Bridge the Gap Between QoS and QoE in Mobile Short Video Service: a CDN Perspective

Chuanqing Lin[1,2], Yangguang Liang, Fuhua Zeng, Zhipeng Huang, Xiaodong Li, Jingyu Yang, Yu Tian[1,2], Gerui Lv[1,2], Qinghua Wu[1,2,4], Zhenyu Li[1,2,4], and Gaogang Xie[2,3]

[1] Institute of Computing Technology, Chinese Academy of Sciences, China
[2] University of Chinese Academy of Sciences, China
[3] Computer Network Information Center, Chinese Academy of Sciences, China
[4] Purple Mountain Laboratories, China

Abstract. Emerging mobile short video services pose different yet stringent performance requirements compared to traditional long video services. Content providers (CPs) aspire to a better user-perceived Quality of Experience (QoE) at the application layer, which is imperceptible to the Content Delivery Network (CDN), which monitors Quality of Service (QoS) at the transport layer. The mismatch between QoS and QoE leads to a complex and diverse mapping correlation between the two metrics. In this paper, we illustrate the QoS-QoE mapping correlation in mobile short video services. Although data-driven QoE prediction models can achieve the desired accuracy, complex scenario features are proven to be necessary, and the prediction model still lacks interpretability. Deeper quantitative analysis shows that the correlation becomes complex and diverse when resources are insufficient. The clustering-based prediction framework can successfully summarize scenario features and perform QoE prediction based on QoS metrics alone. Furthermore, we propose predictive QoE-based CDN scheduling. Experiments show that compared to scheduling with QoS metrics, QoE-aware scheduling achieves an average QoE improvement of 9.9% under comparable QoS quality.

Keywords: QoE prediction · CDN scheduling · Mobile short video.

1 Introduction

Mobile short video services have experienced explosive growth over the past decades. Currently, TikTok alone has over 100 million monthly active users globally [8]. Due to the short playback time of a single video, the performance requirements for such services differ significantly from those of long video services and become more stringent [29, 28]. The Content Delivery Network (CDN) is widely employed to ensure a promised user experience, by deploying nodes closer to users to reduce response time. Quality of Service (QoS) and Quality of Experience (QoE) are commonly adopted metrics to evaluate CDN performance and user experience from different layers and locations. QoS relates to network performance, such as latency and throughput, which can be monitored from servers of CDNs. QoE focuses on the video playback experience, including the startup delay, stall counts, and bitrate selection, which is only obtained from applications on end devices. CDNs focus on QoS to schedule traffic to specific nodes, whereas users and content providers are more concerned with QoE.

Due to the hierarchical layer and locational mismatch between QoS and QoE, the correlation between these two factors often exhibits complex, non-linear characteristics. While it is generally believed that good QoS leads to good QoE, prior works indicate that in certain situations, QoE is insensitive to QoS improvements, and good QoS might even result in degraded QoE [26]. This results in CDNs' efforts for QoS optimization not always leading to QoE improvements, creating a significant imbalance between investment cost and performance gain. As a result, CDNs are strongly motivated to investigate the complex QoS-QoE mapping correlation and thereby optimize the QoE of short video playback.

Prior works [18, 4, 17] have proposed QoS-based objective QoE prediction to optimize QoE. However, these works operated at the user side and focused on a single stream to perform bitrate selection or CDN multihoming. CDNs, focusing on the general correlation between QoE and QoS among multi-user and multi-stream to guide traffic scheduling, cannot benefit from these works.

This paper, from the CDN perspective, investigates the correlation between QoS and QoE. Based on the dataset involving both QoS and QoE metrics (§3.1), we employ an XGBoost model to perform QoE prediction at an aggregated level (§3.2). However, additional scenario features are proven to be necessary for accurate prediction, and the data-driven model still lacks interpretability for practical deployment. Consequently, we conduct quasi-experimental design (QED) to illustrate the correlation between two metrics (§3.3). The correlation pattern is portrayed by pattern clustering under different scenarios, which is shown to become complex and diverse when resources are insufficient during traffic peaks. The clustering approach is validated to summarize the scenario features successfully and perform QoE prediction based on QoS metrics alone.

