Content Creation Competition In Online Recommender Systems

Pawan Jayakumar
pj8wfq@virginia.edu
University of Virginia
Charlottesville, Virginia, USA

Ho Yeon Jeong
hj9yzk@virginia.edu
University of Virginia
Charlottesville, Virginia, USA

Alan Zheng
az4xfp@virginia.edu
University of Virginia
Charlottesville, Virginia, USA

Abstract

Content on recommendation platforms (Youtube, TikTok, etc.) is quickly dominating the attention and pastimes of young adults and children. Creators are constantly faced with the dilemma of what content to produce to best capture views on their platform. Their behavior as a collective and its effect on the user experience are still unanswered questions. This project attempts to construct a simulation for this problem. We found that the EXP3 adversarial bandit algorithm performs significantly better than other bandit algorithms.

Keywords: EXP3, UCB1, Recommendation Systems, Reinforcement Learning

1 Introduction

In this simplified simulation of social media recommendation system environment, there are K possible content types M creators could create for a population of N users. The utility of a content type to a user is represented as a cell in the utility matrix $A \in R^{K \times N}$. This is not known to the creators. At every round t, creators choose a piece of content from K. All of the chosen content is represented by the set C_t. Each user then gets one piece from C_t given by the recommender system based on which piece would give them the most utility. Ties are broken by random choice. At the end of the round, every creator is informed on the number of users who consumed their content. The choice of content to create can be represented as a multi-armed bandit problem and their strategic behavior is thus modeled with the EXP3 algorithm which uses the number of views as the reward. The utility matrix is drawn uniformly from $[0, 1]$. Finally, user welfare over t rounds is defined as

$$\text{Welfare}_t = \sum_{t=1}^{T} \sum_{j=1}^{N} max_{k \in [C_t]} A_{kj}$$

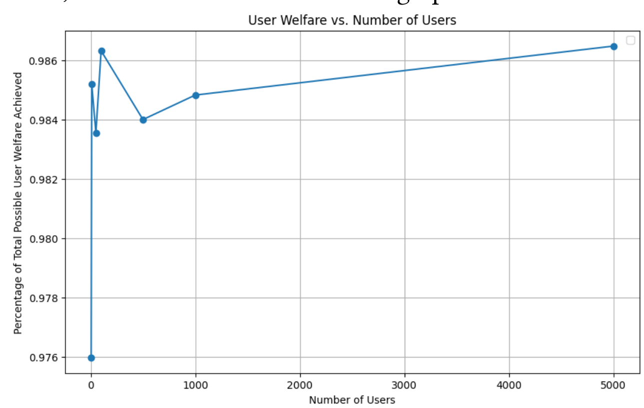
2 Related Work

Our experiment uses the Exp3 algorithm for recommending content. Exp3 stands for Exponential-weight algorithm for Exploration and Exploitation. It works by maintaining a list of weights for each of the actions, using these weights to decide randomly which action to take next, and increasing (decreasing) the relevant weights when a payoff is good (bad). There is also an exploratory factor γ which tunes the desire to pick an action uniformly at random.

3 Experiments

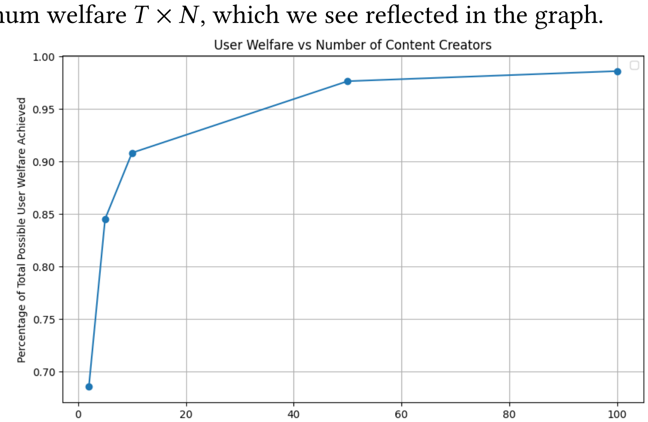
Changing N

Welfare is expected to scale similarly with N in the scenario where EXP3 is able to find the optimal content to create at any number of users. Linearly increasing N simply linearly increases the times we sum up utility multiplied by content, which we see reflected in the graph.



