Enabling QoE Learning and Prediction of WebRTC Video Communication in WiFi Networks

Suying Yan*, Yuchun Guo*, Yishuai Chen*, Feng Xie†, Chenguang Yu‡, Yong Liu‡
* Beijing Jiaotong University, email: {13120210, ychguo, yschen}@bjtu.edu.cn
† ZTE Inc., email: xie.feng@zte.com.cn
‡ New York University, email: {cyu03,yongliu}@nyu.edu

***Abstract*—Real-time video communication is becoming more and more important in our daily life. WebRTC-based video communication technique has attracted a lot of attentions recently. Its user Quality-of-Experience (QoE), however, faces challenges in wireless networks. This paper studies the problem of accurate prediction of QoE of WebRTC in WiFi networks. We first propose a real-time video QoE metric which is based on the time interval between two consecutively played video frames, and prove it correctly reflects playback freezing and video quality. We then conduct 620 experiments in an indoor WiFi environment to evaluate the correlation between video QoE and wireless network conditions. We final build two machine learning models to predict QoE based on wireless network QoS metrics. The first model can be used by a user to estimate her QoE before she initializes a video call, and the second model is for the system to adjust strategy during a video call. Experimental results show that the models are accurate, with F1 scores above 70%. Our results also clearly demonstrate that current WebRTC's QoE problem is mainly related to the volatility of RTT. Our QoE evaluation method, results, and prediction models are beneficial for wireless video communication system design.**

***Index Terms*—WiFi, QoS, QoE, WebRTC, Machine Learning**

I. Introduction

Wireless video real-time communication (RTC) is becoming a killer application on mobile devices. There are specialized video RTC tools for mobile handheld devices, such as Google Hangout, Apple Facetime, and Microsoft Skype, etc. Evaluation results of these applications can be found in [1]. Recently, web real-time communication (WebRTC[1]) technique has attracted lots of attentions in academic and industry. WebRTC enables RTC within webpages. It has been supported in most of the popular web browsers and platforms without installing extra software or plugin. Due to its high-bandwidth and low-delay requirement, it is a big challenge to deliver a satisfactory RTC video service in wireless networks. The existing research focus on congestion control [2] and Quality-of-Service (QoS) measurement [3]. It is Quality-of-Experience (QoE), however, that ultimately determines the user-perceived service quality [3]. To the best of our knowledge, there is no systematic study of WebRTC QoE. There only has been some RTC QoE studies based on Skype [4], [5] and their results cannot be directly used to design and improve WebRTC-based systems. In this paper, we are the first to study WebRTC QoE on WiFi networks.

[1] www.webrtc.org

Specifically, this paper studies and solves the following three problems:

1) *How to choose an appropriate QoE metric for real-time video communication?*

The traditional video quality metrics, e.g., Structural SIMilarity index (SSIM) and Mean Opinion Score (MOS) cannot accurately reflect video freezing, which is found to be the QoE metric that users care the most [6], [7]. Besides, they cannot be obtained easily in real-time. Thus, we propose to use the time interval between two consecutively played video frames (named *QoEIndex*) to evaluate real-time video playback QoE. We find it can reflect video's playback continuity and picture quality. It is also a general metric which can be easily obtained in most of real-time video systems.

2) *How to conduct systematic measurements to collect comprehensive datasets to evaluate the correlation between video QoE and wireless network conditions?*

In this paper, we design and conduct systematic and extensive measurements in an indoor WiFi environment. During the experiments, we collect two types of network QoS metrics: a) wireless signal/link quality metrics, including Signal Quality, received signal strength indicator (RSSI), and Link Quality; b) network data transfer quality metrics, including packet loss rate and Round Trip Time.

3) *How to predict WebRTC's video QoE from wireless network's QoS metrics?*

We systematically evaluated the correlation between video QoE and various wireless network quality metrics, and proposed two video QoE prediction models. The first model can be used by a user to estimate her video QoE before she initializes a video call, and the second can be used for the system to adjust service strategy in real time during a video call. Experimental results show the models are accurate. Their F1 scores are above 70%.

Our QoE evaluation method, measurement results, and prediction models provide valuable insights for wireless WebRTC-based video communication system design. The paper is organized as follows: Section II describes our experimental methodology; Section III introduces our measurement results and the evaluation of the relationship between wireless network quality metrics and the proposed video QoE metric; Section IV presents the QoE prediction models. Section V concludes the paper.

