Suffering from Buffering? Detecting QoE Impairments in Live Video Streams

Adnan Ahmed and Zubair Shafiq
The University of Iowa

Harkeerat Bedi and Amir Khakpour
Verizon Digital Media Services

***Abstract*—Fueled by increasing network bandwidth and decreasing costs, the popularity of over-the-top large-scale *live* video streaming has dramatically increased over the last few years. In this paper, we present a measurement study of adaptive bitrate video streaming for a large-scale live event. Using server-side logs from a commercial content delivery network, we study live video delivery for the annual Academy Awards event that was streamed by hundreds of thousands of viewers in the United States. We analyze the relationship between Quality-of-Experience (QoE) and user engagement. We first study the impact of buffering, average bitrate, and bitrate fluctuations on user engagement. To account for interdependencies among QoE metrics and other confounding factors, we use quasi-experiments to quantify the causal impact of different QoE metrics on user engagement. We further design and implement a Principal Component Analysis (PCA) based technique to detect live video QoE impairments in real-time. We then use Hampel filters to detect QoE impairments and report 92% accuracy with 20% improvement in true positive rate as compared to baselines. Our approach allows content providers to detect and mitigate QoE impairments on the fly instead of relying on post-hoc analysis.**

I. Introduction

An increasingly large number of content publishers now broadcast video content live over the Internet. This growth is a consequence of low costs of content delivery and the adoption of advertisement/subscription based revenue models. The live video content encompasses social networking services (e.g., Periscope, Facebook Live), video game streaming services (e.g., Twitch, YouTube Gaming), broadcast TV networks (e.g. NBC, ABC, CBS), cable TV news networks (e.g., CNN, MSNBC), and cable TV sports networks (e.g., ESPN, NFL network). This migration from traditional broadcast to Internet video creates a need to provide high-quality over-the-top video streaming services.

According to Cisco [4], 73% of the global Internet traffic was video in 2016. Furthermore, the popularity of live video streaming has increased significantly over the last decade due to its emergence on social networks [3], e-sports and video game streaming platforms [14], [15], and online sports and entertainment broadcasts [2]. The "Video Internet" is here to stay—the volume of video traffic is expected increase up to 82% of the global Internet traffic by 2021. Therefore, the stakeholders in the Internet video ecosystem such as content providers, Content Delivery Networks (CDNs), and Internet Service Providers (ISPs) need to frequently upgrade their infrastructure to meet the increasing demands of Internet video. Moreover, as high-definition streaming devices, augmented/virtual reality (AR/VR) streaming, and broadband Internet connectivity becomes more common, user expectations for high-quality and smooth video streaming are continuing to rise. Since content providers generate revenue through advertisements and subscriptions, they strive to maintain good user experience and maximize user engagement [1].

Extensive research has been conducted on various aspects of streaming content delivery to cope with the ever-increasing demands of high-quality Internet video. On one hand, prior literature includes studies on analyzing the impact of Quality-of-Service (QoS) metrics such as bandwidth, packet loss, and bit error rate on the performance of specific applications including streaming video [10], [11], [17], [33]. To this end, researchers used passively collected network data to study video access patterns and viewing behavior of users across different ISPs and edge networks. On the other hand, researchers have used video-specific Quality-of-Experience (QoE) metrics, such as rate of buffering and average bitrate, in the pursuit to directly quantify user experience and understand the effect of "bad QoE" on user engagement [7], [13], [21], [24]. Video streaming services mostly rely on human-in-the-loop and post-hoc analysis to detect and analyze root causes of QoE impairments [21]. To the best of our knowledge, prior literature lacks tools that can be used by operators to automatically detect video QoE impairments in real-time.

In this paper, we focus our attention on analyzing user engagement and QoE impairments for large-scale live video streams. Specifically, we measure and analyze QoE for live streaming of the 87th annual Academy Awards. The event was streamed all over the United States by a large commercial CDN with 9 geographically distributed Points-of-Presence (PoPs). The event amassed over 600 thousand video stream viewers over the duration of 5 hours, with nearly 100 thousand concurrent viewers at peak. Overall, we observe over 21 million minutes of viewing time from users all across the United States. Our objectives are to understand the impact of various QoE metrics on user engagement and to design techniques for automatically detecting QoE impairments in real-time.

To this end, we first quantify video quality in terms of different QoE metrics such as rate of buffering and average bitrate. We then study the cause-effect relationships between user engagement and QoE metrics using a quasi-experimental framework. Finally, we use Principal Component Analysis

978-1-5090-6501-1/17/$31.00

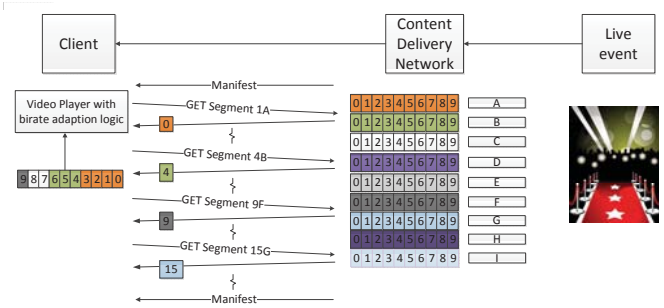


Fig. 1. Architecture of adaptive video streaming for a large-scale live event

(PCA) to detect QoE impairments in an online and real-time fashion. Our analysis of the large-scale live video streaming event reveals several interesting findings and actionable insights for operators. We summarize our key findings and insights as follows.

- We analyze the interplay between different QoE metrics for large-scale live video streaming. Using Kendall rank correlation, we quantify the intrinsic interdependencies between QoE metrics. We find strong positive 89.6% correlation between rate of buffering and buffering ratio. We also find strong negative -40.6% correlation between rate of (bitrate) fluctuation and average bitrate.
- We then employ a quasi-experimental framework to study the impact of QoE metrics on user engagement while accounting for the interdependencies between QoE metrics and other confounding factors such as ISP affiliations and device types. We find that rate of buffering has the most negative impact on user engagement. For example, an increase in rate of buffering by 0.13 buffering events per minute degrades user engagement by as much as 20 minutes on average. We find that average bitrate has a positive impact on user engagement. For example, 1.5 Mbps increase in average bitrate improves user engagement by 10 minutes on average. In contrast, rate of fluctuation has a relatively negligible impact on user engagement.
- We use PCA to help content providers to detect QoE impairments in real-time. We group users based on their AS affiliation and device type and construct "normal" and "residual" subspaces from the users viewing the event for significant duration using PCA. We then project average QoE metrics of users in a time window on the residual subspace and identify the ones with sufficiently large projection magnitudes (L^2-norm) as anomalous users. Generally, we find that anomalous users detected by our methodology have significantly worse QoE metrics compared to normal users. We use the Hampel filters to detect spikes in residual subspace projections in real-time. Our PCA based method provides 92.7% accuracy that represents 20% improvement in true positive rate as compared to baselines.

II. Background & Data

A. Background

Internet video can be classified into three categories, each with its own distinct characteristics: streaming stored video (Netflix, YouTube, Hulu, etc.), streaming live video (Periscope, Twitch, NFL Live, etc.), and interactive live video (Skype, Facetime, Google Hangouts, etc.). On one hand, a stored video is bandwidth-sensitive and is typically streamed from a CDN. On the other, an interactive live video is delay-sensitive and typically uses P2P techniques or relay servers. Lastly, streaming live video falls somewhere in the middle of the spectrum in terms of its delay and bandwidth sensitivity.

