Too Late for Playback: Estimation of Video Stream Quality in Rural and Urban Contexts

Vivek Adarsh[1(✉)], Michael Nekrasov[1], Udit Paul[1], Alex Ermakov[1], Arpit Gupta[1], Morgan Vigil-Hayes[2], Ellen Zegura[3], and Elizabeth Belding[1]

[1] University of California, Santa Barbara, Santa Barbara, USA
{vivek,mnekrasov,u_paul,aermakov,arpitgupta,ebelding}@cs.ucsb.edu
[2] Northern Arizona University, Flagstaff, USA
morgan.vigil-hayes@nau.edu
[3] Georgia Tech, Atlanta, USA
ewz@cc.gatech.edu

Abstract. The explosion of mobile broadband as an essential means of Internet connectivity has made the scalable evaluation and inference of quality of experience (QoE) for applications delivered over LTE networks critical. However, direct QoE measurement can be time and resource intensive. Further, the wireless nature of LTE networks necessitates that QoE be evaluated in multiple locations per base station as factors such as signal availability may have significant spatial variation. Based on our observations that quality of service (QoS) metrics are less time and resource-intensive to collect, we investigate how QoS can be used to infer QoE in LTE networks. Using an extensive, novel dataset representing a variety of network conditions, we design several state-of-the-art predictive models for scalable video QoE inference. We demonstrate that our models can accurately predict rebuffering events and resolution switching more than 80% of the time, despite the dataset exhibiting vastly different QoS and QoE profiles for the location types. We also illustrate that our classifiers have a high degree of generalizability across multiple videos from a vast array of genres. Finally, we highlight the importance of low-cost QoS measurements such as reference signal received power (RSRP) and throughput in QoE inference through an ablation study.

Keywords: QoE · Video streaming · Network measurement · LTE · Digital divide

1 Introduction

More than 60 million people reside in rural regions in the United States [18]. However, cellular deployment is often guided by economic demand, concentrating deployment in urban areas and leaving economically marginalized and sparsely populated areas under-served [27]. Few prior studies have focused on assessing mobile broadband in rural areas of the U.S.; there is a lack of accessible datasets that are not only comprehensive (include network-level and application-level

O. Hohlfeld et al. (Eds.): PAM 2021, LNCS 12671, pp. 141–157, 2021.
https://doi.org/10.1007/978-3-030-72582-2_9

traces) but also representative and inclusive of rural demographics. As a result of the COVID-19 pandemic, the assessment of the quality of experience (QoE) for applications delivered over mobile broadband has become urgent as stay-at-home orders and rapid movement to online schooling and work-from-home protocols increase the demand for applications that are known to be sensitive to network quality, such as video streaming and interactive video chat [50]. As a result, communities without access to usable, high speed broadband, such as many rural communities, are particularly disadvantaged [8,32].

Unfortunately, the evaluation of user quality of experience for video streaming applications accessed over LTE in regions where people are most likely to be smartphone dependent [27,28,34] poses a significant scalability challenge. QoE metric collection over LTE networks in a geographic area requires time and resource intensive measurements for each network provider. As a result, experiments at a single geographic point can be quite lengthy. Moreover, in rural areas, obtaining LTE Internet measurements in places where people are likely to use mobile broadband (e.g., at their homes or along local transportation corridors) can be challenging [49], as places of interest are far apart (requiring more resource intensive targeted measurement campaigns) and less densely populated (prohibiting representative crowd-sourcing measurement efforts). It is in this context that we ask the following research question: *How can we infer the QoE for video streaming applications over LTE at scale?*

While there are few to no existing datasets that measure QoE in rural communities, there are many public and proprietary datasets that report quality of service (QoS) metrics, such as reference signal received power (RSRP) or throughput. These metrics are typically reported independently and are measured over LTE networks in a wide range of locations throughout the U.S. and globally [46,51–53,59,63]. We argue that the wealth of LTE-QoS data points across the U.S. represents a key resource that can be leveraged to broadly assess QoE: while measuring QoE at scale in LTE networks presents significant challenges, measuring QoS at scale in LTE networks has already been demonstrated to be feasible. Hence, *our goal, and key contribution, is a methodology that can leverage low-cost QoS measurements to predict QoE.*

To study the correlation between mobile QoS and QoE performance, a diverse set of network measurements that are representative of a wide-range of conditions is needed. As such, we undertook an extensive measurement campaign to collect 16 datasets comprised of network traces from the Southwestern U.S. for four major telecom operators: AT&T, Sprint, T-Mobile and Verizon. Our datasets vary along two primary axes: population density, and network load. To obtain data from varied population densities, we collected LTE network measurements within multiple rural and urban communities. For variable network load, we collected LTE network traces from crowded events in urban locations that resulted in atypically high volumes of network utilization [5] and, as a result, congestion. We also collected traces from the same urban locations during typical operating conditions as a baseline. Our datasets have broad spatial and temporal variability, but can be classified into three primary categories: under-provisioned (rural), congested (congested urban), and well-provisioned (baseline urban).[1] We leverage these varied datasets to demonstrate the generality of the inference method. Based on our analysis, we show that predictive models can be used to infer video QoE metrics using low-cost QoS measurements, so that QoE can be more easily and scalably determined within difficult to assess regions.

Our key contributions and findings include:

- We collected sixteen measurement datasets[2] from twelve locations through an extensive ; ground measurement campaign within the Southwestern U.S. Our data points are representative of three different network conditions: under-provisioned (rural), congested urban and well-provisioned urban, and include over 32 Million LTE packets. (Sect. 2);
- We develop and evaluate a comprehensive set of predictive models that infer video QoE from low-cost QoS measurements such as RSRP and throughput. Our analysis reveals that predictive models can infer video QoE with an accuracy of at least 80% across all locations and network types (Sect. 3);
- We validate our models across multiple video types from a wide variety of genres. Further, we demonstrate the utility of low-cost RSRP measurements for inferring video QoE (Sect. 3).