Although the specific cluster patterns may vary depending on different configurations from content providers, and the specified prediction model might be only applicable for a single provider, the proposed clustering-based prediction framework demonstrates the generalizability to summarize the scenario features on any given configuration. Therefore, the data sharing between content providers and CDN vendors indicates considerable potential for enhancing users' QoE.

Based on the characterization of the QoS-QoE mapping correlation, we propose QoE prediction-based CDN scheduling (§4). Simulation experiments demonstrate that compared to traditional QoS-based scheduling, QoE-based scheduling achieves an average QoE improvement of 9.9% at stall count under comparable QoS quality, reaching 73.3% of optimal improvement. In summary, the contributions of our work are as follows:

– We observe the complex and non-linear correlation between QoE and QoS. The correlation patterns under different scenarios are illustrated to exhibit diversity when resources are insufficient during traffic peaks.

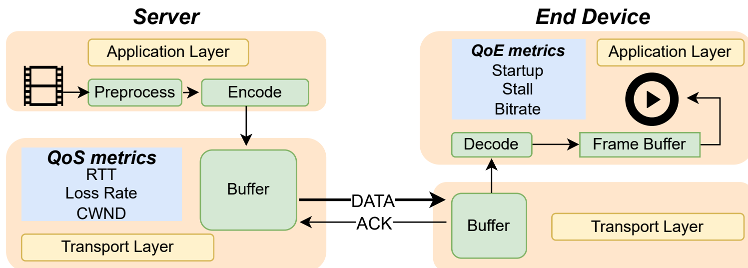

Fig. 1: QoS and QoE for mobile short video services.

- We propose a clustering-based prediction framework to extract and summarize complex scenario features, and thereby predicting the QoE metrics based on QoS metrics alone.
- We present predictive QoE-based CDN scheduling. Simulation experiments show that compared to scheduling for QoS, QoE-based scheduling achieves an average QoE improvement of 9.9% under comparable QoS quality.

2 Background and Motivation

2.1 Background

Mobile short video services. Emerging mobile short video services have imposed significantly different and more stringent performance requirements compared to traditional long videos [28, 29]. First, mobile short video services require shorter startup delays to enable users to continuously browse and discover content of interest. Consequently, short videos typically adopt lower bitrates. Adaptive Bitrate Streaming (ABR) algorithms designed for long videos struggle to converge quickly, thereby exhibiting poor performance. Second, stall events impact user experience more severely in short video services. Even a single stall event during playback reduces the video viewing percentage by 45%, with increased stall event frequency amplifying this decline.

QoS-Based CDN Scheduling. CDN is widely applied in mobile short video services to ensure optimal user viewing experiences. CDNs typically deploy edge nodes closer to end users to minimize video transmission latency and response time. By scheduling traffic from different regions and applications to specific nodes, CDNs manage to optimize users' transmission performance. Such QoS metrics (e.g., latency, bandwidth, and loss rate) operate at the transport layer and can be directly monitored by CDNs. However, users and short video content providers prioritize QoE metrics (e.g., stalls, startup delay, bitrate) at the application layer, which CDNs cannot directly observe.

QoS-QoE Mismatch. As illustrated in Fig. 1, two fundamental mismatches exist between QoS and QoE:

- *Hierarchical mismatch.* QoS is typically implemented within the transport layer of the TCP protocol stack, evaluating data transmission quality over

the network and strongly correlating with network conditions. QoE, conversely, reflects users' experience at the application layer, influenced not only by network quality but also by end-device capabilities and other features.
- *Locational mismatch.* QoS is monitored and managed by the server. The protocol stack marks packets to enable priority handling and service quality assurance. QoE, however, originates from users' interactions with applications on the client side. As the server cannot perceive QoE, it can only optimize QoS. However, the client cares more about QoE than QoS.

2.2 Motivation

Due to the QoS-QoE mismatch, the mapping correlation between QoS and QoE is complex, nonlinear, and influenced by various factors. Although better QoS is generally believed to yield better QoE, prior works [26] indicate that QoE may be insensitive to QoS improvements under certain conditions or even deteriorate. Conversely, degraded QoS also does not necessarily lead to reduced QoE.

