Contextual Bandits

Subrahmanya Swamy Peruru

Photos provided by Pexels

Contextual Bandits – Multiple states

- News article Recommendation systems

- Articles – arms
- Like / Dislike – Reward
- User – State

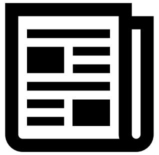
Different users have different preferences for articles

Photos provided by Pexels

Multi-arm Bandits – One state

Arms / Articles

$\boldsymbol{a}$



$\mu(a) = 0.2$

$\boldsymbol{b}$

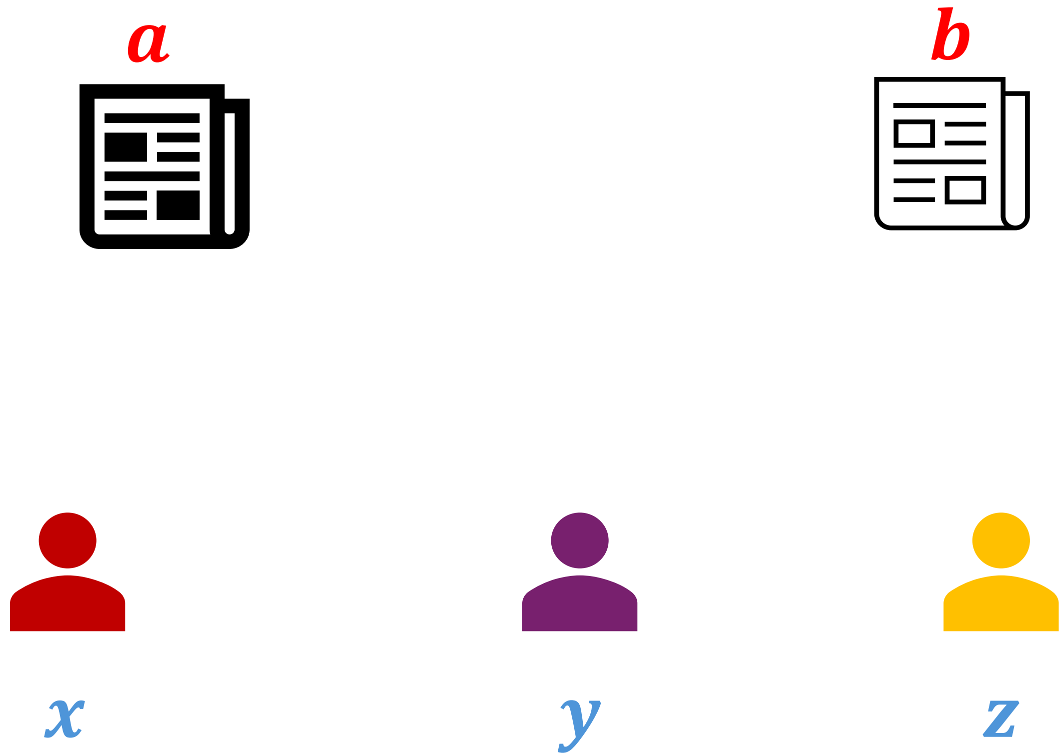
$\mu(b) = 0.9$

- Each arm has only one expected reward associated with it

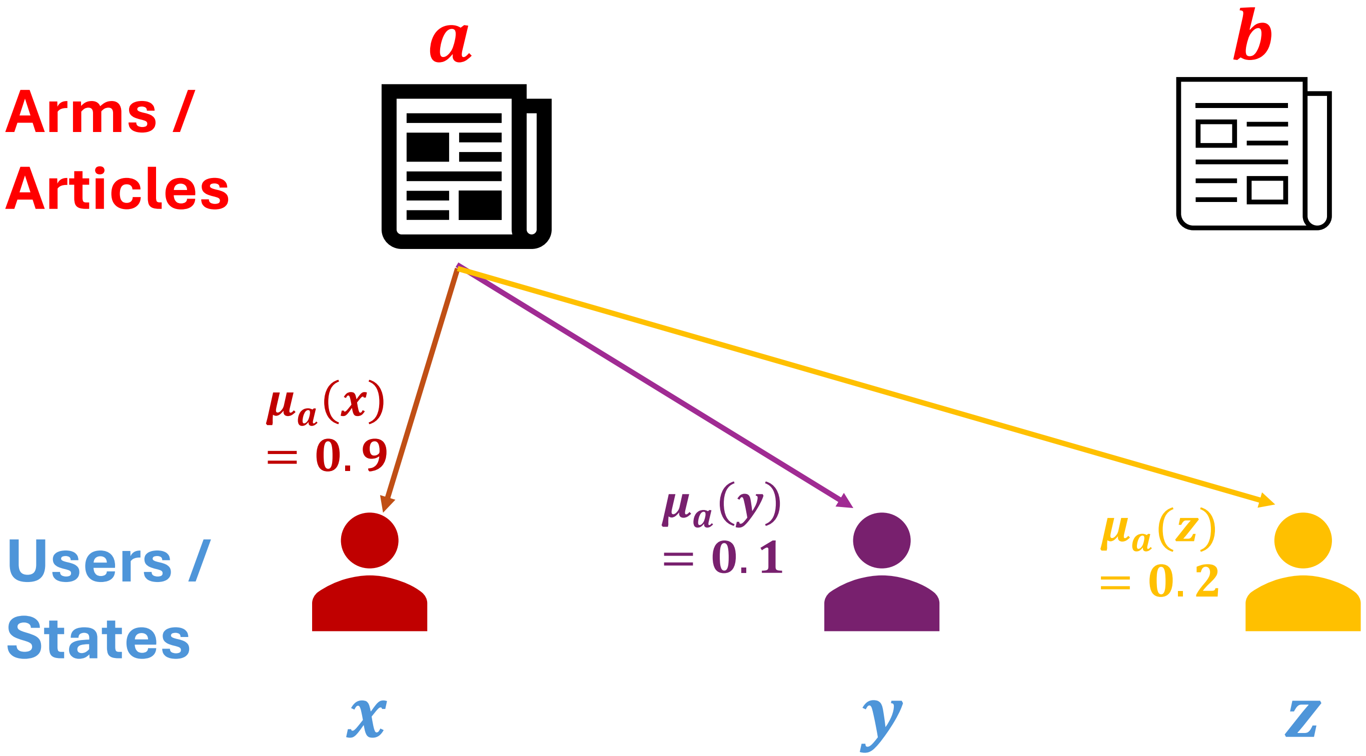
Contextual Bandits – Multiple States

Arms / Articles