2 Methodology and Datasets Overview

QoS metrics, such as received signal strength, latency, throughput, and packet loss, capture the state of network connectivity. However, while QoS provides an indication of network state, there can be a disconnect between QoS and user experience. QoS network metrics are not Pareto-optimal; one element can get better or worse without affecting the other. Consequently, estimation of user experience requires the incorporation of multiple network measures, which may be unique to time, space and application. Note that while the definition of QoE can vary depending on the vantage point from which measurements are taken, we only focus on application-level QoE. Our measurements are active end-user device/passive user as defined in [61].

2.1 QoS and QoE Metrics

In this section, we describe the QoS and QoE metrics we collected (and estimated) for this measurement study, as summarized in Table 1.

Quality of Service Metrics: We collect *reference signal received power (RSRP)* and *throughput* synchronously on the same user equipment (UE). RSRP is defined as the linear average over the power contributions (in Watts) of the

[1] Through extensive analysis, we verified that our datasets are representative of the network characteristics we anticipated: well-provisioned, congested, and/or under-provisioned. We omit that analysis from this paper due to space constraints.

[2] The subset of our dataset that we have permission to release is available at [4].

Table 1. Overview of QoS and QoE metrics at each location, aggregated across available providers.

Type	Metric	Test Interval	Number of Datapoints	Tools
QoS	RSRP	1 second	2160	Network Monitor
	Throughput	1 second	2160	iPerf
QoE	Video resolution	1 second	2160	Selenium, iframe API
	Resolution switches	1 second	2160	Selenium, iframe API
	Rebuffering events	1 second	2160	Selenium, iframe API

resource elements that carry cell-specific reference signals within the measurement frequency bandwidth [2] and, as illustrated by [7], is widely accessible through mobile operating systems. We record instantaneous RSRP readings from the UEs every one second through the Network Monitor application [43]. We measure throughput by fetching a pre-specified 500 MB file from an AWS instance in Virginia using iPerf over TCP to download the file. The large file size allows the data traffic to fill the pipe and to minimize the effect of slow start. We log the packet traces at the client during the iPerf tests in order to sample throughput at 1 s intervals.

Quality of Experience Metrics: We focus on streaming video, currently the most heavily used QoE-centric service in mobile networks [36]. Internet video streaming services typically use Dynamic Adaptive Streaming over HTTP (DASH) [60] to deliver a video stream. DASH divides each video into time intervals known as segments or chunks, which are then encoded at multiple bit rates and resolutions. To analyze video stream quality, we gather two QoE metrics: *resolution switches* and *rebuffering events*. For resolution switches, we compute the number of consecutive samples that had a different resolution as a percentage of the total number of samples collected during the video. We measure at one-second granularity, which captures resolution switches that happen between video *chunks* that are typically 4–5 s long [15]. Finally, a rebuffering event occurs when video pauses while the application buffer waits to accumulate enough content to resume playback. We record the video state (rebuffering event or normal playback) every second.

2.2 Measurement Suite

We run our measurement suite on Lenovo ThinkPad W550s laptops, each of which are tethered to their own Motorola G7 Power (Android 9) via USB in order to measure cellular performance. The cellular plans on all our cellular user equipment (UE) have unlimited data and are hot-spot enabled to effectively achieve the same level of performance as we would on the mobile device. We run our measurement suite on laptops tethered to phones; this configuration gives us the same application performance while facilitating ease of programming, data extraction, and unification of application-level measurements.

We choose YouTube as the streaming platform because of its popularity in the U.S., capturing over 88% of the mobile market [62]. To collect video QoE metrics, we run a 3-min clip of a Looney Tunes video [64], three times across each of the four LTE providers at each location; we exclude from our results the sessions that experienced playback errors during execution. We chose this particular video due its mix of high and low action scenes, which result in variable bitrates throughout the video (typically, high action scenes have a higher bitrate than low action scenes). After testing multiple playback duration, we observed that a 3-min window was adequate for the playback to reach steady state, while long enough to capture rebuffering and/or resolution switches that occur. To infer video QoE, we collect the input features (RSRP and throughput) synchronously, on a separate device so as not to bias the video streaming measurements. Synchronous measurements of throughput, RSRP and QoE metrics are required to train learning algorithms to infer video QoE for a future time instance. We use different servers for throughput and YouTube tests so that we can obtain concurrent QoS and QoE measurements. Our setup reflects the real world scenario where throughput test servers and YouTube servers are separate while simultaneously affected by varying conditions from *within* the cellular network [6]. In LTE, each bearer (connection from a UE) enjoys a relatively isolated data tunnel before the egress from the packet gateway, located inside the core [1]. This reduces contention among UEs competing for resources at a single eNodeB, and as a result we can accurately record QoS and QoE metrics on two separate devices.

To execute this experiment, we first automate the loading and playback of the YouTube video on the Chrome browser using Selenium [58]. The video resolution is set to auto. Then we use YouTube's iframe API [65] to capture playback events reported by the video player. The API outputs a set of values that indicate player state (not started, paused, playing, completed, buffering) using the getPlayerState() function. The API also provides functions for accessing information about play time and the remaining buffer size.

2.3 Description of Datasets

We collect 16 datasets from 12 locations across the Southwestern U.S. Eight of the datasets were collected from rural locations that had sparse cellular deployment.