II. Methodology

This section describes our experimental environment, measurement methods, and the proposed video QoE metric.

A. Testbed

In this paper, we focus on studying WebRTC video communication in WiFi environment. With WiFi access, most of RTC communication takes place indoors. Thus, we consider the typical office usage environment. We set up a testbed which contains two laptops and a 802.11n wireless LAN AP. The laptops are ThinkPad T440p-20ANS00U00, and the AP is TP Link-WDR4320. One and only one client accesses the WiFi AP via its air interface. The other client connects to the AP through its Ethernet port. The clients run the official open-source reference programs of WebRTC. The programs were modified to add video QoE and network QoS monitoring code. We also wrote a python program to run on the clients to collect wireless network related information. To ensure the transmitted video contents are consistent and repeatable, we choose a high-definition (HD) video sequence *Big Buck Bunny* as the video source, as it is widely used in video-related research. We inject the video sequence into the WebRTC clients using a virtual video camera tool[2].

B. Measurement Methodology

We carried out comprehensive measurements within the AP's signal coverage range. Fig. 1 shows the indoor WiFi network environment in our measurement. The AP is placed in room3. The brown thick lines are the walls between rooms, and the light blue blocks are our experiment spaces. Room0 to room4 are typical office rooms which contain desks, chairs, computers, and other office supplies. Besides, each room is covered by several other WiFi APs which work in channels that different from our AP. We conducted independent experiments at each seat in room1 to room4. We also divided the space of corridor into squares with size of one square meter, and conducted experiments in each square. In total, we have 62 experimental positions. At each experimental position, we conducted 10 experiments. Thus, we finally have 620 experiments.

C. Wireless Network Quality Metrics

To characterize the wireless network quality, we use the following metrics:

- Wireless signal/link quality metrics: We use all wireless physical layer metrics reported by Microsoft Windows 7 OS through its API, including *Signal Quality* (SQ) , *received signal strength indicator* (RSSI), and *Link Quality* (LQ), which are recorded by the Python program we write and run on the laptop.
- UDP transportation quality metrics. As video transportation in WebRTC uses RTP over UDP, we also measure *Packet Loss Rate* (Loss) and *Round Trip Time* (RTT), which are recorded by the open-source WebRTC reference program we modified to add these monitoring codes.

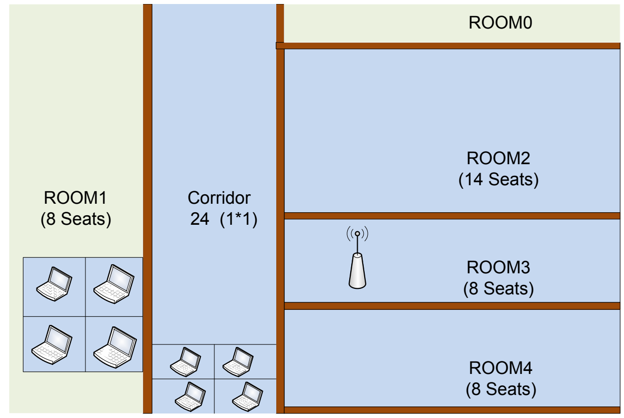

Fig. 1: Indoor measurement environment

D. Video QoE Metric

It is a challenge to record video playback continuity for RTC video. To the best of our knowledge, there is no attempt to track video freezing automatically in RTC applications. Thus, we must design a method to trace user video freezing events. After extensive experiments, we found that the *time interval between two consecutively played video frames* (named *QoEIndex*) is a good QoE indicator. Specifically, when the QoEIndex is larger than one second, users usually feel video freezing. This result is reasonable. For a video with frame rate of 30 frames/second, the regular interval of two consecutively played frames is 33ms. Thus, when the QoEIndex is longer than 33ms, i.e., there are some frames lost or delayed, the video playback continuity is impacted. As visual sensitivity of human is limited, when this interval is abnormal but not too long, users' eyes may not detect it immediately. After lots of tests with human subjects, we find that one second is an appropriate threshold. Such a result is consistent to the measurement result reported in [8], where the video freeze duration is measured by seconds.