Users / States

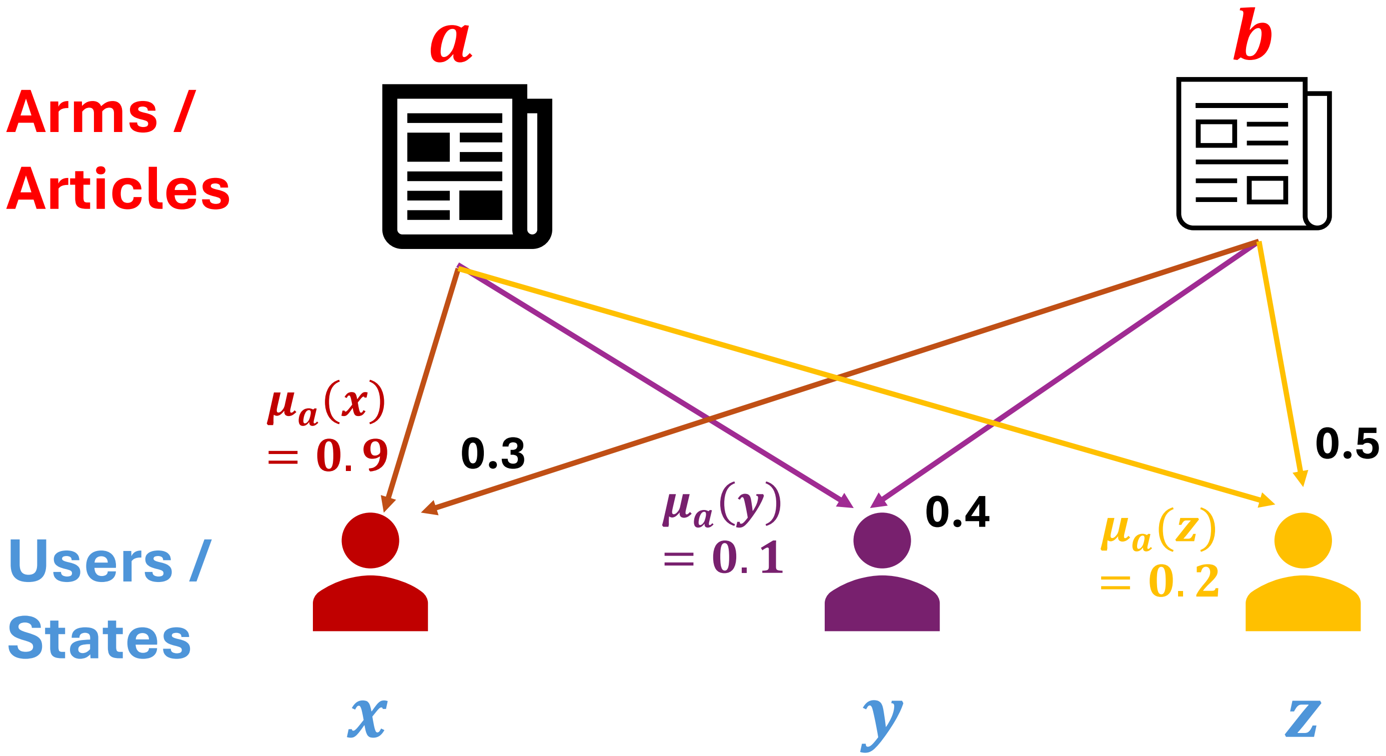


Contextual Bandits – Multiple States



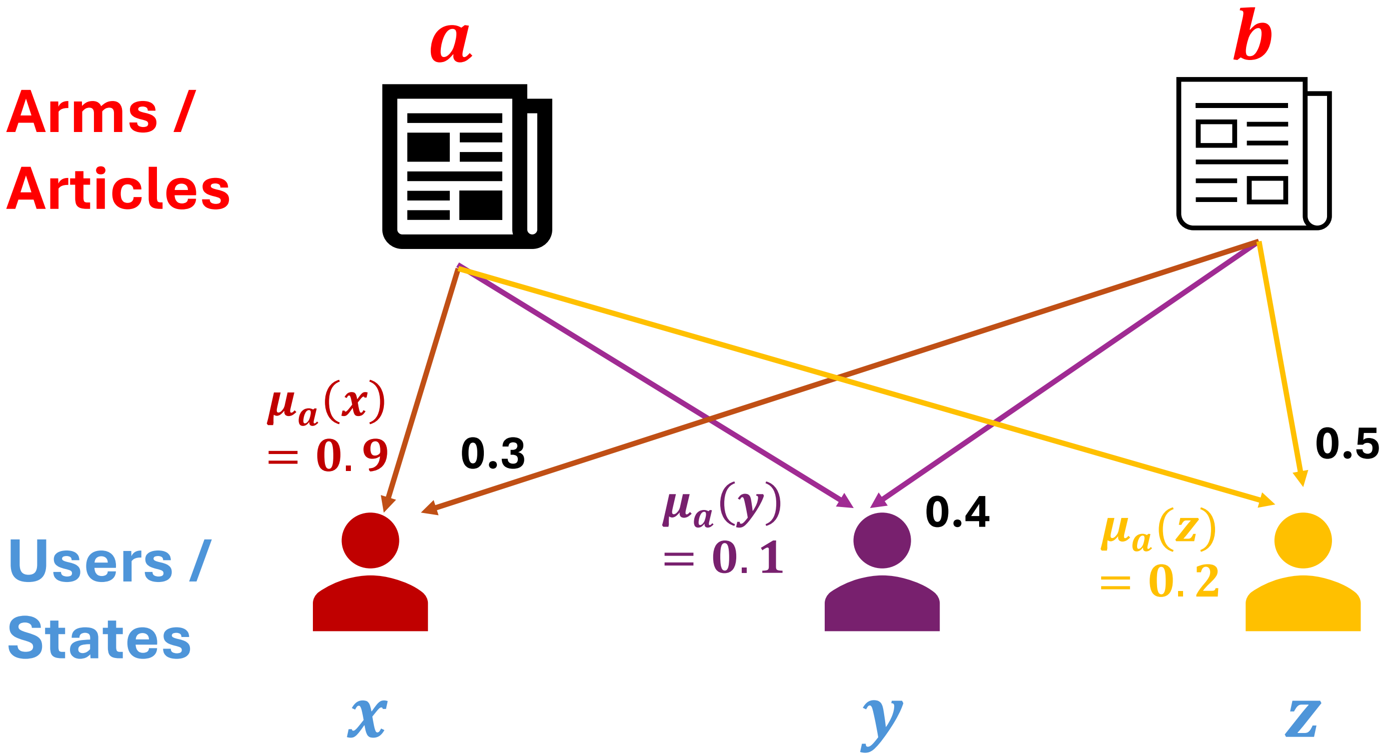
- Expected reward of an arm changes with user $\mu_a(x)$

Contextual Bandits – Multiple States



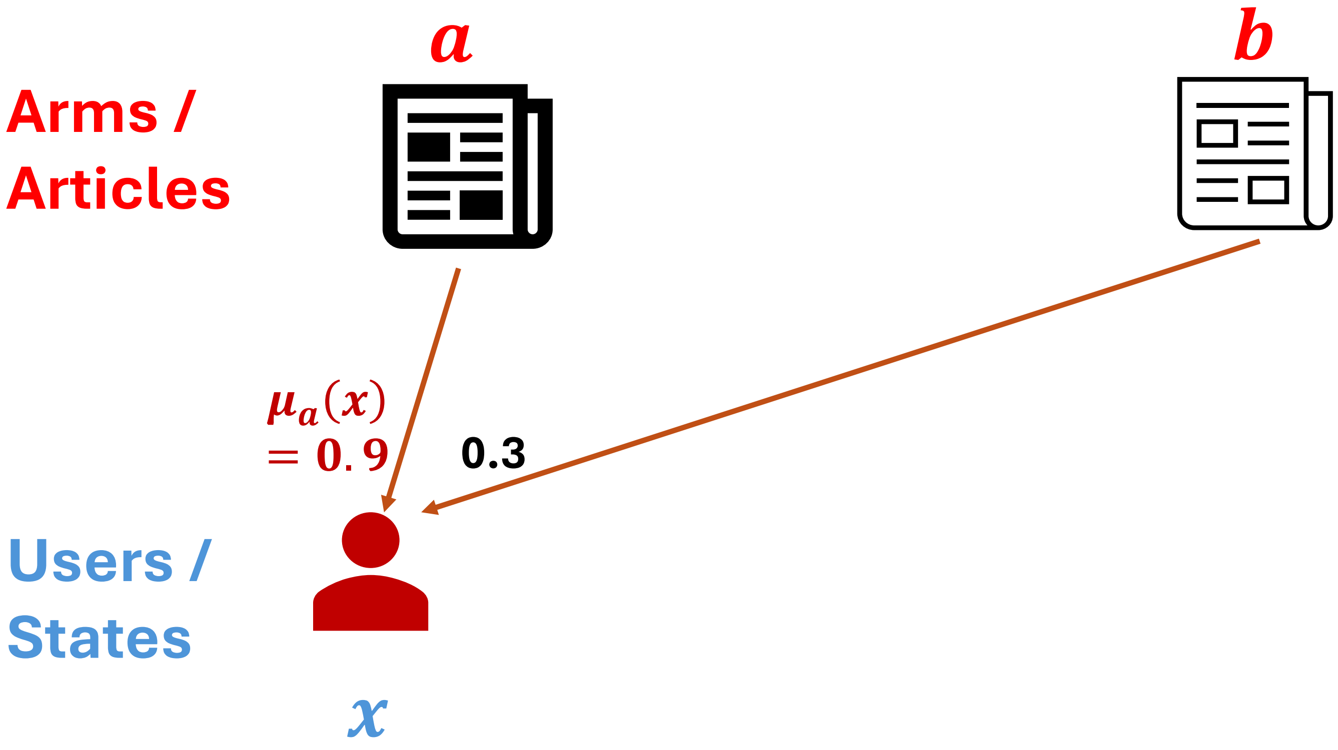
- Expected reward of an arm changes with user $\mu_a(x)$

