$ can be inferred by

$$\hat{r}_{m,t} \sim \mathcal{N}(p_m^\top q_n, \sigma_n^2).$$

Experiment

➤ Data Set

Data Set	Yahoo News	MovieLens (10M)
#users	226,710	71,567
#items	652	10,681
#ratings	280,410,150	10,000,054

➤ Evaluate Method

- Replayer method.

Experiment

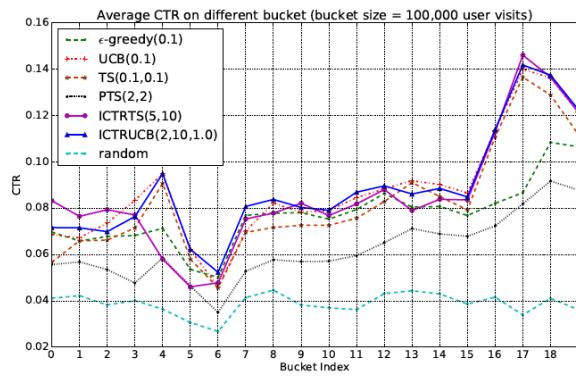


Fig. 2: The average CTR of Yahoo! Today News data is given along each time bucket. All algorithms shown here are configured with their best parameter settings.

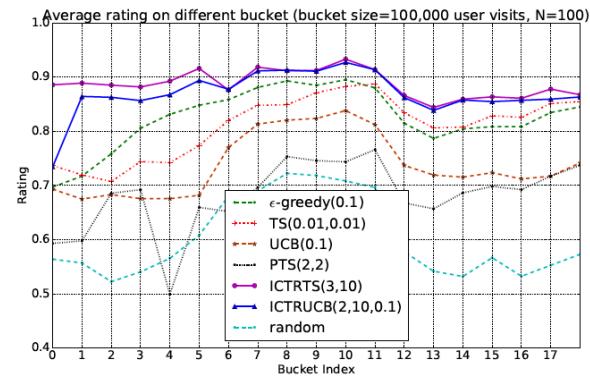


Fig. 3: The average rating of MovieLens (10M) data is given along each time bucket. All algorithms shown here are configured with their best parameter settings.

Conclusion

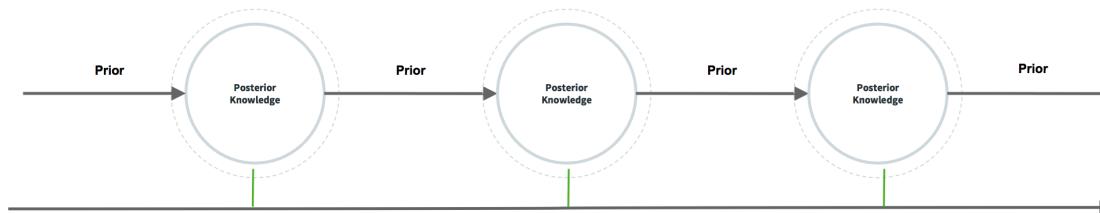
- Only user/item rating matrix is taken into account and model it as a multi-armed bandit problem with dependent arms.
- Propose Interactive Collaborative Topic Regression (ICTR) model to learn the dependence among arms.