We now give the definition of "Good" and "Bad" QoE used in this paper. When QoEIndex > 1s, we say the QoE is bad, otherwise, we say QoE is good. With this method, we further obtain the durations of good and bad QoE. We also find that the QoEIndex is strongly correlated to video's Structural SIMilarity (SSIM) index[3]. In WebRTC, when the network condition gets worse, the video sender decreases its video encoding rate to ensure smooth streaming. The video's playback quality is degraded. Thus, when a bad QoE event occurs, the video's picture quality usually also degrades.

III. Measurement Results

In this section, we present our experimental measurement results and evaluate the relationship between WebRTC's user QoE and wireless network's quality metrics.

[2]e2eSoft. http://www.e2esoft.cn/vcam/

[3]http://www.cns.nyu.edu/~lcv/ssim/

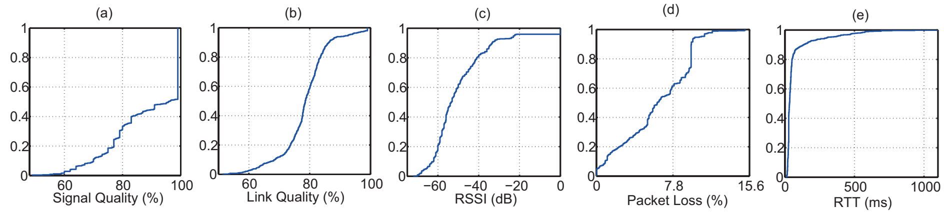

Fig. 2: Distribution of measured wireless network quality metrics.

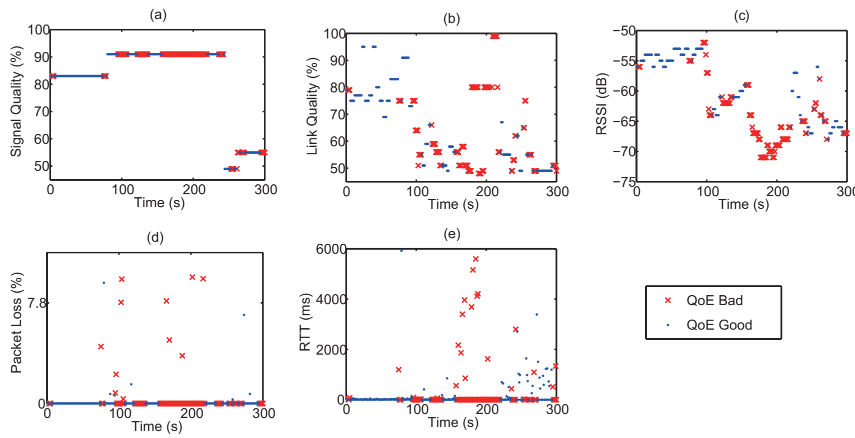

Fig. 3: Variances of wireless network QoS metrics in one experiment.

A. *Distribution of Wireless Network Quality Metrics*

Fig. 2 plots the CDFs (cumulative distribution function) of the measured wireless network quality metrics. As shown in Fig. 2, our measurement covers a wide range of wireless network conditions. For instance, Fig. 2c shows the RSSI ranges from -70dB to 0dB, which is the general working range of WiFi network. Such a result proves the efficiency and coverage of our measurement methodology.

B. *Volatility of Wireless Network Quality*

We then observe the volatility of wireless network quality metrics during our experiments. As an example, we first observe the volatility of the five WiFi network metrics in one randomly picked experiment. Fig. 3 plots the temporal variance of the five QoS metrics and our QoEIndex. The experiment is conducted at a position in room2 and lasts for 5 minutes. As showed in Fig. 3, all these 5 metrics varies considerably during the experiment. For instance, as shown in Fig.3(a), the signal quality drops from 91% to 52% at 240s, which is a significant dropping. Moreover, the link quality and RSSI all vary with some trends. As a result, the variance of link quality within 300 seconds is 133.83 and variance of RSSI is 11.07. The packet loss and RTT, however, keeps low most of time, but have several abrupt increases.

We next observe the volatility of the 5 wireless network QoS metrics across all 620 experiments. Fig. 4 plots the variances of the five QoS metrics over all experiments. The x-axis is position ID. We have 62 positions. For each position, we plot ten variance values, one for each experiment. As shown in Fig. 4, even at a same position, the variance of QoS metrics in different experiments can be quite different. For instance, in Fig. 4, we use red "o" to mark the 20^{th}, 40^{th}, and 60^{th} positions' experiment results. As shown in Fig. 4, while some QoS metrics are stable (e.g., signal quality and RSSI for the 20^{th} position), some metrics have very different variances cross different experiments (e.g., for the 20^{th} position, 8 experiments have stable link quality, but 2 experiments have significantly volatile link quality).