Video service providers typically use HTTP-based adaptive bitrate video streaming techniques. Streaming video is divided into smaller segments (or chunks) of length 2-10 seconds and encoded at different bitrates with different video and audio quality at the server. The clients are made aware of the available bitrates via a manifest file downloaded at the start and during playback. It is worth noting that the content servers are not required to track the playback state at clients. The idea is to dynamically adapt video bitrate based on various factors such as network bandwidth estimation and player buffer occupancy. Since end-to-end network bandwidth fluctuates over time, *adaptive* bitrate controllers [6], [19], [26], [34], [35] are implemented at the client side to estimate network bandwidth, request video at appropriate bitrates and optimize tradeoffs associated with QoE metrics and streaming bitrate. Note that user QoE for streaming video is determined by a wide range of factors such as encoding bitrate, buffering rate, etc. For example, requesting a high streaming bitrate results in frequent buffering events (i.e., stall while client playback buffer is being replenished) and a lower bitrate means that the user watches lower quality video stream. Understanding the relationship between QoE and user engagement is an active research area [7], [13], [24], [33].

B. Data Collection

Figure 1 shows the video delivery architecture and the adaptive bitrate streaming protocol used by the CDN. The CDN back-end servers receive high quality video feed from the live event which is digitally encoded into 4-second segments at 9 different quality levels. The quality levels range from the lowest bitrate (quality A) to the highest bitrate (quality I).[1] The encoded video segments are pushed to the front-end cache servers for delivery to clients. Before the initialization of video streaming, the video player downloads the manifest file from content servers which contain URLs of video segments at multiple quality levels. The client sends a series of HTTP GET requests to the content server to fetch video segments. The video segments from the CDN server are downloaded and stored in the playback buffer of the client video player. The client can then reassemble the segments for rendering a seamless video playback. Recall that the adaptive bitrate controller is responsible for requesting the video segments at appropriate bitrates based on network bandwidth estimation and playback buffer information. The bitrate adaption logic is

[1]The CDN uses variable bitrate encoding. The average bitrate for each quality level is as follows. A: 60 Kbps. B: 124 Kbps. C: 234 Kbps. D: 316 Kbps. E: 710 Kbps. F: 1.2 Mbps. G: 2.5 Mbps. H: 3.7 Mbps. I: 3.9 Mbps

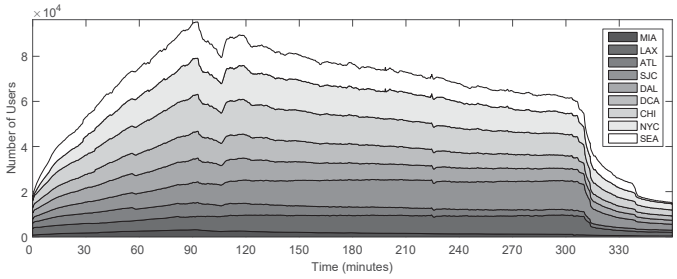


Fig. 2. Timeseries of viewership for different PoPs. The event was streamed by more than 600K users. At its peak, the event was streamed by nearly 100K users.

PoP Name	Location	# Users
MIA	Miami	17,379
LAX	Los Angeles	25,880
ATL	Atlanta	36,584
SJC	San Jose	54,326
DAL	Dallas	63,308
DCA	Washington DC	64,080
CHI	Chicago	91,417
NYC	New York	114,029
SEA	Seattle	158,623

TABLE I
DISTRIBUTION OF VIEWERS ACROSS POPS. THE TOTAL NUMBER OF VIEWERS ACROSS ALL POPS IS 625,626.

not defined in the standards and its implementation is often proprietary.

Our data set is composed of server-side HTTP logs from a commercial CDN which was responsible for online streaming of the live event. The data is collected at multiple Points-of-Presence (PoPs) across the United States. The HTTP logs contain information such as client IP address, port number, and user agents.[2] We also have information about video bitrate and server response time. Using these logs, we can measure and analyze video quality metrics and track engagement statistics for different users.

C. Data Statistics

As shown in Table I, our data set is collected from 9 different PoPs. We note that top-3 PoPs (Chicago, New York, Seattle) account for 60% of total users. Overall, our data set contains 625,626 users and more than 21 million minutes worth of video view time. Figure 2 plots the timeseries of number of users that were watching the event. The event lasted over five hours. We note that the viewership started to rapidly increase and reached its peak at around the 90th minute mark. The peak corresponds to the marquee moment at the event and was followed by a steady decline over time. After about 3 hours, the viewership declined rapidly which corresponds to the end of the main event. Figure 3 plots the distribution of viewing time across all users in our data set. We observe that the viewing time distribution has a long tail [7], [34]. Furthermore, the top 5% of users watch the event

[2]Note that all device and user identifiers (e.g., IP addresses) in the collected data set are anonymized to protect privacy without affecting the usefulness of our analysis. The data sets do not permit the reversal of the anonymization or re-identification of users.

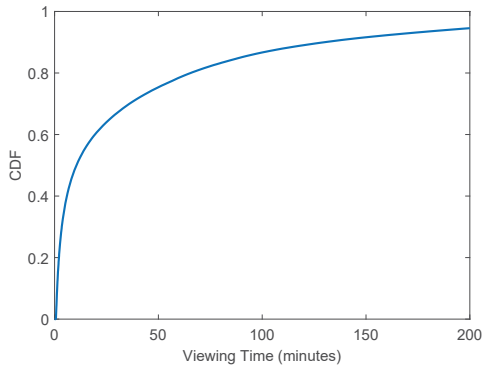


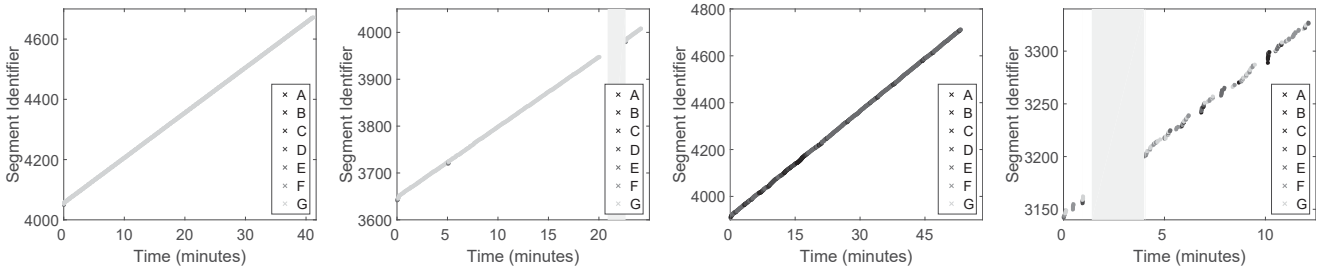
Fig. 3. Distribution of viewing time

for more than 220 minutes. However, the median viewing time is approximately 20 minutes. This is expected because most users may only be interested in different portions of the event and therefore only tune in during those portions. For example, as evident from Figure 2, most users may be interested in the event around the 90th minute mark.