Q & A

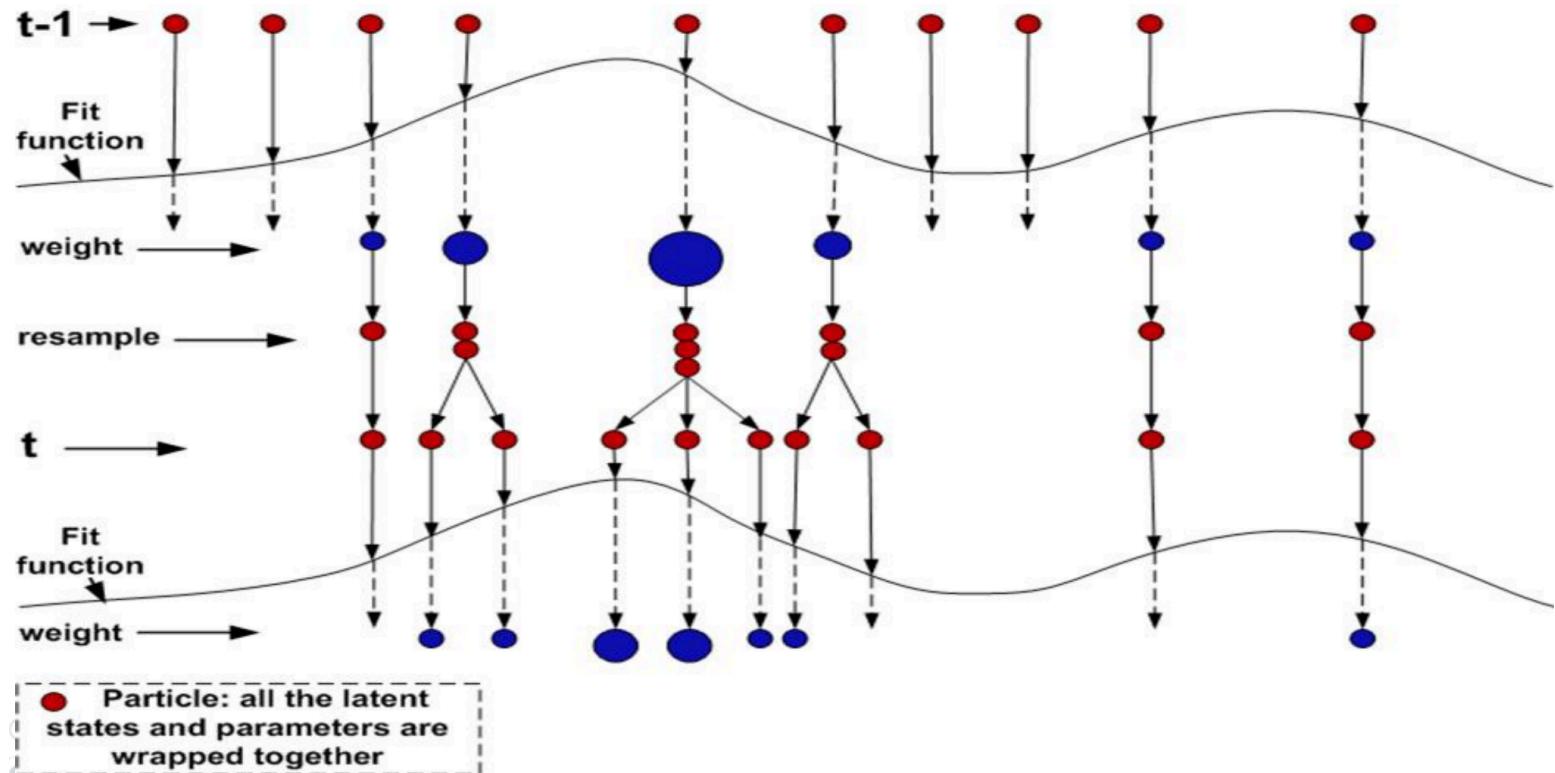


Online Inference of ICTR Model: Particle Learning

Definition 1 (Particle). A particle for predicting the reward $\hat{r}_{m,t}$ is a container that maintains the current status information for both user m and item $x_{m,t}$. The status information comprises of random variables such as p_m , σ_n^2 , Φ_k , q_n , and $z_{m,t}$, as well as the hyper parameters of their corresponding distributions, such as λ , α , β , η , μ_q and Σ_q .



Online Inference of ICTR Model: Particle Learning



Re-sample Particles with Weights

Let $\mathcal{P}_{m,n(t-1)}$ denote the particle set at time $t - 1$ and $\mathcal{P}_{m,n(t-1)}^{(i)}$ be the i^{th} particles given both ticket problem m and automation $n(t - 1)$ at time $(t - 1)$, where $1 \ll i \ll B$. Each particle has a weight, denoted as $\rho^{(i)}$, where $\sum_{i=1}^B \rho^{(i)} = 1$. The fitness of each particle $\mathcal{P}_{m,n(t-1)}^{(i)}$ is defined as the likelihood of the observed data $x_{m,t}$ and $r_{m,t}$. Therefore,

$$\rho^{(i)} \propto p(x_{m,t}, r_{m,t} | \mathcal{P}_{m,n(t-1)}^{(i)}).$$

As further deriving,

$$\rho^{(i)} \propto \sum_{z_{m,t}=1}^K \{N(r_{m,t} | p_m^\top q_n, \sigma_n^2) E(p_{m,k} | \lambda) E(\varphi_{k,n} | \eta)\}$$

where $E(p_{m,k} | \lambda) = \frac{\lambda_k}{\sum_{k=1}^K \lambda_k}$ and $E(\varphi_{k,n} | \eta) = \frac{\eta_{k,n}}{\sum_{n=1}^N \eta_{k,n}}$ represent the conditional expectations of $p_{m,k}$ and $\varphi_{k,n}$ given the observed reward λ and η of $\mathcal{P}_{m,n(t-1)}^{(i)}$.