C. *Correlation between Wireless Network QoS and Video QoE*

This subsection analyzes the correlation between the wireless network QoS metrics and video QoE.

As an example, we first illustrate their correlation in the experiment used to generate Fig. 3. We augment Fig. 3 with QoEIndex of the experiment. The QoE bad event is marked with red color. As shown in Fig. 3, the bad QoEIndex seems correlated with wireless network QoS degradations. For instance, during the time from 200s to 250s, QoEIndex is bad and both the RSSI and link quality (LQ) are low.

We then evaluate the correlation of wireless network quality metrics with video QoE in all experiments. To obtain a macroscopic analysis of all 620 experiments, we define a video session is *unacceptable* when its QoE bad duration is longer than 30% of the whole session length. We then calculate the relative information gain [7] of the mean and variance of the five wireless network QoS metrics to QoE unacceptable indicator of all 620 experiments, respectively. More specifically, X is the QoE unacceptable indicator. For

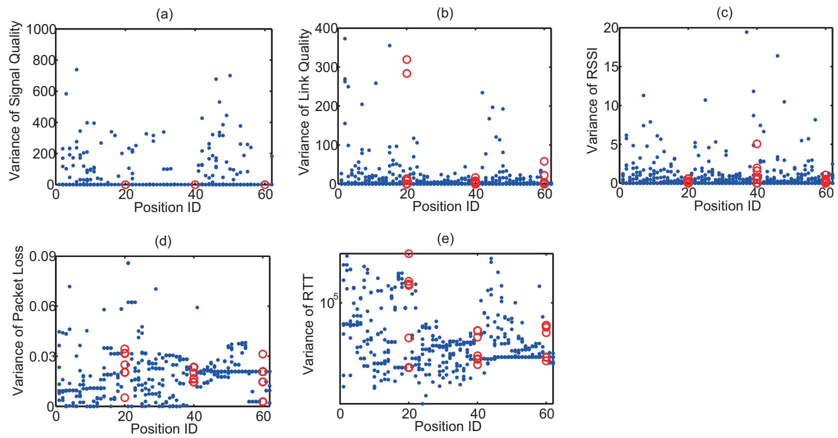

Fig. 4: Variances of wireless network QoS metrics in all experiments.

each QoS metric Y, we calculate the relative information gain of X against Y as:

$$RIG(Y|X) = \frac{H(Y) - H(Y|X)}{H(Y)},$$

where $H(Y)$ is the entropy of random variable Y and $H(Y|X)$ is the conditional entropy of Y given random variable X. The relative information gain quantifies how much uncertainty in each wireless network QoS metrics is reduced by knowing whether the QoE is unacceptable or not. The higher the information gain, the more correlated the QoS metric is to the QoE unacceptable event. To bin measurement results of wireless network quality metrics into discrete bins, we use the method introduced in [9]. Table I shows the result.

As shown in Table I, the relative information gain of QoE unacceptable indicator against the variance of RTT is 0.136, which is the highest value in the table. Then, the mean of RTT and variance of link quality are 0.087 and 0.082, respectively. They are relatively high. Thus, we conclude that *the video QoE are correlated to the wireless network quality metrics, in particular the variance of RTT*. Such a result suggests that the current WebRTC's video QoE problem is mainly due to the volatility of RTT. It is reasonable, as existing WebRTC congestion control algorithms partially relies on variance of network latency to adjust the video streaming rate.

TABLE I: Relative information gain of network QoS metrics and video QoE unacceptable indicator

Feature	Relative Information Gain
RTT - Variance	0.136
RTT - Mean	0.087
Link Quality - Variance	0.082
RSSI - Mean	0.040
Link Quality - Mean	0.035
Signal Quality - Variance	0.031
RSSI - Variance	0.026
Signal Quality - Mean	0.017
Packet Loss - Mean	0.012
Packet Loss - Variance	0.009

IV. WebRTC QoE Models

A. QoE Models

In this section, we build the following two machine learning models to predict the QoE of a user's WebRTC video communication session from the wireless network quality metrics.