D. Examples

To better understand the data, we illustrate four example adaptive bitrate streaming sessions in Figure 4. Each adaptive bitrate video session involves downloading a sequence of segments, each at one of the distinct quality levels. The x-axis represents the segment arrival time, which starts with the download of the first 4 second segment. Note that the four users shown in Figure 4 join the live video stream at different points in time. The y-axis represents the segment identifier which increases monotonically in a linear fashion due to the live nature of the video stream. The markers represent individual segments, which are color coded to indicate their bitrate. Darker colors represent lower quality levels as compared to lighter colors. The vertical gray strips represent buffering events, and the width of the strip represent buffering duration. Note that the video player skips segments during a buffering event and resumes playback from the most recent video segment from the content provider at the end of the buffering event.

In Figure 4, we note that all users start off the video streaming session at a low bitrate, which is usually done to sufficiently fill up the playback buffer at the beginning [19]. Figure 4(a) shows a user belonging to a cable ISP with smooth video streaming experience at a high bitrate. Figure 4(b) shows another example user from the same cable ISP. This user also watches the video at a high bitrate, but stops receiving subsequent segments around the 20 minute mark resulting in a buffering event represented by the gray strip. The buffering event lasts for over 2 minutes likely due to loss of Internet connectivity. Comparing Figures 4(a) and 4(b), we observe that multiple users in the same ISP may have different video streaming experiences. Such differences can potentially be attributed to variations in the last-mile and last-hop connectivity as well as the device types. Figures 4(c) and 4(d) show users watching the video stream from cellular ISPs.



(a) (b) (c) (d)

Fig. 4. Illustrations of video streaming sessions: (a) High quality streaming, without buffering/fluctuations; (b) High quality streaming, with a buffering event, without fluctuations; (c) Low quality streaming, without buffering, with fluctuations; and (d) Low quality streaming, with a buffering event and fluctuations.

Note that both users generally watch the event at low bitrates and experience bitrate changes likely due to rapid variations in the cellular radio link quality [8], [9].

III. QoE Measurement

Prior research has studied the impact of commonly used network quality (or QoS) metrics such as end-to-end delay and packet loss on application performance [16], [17], [33]. While these QoS metrics have been shown to be useful, they do not directly tie-in to end-user experience. A user may experience video streaming issues despite good network QoS due to unexpected cross-layer interactions [18]. Therefore, it is important to study QoE metrics that directly measure end-user experience. In this section, we analyze video QoE and characterize its impact on user engagement which is defined in terms of video viewing time.[3]

A. QoE Metrics

Below, we define a set of video-specific metrics[4] to quantify QoE [7], [13].

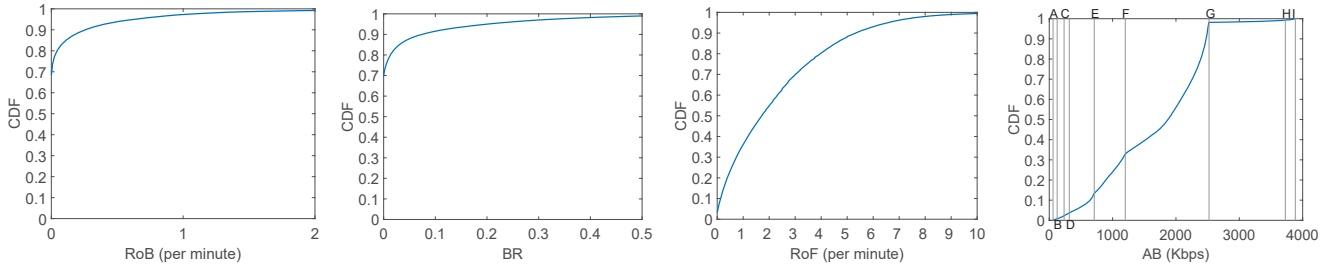
1) **Rate of Buffering (RoB):** We calculate rate of buffering as the number of buffering events (per minute). A buffering event is characterized as the video player stopping the video playback while waiting for the buffer to be sufficiently replenished.
2) **Buffering Ratio (BR):** We calculate buffering ratio as the ratio of total duration of buffering events to the duration of a video session.
3) **Rate of Fluctuation (RoF):** We calculate rate of fluctuation as the number of changes in bitrate (per minute). Note that we do not use the magnitude of bitrate change.
4) **Average Bitrate (AB):** We calculate the average bitrate as the mean bitrate of all video segments requested by a user.

Before we study the impact of QoE metrics on user engagement, we analyze the distributions of QoE metrics. Figure 5 plots the cumulative distribution functions (CDFs) of QoE metrics. In Figure 5(a), we note that a vast majority of users experience relatively low rate of buffering. For example, almost 80% of users experience less than 0.1 buffering events per minute. Similarly, in Figure 5(b), we note low buffering ratios for most users. For example, more than 80% of users have buffering ratios lower than 0.013. In Figure 5(c), we observe that users do experience relatively frequent bitrate fluctuations. For example, about 50% of users experience more than 2 bitrate fluctuations per minute. In Figure 5(d), we note that a majority of users watch the video stream at a high bitrate. For example, more than 60% of users watch the video stream at a bitrate of at least 1 Mbps (i.e., quality level F) on average. Overall, we observe that a small fraction of users in the tail of the distributions suffer from frequent and prolonged buffering events, frequent bitrate fluctuations, and low average bitrate. We expect these "anomalous" users to have relatively lower engagement as compared to other users. To test this hypothesis, we next analyze the relationship between QoE metrics and user engagement.

Figure 6 illustrates the relationship between QoE metrics and user engagement. We quantify user engagement in terms of viewing time (in minutes). Naturally, higher viewing time indicates better user engagement. From Figures 6(a) and 6(b), we note that buffering events seem to have a strong impact on user engagement. We observe a sharp drop in viewing time for increasing values of rate of buffering and buffering ratio. For example, viewing time decreases from a high of more than 210 minutes for users who experience little/no buffering to less than 30 minutes for users experiencing 0.5 buffering events per minute. Moreover, viewing time decreases from a high of about 150 minutes for users who experience very low buffering ratio to about 30 minutes for users experiencing 5% buffering ratio. This is expected because frequent and prolonged buffering events negatively impact user engagement and decrease user viewing time [13]. In Figure 6(c), we note that rate of fluctuation also has a negative impact on user engagement. For instance, viewing time decreases to 30 minutes when rate of fluctuation exceeds 1.5 bitrate fluctuations per minute. We note that frequent bitrate changes as a result of client-side bitrate adaption impact user experience and degrade user engagement [19]. In Figure 6(d), we observe a non-monotonic relationship between average bitrate and user engagement. More specifically, we observe spikes in viewing time when average bitrate is close to the discrete quality

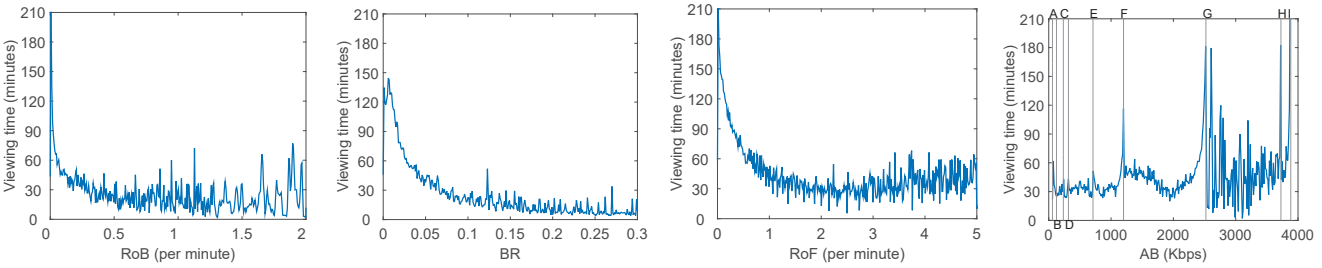
[3]Note that user engagement is also impacted by user interest in content. We argue that lack of interest generally results in early abandonments. To account for early abandonments, we filter out sessions that are abandoned within the first 10 segments.

