# Online Clustering for News Recommender System

Minh N.H. Nguyen and Choong Seon Hong
Department of Computer Science and Engineering, Kyung Hee University
Email: {minhnhn, cshong}@khu.ac.kr

**Abstract**: Nowadays, intelligent recommender systems in web applications provide helpful information for online users. The more useful and related items are suggested to users, the more interesting and better feedback from users. However, traditional recommender systems become inefficient in the large scale scenarios and require high computation cost. Recommender agents need to make real-time decisions for millions of incoming user requests per day. Based on limited information from user context, we propose algorithm OC-MAB with integrated clustering technique and online learning. This approach can perform as an adaptive learning in dynamic environment with scalability, cheap storage and computation cost for news recommendation systems.

## 1. Introduction

Large-scale streaming input is an essential issue for intelligent Web recommendation service nowadays. In this type of application, recommender systems handle sequential streaming inputs from users requests with different contexts. Whereas, the traditional batch learning and offline machine learning approaches could not deal with dynamic real time scenarios in a practical large scale system. Fortunately, the era of online learning approach under theoretical guarantee which provides a promising adaptive approach for making decision in unknown environment. The learning process helps the agent to update knowledge from current and past observations. One example is Yahoo! Today module website [4, 5] which is running online algorithms to show suitable highlight news for users. The goal of recommender system is achieving as high as possible click through rate from their suggested highlight stories.

Many online learning proposals invest in the general decision making framework Multi-Armed Bandit (MAB). This framework could provide a robust solution in intelligent real time systems with variant settings based on unknown environment. Using this approach, we can design a recommender agent for suggesting good articles or products to users that can improve user click through rate (CTR) or user purchase.

## 2. Preliminaries

Traditional recommender system such as Content-based filtering [2] locally recommends users based on matching user profiles with item properties. Another approach is collaborative filtering [3] which discovers similarity patterns across users or items based on history purchase information. These recommender methods require high computational cost and cannot be efficiently used for real time streaming input service of millions of users.

**Online learning** [1] is considered as a repetitive game that the agent has to make decision based on sequential one by one data input. This process learns rewards from feedback of users which are clicked/not clicked in recommender application.

**Multi-Armed Bandit framework (MAB):** In this framework setting, we have multiple actions along with unknown reward distribution function of environment. At each time, we can only choose one action to perform and observe reward from that action. With the unknown feedback function follows stochastic process, the setting has name stochastic multi-armed bandit problem [6]. The goal of the agent is maximizing the total reward from decisions. Some context-free algorithms in online learning with MAB framework are **ε-greedy** [6]**, UCB1** [6]. For an efficient recommender system, valuable user contexts are grouped based on similar interests. Recommender agent treats the same for each user in a group.

In this work, we refine the original offline k-means clustering technique for online clustering. After grouping user contexts, we apply existing MAB online strategies for individual clusters. These strategies have cheap computation, scalability and can be implemented distributed learning process for each cluster.

## 3. Problem Formulation:

### a) Contextual Multi-armed Bandit Setting

We define set $\mathcal{A}$ is the set of news, set $\mathcal{C}$ is the set of users context. A contextual-recommender agent will interact with user and update algorithm models through **T** discrete trials:

i. Agent receives context vector (feature vector) $x_t$ from context space $\mathcal{C}$. Based on context $x_t$, algorithm **A** will make decision for displaying a news $a_t$ to user.
ii. Agent observes reward $r_{t,a_t}$ which is clicked or not from unknown distribution of user feedback.
iii. Agent updates its strategy and parameter model to improve for the next prediction.

Expected total regret of contextual MAB setting is defined after **T** trials [2, 3] to measure the different between the best fixed total expected reward with total reward of algorithm A:

$$R_{\mathbf{A}}(T) = E\left[\sum_{t=1}^{T} r_{t,a_t^*}\right] - E\left[\sum_{t=1}^{T} r_{t,a_t}\right]$$

### b) Online Clustering Multi-Armed Bandit (OC-MAB)

According to algorithm 1, we denote each cluster has a representative context $C_i$ where $i$ is the index of context cluster. Cluster $C_i$ holds the summary information such as the number of context $n_i$ belong to it. Based on representative $C_i$ and number of context $n_i$, the algorithm try to find out the closest cluster depends on Euclidean distance between context $x_t$ and $C_i$. Then clusters are restructured and updated from incoming context. Besides, each cluster maintains its individual online learning process from list of news. Each news $a$ in cluster $C_i$ has $n_{a,i}$ number of selected times and cumulative reward $r_{a,i}$.

After searching the closest clusters in cluster update step using Euclidean distance, the centroid center of that cluster is updated based on the current center and number of contexts in the cluster. Then each cluster runs exploration and exploitation stage which actually provide online learning ability. In this work, we use two well-known strategies: **ε-greedy** [6] and **UCB1** [6].