Contextual Bandits – Multiple States



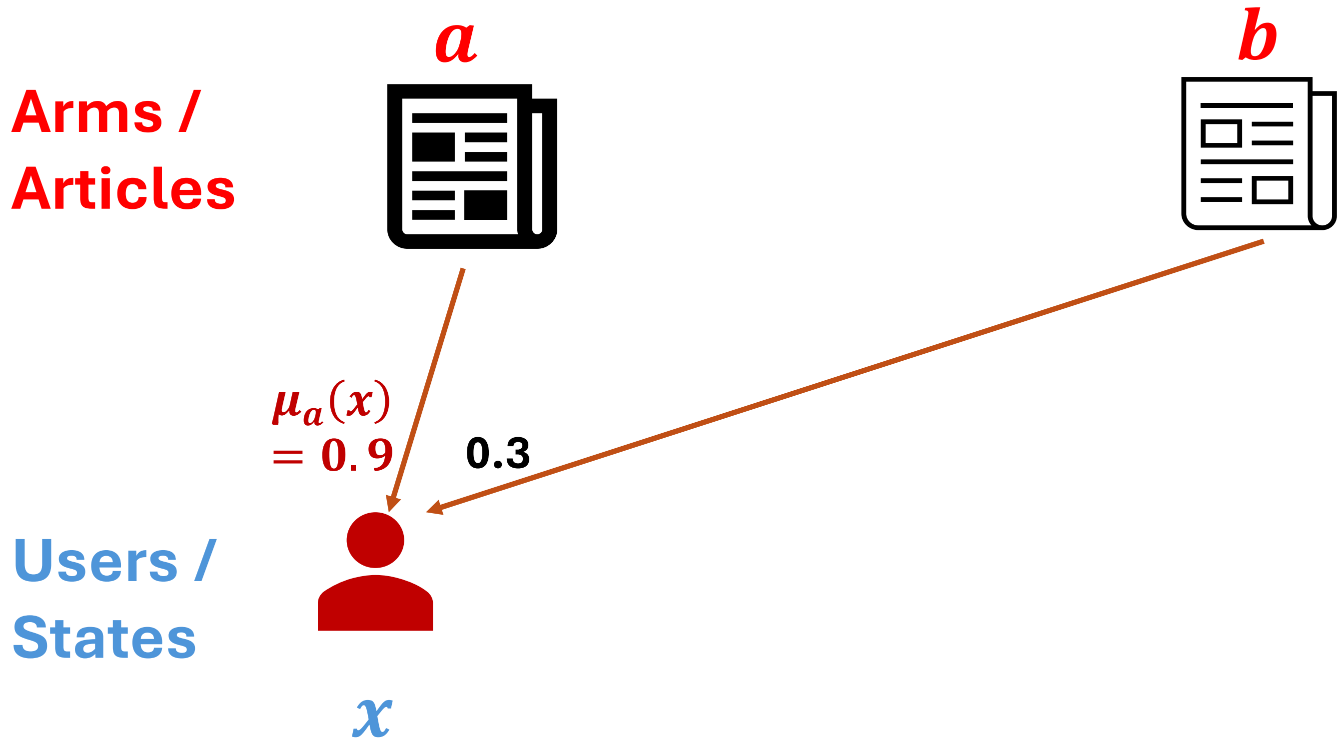
- Expected reward of an arm changes with user $\mu_a(x)$
- **How to deal with it?**

