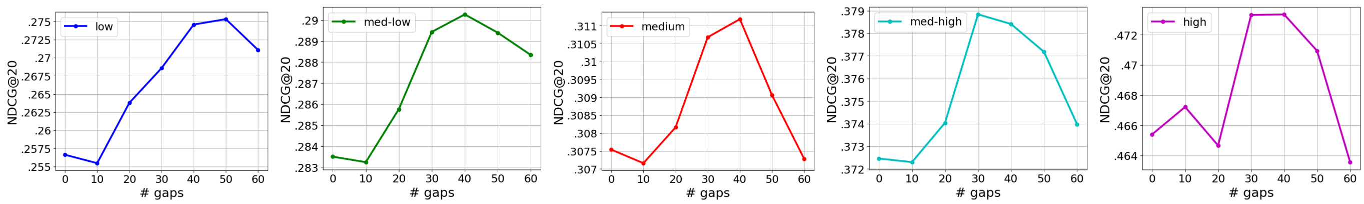
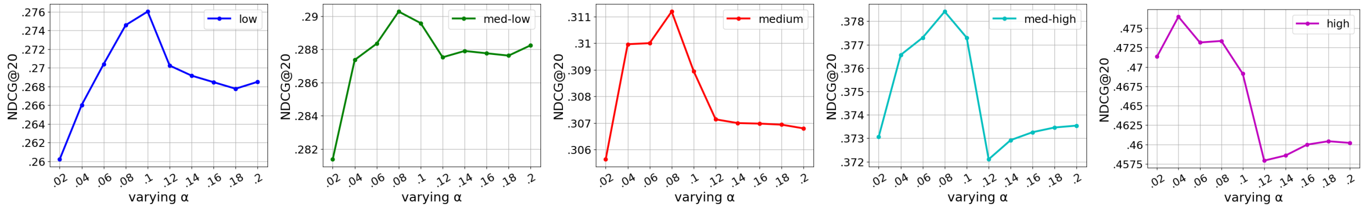


(a) low (b) med-low (c) medium (d) med-high (e) high

Figure 4: Performance of TALL on five subgroups of users on ML1M dataset with varying *#gaps*.



(a) low (b) med-low (c) medium (d) med-high (e) high

Figure 5: Performance of TALL on five subgroups of users on ML1M dataset with varying α.

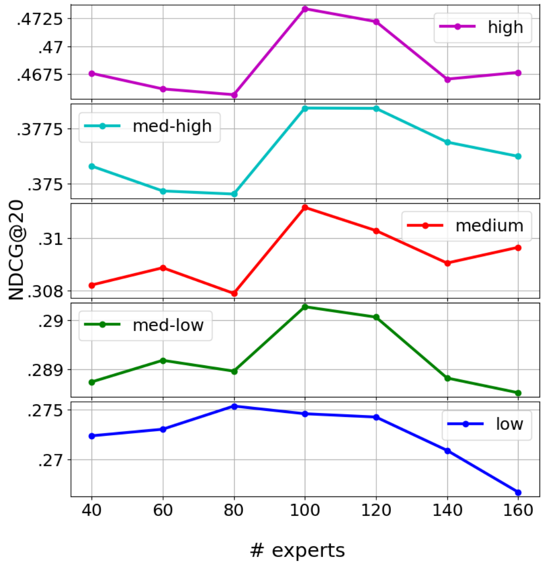


Figure 6: Performance of TALL on five subgroups of users with varying *#experts*.

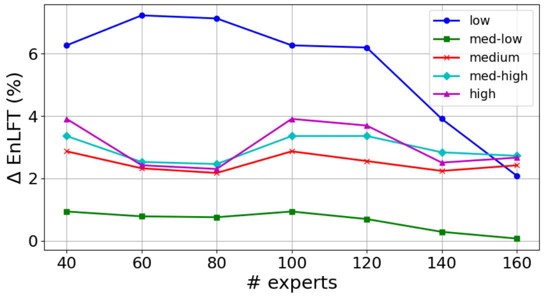


Figure 7: Compare TALL with EnLFT with varying *#experts*.

A HYPER-PARAMETER STUDY

Here, we study the impacts of three hyper-parameters in TALL: (1) the gap window in the adaptive weight module; (2) α in the adaptive weight module; and (3) the number of experts.

A.1 Number of Epochs in Gap Window

The goal of the gap mechanism is to avoid the unstable loss problem at the beginning of model training. Thus, a small gap window could lead to unstable training, while a large gap window results in insufficient epochs of applying the adaptive weight, degrading the effectiveness. Here, we run TALL on the ML1M dataset with the number of epochs in the gap window (denoted as *#gaps*) varying in {0, 10, 20, 30, 40, 50, 60} and other settings the same as in Section 4.2. The overall NDCG@20 results for five subgroups are presented in Figure 4. Our findings show that the utility first increases and then decreases along with increasing *#gaps* for all five subgroups. Additionally, we can observe that the optimal *#gaps* to achieve the best NDCG@20 vary for different subgroups. Specifically, the niche users show the best performance when *#gaps* is around 50, and the mainstream group exhibits the best performance when *#gaps* is around 30. This is because mainstream users are easier to learn and more quickly enter a stable training phase than niche users, also empirically verifying the problem of unsynchronized learning we observed in Figure 1. Although the peaks are different for different user groups, we can take $\#gaps = 40$ for TALL to deliver an overall effective result for all users.

A.2 Strength of Adaptive Weight

This hyperparameter α in Equation 3 controls the strength of the adaptive weight module: a small α leads to a stronger adaptive weight module that assigns higher weights for users with large loss changes. In other words, a small α makes the model devote more effort to calibrating the learning paces of different users. In this experiment, we vary α from 0.02 to 0.2 with step 0.02 and evaluate the average changes in NDCG@20 for five subgroups in

Figure 5. In this figure, we can see that while increasing the value of α, NDCG@20 initially increases for all five subgroups, reaches a peak, then decreases to converge. The peak performance for each subgroup is achieved at different values of α: the niche user group achieves peak when α is around 0.1, and when α is around 0.04, the mainstream group achieves the best performance. These findings are consistent with the observations from Section A.1, indicating that different types of users have distinct behaviors and characteristics during the training of a model. Empirically, we can take $\alpha = 0.08$ for TALL to deliver an overall effective result for all users.

A.3 Number of Experts

Last, we study how the number of experts in TALL influences the performance, and whether TALL possesses a consistent advantage over baselines when the number of experts changes. Generally, within a reasonable range, more experts lead to larger model complexity, resulting in better performance. Here, we conduct experiments on TALL and vary the number of experts (denoted as $\#experts$) in {40, 60, 80, 100, 120, 140, 160} shown in Figure 6. We can observe that the performance of TALL first increases and then decreases with larger $\#experts$, and the peak performance is achieved when $\#experts$ is around 100 for all types of users.

Besides the experiment above, we also conduct the experiment with the number of experts (denoted as $\#experts$) varying in {40, 60, 80, 100, 120, 140, 160} on the local learning baseline EnLFT, which performs the best among all baselines with the hyperparameter $\#experts$ on the ML1M dataset according to Table 2. We show the improvement rate of our TALL over EnLFT for all five user groups with different $\#experts$ in Figure 7. We observe that TALL consistently outperforms EnLFT for all user groups with different $\#experts$. However, we see that the improvement rate decreases when $\#experts$ increases for niche users ('low' and 'med-low' groups), while the improvement rate maintains a similar lever for other users. This is because niche users can benefit from higher model complexity in the baseline model, but the novel design of the proposal TALL can still perform better than EnLFT with larger model complexity.