**Algorithm 1** Online Clustering Multi-Armed Bandit (OC-MAB)

```
Initialization: Number of cluster is $K$
Create news lists for each cluster $C_i$
Input the first $K$ contexts as cluster representatives $C_i$
Running: Context $x_t$ from user come
    Update Cluster: $C_i^* = \arg\min_{C_i \in \mathbf{C_K}} \|x_t - C_i\|$
        Assign $x_t$ to $C_i^*$
    Update Centroid: $C_i^* = \frac{x_t + n_i C_i^*}{n_i + 1}$
        $n_i = n_i + 1$
    Exploration – Exploitation process:
        ε-greedy / UCB1 strategy
```

For **ε-greedy** strategy, it runs a Bernoulli randomized selection with probability **ε** for exploration and **1 - ε** for exploitation. In exploitation stage, agent selects the best expected number of clicked news $\mu_{a,i}^*$. While using **UCB1** strategy, agent selects the best expected number of clicked news plus an uncertain amount. This strategy is an optimistic strategy that chooses the possible best news in the future with upper confident bound of each news:

$$\hat{\mu}_{a,i} = \mu_{a,i} + \alpha\sqrt{\frac{2\ln n_i}{n_{a,i}}}$$

In Fig. 1, at time t contexts space has 4 existing clusters and agent receives context $x_t$. Based on the distance to cluster centers, the closest cluster is $C_2$. User with context $x_t$ is assigned to cluster $C_2$ then updating cluster center $C_2$ to new center $C_2^{'}$. Due to only one cluster is selected at each round, cluster $C_2$ run exploration-exploitation process to decide which news to display. Finally, news B is chosen to display for user at this round.

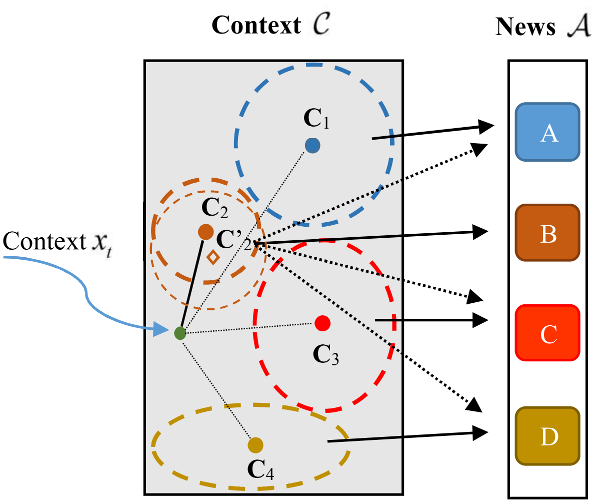

*Figure 1: OC-MAB learning with 4 context clusters* $C_1$, $C_2$, $C_3$, $C_4$ *and 4 news A, B, C, D*

### 4. Numerical results

**Simulation environment:** User context is randomly generated with 50 binary features. In the simulation, we generate 2000 user contexts sequentially arrive at agent. In order to simulate unknown randomized user feedback, we use two different kind of distributions are Bernoulli and Gaussian. Based on the assumption that users within a cluster has the same interest on news. The offline k-means is applied to cluster similar user into 10 clusters due to news pool has 10 news. Each users cluster is interested in one news more than others. The more interest user the higher probability user clicks on one news. Therefore, the unknown distribution of clicked news rewards are controlled by high probability click for Bernoulli distribution or high mean value of click in Gaussian distribution.

For every incoming context, agent decides to display one news and observe user will be click or not on that news. In order to select news, agent uses algorithm OC-MAB clustering user contexts and online learning from user feedback. User feedback which is generated by Bernoulli distribution with probability for the most interesting news of each cluster is 0.8 and the remaining news are 0.3. While user feedback which is generated by

Gaussian distribution has mean 0.8 for the most interesting news and 0.3 for the remaining ones.

