UBR: User-Centric QoE-Based Rate Adaptation for Dynamic Network Conditions

Wangyu Choi and Jongwon Yoon*
{wangyu92,jongwon}@hanyang.ac.kr
Hanyang University, Republic of Korea

ABSTRACT

The prevalence of video streaming applications has led to an escalation in users' demands for high-quality services. Numerous endeavors have been undertaken in the realm of quality-of-experience (QoE) models and adaptive bitrate (ABR) algorithms to fulfill this demand. Nevertheless, the existing QoE models exhibit a significant gap with users' actual experience. ABR algorithms are vulnerable in dynamic network environments. We present an integrated system with an accurate QoE model and an environment-robust adaptation algorithm to ensure high user satisfaction in dynamic network conditions. We define a QoE model that accurately estimates the user's QoE by considering the viewing environment and video content. We then design a meta-reinforcement learning-based adaptation algorithm that adapts to dynamic network conditions. We systematically integrate them, allowing it to update its policy with QoE feedback within a few shots.

KEYWORDS

Meta-learning, QoE model, Rate adaptation, Video steaming

ACM Reference Format:
Wangyu Choi and Jongwon Yoon. 2023. UBR: User-Centric QoE-Based Rate Adaptation for Dynamic Network Conditions. In *The 29th Annual International Conference on Mobile Computing and Networking (ACM MobiCom '23), October 2–6, 2023, Madrid, Spain.* ACM, New York, NY, USA, 3 pages. https://doi.org/10.1145/3570361.3615756

*This work was supported by Basic Science Research Program through the National Research Foundation of South Korea (NRF) funded by the Ministry of Education NRF-2022R1A2C1008743 and the MSIT(Ministry of Science and ICT), Korea, under the Grand Information Technology Research Center support program(IITP-2023-2020-0-01741) supervised by the IITP.

ACM MobiCom '23, October 2–6, 2023, Madrid, Spain

ACM ISBN 978-1-4503-9990-6/23/10…$15.00
https://doi.org/10.1145/3570361.3615756

1 INTRODUCTION

Video streaming apps have grown in popularity, increasing the need for enhanced video streaming services. Extensive efforts have been made in various aspects such as adaptive bitrate (ABR) algorithms [4–6, 9], video quality modeling [1, 2], and bandwidth estimation [4]. These efforts focus on investigating ABR algorithms and quality of experience (QoE) models that directly improve user QoE.

Many efforts [1, 2] have been made to accurately estimate the QoE, but there is still a significant gap due to indirect approaches. Rather than measuring the quality of the rendered video, existing QoE models often focus on the playback buffer, such as downloaded segment size and buffer depletion. Nevertheless, the QoE for the user is considerably influenced by various factors that are inherent to the viewing process (e.g., the viewport's resolution, and motion size). To maximize user QoE, many ABR algorithms have been proposed, in particular, learning-based approaches [4, 6] have emerged as state-of-the-art solutions with the large-scale datasets. However, they have the limitation of environmental sensitivity, making them vulnerable to network dynamics.

Unfortunately, there is a dearth of collaboration between QoE models and ABR in the current literature. Despite the advent of more sophisticated QoE models, contemporary ABR solutions continue to target antiquated QoE models that exhibit a significant gap with the user's actual QoE. For instance, the state-of-the-art algorithms, Pensieve [6], and Fugu [4] try to maximize the sum of bitrate and adaptive penalty, which is not directly related to user QoE.

We focus on key design points: (i) direct approach for QoE modeling, (ii) robust learning-based algorithm, (iii) systematic integration of them. To this end, we first define a user-centric QoE model, that directly estimates the actual QoE. We then design a meta-reinforcement learning-based adaptation algorithm, which is robust to dynamic environments. Finally, we introduce an integrated videos streaming system, UBR, to maximize the user QoE directly. We compare UBR to state-of-the-art ABR algorithms in a variety of environments. UBR accurately estimates user QoE and is robust to adapt within a few shots in user-specific networks.

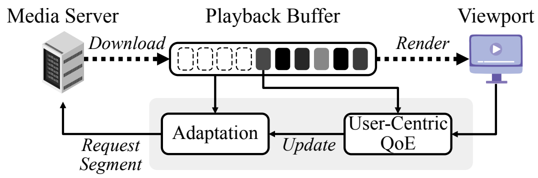

Figure 1: System overview.

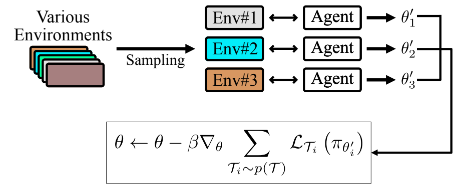

Figure 2: The training process of meta-RL in UBR.

2 DESIGN

Our goal is to design and implement a video adaptation system that enhances the user's actual QoE while ensuring robustness in dynamic network conditions. Toward this, we first define a user-centric QoE model, which accurately models the user's perceived quality. We then design a robust adaptation algorithm to dynamic network conditions while maximizing the QoE model. We illustrate the system overview of UBR in Figure 1. The user-centric QoE adopts LSTM to capture the user's past experience while considering factors belonging to the viewing process.