[4]We do not consider join time because we do not have client-side player information.



(a) Rate of buffering (b) Buffering ratio (c) Rate of fluctuation (d) Average bitrate

Fig. 5. Distributions of QoE metrics. Overall, most users experience few buffering events at high average bitrate. We do observe a tail of users experiencing higher buffering, more fluctuations, and lower bitrate than other users.



(a) Rate of buffering (b) Buffering ratio (c) Rate of fluctuation (d) Average bitrate

Fig. 6. Impact of QoE metrics on user engagement. We note that an increase in buffering events leads to decreased user engagement. We also note that higher average bitrate increases user engagement, but more fluctuations in video quality lead to decreased user engagement.

levels. We surmise that users with average bitrates between two video quality levels (e.g, between F and G) experience more fluctuations which result in lower user engagement. For example, viewing time for users with the average bitrate at 1.2 Mbps (quality F) is about 120 minutes. Similarly, viewing time for users with the average bitrate at 2.5 Mbps (quality G) is about 180 minutes. However, viewing time decreases to almost 30 minutes for users with the average bitrate at 2 Mbps.

While we note that all QoE metrics seem to impact user engagement, we are interested in quantifying the relative impact of QoE metrics on user engagement. However, as reported in prior literature [13], [24], we note that QoE metrics have intrinsic interdependencies. For example, changes in video bitrate affect multiple QoE metrics such as average bitrate and rate of fluctuation. Therefore, a naive application of regression models to interdependent QoE metrics will lead to models that lack meaningful interpretation.

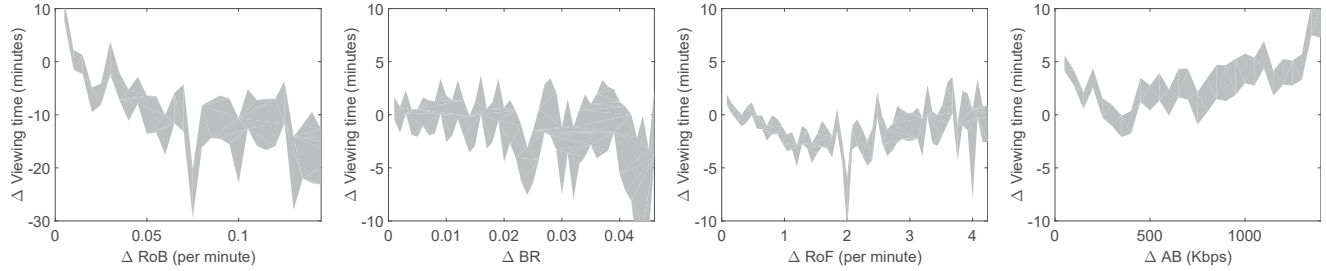
We use Kendall rank correlation to quantify the interdependencies among QoE metrics. Kendall rank correlation coefficient determines the direction and magnitude of the dependency between a pair of variables. Moreover, it does not make any assumptions about the underlying distributions of the variables which makes it suitable to capture non-linear interdependencies among QoE metrics. Table II lists Kendall rank correlation coefficients between all pairs of QoE metrics. Note that high values of Kendall rank correlation coefficients are marked as bold. We observe that the following two pairs of QoE metrics are highly correlated. First, as expected, rate of buffering and buffering ratio exhibit high positive correlation. Therefore, the impact of buffering ratio on user engagement in Figure 6(a) can partially be attributed to rate of buffering and vice versa. Second, rate of fluctuation and average bitrate exhibit high negative correlation. This finding can be explained by the observation that users in good quality networks tend to stream videos at high stable bitrates (high bitrate, low rate of fluctuation) while users in bad quality networks stream videos at low unstable bitrates (low bitrate, high rate of fluctuation). Therefore, the impact of rate of fluctuation on user engagement in Figure 6(c) can partially be attributed to average bitrate and vice versa. We take care of such correlations by conducting quasi-experiments to identify cause-effect relationships between QoE metrics and user engagement.

B. Causal Analysis

To measure the *causal* impact of each QoE metric on user engagement, we employ the quasi-experimental framework [31]. Quasi-Experiment Design (QED) is a well known technique in social/medical sciences and has been previously used by Krishnan et al. to study QoE [24]. In this study, we extend their application of QED to our measurements for a large-scale live video streaming event and study the causal impact of QoE metrics on user engagement. QED allows us to *control* for confounding factor and correlated QoE metrics, and study the cause-effect relationships between QoE metrics and user engagement. In particular, we use matched design QED to study the impact of a target QoE metric on user engagement by comparing two randomly selected users with same values of the variables we decide to control for. Specifically, we control for confounding factors such as device type and ISP affiliation. We also control for highly correlated QoE metrics identified in Table II. For example, when studying the impact

	Rate of Buffering	Buffering Ratio	Rate of Fluctuation	Average Bitrate
Rate of Buffering		**89.6%**	-8.5%	-2.1%
Buffering Ratio	**89.6%**		-7.5%	-1.8%
Rate of Fluctuation	-8.5%	-7.5%		**-40.6%**
Average Bitrate	-2.1%	-1.8%	**-40.6%**	

TABLE II
KENDALL CORRELATION COEFFICIENTS (IN PERCENT) BETWEEN QOE METRICS.



(a) Rate of buffering (b) Buffering ratio (c) Rate of fluctuation (d) Average bitrate

Fig. 7. Causal impact analysis of QoE metrics on user engagement. We note that rate of buffering and average bitrate impact user engagement the most. On the other hand, rate of fluctuation does not significantly impact user engagement.

of rate of buffering on user engagement, we select two users with the same ISP, device type, and similar buffering ratio. As another example, when studying the impact of average bitrate on user engagement, we select two users with the same ISP, device type, and similar rate of fluctuation. Note that we control for only one highly correlated QoE metric in order to limit sparsity. We then assign users in bins based on their confounding factors and the discretized QoE metric that we want to control for. As a result, all users in a bin have same values of the control QoE metric and confounding factors. To quantify the causal impact of a target QoE metric on user engagement, we randomly pair clients in each bin. For each pair, we calculate the difference in user engagement as a function of change in the target QoE metric.

