Beyond the Content: Considering the Network for Online Video Recommendation

Lihui Lang
Hunan University

Meiqi Hu
Hunan University

Changhua Pei
CNIC-CAS

Guo Chen*
Hunan University

ABSTRACT

Online recommendation systems play critical roles in enhancing user experience by helping them find the most interesting videos from a vast amount of content. However, the existing recommendation modules and video transmission modules in the industry often operate independently, resulting in the recommendation model providing some videos that cannot be transmitted within the specified deadlines successfully. This can lead to an inferior watching experience for users and resource waste for video providers. To address this, we propose a novel framework called *NetRec*, which for the first time optimizes the recommendation quality by jointly considering the network transmission. We accomplish this by re-ranking the top-N videos obtained from the recommendation system and selecting the top-M (M is approximately half of N) videos that provide the maximum overall revenue, *e.g.*, video playing time while considering the network status. The entire system comprises network measurement, video quality estimation, and multi-objective optimization modules. Real-world Internet results show that our framework can increase users' video playing time by 20% to 160%. Furthermore, we provide several promising directions for further improving the video recommendation quality under our NetRec framework, which jointly considers the network for the recommendation.

CCS CONCEPTS

• **Networks → Application layer protocols**; *Network measurement*.

KEYWORDS

Network measurement, multi-objective recommendation, video quality estimation

ACM Reference Format:
Lihui Lang, Meiqi Hu, Changhua Pei, and Guo Chen. 2023. Beyond the Content: Considering the Network for Online Video Recommendation. In *7th Asia-Pacific Workshop on Networking (APNET 2023), June 29–30, 2023, Hong Kong, China.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3600061.3600075

*Corresponding author

APNET 2023, June 29–30, 2023, Hong Kong, China

ACM ISBN 979-8-4007-0782-7/23/06…$15.00
https://doi.org/10.1145/3600061.3600075

1 INTRODUCTION

Online short videos (TikTok, Shorts, *etc.*) have begun to occupy the majority of the Internet traffic [2] and proven to have significant commercial value [20]. Existing short video platforms have implemented recommendation systems to encourage users to spend more time on them [6]. Video recommendation models have shown strong abilities to better match user interests and increase platform revenue over the past decade. However, improving the recommendation model accuracy is now facing the problem of diminishing marginal utility [16]. In this paper, we point out that considering users' network status when recommending videos and taking into account both the probability of successful transmission *and* the video content is a highly promising direction to further improve the video recommendation revenue[1]. We find that even with commonly used algorithms, considerable revenue gains can be achieved, which is difficult to achieve solely relying on existing recommendation models.

Specifically, for online short video platforms [6], users may play a video for a while if they are interested in it, or just quickly "swipe" to another one if they are not attracted by the video. The recommendation system helps the server to dynamically choose several possibly attractive videos according to the users' real-time feedback and transmit them to users through the network. These recommended videos need to be transmitted before a very short deadline (before the previous one is swiped), otherwise, if exceeding the deadline, users can only swipe to some locally-cached videos which may be much less attractive and users may leave. For example in Kuaishou [5] (one of the largest short video platforms in China), a server tries to recommend and transmit 6 videos to a user within a 3-second deadline.

Due to the volatility of network status, however, not all the recommended videos may be successfully received by the user. Fig. 1 shows that if not considering the network, the existing recommendation model may lose more than 90% video playing time under poor network conditions (experiment details in §4). Although there are previous works [1, 4, 8, 12] attempting to accelerate the video transmission by dynamically choosing the video bitrate based on the user's network status, they rarely consider the selection of videos. Particularly, since the video sizes may vary, even a video that may be more attractive according to the recommendation model may not be able to be transmitted to the user due to the poor

[1] In practice, the recommendation revenue is evaluated as the total video viewing time of all users [5].

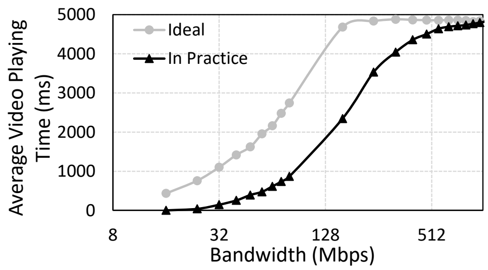


Figure 1: Average video playing time of an existing recommendation model which does not consider network status (in practice) and an ideal one that jointly considers network status (ideal).

network. In this paper, we will, for the first time, focus on how to select videos that can better balance video attractiveness and the transmission success rate to improve the overall recommendation revenue.