- *QoE mapping model*: In this model, we use the current wireless network QoS metrics to predict whether a user's WebRTC video communication session will have acceptable QoE. It aims to establish the relationship between the current WebRTC QoE and current QoS metric. The model can be used by a user to estimate her video QoE before she initializes a video call, which is very important to improve user satisfaction in video RTC services in wireless network and video services. [10] works in this direction also, but they use QoS data collected at base station of cellular network. For comparison, we use the QoS data collected on the mobile terminal itself in a WiFi network.
- *QoE prediction model*: In this model, we use the measured wireless network QoS metrics in a limited time window (say window A) to predict the video *unacceptable* event in the next time window (say window B). As WebRTC use a 10 second video jitter buffer at the receiver side, we use 10 seconds as the size of window B. We tune the size of window A to obtain the best prediction performance in our model training. The training and prediction can be done online. During a user's video communication process, we can keep collecting wireless network quality metrics, predicting user QoE in the next time window, and then adjusting WebRTC's rate control algorithm to improve video playback continuity.

B. Methods

We use the wireless network quality features listed in Table II to train our QoE mapping model, and the features listed in Table III to train the QoE prediction model. For QoE mapping model, we treat each experiment as a sample. Then, among all 620 samples, 496 samples are randomly selected as the training set and the remaining 124 samples are used as testing set. For each experiment, we calculate the mean and variance of each metric, and then use them as the features. Thus, we totally have 10 features. For QoE prediction model, we use the same 620 experiments' measurement results but extract samples by sliding windows. Besides the features used for the QoS mapping model, we also use the current QoS metric value (i.e., the last value in the sliding window) as a feature. Thus, we totally have 15 features. We use *Decision Trees (DTs), Random Forests (RandF), Support Vector Machines (SVM)* and *Extra-Trees classifier (ExtraT)* to train our models and compare their performance.

C. Performance

We evaluate the effectiveness of classification methods in terms of *Precision, Recall* and *F1 score*. Among them, we use F1 score as the main metric, as it is a comprehensive index which includes precision and recall. Moreover, the prediction accuracy of QoE bad is more important to avoid users' frustration of wrong prediction. Thus, we mainly compare the algorithms' F1 score of QoE bad prediction results. Table IV

TABLE II: Features used in QoE mapping model and their importances

Feature	Importance
RTT - Variance	0.44
Link Quality - Variance	0.14
Link Quality - Mean	0.12
RSSI - Mean	0.09
RSSI - Variance	0.07
RTT - Mean	0.04
Packet Loss - Mean	0.04
Packet Loss - Variance	0.04
Signal Quality - Mean	0
Signal Quality - Variance	0

TABLE III: Features used in QoE prediction model and their importances

Feature	Importance
RTT - Mean	0.16
RSSI - Mean	0.13
RTT - Variance	0.12
Link Quality - Mean	0.09
Link Quality - Variance	0.06
RSSI - Variance	0.06
Link Quality - Last value	0.05
Signal Quality - Mean	0.05
Signal Quality - Last value	0.04
Packet Loss - Mean	0.04
RSSI - Last value	0.04
Packet Loss - Variance	0.03
RTT - Last value	0.03
Signal Quality - Variance	0.02
Packet Loss - Last value	0.01

TABLE IV: Performance of different algorithms for QoE mapping model

Algo.	QoE-good result			QoE-bad result		
	precision	recall	F1	precision	recall	F1
SVM	0.98	0.89	0.94	0.5	0.86	0.63
RandF	0.92	0.94	0.93	0.73	0.67	0.70
DTs	0.94	0.92	0.93	0.71	0.77	0.74
ExtraT	0.93	0.98	0.96	0.83	0.56	0.67

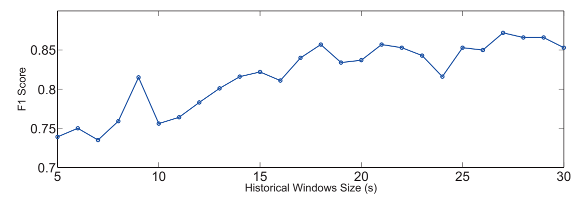

Fig. 6: Performance of Random Forests QoE prediction model.

shows the performance evaluation results of the QoS mapping model. As showed in Table IV, the Decision Tree method has the highest F1 score for QoE bad prediction, i.e., 0.74, which means the model has high accuracy.