An additional eight datasets were collected from four urban locations. In each urban location, we collect two datasets: one during a large event or gathering, in which we expect cellular network congestion to occur (these datasets are marked with **_Cong**); and a second during typical operating conditions. We call the latter dataset the baseline for that location (these datasets are marked with **_Base**). Hence, our 16 traces are broadly classified into three categories: rural, congested urban, and baseline urban. The details of each dataset are summarized in Table 2. The designation of each location as rural or urban is based on Census Bureau data [57]. Through these measurement campaigns, we collect and analyze over 32.7 Million LTE packets. Note that the "Number of Datapoints" column shown in Table 1 indicates the QoS/QoE datapoints gathered by the application, while the "# LTE Packets" column in Table 2 refers to the number of packets collected in the trace files.

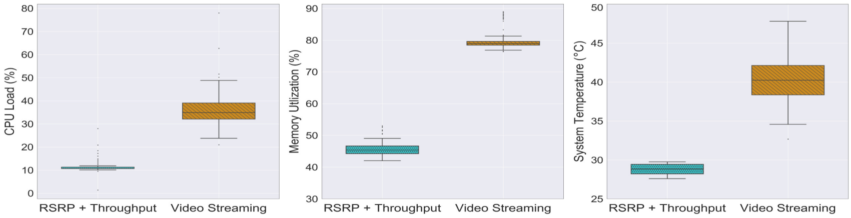
Table 2. Summary of datasets

Location	Date	# LTE Packets	Type	Carriers*
Rural_1	May 28 2019	3.18 Million	Rural	V,A,T,S
Rural_2	May 29 2019	1.38 Million	Rural	V,T
Rural_3	May 28 2019	2.03 Million	Rural	V,A,T,S
Rural_4	May 30 2019	2.16 Million	Rural	V,A,T,S
Rural_5	May 30 2019	2.27 Million	Rural	V,A,T,S
Rural_6	May 31 2019	2.33 Million	Rural	V,A,T,S
Rural_7	May 31 2019	1.26 Million	Rural	V,T
Rural_8	Jun 01 2019	2.83 Million	Rural	V,A,T,S
Urban_1_Cong	Sep 22 2019	2.25 Million	Urban, Congested	V,A,T,S
Urban_1_Base	Sep 28 2019	1.92 Million	Urban, Baseline	V,A,T,S
Urban_2_Cong	Sep 29 2019	2.51 Million	Urban, Congested	V,A,T,S
Urban_2_Base	Sep 30 2019	1.97 Million	Urban, Baseline	V,A,T,S
Urban_3_Cong	Sep 21 2019	2.65 Million	Urban, Congested	V,A,T,S
Urban_3_Base	Sep 30 2019	2.13 Million	Urban, Baseline	V,A,T,S
Urban_4_Cong	Sep 25 2019	2.18 Million	Urban, Congested	V,A,T,S
Urban_4_Base	Sep 26 2019	2.08 Million	Urban, Baseline	V,A,T,S

*This column lists mobile carriers in each data set (some areas had no coverage for particular network operators). V: Verizon, A:AT&T, T:T-Mobile, S: Sprint.

2.4 Video QoE Measurement Scalability Challenges

Collection of *ground-truth* cellular network measurements, as we explore further in Sect. 4, is a challenging task for multiple reasons. First, it requires physical placement of measurement device at the location to be studied. While there are many large, publicly accessible datasets that incorporate some QoS measurements, QoE measurements, particularly in remote regions, are much more difficult. Second, gathering ground truth data to assess video QoE requires an active connection to stream a large encoded video file. This consumes a substantial amount of bandwidth, computational power, memory, and battery, due to the simultaneous use of LTE modems, display, CPU, and GPU [21] on the user device. For instance, streaming applications consume memory to load the video and require accelerated processing to decode and display the stream from the video server. Unlike QoS metrics, which can often be collected in the background through execution by back-end scripts, the high resource cost of QoE measurements for the end user makes this data difficult to crowd-source. In Fig. 1 we show the resource consumption during one hour of RSRP and throughput (QoS) measurements, compared to one hour of video streaming (QoE), on our data collection phones. As can be seen in the figure, the resources consumed by the QoE measurements were significantly higher, both preventing background data collection and more rapidly draining the device battery.



a. CPU Load b. Memory utilization c. System temperature

Fig. 1. Device resource consumption during either RSRP and throughput measurements only, or during video streaming.

Rural regions span large geographic areas with terrain that is often hard to access. QoS data from public sources already struggles to cover these areas. In particular, crowd-sourced datasets are data-rich in regions where there are higher density populations. These regions tend to be either urban areas, or other areas frequented by travelers (i.e. highways, national parks, etc.). Rural communities, by contrast, with their lower population densities, are often under-represented in crowd-sourced datasets. Yet it is exactly these regions where under-provisioned networks typically exist and hence where data is urgently needed. In order to effectively assess QoE in these remote areas, we need a method to improve QoE measurement scalability. We address this challenge in the next section, where we show how predictive models can use the less resource expensive QoS measurements to infer QoE for streaming video on mobile broadband networks in a variety of environments.

3 Inferring QoE Metrics for Video

As discussed in Sect. 2.4, the collection of QoS measurements is less resource consumptive, and hence more scalable, than video QoE measurements. We now describe our approach to infer QoE metrics for video streaming sessions using low-cost QoS metrics.