Contextual Bandits – Multiple States



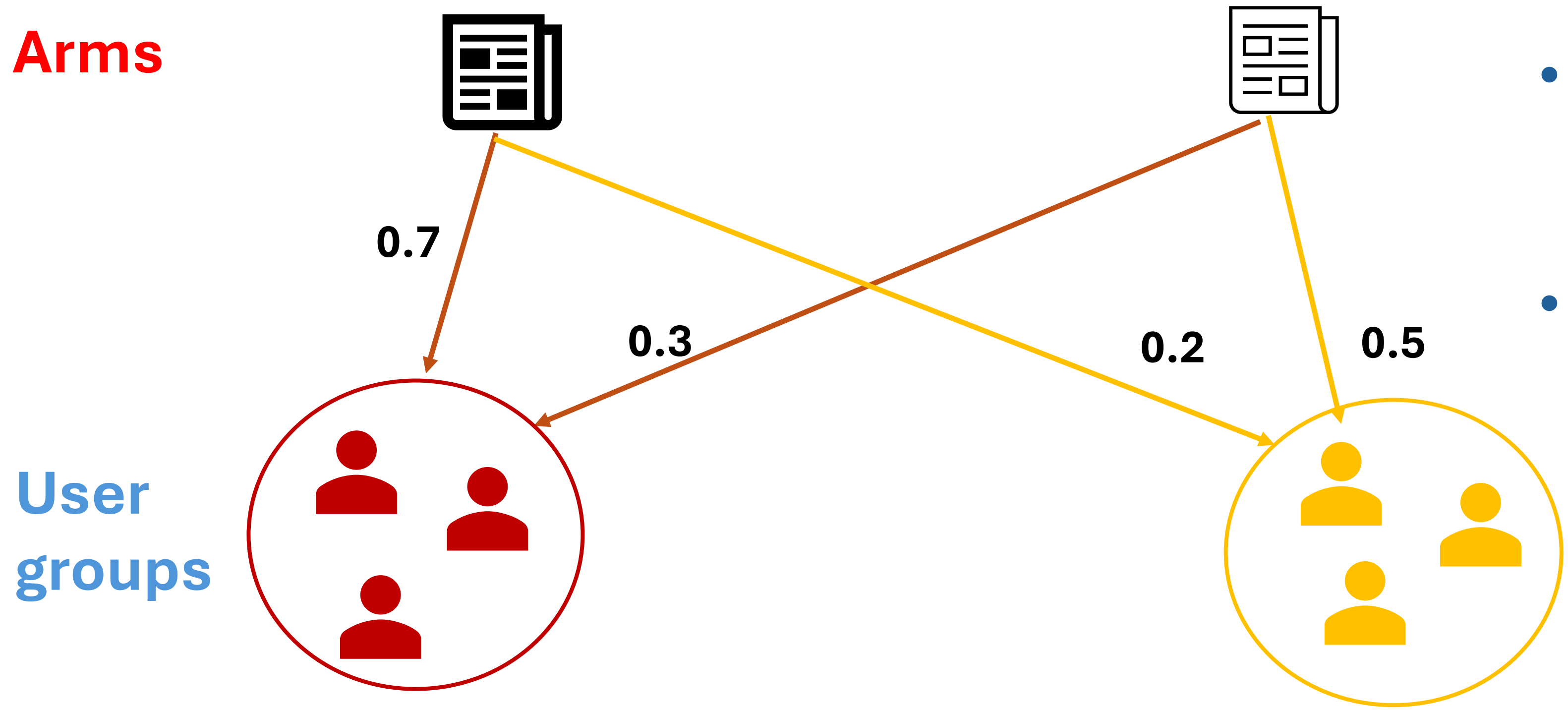
- Expected reward of an arm changes with user $\mu_a(x)$
- Treat each user as a separate bandit problem