Figure 7 plots the results of the QED analysis for all QoE metrics. The x-axis represents the change in target QoE metric across all bins. The y-axis represents the corresponding average change in user engagement. To ascertain statistical significance, we plot 90% confidence intervals for the difference in user engagement. Figure 7(a) shows that rate of buffering affects user engagement the most. For example, an increase in rate of buffering by 0.13 buffering events per minute degrades user engagement by as much as 20 minutes on average. Figure 7(d) shows that average bitrate also impacts user engagement. For example, around 1.5 Mbps increase in average bitrate improves user engagement by 10 minutes on average. While buffering ratio and rate of fluctuation seem to affect user engagement in Figures 6(b) and 6(c), they are not causally impactful according to the QED results in Figures 7(b) and 7(c). We suspect that the apparent effect of buffering ratio and rate of fluctuation on user engagement observed in Figures 6(b) and 6(c) is in fact due to their high correlation with rate of buffering and average bitrate, respectively. Prior work [24] has reported the causal impact of buffering ratio on user abandonment; however, we find that while buffering ratio plays some role in impacting user experience, rate of buffering and average bitrate have the most causal impact on user engagement.

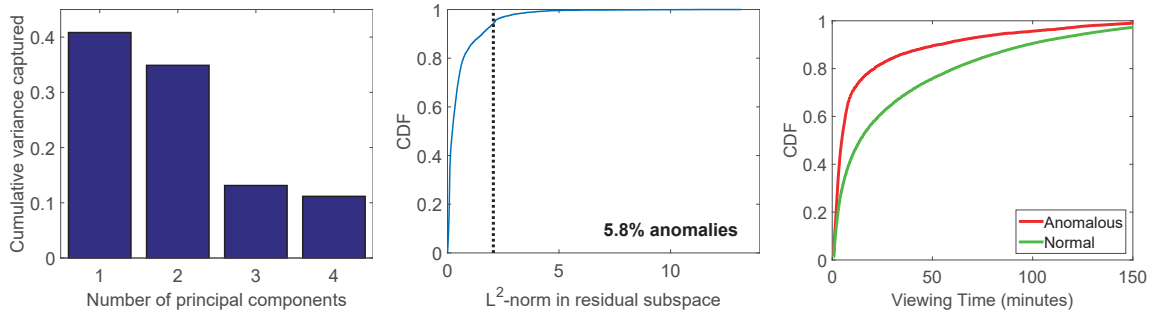
IV. QOE IMPAIRMENT DETECTION

In the previous section, we conducted a post-hoc impact analysis of different QoE metrics on user engagement. We now want to focus our attention towards *real-time* detection of QoE impairments. Real-time QoE impairment detection allows content providers to track QoE impairments over time and take suitable mitigation actions on the fly. QED allowed us to analyze the causal impact of each QoE metric on user engagement. However, QED is primarily used to study after-the-fact cause-effect relationships between variables and outcomes. Therefore, QED is a post-hoc analysis technique which cannot be used for real-time QoE impairment detection.

We propose a real-time QoE impairment detection method using Principal Component Analysis (PCA). Specifically, we formulate the QoE impairment detection as an anomaly detection problem. Our insight is that users experiencing QoE impairments will stand out as anomalous among all users because a majority of users do not experience QoE impairments. We propose to use PCA to detect these anomalies over time [22]. PCA has been widely used in prior literature for anomaly detection [25], [30]. PCA transforms a set of input variables to a set of orthogonal principal components such that a small number of principal components ("normal subspace") explain a majority of variability in the data. The remaining principal components ("residual subspace") can be used to detect anomalies in the data. Using PCA for QoE impairment detection has some advantages. First, PCA takes care of collinearity between QoE metrics by transforming them into a set of linearly uncorrelated principal components. Second, while most statistical anomaly detection methods assume an underlying distribution of "normal", PCA does not require such assumptions. Finally, unlike QED, we can use PCA to detect QoE impairments in real-time by analyzing temporal variations in the residual subspaces. Below we provide a brief background of PCA for anomaly detection.

A. PCA Background

PCA is typically used to transform correlated QoE metrics into a set of orthogonal, linearly uncorrelated principal



(a) Variance captured by principal components (b) L^2-norm distribution (Cable ISP) (c) Viewing time

Fig. 8. Implementation of PCA to detect QoE impaired (anomalous) users.

components. The principal components are ordered by the amount of variance in the data captured by them. The principal components with the least amount of captured variance can be used to constitute the residual subspace. Therefore, we can detect anomalous users experiencing QoE impairments by analyzing the residual subspace. More specifically, we project users into the residual subspace and identify the users with large residual projection magnitude as anomalous users.

First, we divide users in groups based on their ISP affiliation and device type to mitigate potential confounding factors like interests and expectations. We use an observation matrix $\mathbf{Z}^{xy}$ for each user group, of size $n \times m$ to represent QoE metrics, where n is the number of users in ISP x using device y, and m is the number of QoE metrics. Element z_{ij} of the matrix $\mathbf{Z}^{xy}$ is the value of the j^{th} QoE metric for the i^{th} user in user group xy. Thus, each row represents QoE metrics of a user while each column represent the values of a particular QoE metric across all users in the group. We standardize $\mathbf{Z}^{xy}$ so that its columns have zero mean and unit variance. This ensures that principal components are not skewed in the direction of QoE metrics with large magnitude. PCA yields a set of m principal components, $\{\mathbf{v}_i\}_{i=1}^{m}$. Each principal component points along the maximum variance in the remaining data, given the variance captured by the preceding components.

Figure 8(a) plots the variance captured by each principal component in our QoE data. We observe that more than 40% of the variance in our data is captured by only the first principal component, $\mathbf{v}_1$. In addition, the second principal component $\mathbf{v}_2$ captures more than 30% variance. We use the vector space spanned by $\mathbf{v}_1$ and $\mathbf{v}_2$ as normal subspace *S*. More specifically, for a user group xy, we have the matrix $\mathbf{A}^{xy} = (\mathbf{v}_1, \mathbf{v}_2)$ of size $m \times 2$. QoE metrics for a user i, $\mathbf{z}_i^{xy}$ from ISP x and using device y can be decomposed as:

$$\mathbf{z}_i^{xy} = \mathbf{A}^{xy}(\mathbf{A}^{xy})^T\mathbf{z}_i^{xy} + (\mathbf{I} - \mathbf{A}^{xy}(\mathbf{A}^{xy})^T)\mathbf{z}_i^{xy},$$

where $\mathbf{A}^{xy}(\mathbf{A}^{xy})^T\mathbf{z}_i^{xy}$ and $(\mathbf{I} - \mathbf{A}^{xy}(\mathbf{A}^{xy})^T)\mathbf{z}_i^{xy}$ respectively represent the projection of QoE metrics of a user in user group xy on the *normal* and *residual* subspaces. Here, $\mathbf{A}^{xy}(\mathbf{A}^{xy})^T$ and $(\mathbf{I} - \mathbf{A}^{xy}(\mathbf{A}^{xy})^T)$ are linear operators that we can use to project QoE metrics of a user on normal *S* and residual $\hat{S}$ subspaces, respectively. Using this methodology, we synthesize the principal and residual subspaces for each user group xy.