We propose a novel recommendation framework called **NetRec**, which works on the server-side along with the existing recommendation model but jointly considers network status. First, NetRec applies a passive bandwidth measurement system and monitors the previous round's video download bandwidth in real time as the input for the next round's algorithm. Second, we design a multi-objective video selection module based on the knapsack algorithm [15] to re-rank the initial video list given by the recommendation model under the constraints of network bandwidth. We evaluate the effectiveness of our framework based on a real-world open source dataset from a top-tier short video provider [5]. The experimental results show that our algorithm achieves significant gains both in the lab, on campus, and on the Internet. Real-world Internet results show that NetRec can increase users' video playing time by 20% to 160% even only using simple algorithms. NetRec has shown the possibility of a novel and important direction for network-and-recommendation joint optimization. Moreover, through both theoretical and experimental analysis, we show that there are rich opportunities to achieve even better performance in this direction, including more accurate network status prediction and advanced selection algorithms that are robust to network fluctuations.

2 RELATED WORK

Recommendation Algorithms: There are a lot of algorithms mainly focusing on recommending the most attractive videos, through optimizations methods based on click-through rate [7, 10], graph learning [11], causal inference [3, 20], debiasing [14], and collaborative filtering [13, 19], *etc.* However, they have not considered the network status for transmitting the recommended videos.

Video Bit Rate Adjustment: Previous works [1, 4, 8, 12] adaptively adjust the video quality according to the network status during video playing. They may improve users' viewing experience, however, they do not affect which videos would be recommended to the user.

Network Bandwidth Estimation: Network bandwidth estimation is commonly used to optimize transport layer protocols by adjusting congestion control algorithms based on the current network condition [17, 18]. [9] also uses bandwidth estimation in video transmission to cut down unnecessary video stalls and quality drops. However, none of them uses network information for video selection.

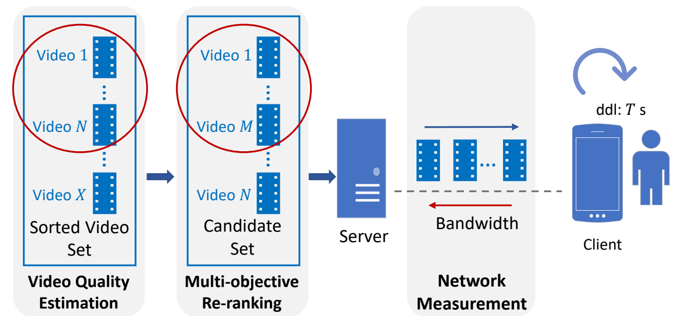


Figure 2: Overview Design.

3 SYSTEM DESIGN

3.1 Overview

We propose a framework named *NetRec* which integrates network information into a recommendation system for better user engagement. NetRec leverages network information such as bandwidth to calculate the remaining transmission time for each video. Knowing the transmission time of each video, the server re-ranks the initial list given by the prior recommendation algorithm to select M new videos which not only have higher estimated playing time but also can be successfully delivered to users within a deadline T, thus achieving better performance.

Our design is divided into three main parts, as shown in Figure 2. The first part is **video quality estimation**, which predicts the quality of videos. In this paper, quality is represented by the estimated playing time, but other metrics such as the probability of a user clicking the like button can also be used. In existing video recommendation systems, the top M videos with the highest estimated playing time are directly delivered to users. In our NetRec, we consider network status by expanding the video set from M videos to N videos and feeding the top N videos to the next step. The second part is **multi-objective re-ranking**, which takes the top N videos from the video quality estimation module as input. For each video in this phase, the remaining transmission time is calculated based on bandwidth information and video size, in addition to the estimated video quality. We propose a multi-objective re-ranking algorithm to select the top M videos whose total playing time is the largest under certain network conditions. Finally, the third part is **bandwidth measurement**, which measures dynamic network bandwidth and provides this information to the re-ranking module for better decision-making.