Latent State Inference

Provide with new observation $x_{m,t}$ and $r_{m,t}$ at time t , the random state $z_{m,t}$ can be any one of K topics. The posterior distribution of $z_{m,t}$ is shown as follows, where $\theta \in \mathcal{R}^K$:

$$z_{m,t} | x_{m,t}, r_{m,t}, \mathcal{P}_{m,n(t-1)}^{(i)} \sim Mult(\theta),$$

θ can be computed by

$$\theta \propto E(p_{m,k} | r_{m,t}, \lambda) \cdot E(\Psi_{k,n} | r_{m,t}, \lambda)$$

$$E(p_{m,k} | r_{m,t}, \lambda) = \frac{\mathcal{I}(z_{m,t} = k)r_{m,t} + \lambda_k}{\sum_{k=1}^K [\mathcal{I}(z_{m,t} = k)r_{m,t} + \lambda_k]},$$
$$E(\Phi_{k,n} | r_{m,t}, \eta) = \frac{\mathcal{I}(x_{m,t} = n)r_{m,t} + \eta_{k,n}}{\sum_{n=1}^N [\mathcal{I}(x_{m,t} = n)r_{m,t} + \eta_{k,n}]}.$$

Where $\mathcal{I}(\cdot)$ is an indicator function, returns 1 when the input Boolean expression is true and otherwise return 0.

Parameter Statistics Inference

Assume $\mu'_{\mathbf{q}'}, \Sigma'_{\mathbf{q}'}, \alpha', \beta', \lambda',$ and η' are the sufficient statistics at time t , which are updated on the sufficient statistics $\mu_{\mathbf{q}}, \Sigma_{\mathbf{q}}, \alpha, \beta, \lambda, \eta$ at $t-1$, and new observation data $x_{m,t}$ and $r_{m,t}$ at time t as follows.

$$\begin{aligned}\Sigma'_{\mathbf{q}_n} &= (\Sigma_{\mathbf{q}_n}^{-1} + \mathbf{p}_m \mathbf{p}_m^\top)^{-1} \\ \mu'_{\mathbf{q}_n} &= \Sigma'_{\mathbf{q}_n} (\Sigma_{\mathbf{q}_n}^{-1} \mu_{\mathbf{q}_n} + \mathbf{p}_m r_{m,t}) \\ \alpha' &= \alpha + \frac{1}{2} \\ \beta' &= \beta + \frac{1}{2} (\mu_{\mathbf{q}_n}^\top \Sigma_{\mathbf{q}_n}^{-1} \mu_{\mathbf{q}_n} + r_{m,t}^\top r_{m,t} - \mu_{\mathbf{q}_n}^\top \Sigma_{\mathbf{q}_n}'^{-1} \mu'_{\mathbf{q}_n}) \\ \lambda'_k &= \mathcal{I}(z_{m,t} = k) r_{m,t} + \lambda_k \\ \eta'_{k,n} &= \mathcal{I}(x_{m,t} = n) r_{m,t} + \eta_{k,n}\end{aligned}$$

At time t , the sampling process for the parameter random variables $\mathbf{q}_n, \sigma_n^2, \mathbf{p}_m, \Phi_k$ is summarized as below:

$$\begin{aligned}\sigma_n^2 &\sim \mathcal{IG}(\alpha', \beta'), \\ \mathbf{q}_n | \sigma_n^2 &\sim \mathcal{N}(\mu'_{\mathbf{q}_n}, \sigma_n^2 \Sigma'_{\mathbf{q}_n}), \\ \mathbf{p}_m &\sim Dir(\lambda'), \\ \Phi_k &\sim Dir(\eta').\end{aligned}$$

Integrate with Policies: Thompson sampling

Without new observation $x_{m,t}$ and $r_{m,t}$, the particle re-sampling, latent state inference and parameter statistics inference for time t , therefore, we utilize the latent vectors p_m and q_n sampled from their posterior distributions at time $t-1$ predicting the reward for each arm.

In our model, each item has B independent particles. Based on Thompson sampling, the policy select an arm $n(t)$ using the following equation:

$$n(t) = \arg \max_n (\bar{r}_{m,n}),$$

Where $\bar{r}_{m,n}$ denotes the average reward:

$$\bar{r}_{m,n} = \frac{1}{B} \sum_{i=1}^B \mathbf{p}_m^{(i)\top} \mathbf{q}_n^{(i)}.$$

Integrate with Policies: UCB

According to UCB policy, it select an arm $n(t)$ based on the upper bound of the predicted reward.
Assuming that

$$r_{m,t}^{(i)} \sim \mathcal{N}(\mathbf{p}_m^{(i)\top} \mathbf{q}_n^{(i)}, \sigma^{(i)2})$$

$$\bar{r}_{m,n} = \frac{1}{B} \sum_{i=1}^B r_{m,t}^{(i)}$$

the UCB is developed by the mean and variance of predicted reward.

$$n(t) = \arg \max_n (\bar{r}_{m,n} + \gamma \sqrt{\nu}),$$

where $\gamma \gg 0$ is a predefined threshold, and the variance is

$$\nu = \frac{1}{B} \sum_i \sigma^{(i)2}$$