3.1 Learning Problem

Our learning problem's goal is to infer QoE metrics using a sequence of throughput and RSRP (QoS metrics) data input. The objective is to build models with appreciable performance that would work in a wide variety of network conditions and different region types (e.g., rural and urban locations). These models could be used to predict application QoE (in our case, video streaming) at a particular location. We use supervised learning to train two different binary classifiers. The first classifier infers whether the video's state is stalled or normal; the second infers whether there is any change in video resolution. Both models perform the classification task every one second.

Input: The learning model takes a sequence of RSRP and throughput values as input. Both of these metrics are low-cost measurements and easily accessible. Given how adaptive bitrate (ABR) video streaming players operate, the changes in throughput and RSRP values have a delayed impact on QoE metrics. For example, a decrease in available throughput will force the video streaming player to use the buffered data before stalling.

As part of feature engineering, we had to determine how many RSRP and throughput values to use as input for the learning model. Intuitively, the use of longer sequences will improve accuracy. However, longer sequences also increases the complexity of the learning model, which requires more training data to avoid over-fitting. After varying $n = 0 \rightarrow 180$ (total playback time of a session), we found that using a sequence of *three* throughput/RSRP values enabled us to strike a balance between model complexity and accuracy. A typical approach to assessing throughput would be to log continuous measurements for a long duration of time and analyze the resulting mean/mode of the distribution. However, our results (Sect. 3.3) indicate that we can infer the video quality from only a 3-s sample. This has the added benefit of reducing the resource utilization at the client device, such as data consumption and battery drainage, while accurately inferring the video stream quality.

Output: We train two separate binary classifiers to predict the video state and change in resolution at the granularity of one second. Predicting QoE metrics at such fine granularity enables opportunities to infer QoE with limited training data. Given the input features, our models infer how likely it is for the video stream to experience either a video stall or a resolution change in the next instant.

Training Data: Our dataset consists of 32,596 data points. Each data point has input values: a sequence of three RSRP and throughput values, as well as two boolean labels: video state (playing or stalled) and resolution switches (yes–resolution will change; no–resolution will not change). We collected this dataset through our measurement campaign by conducting a total of 181 video streaming sessions across multiple locations (Sect. 2.3). For each classifier, we label the output training samples into either of the two classes: class 0 is when playback is normal and devoid of any event (rebuffering or resolution switch), and class 1 is when there is an event. We carried out the classification task by splitting the entire dataset into a ratio of 70:30 training to test sets, as described in Table 3. We split the overall training dataset into training and validation sets (80:20). We chose the samples proportionate to the size of each dataset category (rural, congested urban, and baseline urban). We present the models' performance per location, where we train the models on specific locations and then test on others not included in the training. We do not make any distinctions between operators since an operator-agnostic evaluation is a more comprehensive reflection of coverage and QoE at a particular location.

Table 3. Breakdown of training and test set samples for both classifiers.

		Training Set		Test Set	
Classifier Type	Target Metric	Class 0	Class 1	Class 0	Class 1
Classifier 1	Rebuffering Event	22,175	642	9,504	275
Classifier 2	Resolution Switching	22,490	327	9,639	140

3.2 Learning Algorithm

We now present the learning models we used for the learning problem, our model training approach, and the method for addressing the inherent class-imbalance.

Learning Models: We trained a wide range of off-the-shelf classifiers for this learning problem in order to identify the classifier that strikes the best balance between performance (precision, recall, etc.) and generalizability. First, we trained simpler classifiers, such as gradient boosting [29], bagging [13], random forest [14], ARIMA [12], AdaBoost [30], etc. These classifiers offer better generalizability at the cost of performance. We also trained neural-network (NN)-based classifiers, such as a convolutional neural network (CNN) [41] and recurrent neural network (RNN) [37] (in particular, LSTMs [35] and GRUs [23]), that offer higher accuracy but require considerable training data to avoid over-fitting.

Setup: We ran all the classifiers on a local machine that runs Ubuntu 18.04, powered by a 4-core i7-7700 CPU (3.60 GHz) with 64,GB RAM and 8 GB NVIDIA RTX 2080 GPU. We implemented the simpler classifiers using the scikit-learn 0.21 [56] library of Python, and NN-based models using Keras with Tensorflow backend [24]. We used four fully-connected layers for the NN-based classifiers. For *RNN-LSTM-Focal* (see Table 4), the network utilized 64, 32, and then 16 hidden neurons, in addition to a final output layer with hyperbolic tangent activation function. We used Grid Search [25] to determine the ideal hyper-parameter configuration for each neural network. To avoid over-fitting, we use a dropout of 0.4 while training with the Adam gradient descent optimizer [39]. We ran the RNN-LSTM model for 120 iterations with a batch size of 64.

Class-Imbalance Problem: As rebuffering and changes in the resolution are rare, most of our data points are normal, i.e., they do not have any rebuffering or resolution switching events. As a result, our dataset has the class-imbalance problem, typical for most anomaly detection problems. To address this issue, we applied the sampling technique SMOTE [19] to balance the classes artificially. However, such an approach reduces the number of data points that we can use for training the classifier, which in turn affects the accuracy. With SMOTE, we observed no improvements in accuracy with simpler learning models (e.g., SVM, random forest, etc.), and lower accuracy for NN-based classifiers. Therefore, for the NN-based classifiers, we adapted a new technique that has proven to increase classification accuracy in datasets that suffer from the class-imbalance issue for the object detection problem [42]. This technique addresses the class-imbalance problem by reshaping the standard cross entropy loss in such a way that it lowers the weights for the majority class [42]. It also introduces the concept of *focal loss* that prevents the majority class from overwhelming the classifier during training. The focal loss can be represented as:

$$FL(p_j) = \alpha(1 - p_j)^{\gamma} log(p_j) \tag{1}$$

Here, FL is the focal loss function, and p_j is the softmax probability of the j^{th} class for a particular observation. α and γ are two regularizing parameters. This loss function adds more importance when the network predicts a minority sample as opposed to the overly represented sample—making it ideal for performing classification on an imbalanced dataset.