CDNs improve user access performance and stability by scheduling traffic to the nearest edge nodes with optimal QoS. However, the aforementioned complexities lead to two critical issues in QoS-based CDN scheduling: *(1) Suboptimal video QoE for users.* The QoS-based scheduling approach fails to achieve the intended QoE optimal scheduling results. *(2) Inefficient resource utilization for CDNs.* CDNs' efforts to construct high-performance nodes are not effectively converted to a superior user experience, resulting in a significant imbalance between investment costs and performance gains.

These challenges motivate CDNs to investigate the QoS-QoE mapping correlation in mobile short video applications, thereby guiding QoE-based scheduling.

3 QoS-QoE Mapping

3.1 Dataset

We first introduce the QoS and QoE datasets in this work. The QoS data was collected from a leading CDN vendor in China at the server side, while the QoE data was collected from a leading short video application at the user side. The dataset contains more than 4 million entries, covering requests from 3 short video domain names in 3 ISPs to 276 CDN nodes in 31 provinces over 2 months. Due to the different data collection granularity, all data was aggregated to the average value at the granularity of <domain name, ISP, user province, node province> at each 10-minute time interval.

Metadata. Both the CDN and the short video application recorded metadata of user requests during the covered period, including server location, client region, requested domain name, and request timestamp. Although the granularity differs between the two sources, the data can be matched after aggregation at the same granularity.

QoS metrics. Major QoS metrics include connection setup time, retransmission rate, and throughput. Since QoE information is inaccessible, CDNs analyze

Table 1: Prediction accuracy with different input features

Features	Startup Delay	Stall Count
QoS	6.0%	23.7%
QoS + Time	4.8%	14.7%
QoS + ISP	5.7%	20.1%
QoS + Domain name	5.0%	15.5%
QoS + User location	4.2%	14.2%
QoS + Time + ISP + Domain + User location	3.1%	8.7%

QoS metrics from the TCP stack on their nodes. At the start of video transmission, the user sets up a TCP connection through a three-way handshake with the CDN node. The CDN node measures connection setup time from receiving the SYN packet to receiving the first ACK packet (approximately one Round-trip time), serving as an indicator of the link latency. Throughout transmission, the link's loss rate is calculated as total retransmitted data divided by total transmitted data. Link bandwidth is represented by throughput, calculated as total transmitted data divided by total transmission time. These QoS metrics reflect link performance at the transport layer.

QoE metrics. Major QoE metrics include startup delay and stall count. Startup delay refers to the delay from when a video is requested to when the first frame begins playing, indicating user wait time. Stall count indicates the average number of stalls per 100 seconds of playback, reflecting video fluency. These two metrics represent application-layer playback quality. The dataset does not include the bitrate metric, since it is not prioritized in short video services, as discussed in §2.1. Future work could incorporate bitrate into consideration to explore the three-way trade-off between startup delay, stalling, and video quality.

3.2 Data-driven Prediction Model

To investigate the relationship between QoS and QoE metrics, we first attempt to predict QoE using existing QoS metrics. To achieve it, we employ XGBoost [5] as the mapping relationship prediction model. For the loss function, Mean Absolute Percentage Error (MAPE) [20] was selected to evaluate prediction performance. A smaller MAPE value indicates smaller error and higher prediction accuracy. We train two independent models for two QoE metrics with the same input.

As shown in the Tab.1, prediction using QoS metrics alone yields significant errors, suggesting other scenario factors may influence QoE performance. To analyze the impact of scenario factors, we calculate the mutual information between scenario features and QoS/QoE metrics, as shown in Fig. 2.

Higher mutual information (i.e., larger than 0.1) indicates a stronger relationship between the feature and metric. Temporal features are more significant at daily and hourly granularity, which primarily affects QoS metrics due to traffic peaks. Conversely, spatial features, including the location of users and nodes, exhibit a strong correlation with both QoS and QoE metrics. The domain name variation, which is attributed to the access type (i.e., Wi-Fi and cellular), and the

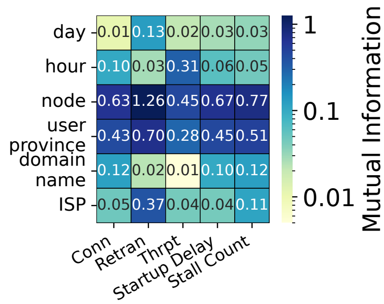

Fig. 2: Mutual Information between features.