Contextual Bandits – Multiple States



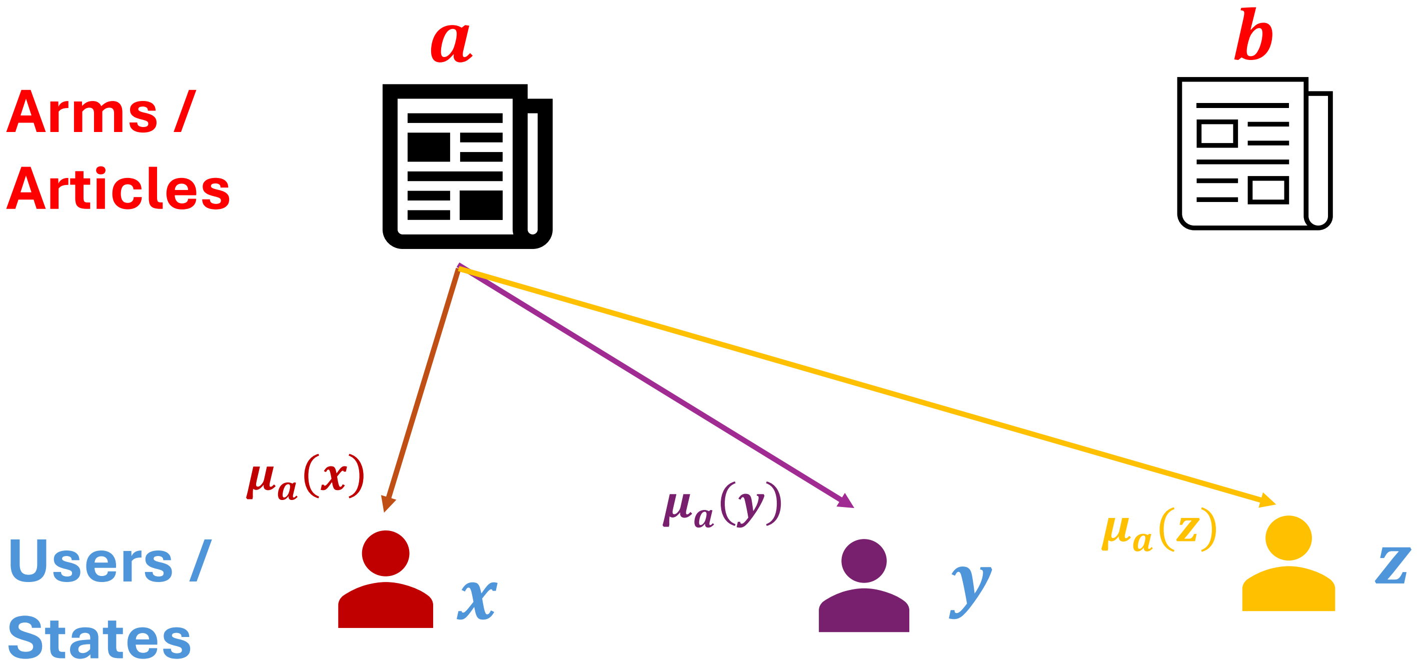
- Expected reward of an arm changes with user $\mu_a(x)$
- Treat each user as a separate bandit problem
- Practically infeasible with millions of users!

Bandits + Unsupervised Learning

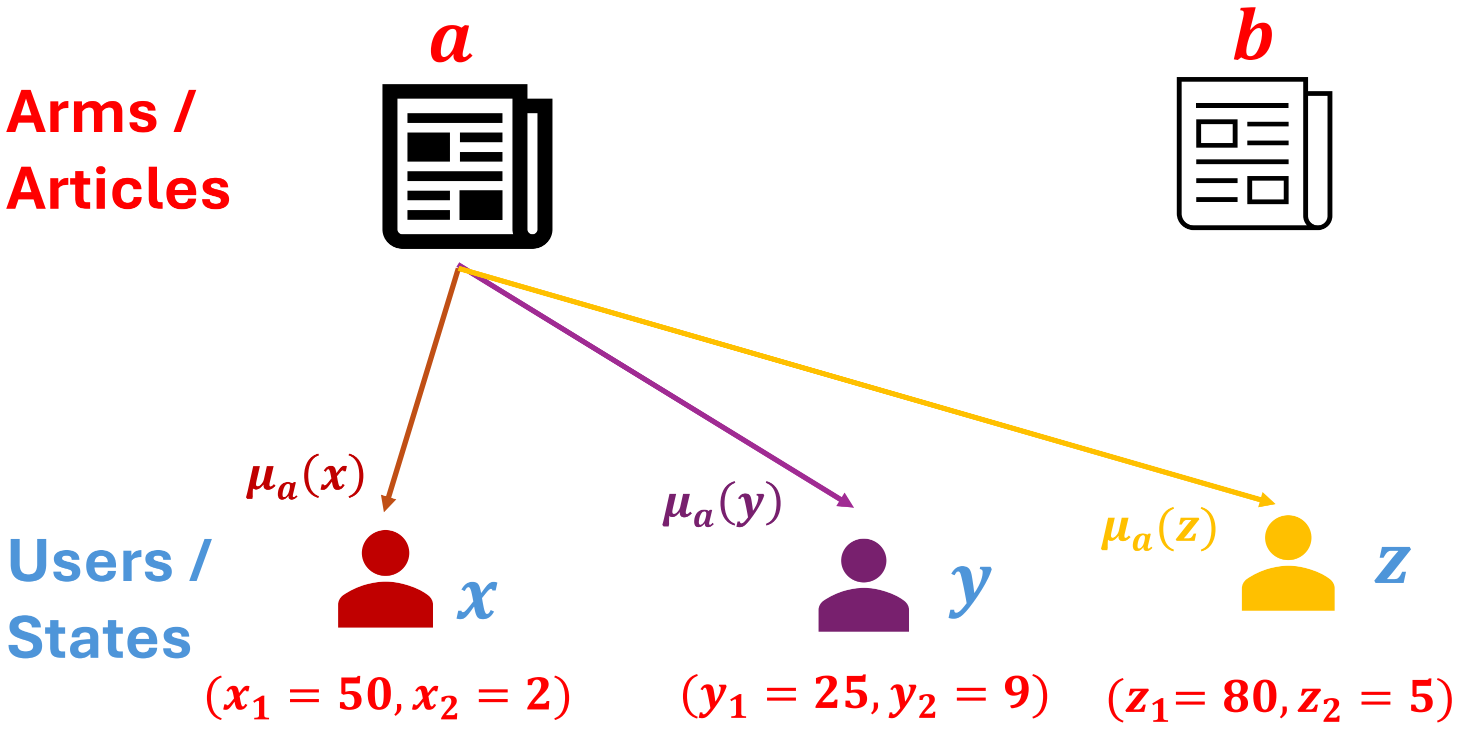


- Form user groups by clustering similar users together
- Solve a separate bandit problem for each cluster