2.1 User-Centric QoE Model

Visual quality. We train a neural network on large-scale datasets. Specifically, we train the function $q_v(\cdot)$ which maps the factors that determine visual quality, $\boldsymbol{x}_v$, to the visual quality, y_v. The $\boldsymbol{x}_v$ vector is made up of the following elements: bitrate, frame rate, perceived motion size, and perceived resolution. A neural network for visual quality estimation is composed of an input layer, two hidden layers with dropout to mitigate overfitting, and a fully-connected layer. We adopt standard supervised learning to train a visual quality estimation model while minimizing the mean squared error (MSE) between the output visual quality and the subjective scores using stochastic gradient descent.

Adaptive penalty. After the establishment of the visual quality estimation model, the subsequent step involves the integration of the adaptive penalty to finalize the QoE. We adopt LSTM network that takes into account information on adaptive incidents as inputs and generates the estimated QoE while considering past information. In particular, adaptive incident information includes (i) a rebuffering indicator and (ii) the time elapsed since the last rebuffering event. The rebuffering indicator represents the duration of rebuffering for the segment, which is equal to or greater than 0.

Let $\boldsymbol{x}_t$ be the set of rebuffering indicators and time elapsed since the last rebuffering event for the t-th segment. Then, the estimated QoE y_t for $\boldsymbol{x}_t$ is given by: $y_t, c_t, h_t = \mathrm{LSTM}(\boldsymbol{x}_t, c_{t-1}, h_{t-1})$ where, c_{t-1} and h_{t-1} represent the previous cell state and hidden state, which contain past information, respectively. We train a neural network to minimize the MSE loss between the estimated y_t and the actual QoE $\hat{y}_t$ obtained from several human-evaluated datasets.

2.2 Robust Adaptation

We aim to design an adaptation algorithm that maximizes the QoE model while ensuring robustness to dynamic network conditions. To effectively cope with dynamic network conditions, we employ meta-learning in our model that continuously/quickly adapts to new environments.

Neural network. We utilize an actor-critic-based neural network architecture, which comprises an input layer, a hidden layer, an actor head, and a critic head. The neural network takes inputs $\boldsymbol{s}_t$ and uses its parameter θ to generate output $\boldsymbol{\pi}\theta(\boldsymbol{s}_t, st)$ and $v^{\pi\theta}(\boldsymbol{s}_t, \boldsymbol{a}_t)$ of the actor head and the critic head, respectively. The input layer is composed of fully connected (FC) layers and 1D convolutional neural networks to extract the features. The 1-CNN has 64 filters of sizes 2 and 1, and FC has 64 neurons. All features are concatenated and pass through the hidden layer, which is composed of 1 FC with 64 neurons and branches to each head.

Training. The ABR policy is designed to rapidly adapt to new and unforeseen user-specific environments, distinguishing it from traditional RL algorithms. Figure 2 provides an overview of the training process, which involves constructing various environments that closely mimic real-world scenarios. In this process, n-th RL agent samples one environment from the pool and updates its parameters to θ'_n through interaction with the environment. We then perform a meta-update by aggregating the updated parameters across different environments. This meta-update specifically focuses on extracting the sensitive parameters θ that significantly affect the policy among the neural network parameters.

To find a parameter θ, we adopt a Model-Agnostic Meta Learning (MAML) algorithm [3] that combines the models obtained after training in different environments. The training process is divided into two phases: internal adaptation and external adaptation. In the internal adaptation phase, the RL agent updates its parameters to θ'_i in the environment pool $p(\mathcal{T}_i)$. We formulate this process as follows:

$$\theta'_i = \theta - \alpha \nabla_\theta \mathcal{L}_{\mathcal{T}_i}(\pi_\theta) \tag{1}$$

$$\mathcal{L}_{\mathcal{T}_i}(\pi_\phi) = -\mathbb{E}_{\mathbf{s}_t, \mathbf{a}_t \sim \pi_\phi, q_{\mathcal{T}_i}}\left[\sum_{t=1}^{H} R_i(\mathbf{s}_t, \mathbf{a}_t)\right] \tag{2}$$

where, $\mathbf{a}_t$ is the model's output (i.e. bitrate), R_i is the reward function, and H is the length of the episode. To update equation (2), we use Proximal Policy Optimization (PPO) [8], one

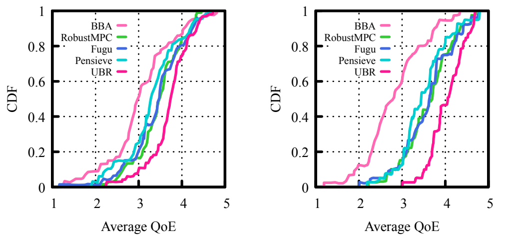

Figure 3: Performance comparison under user-specific network conditions.

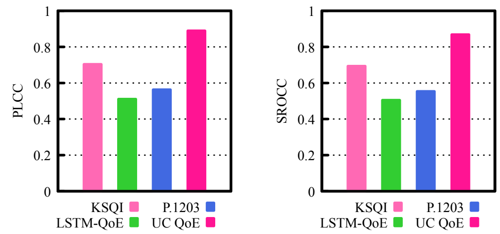

Figure 4: Accuracy comparison QoE model. UC QoE stands for user-centric QoE model.