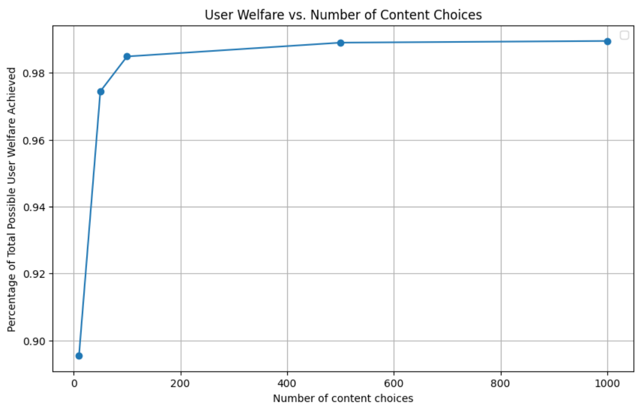
Changing M

User welfare is expected to be maximized if there are more content creators to supply their needs, meaning increased chances of higher utility content available. Thus, increasing M should asymptotically approach the boundary of maximum welfare $T \times N$, which we see reflected in the graph.



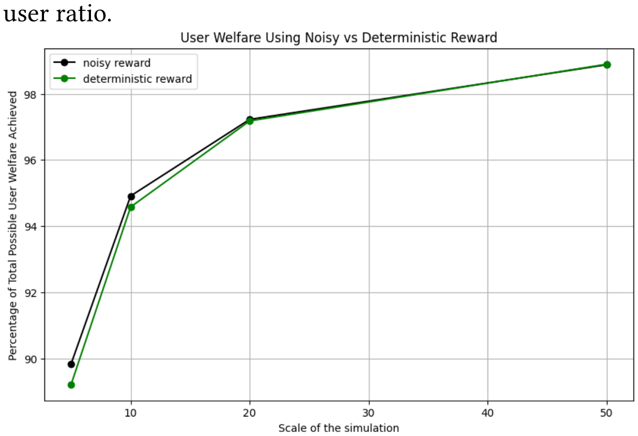
Changing K

Finally, user welfare should also be maximized if there are more content options. Essentially, if there are more content options, it is more likely that a user will have some sort of content they really like that EXP3 can appeal to. Again, increasing K should asymptotically approach the boundary of maximum welfare $T \times N$, which we see in the graph.



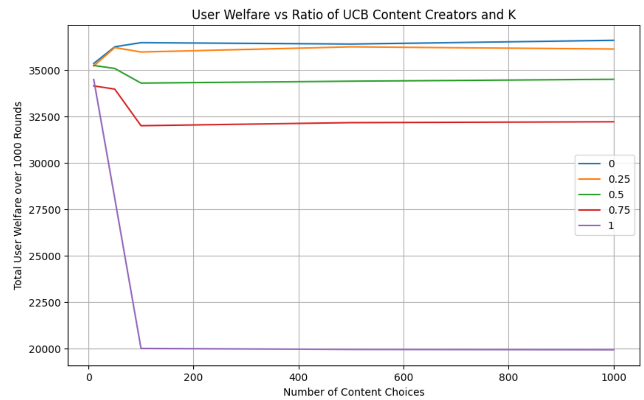
Noisy Utility

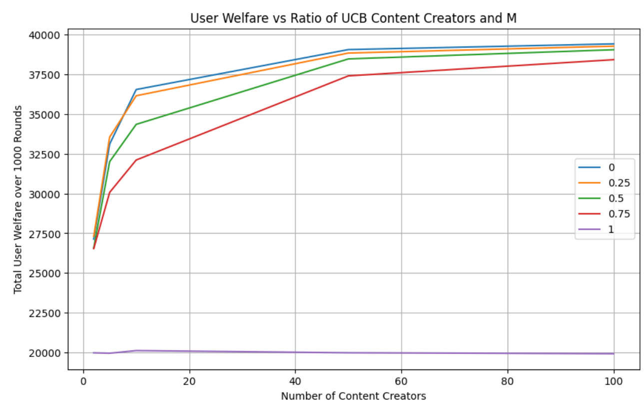
Instead of giving creators information based on the number of users matched to them, a noisy version of the utility of the content was presented to them. Comparing this to the default system demonstrated what information is most valuable for creators to create the best content for users. In 500 trials using $T = 100, K = 20, N = 10$, and $M = 400$, an average difference in welfare of only .944 was found. Compared to the welfare of a round of $T \times N = 2000$, there is an extremely negligible difference. In the graph below, multiple scales of the simulation were also tried. In the x-axis, scale describes a simple factor multiplied to a base ratio of K:M:N. This ratio is 2:1:40, which is derived from Youtube's creator to user ratio.