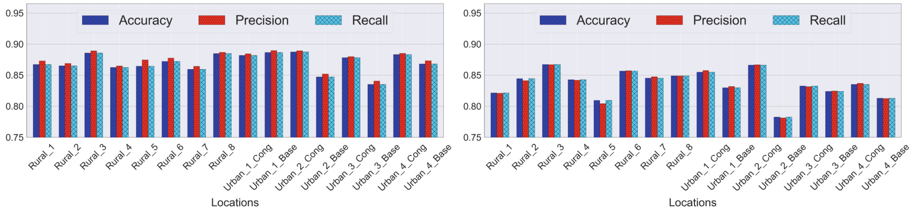
3.3 Results

We now present the performance of the different classifiers we used for this learning problem. For those that performed well, we also quantify their performance across different locations and video types. Finally, we quantify the contribution of an LTE-specific QoS metric, RSRP, in improving the accuracy of our learning models.

Table 4. Performance metrics of the classification models.

	Rebuffering Events			**Resolution Switching**		
Models	Accuracy	Precision	Recall	Accuracy	Precision	Recall
Boosting	0.87	0.88	0.88	0.84	0.85	0.84
Bagging	0.80	0.82	0.82	0.71	0.73	0.72
Random Forest	0.85	0.87	0.86	0.79	0.80	0.80
ARIMA	0.81	0.81	0.81	0.77	0.78	0.78
Decision Trees	0.80	0.80	0.98	0.75	0.75	0.75
Extra Randomized Tree	0.77	0.78	0.77	0.72	0.73	0.72
AdaBoost	0.62	0.60	0.63	0.51	0.55	0.53
Support Vector Machine	0.72	0.72	0.73	0.70	0.71	0.70
K-nearest neighbors	0.60	0.56	0.62	0.58	0.57	0.49
CNN	0.72	0.73	0.73	0.68	0.69	0.69
CNN - Focal	0.84	0.85	0.84	0.81	0.81	0.81
RNN - LSTM	0.82	0.83	0.83	0.80	0.79	0.80
RNN - LSTM - Focal	0.89	0.89	0.89	0.86	0.86	0.87
RNN - GRU	0.82	0.82	0.84	0.80	0.82	0.82
RNN - GRU - Focal	0.86	0.86	0.85	0.83	0.84	0.84

Performance: We analyze the performance of learning models in terms of accuracy, precision, recall, and training time. Table 4 summarizes the performance of all classifiers we explored. We observe that the accuracy of the rebuffering-event classifier is better than the resolution-switching one, as depicted in Fig. 2. This difference is attributable to the smaller number of anomalous data points (resolution switches) in the data (see Table 3). In terms of accuracy, *RNN-LSTM-Focal* performs best. This is expected as this model makes the best use of the sequence of throughput and RSRP values and is best suited to handle the class imbalance problem. On the other hand, though *RNN-LSTM-Focal* has the highest accuracy, the accuracy gains are marginal when compared to simpler learning models, especially *Boosting*. Given these marginal gains and the complexity of training NN-based classifiers (5 vs. 214 s), we use the *Boosting* classifier to characterize the performance across different network and video types.



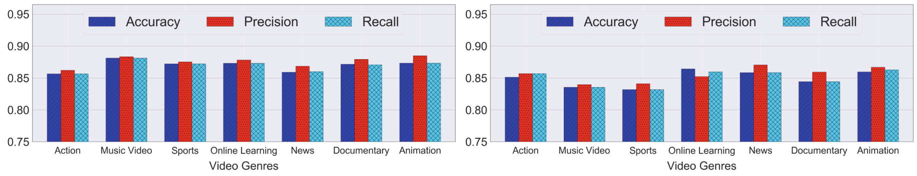
a. Rebuffering events

b. Resolution switching

Fig. 2. Performance of *Boosting* across different locations.

Generalizability: We now quantify the generalizability of the *Boosting* classifier. First, we show how its performance varies across different network types. Figure 2 depicts the performance of inferring video rebuffering using *Boosting* at each location. We observe that the performance differences across different network types are marginal (<2% deviation between categories). We saw similar trends for the *Boosting*-based classifier when inferring resolution switching.

Our initial measurements only collected the QoE metrics for the Looney Tunes video. To verify that our results generalize for other video types, we collected the QoS/QoE data for 108 additional video streaming sessions (a total of 48,825 new data points) at our research facility (baseline-urban). We selected 18 different videos from seven genres: action (trailers/movie clips), music videos, sports, online learning content, news, documentary, and animation (including the original Looney Tunes video) [16]. We selected top trending videos for each genre. Given that the videos were of varying duration, we capped each measurement to a maximum of ten minutes. We streamed each video over three different telecom providers (AT&T, T-Mobile, and Verizon); we were not able to obtain Sprint measurements because of closures of Sprint retail outlets due to the COVID-19 pandemic. Figure 3 shows the performance of *Boosting* for both video rebuffering and resolution switching. We observe marginal variations (<1.5% and <3% deviation for rebuffering and resolution switching, respectively) in accuracy across different video genres, implying that our learning model generalizes reasonably well to different video types. Note that we do not claim that these results generalize for other video players (e.g., Hulu, Netflix), client platforms or devices; we plan to quantify the performance of our learning models for other platforms, devices and non-YouTube videos in the future. Finally, we do not claim to have developed models that generalize across other locations or network conditions – rather we use this study to demonstrate the feasibility of inferring video QoE *at scale* within a limited, but diverse, dataset.



a. Rebuffering events

b. Resolution switching

Fig. 3. Performance of *Boosting* across different video genres.