We use the L^2-norm of $(\mathbf{I} - \mathbf{A}^{xy}(\mathbf{A}^{xy})^T)\mathbf{z}_i^{xy}$ to detect QoE impairments in a user group xy. Figure 8(b) plots the distribution of residual L^2-norm ($||(\mathbf{I} - \mathbf{A}^{xy}(\mathbf{A}^{xy})^T)\mathbf{z}_i^{xy}||^2$) for users in a popular cable ISP. Note that a majority of users have small values of residual L^2-norm. Some users, however, have very large values of L^2-norm beyond the "knee" of the curve. We identify the L^2-norm threshold for every user group (l^{xy}) from the knee of the curve [32] and mark tail users as anomalous.

Using this methodology, we separate out anomalous and normal users in a popular cable ISP and plot their viewing time distributions in Figure 8(c). We observe that the users tagged as anomalous view the video for less duration as compared to the users tagged as normal. For instance, median viewing time for anomalous users is approximately 10 minutes less than that of normal users. The difference in viewing time increases to more than 30 minutes at the 80^{th} percentile. We further use the Kolmogorov-Smirnov test [28] to confirm the statistical significance of the difference in viewing time between normal and anomalous users.

B. Real-time QoE Impairment Detection

1) Proposed Approach: In the last section, we constructed normal and residual subspaces for each user group xy. We want to detect QoE impairments in real-time by analyzing the projections of QoE metrics on the anomalous subspace. To this end, we split the QoE timeseries in time windows of length 1 minute and calculate the average values of the QoE metrics in the time windows. We detect anomalous users by projecting the QoE metrics on the residual subspace and calculating the residual L^2-norm in every time window.

$\mathbf{Z}_t^{xy}$ is an $n \times m$ observation matrix at time window t, where n is the number of users in ISP x using device y and m is the number of QoE metrics. For every time window, the L^2-norm in the residual subspace is calculated as $||(\mathbf{I} - \mathbf{A}^{xy}(\mathbf{A}^{xy})^T)(\mathbf{z}_i^{xy})_t||^2$ Here, $(\mathbf{z}_i^{xy})_t$ is the average QoE metrics of a user i from user group xy in time window t. We then tag users with values of L^2-norm larger than the threshold value l^{xy} as anomalous users.

We want to track the QoE impairments among anomalous users over time and raise an alarm when the QoE metrics degrade significantly. Specifically, for real-time QoE impairment detection across a user group xy, we analyze temporal

variations in the average residual norm of the anomalous users calculated as:

$$(L^2_{avg})^{xy}_t = \frac{\sum_{u \in anomalous(xy)} ||(\mathbf{I} - \mathbf{A}^{xy}(\mathbf{A}^{xy})^T)(\mathbf{z}^{xy}_u)_t||^2}{n_{anomalous(xy)}}$$

where $anomalous(xy)$ encapsulates all users tagged as anomalous in the time window. Note that $(L^2_{avg})^{xy}_t$ portrays the severity of QoE impairments for the anomalous users in a group xy over time. We use $(L^2_{avg})^{xy}_t$ to detect significant degradation in overall QoE in a group.

Figure 9 plots the timeseries of average residual norm for anomalous users ($(L^2_{avg})^{xy}_t$) and QoE metrics for an example ISP-device group. The x-axis represents the time relative to the start of the live video streaming event. The y-axis of the top subgraph represents the $(L^2_{avg})^{xy}_t$ of the anomalous users in the residual subspace. The y-axis of the remaining subgraphs represents the average values of QoE metrics for both anomalous and normal users. Note that the QoE metrics for anomalous users are significantly worse than the QoE metrics for normal users. Specifically, rate of buffering, buffering ratio and rate of fluctuation for anomalous users are significantly larger than those for normal users. Furthermore, average bitrate for anomalous users is less than that for normal users.

Generally, we note that the spikes in residual $(L^2_{avg})^{xy}_t$ follow the degradations in QoE metrics for anomalous users. We detect these events using Hampel filters. Hampel filters have been used in prior literature for robust and efficient outlier detection [12], [29]. Hampel filters identify outliers based on the input deviation from the median input in a moving window of length K. The detected anomalies help content providers to monitor video QoE impairments in real-time and facilitate alarming applications. For our evaluation, we implement a moving window Hampel filter using the threshold $T = 2$ and time window of length $K = 10$.

2) Evaluation: To evaluate the performance of the aforementioned Hampel filter based methodology, we calculate standard ROC metrics in this section.

Ground Truth. In order to calculate the ROC metrics for our methodology, we need ground truth to identify whether the QoE impairments were correctly detected. Since QoE impairments are subjective and difficult to model, there is no definitive ground truth for QoE impairments. We surmise that users tend to abandon the video when they experience severe QoE impairments. Therefore, we use user abandonment rate as ground truth for QoE impairments. Specifically, we tag a detected QoE impairment as a true positive if the user abandonment rate increases significantly in a time window of 10 minutes following the detection. To this end, we use Hampel filters and detect points of significant increase in user abandonment rate.

Results. We compare our PCA based scheme with several baselines. We use individual QoE metrics and user utility equation [27] for baseline comparison. Note that the user utility is a measure of user experience (see [27] for details)

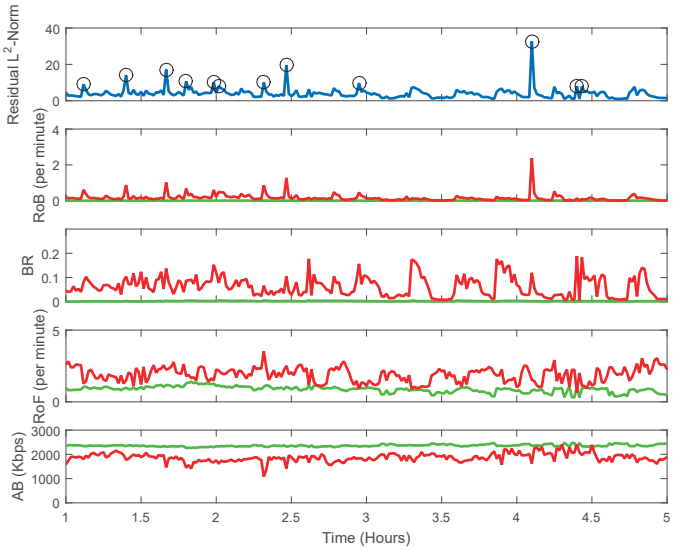


Fig. 9. Online QoE impairment detection results for an example group (ISP affiliation and device type). The circles in the timeseries of residual L^2-norm represent anomalies that are automatically detected by PCA and the Hampel filter. We observe substantial degradation across multiple QoE metrics corresponding to the detected anomalies.

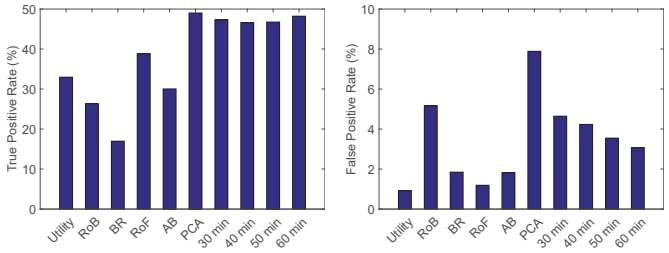
and depends on average bitrate and buffering ratio as: $-370\times$ Buffering Ratio + Average Bitrate/4.