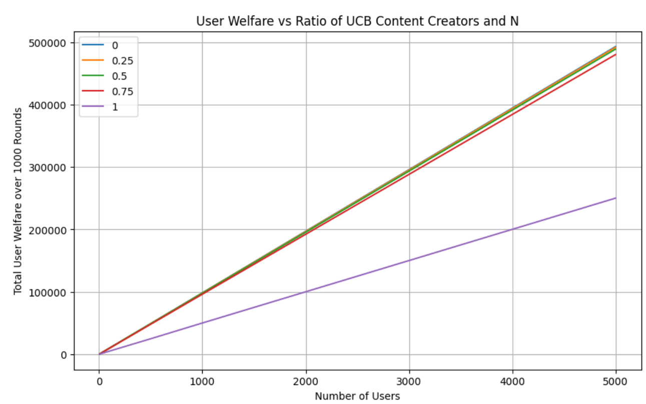


Overall, both rewards allow content creators to produce the optimal content, maximizing user welfare.

Adding UCB1 Creators By default, the simulation assumes all content creators follow the EXP3 algorithm to decide which content to use. EXP3 is designed to perform well in an adversarial multi-arm bandit setup, which this problem is modeled as. In contrast, UCB1 is not designed to handle such an environment. This becomes evident in the results of the simulation, where we ran the simulation with varying fractions of creators using the UCB1 algorithm instead. As the graph shows, the total user welfare goes down as the number of content creators who use UCB1 over EXP3 goes up. Interestingly, as long as some content creators use EXP3, the performance isn't significantly worse, but there is a huge drop off when all content creators use UCB1. This conclusion held when varying K, M, and N.

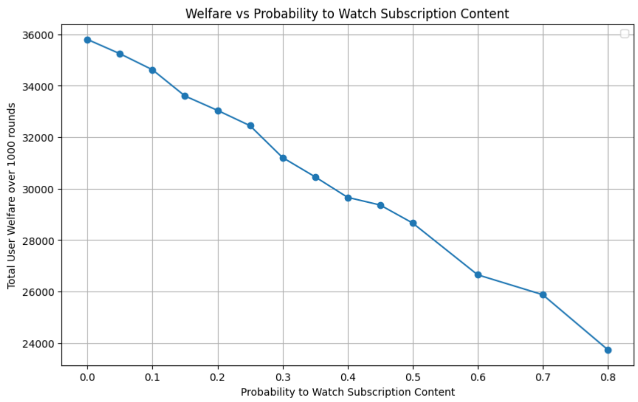




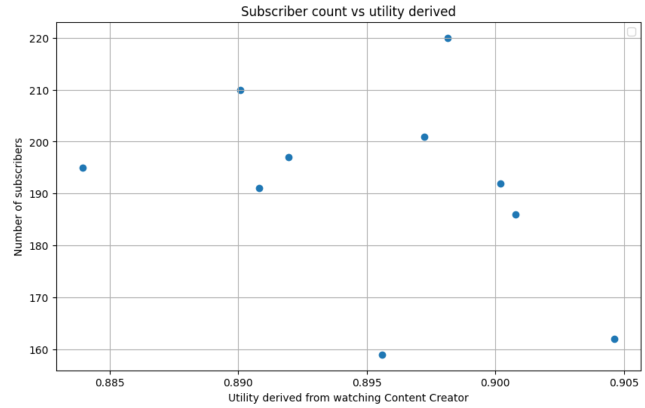


Subscription Mechanism

To further explore the problem, the simulation was adapted to further mimic a real content platform. A core feature of these platforms is some kind of subscription functionality. If the viewer enjoys the content enough (i.e. their utility for the content is high enough) and they view this content from the same creator enough times, they will subscribe to them. Once subscribed, the probability that the user would choose to view subscribed content over the system's recommendation was modulated using a hyper-parameter. As the probability increased, user welfare decreased.



This is to be expected as viewing subscribed content leads to viewing sub-optimal content as the algorithm is already almost perfect at matching content to users.



Another interesting result was that there was no correlation between the number of subscribers a content creator had and the average utility viewers received from watching them. This is not reflected in the real world because users don't exactly know the utility they have for types of content and users are not independent (friends tend to watch similar content).

4 Limitations and Future work

Due to computational constraints, there were many simplifications made to allow us to explore this environment. For example, the values of n, m, and k are much higher in the real world. Also, users are not independent actors because they exchange information about content with each other. Further work can model this behavior using contextual bandits.