3.2 Video Quality Estimation

The primary objective of short video recommendation is to estimate video quality for a given user. Video quality reflects the user's preference for the video. There are various metrics to measure

Algorithm 1: Knapsack Algorithm

Data: N: candidate set size, T: transmission deadline, bw: network bandwidth, br: video bit rate, dur: video duration, $ePlayT$: estimated playing time
Result: $dpMaxVal[i][j]$ $(1 \le i \le N, 1 \le j \le T)$: max playing time when $video_i$ is selected under the condition that the total transmission time ≤ j

Function `Knapsack`(T, bw, br)**:**
  Initialization;
  for $i \leftarrow 1$ *to* N **do** $txTime[i] = \frac{dur[i] \times br[i]}{bw}$;
  for $i \leftarrow 1$ *to* N **do**
    for $j \leftarrow 1$ *to* T **do**
      $dpMaxVal[i][j] = dpMaxVal[i-1][j]$;
    for $j \leftarrow 1$ *to* T **do**
      $temp = dpMaxVal[i-1][j - txTime[i]] + ePlayT[i]$;
      if $txTime[i] \le j$ *and* $temp > dpMaxVal[i-1][j]$ **then** $dpMaxVal[i][j] \leftarrow temp$;
    end
  end
  return $dpMaxVal$

Algorithm 2: Multi-objective Re-ranking Algorithm

Data: N: candidate set size, M: number of video to be selected, T: transmission deadline, bw: network bandwidth, br: video bit rate, dur: video duration, $ePlayT$: estimated playing time, $cand$: candidate set
Result: $ret[1...M]$: selected top M videos

Function `videoSelection`$(cand, T, bw, br, M)$**:**
  $dpMaxVal \leftarrow Knapsack(T, bw, br)$
  for $i \leftarrow N$ *to* 1 **do**
    if $dpMaxVal[i][j] \neq dpMaxVal[i-1][j]$ **then**
      ret.insert (i) ;
      $cnt \leftarrow cnt + 1$;
      $j \leftarrow j - cand[i].txTime$;
  end
  if $cnt < M$ **then**
    choose Top$(M - cnt)$ videos with larger $ePlayT$ in $cand - ret$;
  else if $cnt > M$ **then**
    choose Top-M videos with larger $ePlayT$ in ret ;
  return ret

video quality, such as estimated playing time, liking probability, or sharing probability. In this paper, we use estimated playing time as the metric to measure video quality. Numerous models have been proposed to estimate video quality accurately, taking into account video features and the user's viewing, liking, or sharing history. Training an accurate model is beyond the scope of this paper. Our NetRec framework can easily integrate with existing recommendation systems and reuse their video quality estimation models. We evaluated our framework on a real-world open-source dataset from a top-tier video recommendation platform [5] to provide video quality estimation results.

3.3 Multi-objective Re-ranking

The main goal of this sub-module is to re-rank the initial video list selected by the existing recommendation algorithm. Our re-ranking algorithm needs to work in a multi-objective manner, taking into account not only the video quality but also the likelihood of successful delivery to the user. The problem of the multi-objective re-ranking module can be formulated as follows:

How to select a set of M videos that can achieve the highest total video quality and be transmitted before strict T deadline under certain network bandwidth and video size?

Based on the above formulation, the problem can be regarded as the optimization problem and be solved using Knapsack algorithm [15], which is shown in Algorithm 1. We iterate through each video and consider two choices when putting it into the knapsack: either to put it or not to put it. We update the $dpMaxVal$ with the optimal choice between these two options. The total re-ranking algorithm is shown in Algorithm 2.

Scenario 1: When $x < M$, NetRec picks several videos $(M - x)$ with the highest estimated playing time from the remaining videos that are not picked by the Knapsack Algorithm to fill up to M videos.

Scenario 2:When $x = M$, NetRec has already obtained the optimal combination of N videos.

Scenario 3: When $x > M$, NetRec re-sorts the selected videos and chooses the top M videos with the longest estimated playing time.

3.4 Network Measurement

In order to provide accurate network information for re-ranking algorithms to make better decisions, continuous measurement of real-world network bandwidth is necessary. To minimize measurement computation burden and traffic, we propose a passive monitoring method that estimates bandwidth by tracking total transmitted bytes and the time spent on transmission in a specific user request. The result is recorded and utilized for future user requests. The passive measurement approach offers two key advantages: firstly, there is no need for additional measurement traffic or software, and secondly, the resulting bandwidth estimate is an accurate reflection of the achievable bandwidth of the link with the video provider.