Figure 10 plots the true positive rate (TPR) and false positive rate (FPR). Here, PCA represents the QoE impairment detection using $(L^2_{avg})^{xy}_t$. We note that our PCA based scheme detects QoE impairments with a highest TPR of around 50%. However, the FPR for our PCA based scheme in Figure 10(b) is around 8%. We further note that if we detect QoE impairments using the baseline methodologies, we can potentially get better FPR. Specifically, we note that we can get up to 35% and 40% TPR, if we detect QoE impairments from utility equation and RoF respectively. However, the FPR for baseline methodologies is significantly small compared to our PCA methodology. We argue that this is due to the fact that our PCA methodology learns the principal components from all users in a user group. Note that some users are tolerant towards QoE impairments and continue to watch the video regardless of QoE degradation. On the other hand, some users abandon the video despite good QoE due to lack of interest. Therefore, we need to identify and learn the principal components from the users who watch the video for a significant duration.

C. Supervised PCA

We now modify the PCA based approach by learning the principal components from users with significant viewing duration.

1) Proposed Approach: We argue that users who watch the video for a significant duration tend to have good video QoE while users who experience QoE impairments tend to abandon the video streaming session. Therefore, we consider users with viewing duration more than T_v minutes in every user group (identified by AS and device type). We then use PCA to learn principal components from the smaller subset of users and construct principal and residual subspaces for each user group. We then characterize users as anomalous based



(a) True positive rate (b) False positive rate

Fig. 10. ROC statistics for various QoE impairment detection schemes. While unsupervised PCA based scheme exhibits high true positive rate (TPR), false positive rate (FPR) is also high. We note that supervise PCA based schemes get higher TPR and low FPR rates compared to baselines, as we train the model on users with higher viewing time.

on the L^2-norm of the projection of their QoE metrics on the residual subspace in each time window. We further track the average L^2-norm of anomalous users ($(L^2_{avg})^{xy}_t$) over time and detect anomalies using Hampel filters. For our evaluation, we use spikes in user abandonments to characterize a detected impairment as either true positive or a false positive.

2) Evaluation: We calculate the TPR and FPR of the supervised PCA based QoE impairment detection methodology and plot the results in Figure 10. We vary the viewing duration threshold T_v during PCA training and plot the TPR and FPR in Figures 10(a) and 10(b). Specifically, we vary T_v from 30 minutes to 60 minutes in increments of 10 minutes. The last 4 bars in Figures 10(a) and 10(b) represent the TPR and FPR of the supervised PCA based method with varying T_v. We note in Figure 10(a) that TPR for supervised QoE impairment detection are comparable to unsupervised QoE impairment detection. For instance, we observe 47% TPR for supervised detection with $T_v = 30$ minutes as compared to 50% TPR with unsupervised detection. We further note that the TPR increases only slightly with increasing T_v with a maximum of 48% at $T_v = 60$ minutes.

From Figure 10(b), we note that the FPR decreases significantly with supervised detection compared to unsupervised detection. For instance, we observe 4.6% FPR with $T_v = 30$ minutes compared to the 8% FPR with unsupervised PCA based QoE impairment detection scheme. Furthermore, the FPR for supervised detection scheme decreases to 3% with $T_v = 60$ minutes. Overall, we observe more than 90% accuracy across our PCA based schemes and baselines. This is because as we increase T_v, the principal components capture the QoE metrics of the users who tend to watch the video longer. To conclude, we note that our supervised QoE impairment detection scheme provides better FPR compared to unsupervised QoE impairment detection. Our scheme allows content providers to be notified of QoE impairments in real-time and take mitigation actions on the fly.

V. Related Work

Extensive research has been conducted on different aspects of streaming video. However, prior studies of live video streaming are limited to mainly characterizing user access patterns and viewing behaviors rather than analyzing QoE impairments in light of user engagement. Below, we review related work on video QoE measurement and diagnosis.

Researchers have analyzed network-oriented QoS metrics (e.g., delay, packet loss, throughput) to study video streaming performance and user experience. Gill et al. characterized YouTube traffic patterns on a campus network [17]. They found that caching Web 2.0 metadata can improve bandwidth utilization and user experience. Finamore et al. found that most users abandon a video quickly and typically use default video player configurations [16]. Because of these early abandonments, a lot of data is spuriously downloaded in the client buffer due to an aggressive downloading scheme. With the widespread use of adaptive bitrate controller, however, this concern can be largely mitigated. Shafiq et al. analyzed QoS metrics such as throughput and radio signal-strength to study video abandonment in a cellular network [33]. They also proposed a predictive model to forecast individual user abandonments based on these QoS metrics. Casas et al. detected YouTube buffering events in a HSPA/3G network from network-layer measurements and mapped them to MOS values [10]. Chen et al. analyzed the impact of service quality metrics (e.g., buffering), network quality metrics (e.g., physical-layer data rate), video content (e.g., video length), and viewer demography (e.g., gender) on user engagement in a Wi-Fi network [11]. In contrast to these QoS-based studies, we analyze QoE metrics for adaptive live streaming video.

Some researchers have conducted small-scale user studies to measure and analyze QoE. Joumblatt et al. in [23] studied user satisfaction for a wide range of applications (e.g., YouTube, Firefox, etc.) and correlated them with end-host QoS measurements (e.g., RTT, jitter, etc.) and application contexts. Based on data from 19 users, they used supervised learning techniques to build user satisfaction predictors from lower layer metrics. In [20], Jackson et al. conducted a user study to measure QoE for streaming video. They calculated Mean Opinion Score (MOS) for videos that start from good quality but degrade over time and also for videos that start from bad quality but improve over time. They found that users reported higher satisfaction when video quality improved over time.

Other researchers (including ourselves [5]) have conducted large-scale "in the wild" studies to measure and analyze QoE. In [13], Dobrian et al. conducted a seminal study to understand the impact of video quality on user engagement. In our work, we borrow the QoE metrics proposed in their paper and also include rate of bitrate fluctuation as an additional QoE metric. In [24], Krishnan et al. noted that while QoE metrics impact user engagement, confounding factors render such correlational analysis inaccurate. We build on their work by leveraging a quasi-experimental framework to account for interdependencies among QoE metrics as well as other confounding factors such as device type and ISP affiliation. To complement their findings, we note that while buffering ratio (or rebuffer delay) impacts user engagement, rate of buffering and average bitrate exhibit the most causal impact on user engagement. Researchers have also tried to understand the root-causes of video quality problems. In [21], Jiang et al.

showed that most QoE impairments can be attributed to a small set of features such as CDN-AS combinations. They conducted an offline analysis of QoE impaired video streaming sessions and proposed mitigation strategies. In contrast, we propose an online technique to automatically detect QoE impairments. In [7], Balachandran et al. proposed machine learning based models to predict individual user abandonments. On the other hand, we use PCA for real-time, online detection of QoE impairments across groups of users. Furthermore, our technique allows content and network providers to learn principal components with data from past streaming events and detect QoE impairments for the current video streaming event.