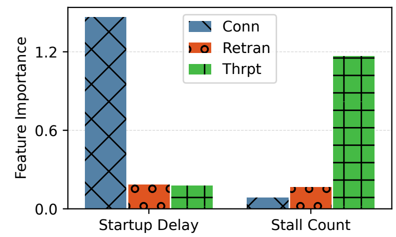

Fig. 3: Feature importance in the prediction model.

ISP variation, which is attributed to the infrastructure construction, influence the corresponding QoS metric and, thereby, the corresponding QoE metric.

Based on this analysis, attempts that combine hourly time features, user location features, and domain features with QoS metrics for model prediction achieve MAPE below 10%, as shown in Tab.1. Compared to predicting by QoS alone, incorporating scenario factors significantly reduces startup delays and stall count prediction error by 48.3% and 63.3%, respectively. This demonstrates that scenario factors significantly impact the QoS-QoE mapping.

3.3 QED Analysis

The previous data-driven prediction model lacks interpretability for deployment, and the correlation between QoE and QoS remains unrevealed. We further investigate how QoS influences corresponding QoE in different scenarios.

Univariate mapping function. Intuitively, different QoS metrics have varying importance for distinct QoE metrics. For example, startup delay is mostly impacted by connection setup time to fetch the first frame, while stall events are caused by insufficient playback buffer under low throughput. The feature importance analysis illustrated in Fig.3 confirms this observation.

We employ Quasi Experimental Design (QED) [14] to study how scenario and temporal features affect the quantitative relationships in two univariate functions: <connection setup time, startup delay> and <throughput, stutter>. For a fixed network link (static domain name, ISP, node region, user region), we derive a univariate function f that maps a single QoS metric to a single QoE metric under controlled scenario conditions:

$$QoE_i = f(QoS_i | scenario) \tag{1}$$

This conditional function can be plotted as a 2D curve. With our dataset and prediction model, we generated 1,872 conditional functions and plotted their curves, representing QoS-QoE mappings under diverse scenarios.

Clustering. To identify different and complex patterns, we cluster these curves based on shape similarity and distance. The shape similarity indicates the QoS-QoE mapping correlation, while the distance variation indicates the scenario features, especially the geographical distance of a link. Compared to Euclidean distance, we employ Dynamic Time Warping (DTW) distance [3],

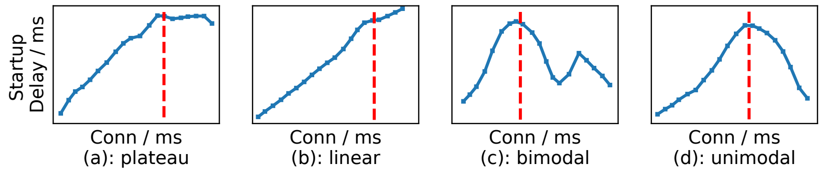

Fig. 4: Centers of shape clustering for <connection setup time, startup delay>.

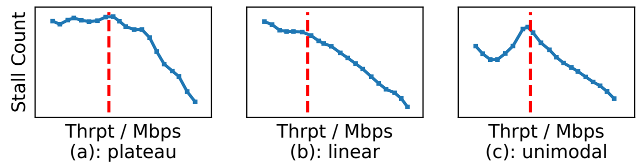

Fig. 5: Centers of shape clustering for <throughput, stall count>.

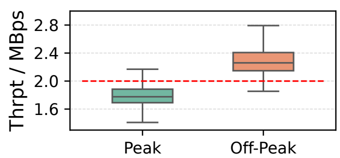

Fig. 6: Throughput division.