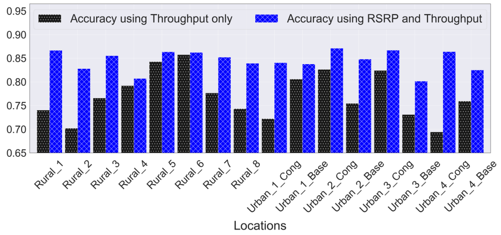


Fig. 4. Inferring video rebuffering using *Boosting* with and without RSRP as an input feature.

Ablation Study: To better understand the impact of an LTE-specific metric (i.e., RSRP) in inferring QoE metrics, we performed an ablation study. Figure 4 compares the accuracy of the *Boosting* classifier in inferring rebuffer events with and without the RSRP values. We observe that the average increase in accuracy, with RSRP as an input, is 9.28%, while the maximum gain is 18.61%. This result could be attributed to the exposition of the relationship, by the non-linear models, between RSRP and throughput to identify the target metrics at any given location successfully. This study highlights the importance of LTE-specific RSRP measurements in accurate prediction of rebuffering and resolution switching.

4 Related Work

Prior work most similar to ours, which focuses on quantifying the user experience, typically infers the QoE of video streaming from QoS of fixed broadband networks [22,31,38]. In contrast, our work focuses on mobile broadband, which often exhibits a wide variation in performance over time and space. Some past work on mobile broadband, such as [3,11,20,54], has examined metrics solely from the application and network layers. [15,26,33,40,44,45] require direct access to (encrypted or unencrypted) network traffic to infer video QoE. In contrast, our approach is independent of network traces and incorporates low-cost signal and throughput measurements for rapid QoE prediction. Few publicly available QoS datasets include synchronous RSRP measurements. [17,48,63] analyze network traces that contain performance indicators captured during streaming sessions, and experiment metadata from mobile broadband networks. All of these datasets, however, have limited types of datapoints (primarily from dense, urban locations); the datasets have minimal to no measurements from networks that are under-provisioned or located in remote regions. We believe it is challenging to utilize existing prior datasets (from primarily urban scenarios) to evaluate diverse network conditions in the context of the measurements examined in this work, either due to non-overlapping and non-scalable nature of prior measurements or lack of a comprehensive and representative dataset. Further, the accuracy of our models, given the inexpensive measurements, indicates the feasibility and scalability of our approach.

Prior work that has focused on charting the relationship between RSRP and QoE has important limitations. For instance, [10] presents a mapping of RSRP and video QoE that is derived using only simulated experiments. The authors of [47] explore the effect of radio link quality, such as RSRP, on streaming video QoE. The presented results are limited in scope as their setup streams a custom video hosted on their own server; by omitting evaluation of a popular streaming service, such as YouTube or Netflix, the work does not accurately capture the application and network performance experienced by actual users. [9] undertakes a study similar to ours, however, with a modest dataset that is limited to a small portion of a local transit route and is thus difficult to generalize.

5 Conclusion

Through an extensive measurement campaign, we collect 16 datasets with widely varying performance profiles. Our dataset includes representation of: i) the variability of mobile broadband performance as a consequence of either sparse deployments or network congestion, and ii) the communities most likely to be dependent on mobile broadband (rural areas). Through our analysis, we highlight the challenges of quantifying QoE metrics at scale, particularly in remote locations. To address this challenge, we develop learning models that use low-cost and easily accessible QoS data (LTE-specific RSRP and throughput) to predict QoE metrics. Our models can be generalized to video content from different genres, as well as to other locations that share network characteristics similar to those of our dataset. The observed efficacy of the models indicates that video QoE can be more easily and scalably determined within difficult to assess regions, using low-cost QoS measurements. For instance, given the increased load on video streaming platforms during COVID-19 [50], cellular operators could employ our approach to detect sectors with possible bottlenecks without having to rely on user feedback/complaints, particularly in remote locations. This has the potential to lead to faster turnaround times for network troubleshooting [55], and therefore may lower outage periods for users heavily dependent on video streaming.

Acknowledgments. Our work would not have been possible without the incredible support of Jerrold Baca. We wish to thank our shepherd, Marco Fiore and the anonymous PAM reviewers for their valuable feedback on the paper. This work was funded through the National Science Foundation Smart & Connected Communities award NSF-1831698.