Bandits + Supervised Learning

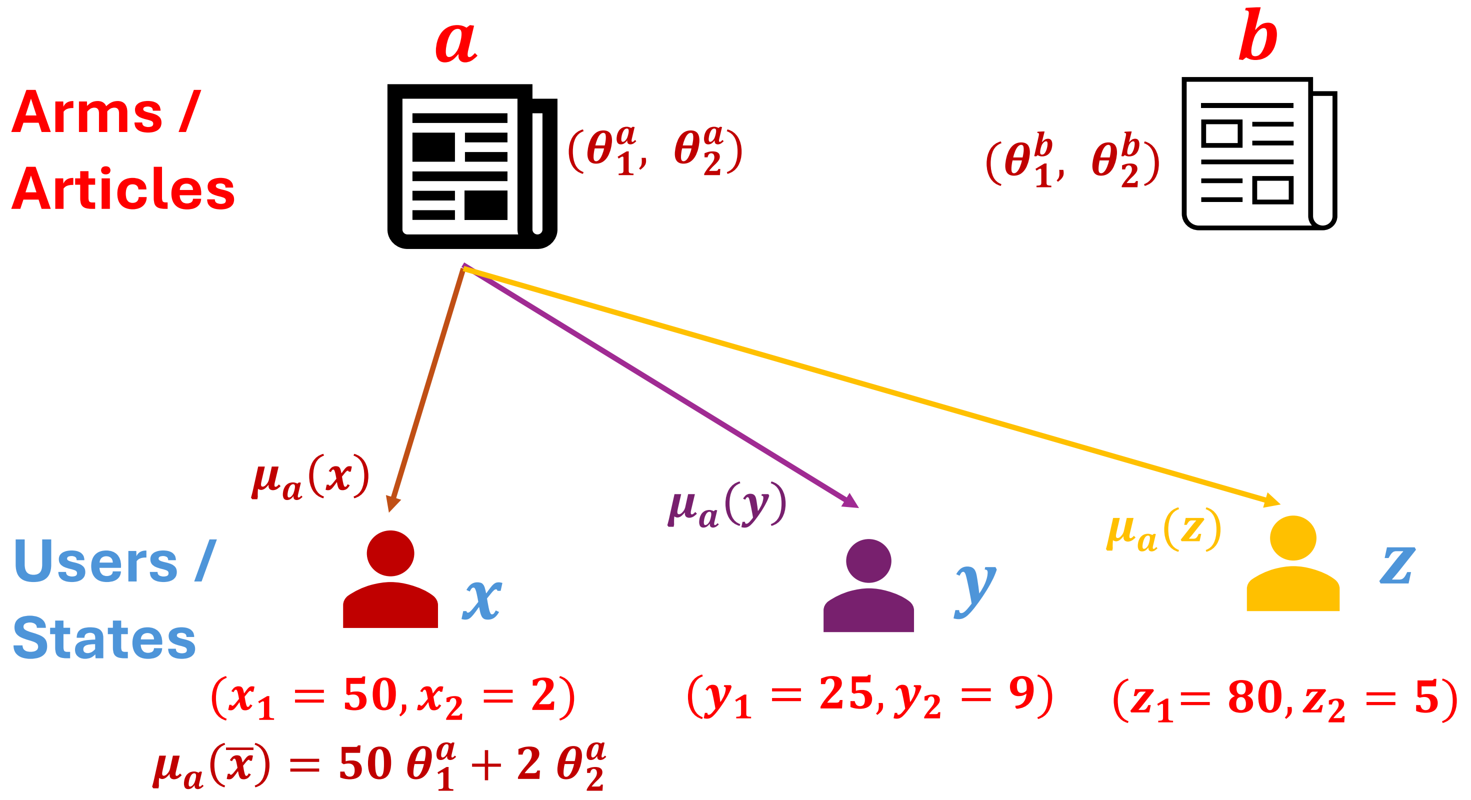


Bandits + Supervised Learning



- User (state) represented by features such as age, income $\bar{x} = (x_1, x_2)$

Bandits + Supervised Learning



- User (state) represented by features such as age, income $\bar{x} = (x_1, x_2)$
- Model the expected reward for user $\bar{x}$ for pulling arm a as $\mu_a(\bar{x}) = \theta_1^a x_1 + \theta_2^a x_2$

Contextual (Linear) Bandits

- Users (state) represented by features such as age, gender
  - State feature vector: $\bar{x} = (x_1, x_2)^T$ Eg: (50, 1), (25, 0), (80, 1)

- Each article (arm) has a different expected reward associated with each user (state)
  - The expected reward of an arm is characterized by **unknown** $\overline{\theta^a} = (\theta_1^a, \theta_2^a)^T$
  - State-specific expected reward: $\mu_a(\bar{x}) = \theta_1^a x_1 + \theta_2^a x_2$ (Linear Bandits)

- The reward for playing arm a_t under state $\overline{x_t}$ is $R_t = \mu_{a_t}(\overline{x_t}) + \epsilon_t$, where ϵ_t is independent mean-zero noise.