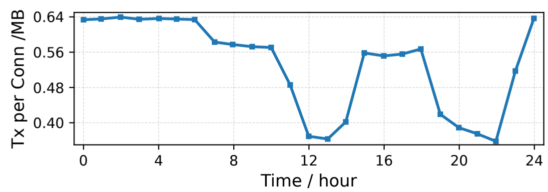

Fig. 7: Daily data transmitted per TCP stream.

which prioritizes shape alignment over absolute position. K-Medoids clustering [12], which is robust against outliers, is applied with the optimal cluster count determined by the elbow method using the Sum of Squared Errors (SSE).

The clustering results for <connection setup time, startup delay>, <throughput, stall count> are shown in Fig. 4 and Fig. 5. For connection setup time and startup delay, four categories can be identified based on shape similarity, which can be further divided into eight categories based on both shape and distance. The shape of the mapping curve can be categorized as one of four types: *(a)* plateau, *(b)* linear, *(c)* bimodal, or *(d)* unimodal. Correlation between throughput and stall count can be categorized into three classes based on shape and further divided into six classes based on both shape and distance. The shape of the mapping curve can be categorized as: *(a)* plateau, *(b)* linear, or *(c)* unimodal.

Clustering result analysis. As we can find in the two figures, when resources are abundant (low delay or high throughput), QoS-QoE relationships are approximately linear. Under resource scarcity (e.g., high delay or low throughput in peak hours), the relationship becomes nonlinear and complex, sometimes even showing QoE degradation despite QoS improvement.

We observe a transition point (dashed line) which marks the division between traffic peak (11h-14h, 19h-22h) and off-peak QoS metrics. As shown in Fig.6, the transition point can be defined as the average of the upper quartile (Q3) of throughput at peaks and the lower quartile (Q1) of throughput at off-peaks.

$$Thrpt_{tran} = \frac{Q_3(Thrpt_{peak}) - Q_1(Thrpt_{off-peak})}{2} \tag{2}$$

The transition point represents the threshold where network resources shift from abundant to constrained, which we empirically observed to be the inflection

Table 2: Prediction accuracy of different clusters.

Cluster	1	2	3	4	5	6	7	8
Startup delay	3.2%	2.3%	2.3%	2.8%	3.2%	2.5%	4.4%	2.5%
Stall count	7.7%	10.1%	9.8%	6.9%	7.0%	8.0%		

point where application-layer bitrate adaptation strategies become most active. Fig. 7 illustrates the average amount of transmitted data per TCP stream in a day. During peak hours, the amount of data transmitted by users decreases significantly. In order to ensure video stability and an optimal viewing experience, applications may have adopted a lower bit rate for short videos. This explains why QoE metrics may improve even when the QoS metrics are poor in bimodal and unimodal patterns.

Clustering validation. Clustering groups scenario features (e.g., domain names, user and node provinces) which have a similar impact on QoE metrics into a single category. As a result, QoE metrics within each clustering group do not rely on the scenario features and can be accurately predicted using only QoS metrics. To validate this, separate models were trained for each clustering group to predict corresponding QoE metrics. The prediction errors for each cluster in terms of the startup delay and the stall count are shown in Tab. 2.

The average QoE metric prediction errors of all clusters reach 2.9% for the startup delay and 8.25% for the stall count, which is lower than the prediction error of the model based on QoS and scenario factors in §3.2. This result successfully proves that the clustering approach extracts and summarizes scenario features. Consequently, the correlation between QoS and QoE is straightforward within each clustering group.

3.4 Summary

Complex QoS-QoE mapping correlation. Data collected from CDNs and short-video applications indicates a complex correlation between QoS metrics and QoE metrics. Quantitative analysis and clustering show that when resources are limited, the mapping correlation becomes diverse due to the intricate bit rate strategies employed by applications.
Clustering-based predictive framework. Scenario features can be extracted and summarized through clustering. The straightforward QoS-QoE mapping can be subsequently established inside each cluster.

4 QoE-aware CDN Scheduling

4.1 Implications for CDN Scheduling

Based on prior analysis, the relationship between QoS and QoE is not a simple linear mapping and may vary with other scenario factors such as time and user region. Consequently, solely QoS-based CDN scheduling does not always guarantee optimal QoE. Insights from QoS-QoE quantitative analysis can optimize CDN scheduling in two aspects:

Algorithm 1: QoE-based CDN scheduling.