3.5 Video Fetching and Feedback Collection

After applying the multi-objective re-ranking module, the top M video IDs are sent to the client to retrieve the content from the service providers or nearby content providers. We implement a timeout mechanism at the beginning of each request, and any video fetched beyond the deadline (*e.g.*, 3 seconds) is not displayed to the user. The real playing time of the displayed videos is recorded as user feedback to evaluate the effectiveness of our model. Notably, this feedback can also be utilized to train the video quality estimation models.

4 EVALUATION

In this section, we aim to answer the following research questions through various testbed experiments.

- RQ1: Can NetRec improve the overall recommendation revenue in different network conditions?
- RQ2: How does the bandwidth estimation algorithm affect the overall performance?
- RQ3: How does the multiobjective re-ranking algorithm affect the overall performance?
- RQ4: How does the accuracy of the content recommendation algorithm affect the effectiveness of NetRec? Can NetRec still be effective as the content recommendation algorithm becomes increasingly accurate?

Dataset. To realistically evaluate the effectiveness of NetRec, we use a publicly available dataset from Kuaishou [5], which records the information of over 10K videos watched by millions of users in one of the largest online video providers in China during two months (from July 5, 2020 to September 5, 2020). For each video, the dataset records its original time length ($videoDuration$), the time watched by a user in each recommend ($playingTime$). The total playing time ($playingDuration$) and the total number of displays ($showCnt$) during the measured period are provided. The estimated playing time of a recommendation model can be approximated by $\frac{playingDuration}{showCnt}$.

Metric. We evaluate the total video playing time as the evaluation metric, called the *revenue*. Specifically, *revenue* is calculated by summing the playing time of all the videos that are recommended and successfully transmitted to the user through the network, during each evaluation round.

Schemes Compared. We compare NetRec with the traditional mechanism that only considers the video content in recommendation (we'll refer to it as the baseline in the rest of the paper). Traditional recommendation systems select N videos as candidates and recommend the top M with the longest estimated playback duration ($estimatedPlayingTime$) without considering network conditions. $estimatedPlayingTime$ is typically calculated by dividing its $playingDuration$ by its $showCnt$.

Testbed Setup. Figure 3 displays the process of evaluation. When the client requests the video from the server, it attaches the network information and starts a timer. After the request arrives at the server, the *userID* and network information are recorded. The server selects the videos to be transmitted according to the network information and the existing information of the video. In our experiment, We count the playing time of the video received before the timer expires. In our experiment, we used three servers as the server-side and two regular computers as the client-side. Because every user in the data set watches more than 2k videos, we can get a fair evaluation even if we only choose one user. So we randomly select users from the dataset for our evaluation, and the selected user watches a total of 3234 videos in the dataset. We select 12 videos as candidates for each request and for each round we send 269 requests to measure the revenue under NetRec and Baseline. All the results below are collected from more than 300 rounds of experiments conducted from Dec. 2022 to Mar. 2023.

Network Environments. The experiments are done under three network environments. **1) Lab LAN:** Both the server and the client are deployed in a small laboratory network, and they are directly connected through 1 Gbps cables and several switches. **2) Campus Network:** The server and client locates in a college campus network that has about 40K users. The network contains about 10K wired and wireless switches and routers. The server accesses the campus network through a 1 Gbps cable, and the client accesses the campus network through a WiFi 5 wireless router. **3) Internet:** The server and the client locates in Nanjing and Changsha, respectively, which are two cities in different provinces in China. They communicate with each other through the network offered by ISPs in China. The server uses wired cables as the access link, and the client is accessed through home wireless routers.