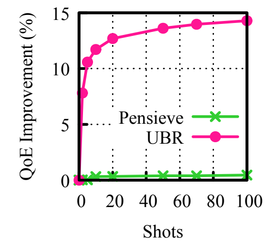

Figure 5: Adaptability of UBR

of the policy gradient algorithms known for its efficiency in many complex tasks. In external adaptation, we calculate the final meta parameter θ through meta update using θ'_i obtained from the internal update.

3 EVALUATION

Performance under user-specific networks. We evaluate the performance of UBR under user-specific network conditions. We collect the network traces of the two top viewers from the Puffer website [4]. For each trace, we execute UBR and the baseline algorithm for comparison. We consider the following 4 baselines: BBA [5], RobustMPC [9], Fugu [4], and Pensieve [6]. For the QoE metric, we use our user-centric QoE model, which closely reflects the user's QoE. Figure 3 shows the comparison results of the baseline algorithms and UBR. We can see that UBR outperforms the baseline algorithms in both network traces. The findings indicate that UBR adapts to each user-specific network condition and provides direct QoE improvements to users.

Accuracy of user-centric QoE model. To demonstrate the effectiveness of our user-centric QoE model, we compare it to existing QoE models. We consider the following as baseline QoE models: KSQI [1], P.1203 [7], LSTM-QoE [2], which are the recently proposed state-of-the-art QoE models. We use Pearson Linear Correlation Coefficient (PLCC) and Spearman Rank Order Correlation Coefficient (SROCC), which measure the correlation between any two datasets. Figure 4 shows a comparison of the accuracy of the user-centric QoE model and the traditional QoE model. We can see that the user-centric QoE model has higher accuracy than the traditional QoE model for both metrics (user-centric QoE is above 0.8 for both, while the baseline algorithms are below 0.7). This result indicates that user-centric QoE is a more direct reflection of the actual user's perceived quality.

Adaptability of UBR. To demonstrate the adaptability of our algorithm based on meta-RL, we measure how quickly the algorithm improves QoE with new data under new network conditions. For comparison, we adopt a traditional RL-based Pensieve. Figure 5 compares the adaptability of UBR and Pensieve. UBR improves QoE after only a few updates (About 10% for 5 shots), while Pensieve has barely improved QoE (0.4%) after 100 updates. Based on these results, we confirm the superiority of UBR based on meta RL. This adaptability of UBR ensures high QoE for users in new, unseen and time-varying environments robustly.

4 CONCLUSION

We propose a user-centric QoE model that directly estimates the user's experience in video streaming, and design and implement a robust adaptation algorithm, UBR, under dynamic network conditions. The user-centric QoE model considers the user's viewing environment and video content. The UBR leverages meta-RL to maximize the user's QoE, thus providing fast adaptation in dynamic network conditions.

REFERENCES

[1] Z. Duanmu, W. Liu, D. Chen, Z. Li, Z. Wang, Y. Wang, and W. Gao. 2019. A Knowledge-Driven Quality-of-Experience Model for Adaptive Streaming Videos. (Nov. 2019). arXiv:1911.07944 [cs.MM]

[2] N. Eswara, S. Ashique, A. Panchbhai, S. Chakraborty, H. P. Sethuram, K. Kuchi, A. Kumar, and S. S. Channappayya. 2020. Streaming Video QoE Modeling and Prediction: A Long Short-Term Memory Approach. *IEEE Trans. Circuits Syst. Video Technol.* 30, 3 (March 2020), 661–673.

[3] C. Finn, P. Abbeel, and S. Levine. 2017. Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks. arXiv:1703.03400 [cs.LG]

[4] F. Y. Yan and H. Ayers, C. Zhu, and S. Fouladi, J. Hong, K. Zhang, P. Levis, and K. Winstein. 2020. Learning in situ: a randomized experiment in video streaming. In *Proceedings of NSDI.*

[5] T-Y. Huang, R. Johari, N. McKeown, M. Trunnell, and M. Watson. 2014. A Buffer-Based Approach to Rate Adaptation: Evidence from a Large Video Streaming Service. In *Proceedings of SIGCOMM.* 187–198.

[6] H. Mao, R. Netravali, and M. Alizadeh. 2017. Neural Adaptive Video Streaming with Pensieve. In *Proceedings of SIGCOMM.* 197–210.

[7] W. Robitza, M-N. Garcia, and A. Raake. 2017. A Modular HTTP Adaptive Streaming QoE Model – Candidate for ITU-T P.1203 ("P.NATS"). In *Proceedings of QoMEX.* 1–6.

[8] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov. 2017. Proximal Policy Optimization Algorithms. arXiv:1707.06347 [cs.LG]

[9] X. Yin, A. Jindal, V. Sekar, and B. Sinopoli. 2015. A Control-Theoretic Approach for Dynamic Adaptive Video Streaming over HTTP. In *Proceedings of SIGCOMM.* 325–338.