Input: Traffic demand T_j; Node capacity C_i; QoS Metrics QoS_{ij}; Scenario Factors F_{ij}.
Output: Scheduling strategy X_{ij}.

```
for j ∈ M, i ∈ N do
    Model_ij ← Clustering(F_ij);
    D_ij ← Model_ij(QoS_ij)
end
Solve the optimization problem Obj.3 to find X_ij.
```

Performance differentiated node utilization strategies. Fig.4 and Fig.5 illustrate distinct curve shapes for different mapping types. The effectiveness of QoS improvements on QoE enhancement varies across these curves. Conversely, reducing QoS in certain scenarios does not always degrade QoE. Thereby, CDNs can actively schedule specific traffic to nodes with poorer network quality without compromising user experience, while ensuring efficient resource utilization simultaneously.

Time differentiated scheduling strategies. The mapping correlation pattern is divided into two segments according to the resource supplement at different time periods. As a result, CDNs must adjust scheduling strategies accordingly. For example, for *plateau* shaped correlation (Fig. 4(a) and 5(a)), aggressive scheduling strategies (allocating high-QoS resources) significantly improve QoE during off-peak hours. Whereas during peak hours, conservative scheduling strategies (using low-QoS resources) minimally impact their QoE, saving high-QoS resources for other traffic.

4.2 Scheduling Experiment

To validate the improvement in user experience through QoE-aware scheduling, we conducted simulated experiments using the same dataset from §3.1, which covers 3 ISPs, 31 provinces, and 276 CDN nodes, constituting a total of four million data entries. We ran the CDN scheduling problem six times each day for a consistent 30-day period in a month, during traffic peaks (12 AM, 8 PM, and 10 PM) and off-peak times (9 AM, 3 PM, and 5 PM).

CDN Traffic Scheduling Problem. For given traffic demand T_j in each province $j \in M$, CDN scheduling computes the scheduling strategy X_{ij} that indicates the amount of traffic from province j scheduled to the specific node $i \in N$. In general, the traffic scheduling problem can be formulated as follows.

$$\begin{aligned} \min_X \sum_{i,j} & D_{ij} \cdot X_{ij} \\ s.t. \sum_j & X_{ij} \leq C_i, \ \forall i \in N; \\ \sum_i & X_{ij} = T_j, \ \forall j \in M. \end{aligned} \tag{3}$$

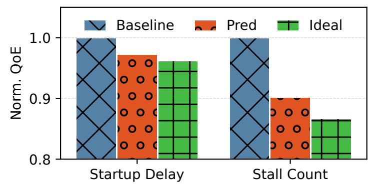

Fig. 8: Norm. QoE quality.

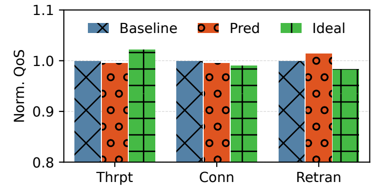

Fig. 9: Norm. QoS quality.

where C_i denotes the capacity of each node i and the quality matrix D characterizes the quality penalty (i.e., latency, throughput) for traffic from province j to node i. The optimization problem aims to find the optimal strategy X_{ij} that minimizes the overall quality penalty while ensuring no nodes are overloaded and all traffic demands are scheduled.

Baseline. The ultimate goal for CDN scheduling is to optimize real QoE performance, which is only collectible after the transmission. Thereby, the optimization goal Obj.3 can be formulated in multiple ways, according to the definition of the quality matrix D. Specifically, three types of D are defined in the simulation. *(1) QoS-based:* D_{ij} is computed by the connection setup latency $Conn_{ij}$ and throughput $Thrpt_{ij}$ monitored by CDN. We set this method as the baseline because such metrics optimize the overall QoS performance, which is adopted by CDNs currently. *(2) predicted QoE-based*: D_{ij} is predicted based on QoS performance, through the pre-trained prediction model corresponding to the scenario features, as introduced in §3.3. This method promises CDNs to optimize the overall QoE performance directly under limited prediction error. *(3) real QoE-based*: D_{ij} is collected from QoE metrics for real traffic from province j to node i, according to the dataset. This method shows the optimal QoE performance can be achieved when the QoE prediction model predicts completely accurately. The workflow of the scheduling is illustrated in Alg.1.