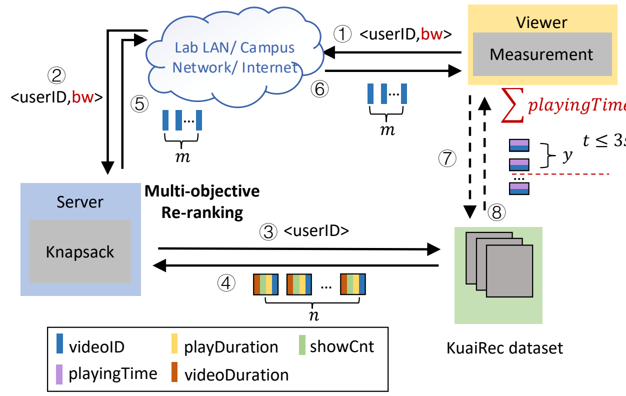
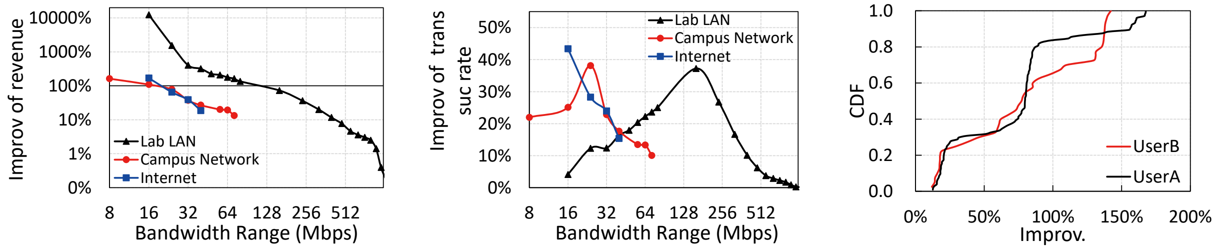


Figure 3: Testbed Setup.

4.1 RQ1: Overall Performance

Figure 4(a) shows the revenue improvement under various network environments, compared with traditional recommendation systems, as the actual network bandwidth between the server and the client varies. NetRec improves the revenue by 109.29%-12357.31%, 13.26%-1545.86% and 0.18%-72.82% when the network bandwidth is 0-2 MBps, 2-10 MBps and 10-150 MBps, respectively. Generally, it improves the revenue more when the network bandwidth is small and improves less when it grows higher. This is because when there is enough network bandwidth, the recommended videos will have greater chances to be transmitted successfully, even if the network condition has not been considered during the recommendation. However, when the network is slow, it needs to carefully consider the transmission condition when recommending videos, otherwise, the videos may not be received by the client due to the network limit. We output the improvement of the successful transmission rate of our algorithm relative to the baseline, as shown in Figure 4(b). It can be seen that regardless of the network environment and bandwidth, our algorithm can effectively reduce the timeout ratio. A lower timeout ratio means that our NetRec can ensure that more videos are successfully presented to users. This also explains why our algorithm can generate higher revenue compared to the baseline.

Note that even though the measured network bandwidths are similar, the improvement degree may be different under different network environments, due to the bandwidth fluctuations in actual networks.



(a) Revenue Improv. (b) Trans. Ratio Improv. (c) Revenue Improv. for Different Users

Figure 4: Improvement of revenue 4(a) and improvement of successful transmission ratio 4(b) under vairous network environments, compared with traditional recommendation systems, as the actual network bandwidth between the server and the client varies. 4(c) shows the detailed revenue improvements under campus network of two randomly selected users from the dataset, where each user has viewed over 2000 videos.

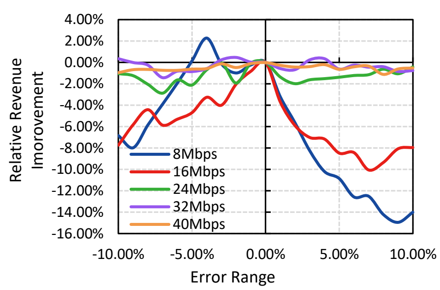
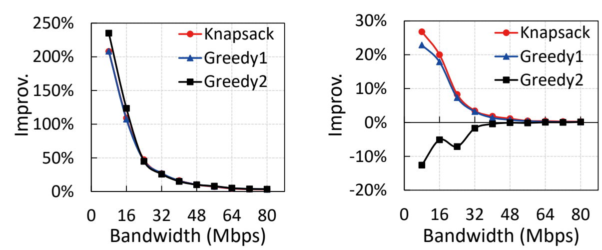


Figure 5: Revenue changes versus error-free scenario under different bandwidth estimation error ranges and network conditions.