- If T arbitrary users visit our website sequentially, How to maximize the total reward $\sum_{t=1}^{T} R_t$?

Photos provided by Pexels

Multi-arm Bandits: Parameter Estimation

- Explore each arm N times and
- Estimate the mean parameters based on sample rewards

One-state Multi-arm Bandits

$\mu(a) = 2$ Unknown parameter

Sample reward: $R_t = \mu(a) + noise$

Rewards from 3 rounds of exploration

$$\mu(a) \approx R_1 = 2.7$$
$$\mu(a) \approx R_2 = 1.6$$
$$\mu(a) \approx R_3 = 2.1$$

Best estimate: $\hat{\mu}(a) = \arg\min_x \, (R_1 - x)^2 + (R_2 - x)^2 + (R_3 - x)^2$

Optimal solution: $\hat{\mu}(a) = \overline{\mu(a)} = \frac{R_1 + R_2 + R_3}{3}$

Photos provided by Pexels

Linear Bandits – Parameter Estimation

Unknown parameter: (θ_1^a, θ_2^a)
Sample reward: $R_t = \mu_a(x) + noise$
$= \theta_1^a x_1 + \theta_2^a x_2 + noise$

Exploration:

3 users with features: (50, 1), (25, 4), (80, 7)
Observed rewards: 0.7, 0.4, 0.5

$$D_a \, \theta^a = b_a$$

$$R_1 = 0.7 \approx \theta_a^1 \, 50 + \theta_a^2 \, 1$$
$$R_2 = 0.4 \approx \theta_a^1 \, 25 + \theta_a^2 \, 4$$
$$R_3 = 0.5 \approx \theta_a^1 \, 80 + \theta_a^2 \, 7$$

$$\begin{bmatrix} 50 & 1 \\ 25 & 4 \\ 80 & 7 \end{bmatrix} \begin{bmatrix} \theta_a^1 \\ \theta_a^2 \end{bmatrix} = \begin{bmatrix} 0.7 \\ 0.4 \\ 0.5 \end{bmatrix}$$

Best estimate: $\widehat{\theta^a} = \arg\min_{(\theta_1^a, \theta_2^a)} \begin{matrix} (0.7 - \theta_a^1 \, 50 + \theta_a^2 \, 1)^2 + \\ (0.4 - \theta_a^1 \, 25 + \theta_a^2 \, 4)^2 + \\ (0.5 - \theta_a^1 \, 80 + \theta_a^2 \, 7)^2 \end{matrix}$

Optimal Sol (with Regularizer): $\widehat{\theta^a} = \left(\mathrm{D}_\mathrm{a}^\mathrm{T} \, \mathrm{D}_\mathrm{a} + \mathrm{I}_\mathrm{d}\right)^{-1} \mathrm{D}_\mathrm{a}^\mathrm{T} \mathrm{b}_\mathrm{a}$

Photos provided by Pexels

ETC algorithm for Linear Bandits

- Explore each arm N times
- Based on the history $\{(\overline{x_t}\ a_t, R_t)\}_{t=1}^{NK}$, estimate the parameters for all $a \in \mathcal{A}$ using Ridge regression.
  - $\widehat{\theta^a} = \left(\mathrm{D_a^T}\ \mathrm{D_a} + \mathrm{I_d}\right)^{-1} \mathrm{D_a^T b_a}$
  - D_a is $N \times d$ context matrix whose rows represent user feature vectors
  - b_a is $N \times 1$ reward vector with rewards obtained during N exploration rounds
  - $D_a = \begin{bmatrix} x_1^1 & x_2^1 \\ x_1^2 & x_2^2 \\ x_1^3 & x_2^3 \end{bmatrix}$ feature dimension $d = 2$, arm a played $\mathrm{N} = 3$ times
- At any round $t > NK$, play the arm $a_t = \arg\max_a\ x_t^T \widehat{\theta^a}$

Photos provided by Pexels

ϵ —greedy for Linear Bandits

- Explore each arm for d rounds
- Estimate the θ^a parameters for all $a \in \mathcal{A}$ using Ridge regression.
- At any round $t > Kd$, if the user has features x_t
  - With probability $1 - \epsilon$: Play the arm $a_t = \arg\max_a \; x_t^T \widehat{\theta^a}$
  - With probability ϵ: Play any arm at random

Photos provided by Pexels

LinUCB (UCB for Linear Bandits)

- Explore each arm for d rounds
- At time t, based on the history $\{(\bar{x}_s, a_s, R_s)\}_{s=1}^{t-1}$, estimate the θ^a parameters for all $a \in \mathcal{A}$ using Ridge regression.
- Pick the arm $a_t = \arg\max_a \; x_t^T \widehat{\theta^a} + \sqrt{x_t^T (D_a^T D_a + I_d)^{-1} x_t}$

Exploit **Explore**

Photos provided by Pexels