QoE Improvement As shown in Fig. 8, predictive QoE-aware scheduling can effectively improve all of the system's QoE metrics. Since each QoE metric has a different scale, it is normalized to facilitate comparison of the improvement effect among different QoEs. A smaller value is better for each type of metric. The simulation based on predicted QoE achieves an average improvement of 9.9% in stall count and utilizes 73.3% of the optimal improvement space. Meanwhile, the improvement in startup delay is 2.67%. The improvement is moderate because startup delay is highly related to connection latency, which is proportional with geographically traversed distance and thereby hard to improve overall. The optimal startup delay improvement is only 3.8% and predictive QoE-based scheduling utilizes 71.2% of the improvement space. The QoE improvement does not rely on QoS improvement. As shown in Fig. 9, the QoS quality of predictive or real QoE-aware scheduling remains comparable with QoS-based scheduling.

5 Related Work

Subjective QoE prediction. The complex composition of QoE contains parameters in both subjective and objective dimensions [24]. Subjective QoE is strongly correlated with users' subjective perceptions and is difficult to assess. A common method to get subjective QoE is mean opinion score (MOS) [6], which is costly and poorly scalable. [25] tried to establish the correlation between QoS and subjective QoE but did not consider application performance. [10] predicted MOS with objective QoE. Subjective QoE has much less relevance to CDNs.

Objective QoE prediction. Objective QoE is easier to portray than subjective QoE. Existing work has examined the relationship between QoS and objective QoE from several perspectives. [19, 9] estimated QoE performance by active probing. [7, 13, 21, 18] performed QoE prediction in encrypted traffic. [1] used passive measurements of QoS to predict objective QoE. [15, 11, 2, 22, 27, 16, 23] performed the prediction based on machine learning models and achieved better results. However, these works either consider only the overall performance of QoS, which fails to meet CDNs' need for granular quality information, or focus on CDN multihoming or bitrate selection from the user perspective, which is inapplicable to CDNs.

6 Conclusion

This work investigated the QoS-QoE mapping correlation in mobile short video services, which is observed to exhibit complexity and diversity under varying scenarios, especially when resources are insufficient during traffic peaks. A clustering-based prediction framework is presented to extract and summarize scenario features, and thereby predicting the QoE metrics based on QoS metrics alone. The subsequently proposed predictive QoE-based CDN scheduling attains an average QoE improvement of 9.9% under comparable QoS quality.

Acknowledgments. We thank all anonymous reviewers for their constructive feedback. This work was supported in part by the National Key R&D Program of China (2022YFB2901800). Gaogang Xie is the paper's corresponding author.

References

1. Aggarwal, V., Halepovic, E., Pang, J., et al.: Prometheus: Toward quality-of-experience estimation for mobile apps from passive network measurements. In: HotMobile' 14
2. Balachandran, A., Sekar, V., Akella, A., et al.: Developing a predictive model of quality of experience for internet video. In: ACM SIGCOMM (2013)
3. Berndt, D.J., Clifford, J.: Using dynamic time warping to find patterns in time series. In: 3rd KDD (1994)
4. Bronzino, F., Schmitt, P., Ayoubi, S., Martins, G., Teixeira, R., Feamster, N.: Inferring streaming video quality from encrypted traffic: Practical models and deployment experience. SIGMETRICS **3**(3), 1–25 (2019)
5. Chen, T., Guestrin, C.: Xgboost: A scalable tree boosting system. In: ACM SIGKDD. p. 785–794. KDD '16 (2016)
6. Choe, J.H., Jeong, T.U., Choi, H., et al.: Subjective video quality assessment methods for multimedia applications. Journal of Broadcast Engineering **12** (03 2007)