To demonstrate that the revenue improvement of NetRec is not due to the selection of a user dataset that is more advantageous to us, we also randomly selected another two users who watched more than 2K videos during the statistical period. As shown in Figure 4(c), NetRec is still to achieve good revenue despite using different users' datasets, with about 80% revenue improvement in mean for different rounds of evaluation.

4.2 RQ2: Bandwidth Estimation

Given a specific network bandwidth, we can accurately predict whether the Top-M videos selected by NetRec can be successfully transmitted. Therefore, to comprehensively evaluate the impact of the bandwidth estimation algorithm on the entire system, we analyzed the revenue under a range of network bandwidths when the estimation error of the algorithm was within ±10%. The experimental results are shown in Figure 5. As shown in the figure, when the bandwidth environment is better, such as 5MBps, the tolerance of NetRec to the bandwidth estimation method is high. Even when the bandwidth estimation error reaches 10%, the decrease in the performance is within 2%. However, as the network corrupts, the revenue of NetRec is more sensitive to the accuracy of the bandwidth estimation algorithm. It can be seen that when the bandwidth is 1MBps, a 10% estimation error will result in an up to 15% decrease in revenue. We also observed that overestimation errors are more likely to affect the revenue than underestimation.



(a) EstimatedPlayingTime. (b) RealPlayingTime.

Figure 6: Comparison of different re-ranking algorithms with different video quality estimation method (EstimatedPlayingTime or RealPlayingTime).

Implications: Further improving the accuracy of bandwidth estimation is valuable, which may bring ~5% higher recommendation revenue if the bandwidth estimation error is reduced from 10% to 5%.

4.3 RQ3: Multi-objective Re-ranking

The multi-objective re-ranking algorithm is the core part of our NetRec system. In addition to the knapsack algorithm used by NetRec, we propose two greedy algorithms for comparison. The first algorithm (Greedy1) sorts videos greedily according to $\frac{estimatedPlayingTime}{txTime}$, while the second one (Greedy2) only considers *txTime*, ignoring video quality. From Figure 6 (a), we observe that Greedy2 achieves better results than Greedy1, and Greedy1 performs similarly to our knapsack algorithm. We guess that this is because the *estimatedPlayingTime* used by the re-ranking algorithm for selecting M video is inaccurate. The inaccurate input limits the ability of the re-ranking algorithm to select better videos.

To verify our hypothesis, we replace *estimatedPlayingTime* with *realPlayingTime* in NetRec and baseline. Under this scenario, all the re-ranking algorithms can accurately estimate the playing time when the video is delivered to end-users. As shown in Figure 6 (b), the performance of Greedy2 is worse than the baseline algorithm as it does not consider the video quality. The knapsack algorithm performs better than Greedy1 indicating a better trade-off ability between video quality and transmission time.

Implications: As the improvement of the accuracy of estimating video play time in the recommendation models, it brings more benefits to optimize the multi-objective re-ranking algorithm.

4.4 RQ4: Ideal Quality Estimation?

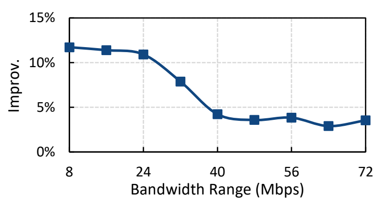


Figure 7: The revenue improvement of NetRec in different bandwidth range in Campus Network, when the video quality estimation achieves 100%.

We conducted a real-world experiment in which the accuracy of video quality estimation reached 100%. To achieve 100% accuracy, we use the posterior playing time to replace the *estimatedPlayingTime*. The result is shown in Figure 7, which indicates the lower-bound of improvements for NetRec. This is because the more accurate of the video quality estimation algorithm, the stronger the baseline algorithm becomes. In a real-world environment, our NetRec achieves higher improvements (shown in Figure 4(a)) because the accuracy of video quality estimation is far lower than 100%.

Implications: Even if the recommendation model achieves ideal 100% accuracy for video quality estimation, jointly considering network status can also bring 3% to 12% revenue improvement.

5 CONCLUSION

In this paper, we propose a novel short video recommendation mechanism that incorporates network bandwidth into the video recommendation system. The mechanism uses a knapsack algorithm to select the video combination that can be successfully transmitted within the specified time and brings the highest real revenue. In situations with poor network bandwidth, the mechanism can bring about 160% improvement compared to traditional recommendation mechanisms. At the same time, this paper also points out that in a network environment with poor bandwidth, the loss of revenue will gradually increase as the bandwidth measurement error increases.