References

1. 3GPP TR 29.281: LTE General Packet Radio System (GPRS) and Tunnelling Protocol User Plane (GTPv1-U), July 2018
2. 3GPP TS 136.214: Evolved Universal Terrestrial Radio Access (E-UTRA); Physical layer - Measurements, April 2010
3. Abdellah, S., Sara, M., El-Houda, M.N., Samir, T.: QoS and QoE for mobile video service over 4G LTE network. In: IEEE Computing Conference, pp. 1263–1269. IEEE (2017)
4. Adarsh, V.: Dataset for QoE Analysis (2021). https://github.com/videostream-ML/urban_rural_qoe
5. Adarsh, V., Nekrasov, M., Zegura, E., Belding, E.: Packet-level overload estimation in LTE networks using passive measurements. In: Proceedings of the Internet Measurement Conference, pp. 158–164 (2019)
6. Adarsh, V., Schmitt, P., Belding, E.: MPTCP performance over heterogenous subpaths. In: 28th International Conference on Computer Communication and Networks (ICCCN), pp. 1–9. IEEE (2019)
7. Alimpertis, E., Markopoulou, A., Butts, C., Psounis, K.: City-wide signal strength maps: prediction with random forests. In: The World Wide Web Conference, pp. 2536–2542. WWW (2019)
8. Amanda Holpuch (The Guardian): US's Digital Divide 'is going to kill people' as COVID-19 exposes inequalities. https://www.theguardian.com/world/2020/apr/13/coronavirus-covid-19-exposes-cracks-us-digital-divide. Accessed 05 Oct 2020
9. Anchuen, P., Uthansakul, P.: Investigation into user-centric QoE and network-centric parameters for YouTube service on mobile networks. In: Proceedings of the 7th International Conference on Communications and Broadband Networking, pp. 28–32 (2019)
10. Awad, N., Mkwawa, I.: The impact of the reference signal received power to quality of experience for video streaming over LTE network. In: Annual Conference on New Trends in Information Communications Technology Applications (NTICT), pp. 192–196 (2017)
11. Begluk, T., Husić, J.B., Baraković, S.: Machine learning-based QoE prediction for video streaming over LTE network. In: 17th International Symposium Infoteh-Jahorina (InfoTeh), pp. 1–5. IEEE (2018)
12. Box, G.E., Pierce, D.A.: Distribution of residual autocorrelations in autoregressive-integrated moving average time series models. J. Am. Stat. Assoc. **65**(332), 1509–1526 (1970)
13. Breiman, L.: Bagging predictors. Machine Learn. **24**(2), 123–140 (1996)
14. Breiman, L.: Random forests. Machine Learn. **45**(1), 5–32 (2001)
15. Bronzino, F., Schmitt, P., Ayoubi, S., Martins, G., Teixeira, R., Feamster, N.: Inferring streaming video quality from encrypted traffic: practical models and deployment experience. In: Proceedings of the Measurement and Analysis of Computing Systems (2019)
16. Bärtl, M.: YouTube channels, uploads and views: a statistical analysis of the past 10 years. Convergence **24**(1), 16–32 (2018)
17. Casas, P., et al.: Predicting QoE in cellular networks using machine learning and in-smartphone measurements. In: 9th International Conference on Quality of Multimedia Experience (QoMEX), pp. 1–6 (2017)
18. Census Bureau: Differences Between Urban and Rural Populations (2016). https://www.census.gov/newsroom/press-releases/2016/cb16-210.html

19. Chawla, N.V., Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P.: SMOTE: synthetic minority over-sampling technique. J. Artif. Intell. Res. **16**, 321–357 (2002)
20. Chen, H., Yu, X., Xie, L.: End-to-end quality adaptation scheme based on QoE prediction for video streaming service in LTE networks. In: 11th International Symposium and Workshops on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks (WiOpt), pp. 627–633. IEEE (2013)
21. Chen, X., Ding, N., Jindal, A., Hu, Y.C., Gupta, M., Vannithamby, R.: Smartphone energy drain in the wild: analysis and implications. In: Proceedings of the ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems. SIGMETRICS (2015)
22. Chen, Y., Wu, K., Zhang, Q.: From QoS to QoE: a tutorial on video quality assessment. IEEE Commun. Surv. Tutor. **17**(2), 1126–1165 (2014)
23. Cho, K., et al.: Learning phrase representations using RNN encoder-decoder for statistical machine translation. arXiv preprint arXiv:1406.1078 (2014)
24. Chollet, F.: Keras (2019). https://github.com/keras-team/keras
25. Cournapeau, D.: Tuning the Hyper-Parameters of an Estimator (2019). https://scikit-learn.org/stable/modules/grid_search.html
26. Dimopoulos, G., Leontiadis, I., Barlet-Ros, P., Papagiannaki, K.: Measuring video QoE from encrypted traffic. In: Proceedings of the Internet Measurement Conference. IMC (2016)
27. Federal Communications Commission: Broadband Deployment Report, February 2018. https://www.fcc.gov/reports-research/reports/broadband-progress-reports/2018-broadband-deployment-report
28. Federal Communications Commission: Broadband Deployment Report, May 2019. https://www.fcc.gov/reports-research/reports/broadband-progress-reports/2019-broadband-deployment-report
29. Freund, Y., Schapire, R., Abe, N.: A short introduction to boosting. Japanese Society Artif. Intell. **14**(771–780), 1612 (1999)
30. Freund, Y., Schapire, R.E.: A desicion-theoretic generalization of on-line learning and an application to boosting. In: Vitányi, P. (ed.) EuroCOLT 1995. LNCS, vol. 904, pp. 23–37. Springer, Heidelberg (1995). https://doi.org/10.1007/3-540-59119-2_166
31. Goran, N., Hadžialić, M.: Mathematical bottom-to-up approach in video quality estimation based on PHY and MAC parameters. IEEE Access **5**, 25657–25670 (2017)
32. Grant Samms (Forbes): As Cities Face COVID-19, The Digital Divide Becomes More Acute, April 2020. https://www.forbes.com/sites/pikeresearch/2020/04/02/as-cities-face-covid-19-the-digital-divide-becomes-more-acute/#277c93e558c5. Accessed 05 Oct 2020
33. Gutterman, C., et al.: Requet: real-time QoE detection for encrypted YouTube traffic. In: Proceedings of the 10th ACM Multimedia Systems Conference, pp. 48–59 (2019)
34. Hansi Lo Wang (NPR): Native Americans On Tribal Land Are 'The Least Connected' To High-Speed Internet, December 2018. https://www.npr.org/2018/12/06/673364305/native-americans-on-tribal-land-are-the-least-connected-to-high-speed-internet
35. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Comput. **9**(8), 1735–1780 (1997)