7. Dimopoulos, G., Leontiadis, I., Barlet-Ros, P., Papagiannaki, K.: Measuring video qoe from encrypted traffic. p. 513–526. IMC '16 (2016)
8. Doyle, B.: Tiktok statistics, wallaroomedia.com/blog/social-media/tiktok-statistics/
9. Ge, C., Wang, N.: Real-time qoe estimation of dash-based mobile video applications through edge computing. In: IEEE INFOCOM 2018 (2018)
10. Ito, Y., Tasaka, S.: Quantitative assessment of user-level qos and its mapping. IEEE Transactions on Multimedia **7**(3) (2005)
11. Jiang, J., Sekar, V., Milner, H., Shepherd, D., Stoica, I., Zhang, H.: Cfa: A practical prediction system for video qoe optimization. p. 137–150. NSDI'16 (2016)
12. Kaufmann, L., Rousseeuw, P.: Clustering by means of medoids. Data Analysis based on the L1-Norm and Related Methods pp. 405–416 (01 1987)
13. Khokhar, M.J., Ehlinger, T., Barakat, C.: From network traffic measurements to qoe for internet video. In: 2019 IFIP Networking (2019)
14. Krishnan, S.S., Sitaraman, R.K.: Video stream quality impacts viewer behavior: Inferring causality using quasi-experimental designs. In: IMC '2012 (2012)
15. Lin, Y.T., Oliveira, E.M.R., Ben Jemaa, S., Elayoubi, S.E.: Machine learning for predicting qoe of video streaming in mobile networks. In: 2017 IEEE ICC (2017)
16. Liu, J., Waz, M., Chung, J., et al.: Estimating video quality using coarse-grained features: Insights and limitations from gaussian mixture models. CoNEXT '24 (2024)
17. Mangla, T., Halepovic, E., Ammar, M., et al.: Using session modeling to estimate http-based video qoe metrics from encrypted network traffic. IEEE TNSM (2019)
18. Mazhar, M.H., Shafiq, Z.: Real-time video quality of experience monitoring for https and quic. In: IEEE INFOCOM 2018 (2018)
19. Miranda, G., Macedo, D.F., Marquez-Barja, J.M.: A qoe inference method for dash video using icmp probing. In: 2020 16th CNSM (2020)
20. de Myttenaere, A., Golden, B., Grand, B.L., Rossi, F.: Mean absolute percentage error for regression models. Neurocomputing (jun 2016)
21. Orsolic, I., Pevec, D., Suznjevic, M., et al.: A machine learning approach to classifying youtube qoe based on encrypted network traffic. Multimedia Tools Appl. (2017)
22. Shah, A., Bran, J., Zarifis, K., Bedi, H.: Ssqoe: Measuring video qoe from the server-side at a global multi-tenant cdn. In: PAM 2022. p. 600–625 (2022)
23. Sharma, T., Mangla, T., Gupta, A., Jiang, J., Feamster, N.: Estimating webrtc video qoe metrics without using application headers. IMC '23, ACM (2023)
24. Wu, W., Arefin, A., Rivas, R., al., N.: Quality of experience in distributed interactive multimedia environments: Toward a theoretical framework. In: 17th ACM Multimedia. p. 481–490. MM '09 (2009)
25. Xiao, Z., Xu, Y., Feng, H., et al.: Modeling streaming qoe in wireless networks with large-scale measurement of user behavior. In: 2015 IEEE GLOBECOM (2015)
26. Zhang, H., Zhou, A., Hu, Y., Li, C., Wang, G., Zhang, X., Ma, H., Wu, L., Chen, A., Wu, C.: Loki: Improving long tail performance of learning-based real-time video adaptation by fusing rule-based models. In: 27th MobiCom. p. 775–788 (2021)
27. Zhang, Y., Cheng, S., Guo, Z., Zhang, X.: Inferring video streaming quality of real-time communication inside network. IEEE TCSVT (2024)
28. Zhang, Y., Li, P., Zhang, Z., Bai, B., Zhang, G., Wang, W., Lian, B.: Challenges and chances for the emerging short video network. In: IEEE INFOCOM 2019 (2019)
29. Zhang, Y., Liu, Y., Guo, L., Lee, J.Y.B.: Measurement of a large-scale short-video service over mobile and wireless networks. IEEE TMC pp. 1–1 (2022)