We will focus on the network measurement part, to reduce the revenue loss caused by network measurement errors in our recommendation mechanism for future work.

ACKNOWLEDGMENTS

This work was supported in part by the National Natural Science Foundation of China under Grant 62222204 and Grant 62172148, in part by the National Key Research and Development Program of China under Grant 2020YFB2104000, in part by the Natural Science Foundation of Hunan Province for Excellent Young Scholars, in part by the Training Program for Excellent Young Innovators of Changsha, in part by the National Natural Science Foundation of China under Grant 62202445 and in part by the National Key Research and Development Program of China under Grant 2022YFB2901800.

REFERENCES

[1] Zahaib Akhtar, Yun Seong Nam, Ramesh Govindan, Sanjay G. Rao, Jessica Chen, Ethan Katz-Bassett, Bruno Ribeiro, Jibin Zhan, and Hui Zhang. 2018. Oboe: auto-tuning video ABR algorithms to network conditions. *Proceedings of the 2018 Conference of the ACM Special Interest Group on Data Communication* (2018).

[2] China Internet Network Information Center. 2022. The 50^{th} Statistical Report on China's Internet Development. (2022). https://www.cnnic.net.cn/NMediaFile/2022/0926/MAIN1664183425619U2MS433V3V.pdf, Last accessed on 2023-03-08.

[3] Sihao Ding, Peng Wu, Fuli Feng, Yitong Wang, Xiangnan He, Yong Liao, and Yongdong Zhang. 2022. Addressing Unmeasured Confounder for Recommendation with Sensitivity Analysis. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '22)*. Association for Computing Machinery, New York, NY, USA, 305–315. https://doi.org/10.1145/3534678.3539240

[4] Tongtong Feng, Haifeng Sun, Q. Qi, Jingyu Wang, and Jianxin Liao. 2020. Vabis: Video Adaptation Bitrate System for Time-Critical Live Streaming. *IEEE Transactions on Multimedia* 22 (2020), 2963–2976.

[5] Chongming Gao, Shijun Li, Wenqiang Lei, Jiawei Chen, Biao Li, Peng Jiang, Xiangnan He, Jiaxin Mao, and Tat-Seng Chua. 2022. KuaiRec: A Fully-observed Dataset and Insights for Evaluating Recommender Systems. In *Proceedings of the 31st ACM International Conference on Information and Knowledge Management (CIKM '22)*. 11. https://doi.org/10.1145/3511808.3557220

[6] Xudong Gong, Qinlin Feng, Yuan Zhang, Jiangling Qin, Weijie Ding, Biao Li, Peng Jiang, and Kun Gai. 2022. Real-Time Short Video Recommendation on Mobile Devices. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management (CIKM '22)*. Association for Computing Machinery, New York, NY, USA, 3103–3112. https://doi.org/10.1145/3511808.3557065

[7] Wei Guo, Can Zhang, Zhicheng He, Jiarui Qin, Huifeng Guo, Bo Chen, Ruiming Tang, Xiuqiang He, and Rui Zhang. 2022. MISS: Multi-Interest Self-Supervised Learning Framework for Click-Through Rate Prediction. In *2022 IEEE 38th International Conference on Data Engineering (ICDE)*. 727–740. https://doi.org/10.1109/ICDE53745.2022.00059

[8] Kerim Hodzic, Mirsad Cosovic, Sasa Mrdovic, Jason J. Quinlan, and Darijo Raca. 2022. Realistic Video Sequences for Subjective QoE Analysis. Association for Computing Machinery, New York, NY, USA. https://doi.org/10.1145/3524273.3532894

[9] Jinsung Lee, Sungyong Lee, Jongyun Lee, Sandesh Dhawaskar Sathyanarayana, Hyoyoung Lim, Jihoon Lee, Xiaoqing Zhu, Sangeeta Ramakrishnan, Dirk Grunwald, Kyunghan Lee, and Sangtae Ha. 2020. PERCEIVE: Deep Learning-Based Cellular Uplink Prediction Using Real-Time Scheduling Patterns. In *Proceedings of the 18th International Conference on Mobile Systems, Applications, and Services (MobiSys '20)*. Association for Computing Machinery, New York, NY, USA, 377–390. https://doi.org/10.1145/3386901.3388911