36. Hoßfeld, T., Seufert, M., Sieber, C., Zinner, T.: Assessing effect sizes of influence factors towards a QoE model for HTTP adaptive streaming. In: 6th International Workshop on Quality of Multimedia Experience (QoMEX), pp. 111–116. IEEE (2014)
37. Jordan, M.I.: Attractor dynamics and parallelism in a connectionist sequential machine. In: Artificial Neural Networks: Concept Learning, pp. 112–127 (1990)
38. Kim, H.J., Choi, S.G.: A study on a QoS/QoE correlation model for QoE evaluation on IPTV service. In: 12th International Conference on Advanced Communication Technology (ICACT), vol. 2, pp. 1377–1382. IEEE (2010)
39. Kingma, D., Ba, J.: Adam: a method for stochastic optimization. In: International Conference on Learning Representations (2014)
40. Krishnamoorthi, V., Carlsson, N., Halepovic, E., Petajan, E.: BUFFEST: predicting buffer conditions and real-time requirements of HTTP (S) adaptive streaming clients. In: Proceedings of the 8th ACM on Multimedia Systems Conference, pp. 76–87 (2017)
41. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems, pp. 1097–1105 (2012)
42. Lin, T., Goyal, P., Girshick, R.B., He, K., Dollár, P.: Focal Loss for Dense Object Detection. CoRR abs/1708.02002 (2017). http://arxiv.org/abs/1708.02002
43. Lubek, B.: Network Monitor. https://github.com/caarmen/network-monitor
44. Mangla, T., Halepovic, E., Ammar, M., Zegura, E.: MIMIC: using passive network measurements to estimate HTTP-based adaptive video QoE metrics. In: 2017 Network Traffic Measurement and Analysis Conference (TMA) (2017)
45. Mangla, T., Halepovic, E., Ammar, M., Zegura, E.: eMIMIC: estimating HTTP-based video QoE metrics from encrypted network traffic. In: 2018 Network Traffic Measurement and Analysis Conference (TMA) (2018)
46. Midoglu, C., Moulay, M., Mancuso, V., Alay, O., Lutu, A., Griwodz, C.: Open video datasets over operational mobile networks with MONROE. In: Proceedings of the 9th ACM Multimedia Systems Conference, pp. 426–431 (2018)
47. Minovski, D., Åhlund, C., Mitra, K., Johansson, P.: Analysis and estimation of video QoE in wireless cellular networks using machine learning. In: 11th IEEE International Conference on Quality of Multimedia Experience (QoMEX), pp. 1–6 (2019)
48. MONROE: MONROE Video Dataset (2018). https://doi.org/10.5281/zenodo.1230448
49. Nekrasov, M., et al.: Evaluating LTE coverage and quality from an unmanned aircraft system. In: Proceedings of the 16th IEEE International Conference on Mobile Ad-Hoc and Smart Systems (2019)
50. Nielsen Insights: Streaming Consumption Rises in U.S. Markets With Early Stay-at-home Orders During COVID-19 (2020). https://www.nielsen.com/us/en/insights/article/2020/streaming-consumption-rises-in-u-s-markets-with-early-stay-at-home-orders-during-covid-19/
51. Ookla: Mobile Speedtest Intelligence Data (2019). https://www.speedtest.net/reports/united-states/
52. Open Signal: Open Signal 3G and 4G LTE Cell Coverage Map (2016). http://opensignal.com
53. OpenCelliD: The World's Largest Open Database of Cell Towers (2020). https://opencellid.org/

54. Orsolic, I., Pevec, D., Suznjevic, M., Skorin-Kapov, L.: A machine learning approach to classifying YouTube QoE based on encrypted network traffic. Multimedia Tools Appl. **76**(21), 22267–22301 (2017)
55. Paul, U., Ermakov, A., Nekrasov, M., Adarsh, V., Belding, E.: #Outage: detecting power and communication outages from social networks. In: Proceedings of The Web Conference, pp. 1819–1829. WWW (2020)
56. Pedregosa, F., et al.: Scikit-learn: machine learning in Python. J. Machine Learn. Res. **12**, 2825–2830 (2011)
57. Ratcliffe, M., Burd, C., Holder, K., Fields, A.: Defining rural at the US census bureau. Am. Community Surv. Geogr. Brief **1**(8) (2016)
58. Selenium: The Selenium Browser Automation Project. https://www.selenium.dev/documentation/en/
59. Skyhook: Skyhook Coverage Area (2019). https://www.skyhook.com/coverage-map
60. Sodagar, I.: The MPEG-DASH standard for multimedia streaming over the internet. IEEE Multimedia **18**(4), 62–67 (2011)
61. Sousa, I., Queluz, M.P., Rodrigues, A.: A survey on QoE-oriented wireless resources scheduling. J. Netw. Comput. Appl. **158**, 102594 (2020)
62. Statista: Most Popular Video Streaming Services in the US (2019). https://www.statista.com/statistics/910895/us-most-popular-video-streaming-services-by-reach/
63. Wamser, F., Wehner, N., Seufert, M., Casas, P., Tran-Gia, P.: YouTube QoE monitoring with YoMoApp: a web-based data interface for researchers. In: Network Traffic Measurement and Analysis Conference, pp. 1–2. IEEE (2018)
64. YouTube: Looney Tunes Summer Vacation! WB Kids (2018). https://www.youtube.com/watch?v=8fKNkiJl_Ro
65. YouTube: YouTube Player API Reference for iframe Embeds (2019). https://developers.google.com/youtube/iframe_api_reference