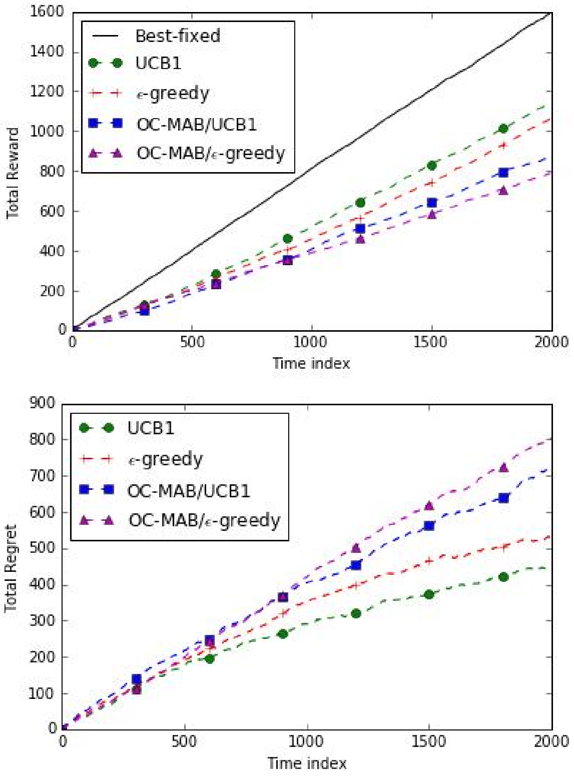

*Figure 2: User feedback follow Bernoulli distribution*

From simulations in Fig. 2, 3, the best-fixed line is the baseline to measure performance of algorithm. **ε-greedy** and **UCB1** strategy with known clusters from offline k-means give better total reward and have sublinear curves in total regret as we expected. UCB1 strategy gives the best reward and the lowest total regret after expense high exploration cost in the beginning. However, this setting is impractical due to we don’t know all contexts before it arrives. After receiving 2000 contexts, OC-MAB strategy agent has the total reward more than 800 clicked events. It achieves around 40% of the number of the total access and 50% of the best-fixed reward. UCB1 strategy gives a little better total reward than ε-greedy. Total regret still forms linear function of **T** in OC-MAB. This fact due to applying online manner only approximates offline K-Means context clusters. There are small gaps between centers of online manner and offline K-Means.

## 5. Conclusion

In this paper, OC-MAB proposes integrated clustering context of users to select news or actions which can be useful for recommender systems. The algorithm provides the scalability of user contexts and cheap computation in each iteration without recording the whole history contexts and user feedback. Based on very limited information, OC-MAB can reconstruct clusters and perform adaptive learning in dynamic environment as user click/not click news.

In the future work, we continue analyzing online manner to achieve closer offline clustering results. The smaller distance between centers could produce better the total number of user clicks and lower total regret.

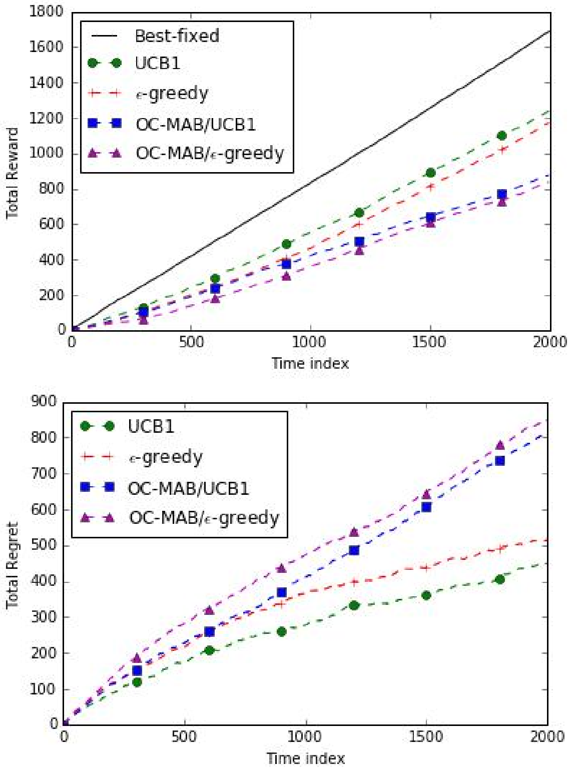

*Figure 3: User feedback follow Gaussian distribution*

## 6. Acknowledgement

"This research was supported by the MSIP, Korea, under the G-ITRC support program (IITP-2016-R6812-15-0001) supervised by the IITP." *Dr. CS Hong is the corresponding author.

### References:

[1] Shalev-Shwartz, Shai, and Shai Ben-David. *Understanding machine learning: From theory to algorithms*. Cambridge University Press, 2014.

[2] Mladenic, Dunja. "Text-learning and related intelligent agents: a survey." *IEEE Intelligent Systems* 4 (1999): 44-54.

[3] Schafer, J. Ben, Joseph Konstan, and John Riedl. "Recommender systems in e-commerce." *Proceedings of the 1st ACM conference on Electronic commerce*. ACM, 1999.

[4] Li, Lihong, et al. "A contextual-bandit approach to personalized news article recommendation." *Proceedings of the 19th international conference on World wide web*. ACM, 2010.

[5] Li, Lihong, et al. "An Unbiased Offline Evaluation of Contextual Bandit Algorithms with Generalized Linear Models." *Proceedings of Workshop and Conference*. JMLR, 2012.

[6] Auer, Peter, Nicolo Cesa-Bianchi, and Paul Fischer. "Finite-time analysis of the multiarmed bandit problem." *Machine learning* 47.2-3 (2002): 235-256.