[10] Fuyuan Lyu, Xing Tang, Huifeng Guo, Ruiming Tang, Xiuqiang He, Rui Zhang, and Xue Liu. 2022. Memorize, Factorize, or be Naive: Learning Optimal Feature Interaction Methods for CTR Prediction. In *2022 IEEE 38th International Conference on Data Engineering (ICDE)*. 1450–1462. https://doi.org/10.1109/ICDE53745.2022.00113

[11] Xuan Rao, Lisi Chen, Yong Liu, Shuo Shang, Bin Yao, and Peng Han. 2022. Graph-Flashback Network for Next Location Recommendation *(KDD '22)*. Association for Computing Machinery, New York, NY, USA, 1463–1471. https://doi.org/10.1145/3534678.3539383

[12] Sujay S Tadahal, Sushma V Gummadi, Kartik Prajapat, S. M. Meena, Uday Kulkarni, Sunil V. Gurlahosur, and Shashidhara B. Vyakaranal. 2021. A Survey On Adaptive Bitrate Algorithms and Their Improvisations. *2021 International Conference on Intelligent Technologies (CONIT)* (2021), 1–7.

[13] Chenyang Wang, Yuanqing Yu, Weizhi Ma, Min Zhang, Chong Chen, Yiqun Liu, and Shaoping Ma. 2022. Towards Representation Alignment and Uniformity in Collaborative Filtering. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '22)*. Association for Computing Machinery, New York, NY, USA, 1816–1825. https://doi.org/10.1145/3534678.3539253

[14] Zimu Wang, Yue He, Jiashuo Liu, Wenchao Zou, Philip S. Yu, and Peng Cui. 2022. Invariant Preference Learning for General Debiasing in Recommendation. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '22)*. Association for Computing Machinery, New York, NY,

USA, 1969–1978. https://doi.org/10.1145/3534678.3539439

[15] Wikipedia. 2023. Knapsack problem — Wikipedia, The Free Encyclopedia. http://en.wikipedia.org/w/index.php?title=Knapsack%20problem&oldid=1145132664. (2023). [Online; accessed 18-March-2023].

[16] Wikipedia. 2023. Marginal utility — Wikipedia, The Free Encyclopedia. http://en.wikipedia.org/w/index.php?title=Marginal%20utility&oldid=1135735461. (2023). [Online; accessed 18-March-2023].

[17] Xinlei Yang, Hao Lin, Zhenhua Li, Feng Qian, Xingyao Li, Zhiming He, Xudong Wu, Xianlong Wang, Yunhao Liu, Zhi Liao, Daqiang Hu, and Tianyin Xu. 2022. Mobile Access Bandwidth in Practice: Measurement, Analysis, and Implications. In *Proceedings of the ACM SIGCOMM 2022 Conference (SIGCOMM '22)*. Association for Computing Machinery, New York, NY, USA, 114–128. https://doi.org/10.1145/3544216.3544237

[18] Xinlei Yang, Xianlong Wang, Zhenhua Li, Yunhao Liu, Feng Qian, Liangyi Gong, Rui Miao, and Tianyin Xu. 2021. Fast and Light Bandwidth Testing for Internet Users. In *18th USENIX Symposium on Networked Systems Design and Implementation (NSDI 21)*. USENIX Association, 1011–1026. https://www.usenix.org/conference/nsdi21/presentation/yang-xinlei

[19] Jiangchao Yao, Feng Wang, Xichen Ding, Shaohu Chen, Bo Han, Jingren Zhou, and Hongxia Yang. 2022. Device-Cloud Collaborative Recommendation via Meta Controller *(KDD '22)*. Association for Computing Machinery, New York, NY, USA, 4353–4362. https://doi.org/10.1145/3534678.3539181

[20] Ruohan Zhan, Changhua Pei, Qiang Su, Jianfeng Wen, Xueliang Wang, Guanyu Mu, Dong Zheng, Peng Jiang, and Kun Gai. 2022. Deconfounding Duration Bias in Watch-Time Prediction for Video Recommendation. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '22)*. Association for Computing Machinery, New York, NY, USA, 4472–4481. https://doi.org/10.1145/3534678.3539092