VI. Conclusion

In this paper, we analyze QoE and its impact on user engagement for large-scale live video streaming. We study QoE for a live video streaming event that amassed over 600 thousand viewers. We make the following key contributions in this paper. First, we use a quasi-experimental framework to quantify the causal impact of different QoE metrics on user engagement. To this end, we control for both confounding factors and interdependencies among QoE metrics. We find that rate of buffering and average bitrate have the most impact on user engagement. Second, we use PCA and the Hampel filter for online detection of users experiencing QoE impairments. We find that users experiencing QoE impairments exhibit anomalous QoE metrics and lower engagement as compared to other users. We use Hampel filters to detect QoE impairments in real-time and validate detection performance by using subsequent user abandonments as ground truth. Our PCA-based approach is useful for content providers to detect QoE impairments and take mitigation actions in real-time.

Acknowledgements

This work is supported in part by the National Science Foundation under grant numbers CNS-1464110 and CNS-1617288.

References

[1] Conviva Consumer Survey Reports. http://www.conviva.com/conviva-customer-survey-reports.

[2] NCAA March Madness Live sets record with more than 80 million streams during 2015 NCAA tournament. http://www.ncaa.com/news/ncaa/article/2015-04-07/ncaa-march-madness-live-sets-record-more-80-million-streams-during-2015, 2015.

[3] Periscope #yearone. https://medium.com/@periscope/year-one-81c4c625f5bc, March 2016.

[4] Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2016-2021 White Paper, February 2017.

[5] A. Ahmed, Z. Shafiq, and A. Khakpour. QoE Analysis of a Large-Scale Live Video Streaming Event. In *ACM SIGMETRICS*, 2016.

[6] S. Akhshabi, A. C. Begen, and C. Dovrolis. An Experimental Evaluation of Rate-Adaptation Algorithms in Adaptive Streaming over HTTP. In *MMSys*, 2011.

[7] A. Balachandran, V. Sekar, A. Akella, S. Seshan, I. Stoica, and H. Zhang. Developing a Predictive Model of Quality of Experience for Internet Video. In *ACM SIGCOMM*, 2013.

[8] J. Brodkin. T-Mobile exempts video from data caps, but lowers resolution to 480p. http://bit.ly/1GW4n86, November 2015.

[9] J. Brodkin. Netflix throttles video on AT&T and Verizon to keep users under data caps. http://bit.ly/1Rq38NK, March 2016.

[10] P. Casas, M. Seufert, and R. Schatz. YOUQMON: A System for Online Monitoring of YouTube QoE in Operational 3G Networks. In *IFIP Performance*, 2013.

[11] Y. Chen, Q. Chen, F. Zhang, Q. Zhang, K. Wu, R. Huang, and L. Zhou. Understanding viewer engagement of video service in Wi-Fi network. *Computer Networks*, 91:101 – 116, 2015.

[12] L. Davies and U. Gather. The Identification of Multiple Outliers. *Journal of the American Statistical Association*, 88(423):782–792, 1993.

[13] F. Dobrian, A. Awan, D. Joseph, A. Ganjam, J. Zhan, V. Sekar, I. Stoica, and H. Zhang. Understanding the Impact of Video Quality on User Engagement. In *ACM SIGCOMM*, 2011.

[14] L. Eadicicco. 10 Facts About Twitch. http://www.businessinsider.com/statistics-about-twitch-2014-8, August 2014.

[15] D. Ewalt. How Big Is Twitch's Audience? Huge. http://www.forbes.com/sites/davidewalt/2014/01/16/twitch-streaming-video-audience-growth/, January 2014.

[16] A. Finamore, M. Mellia, M. M. Munaf, R. Torres, and S. G. Rao. YouTube Everywhere: Impact of Device and Infrastructure Synergies on User Experience. In *IMC*, 2011.

[17] P. Gill, M. Arlitt, Z. Li, and A. Mahanti. YouTube Traffic Characterization: A View From the Edge. In *IMC*, 2007.

[18] T.-Y. Huang, N. Handigol, B. Heller, N. McKeown, and R. Johari. Confused, Timid, and Unstable: Picking a Video Streaming Rate is Hard. In *ACM Internet Measurement Conference (IMC)*, 2012.

[19] T.-Y. Huang, R. Johari, N. McKeown, M. Trunnell, and M. Watson. A Buffer-Based Approach to Rate Adaptation: Evidence from a Large Video Streaming Service. In *Sigcomm*, 2014.

[20] F. Jackson, R. Amin, Y. Fu, J. E. Gilbert, and J. Martin. A User Study of Netflix Streaming. In *International Conference on Design, User Experience and Usability (DUXU)*, 2015.

[21] J. Jiang, V. Sekar, I. Stoica, and H. Zhang. Shedding Light on the Structure of Internet Video Quality Problems in the Wild. In *ACM Conference on Emerging Networking EXperiments and Technologies (CoNEXT)*, 2013.

[22] I. Jolliffe. *Principal Component Analysis*. John Wiley, 2014.

[23] D. Joumblatt, J. Chandrashekar, B. Kveton, N. Taft, and R. Teixeira. Predicting User Dissatisfaction with Internet Application Performance at End-Hosts. In *IEEE INFOCOM*, 2013.

[24] S. S. Krishnan and R. K. Sitaraman. Video Stream Quality Impact Viewer Behavior: Inferring Causality Using Quasi-Experimental Designs. In *ACM Internet Measurement Conference (IMC)*, 2012.

[25] A. Lakhina, M. Crovella, and C. Diot. Diagnosing Network-Wide Traffic Anomalies. In *ACM SIGCOMM*, 2004.

[26] Z. Li, X. Zhu, J. Gahm, R. Pan, H. Hu, A. C. Begen, , and D. Oran. Probe and Adapt: Rate Adaptation for HTTP Video Streaming At Scale. *IEEE Journal on Selected Areas in Communications*, 2014.

[27] X. Liu, F. Dobrian, H. Milner, J. Jiang, V. Sekar, I. Stoica, and H. Zhang. A Case for a Coordinated Internet Video Control Plane. In *SIGCOMM*, 2012.

[28] F. J. Massey. The Kolmogorov-Smirnov Test for Goodness of Fit. In *Journal of the American Statistical Association*, 1951.

[29] R. K. Pearson. Outliers in Process Modeling and Identification. *IEEE Transactions on Control Systems Technology*, 10(1), 2002.

[30] H. Ringberg, A. Soule, J. Rexford, and C. Diot. Sensitivity of PCA for Traffic Anomaly Detection. In *Sigmetrics*, 2007.

[31] P. Rosenbaum and D. Rubin. Constructing a control group using multivariate matched sampling methods that incorporate the propensity score. In *American Statistician, pages 33-38*, 1985.

[32] V. Satopaa, J. Albrecht, D. Irwin, and B. Raghavan. Finding a "Kneedle" in a Haystack: Detecting Knee Points in System Behavior. In *31st International Conference on Distributed Computing Systems Workshops*, 2011.

[33] M. Z. Shafiq, J. Erman, L. Ji, A. X. Liu, J. Pang, and J. Wang. Understanding the Impact of Network Dynamics on Mobile Video User Engagement. In *ACM SIGMETRICS*, 2014.

[34] Y. Sun, X. Yin, J. Jiang, V. Sekar, F. Lin, N. Wang, T. Liu, and B. Sinopoli. CS2P: Improving Video Bitrate Selection and Adaptation with Data-Driven Throughput Prediction. In *Sigcomm*, 2016.

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