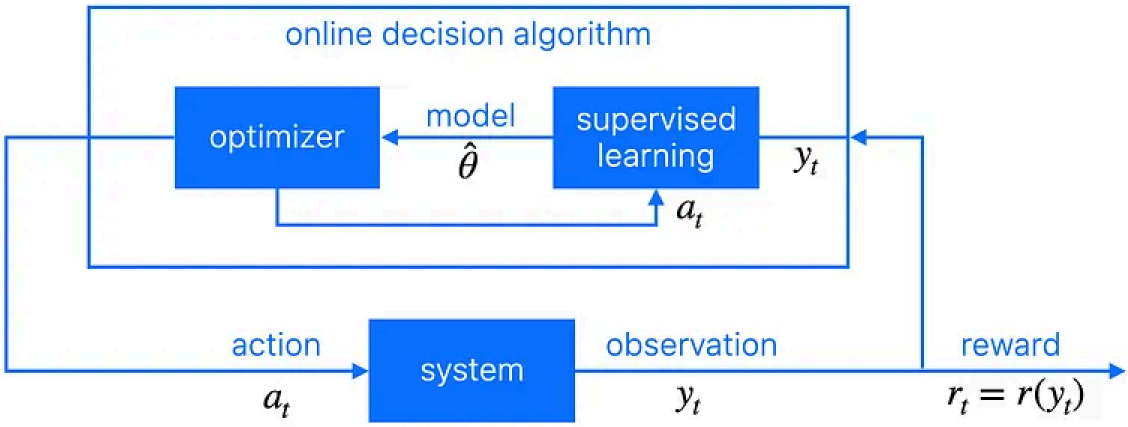
Multi-Arm Bandit for Recommendation Systems

Raaghav P, 21011101094

What is Multi-Arm Bandits (MAB)?

- A type of RL which aims to strike a balance between exploration and exploitation.
- They achieve this by exploring new actions to understand thier potential rewards and then exploiting the current best action to maximize the overall reward.
- The objective is to gain knowledge about and select actions that maximize the total reward while minimizing regret.



ϵ-greedy Algorithm

- At every trial, it randomly chooses an action with probability ε and greedily chooses the highest value action with probability 1 - ε.
- We balance the explore-exploit trade-off via the parameter ε.
  - A higher ε leads to more exploration while a lower ε leads to more exploitation.
  - However, ε-greedy can explore longer than necessary (though this can be mediated by decreasing ε over time).
- Another downside is that ε-greedy doesn't provide guidance on which items to explore and defaults to exploring all items uniformly at random.

Recommendation System

- In terms of recommendation system, MAB can outperform the traditional A/B testing because,
    - they can handle more complex situations.
    - they adapt more quickly to the observed data.
    - they can dynamically allocate more resources to versions that perform better for specific user segments, leading to more personalized experiences.

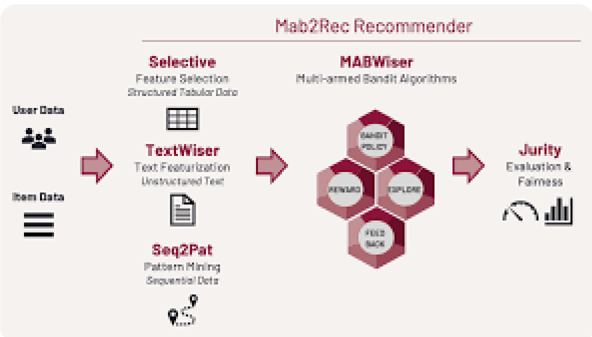
> **Mapping MAB Terms in context of Recommendation System**
>
> - **Action/arm**: recommendation item candidates.
> - **Reward**: customer interaction from a single trial, such as a click or purchase.
> - **Value**: estimated long-term reward of an arm over multiple trials.
> - **Policy**: algorithm/agent that chooses actions based on learned values.

> **GOAL**
>
> - Here, the difference from traditional learning scenarios is the goal, which must be related to *user's satisfaction with the system.*
> - It requires a personalized *action selection policy* π to the users' preferences and tastes identified by the historic of *user's actions* h.
> - For this reason, the item i_t^* should be chosen according to a prediction rule π, which is defined as a function to exploit and explore the current known information about the user until now: $i^* = \pi(h_t)$

Thus, the main goal is also to maximize the expected reward achieved after T times,

$$i^*_{(\cdot)} = argmax \sum^T_{t=1} \mathbb{E}\left[r_{u,i_t} | t\right]$$

- The overall process looks like



Reference

1. # Multi-Armed Bandits in Recommendation Systems: A survey of the state-of-the-art and future directions