We plot the resulting Decision Tree model in Fig. 5, and list its feature importance in Table II. Note that the mean and variance of Signal Quality are not included in the Decision Tree model and thus their importances are marked by "0". As shown in Table II, the importance of variance of RTT is 0.44, which is considerably higher than other features, meaning it is the most important feature for the prediction of WebRTC video QoE unacceptable event. Such a result is consistent with the results of relative information gain shown in Table I, where the relative information gain of the variance of RTT is also the highest in all features.

For QoS prediction model, after extensive experiments, we find that SVM's F1 score is always below 0.3 and the performance of Extra-Trees and Decision Tree fluctuates widely with the size of sliding window A, i.e., the historical time window in which we collect historical wireless network quality metrics for prediction. For comparison, Random Forests method performs well and stably. Such a result is reasonable as Random Forests is ensembles of a number of decision trees and is the most successful general-purpose algorithm [11]. Thus, we finally select Random Forests model. Fig. 6 plots the F1 score of the Random Forests model against the size of sliding window. As shown in Fig. 6, as the size of sliding window A increases, the F1 score of the prediction model gradually increases, meaning the model performs better when using more historical data. Then, when the window size is larger than 17 seconds, the F1 score keeps relatively stable and is always larger than 0.8, means the model has high accuracy. For instance, when the window size is 20 seconds, the precision, recall and F1 score of the model are 0.996, 0.744, and 0.839, respectively. Thus, in practice, we suggest that the size of sliding window can be selected in the range from 20 to 30 seconds.

We list the features' importance of the Random Forests model in Table III. As shown in Table III, the importance of RTT is still the highest: the importance of RTT-Mean is 0.16, and RTT-Variance is 0.12. Such a result is consistent with that of the QoE mapping model. The difference is that the importance of different features are relatively close. Such a result suggests that the online real-time prediction is essentially different from the offline mapping.

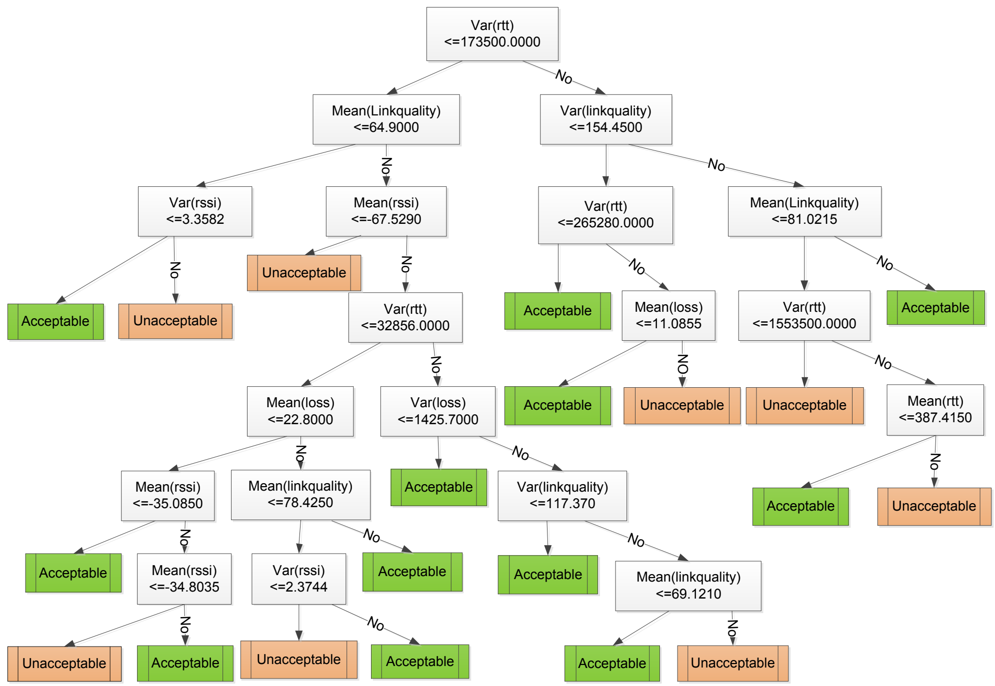

Fig. 5: Decision Tree for QoE mapping

V. Conclusions

In this paper, we studied the problem of accurate prediction of user video QoE of WebRTC in WiFi networks. First, we proposed a new, simple, and efficient QoE metric which is based on the time interval between two consecutively played video frames. Second, we conducted 620 experiments in an indoor WiFi environment and showed the strong correlation of WebRTC user QoE with wireless network QoS metrics. Finally, we built two machine learning models to predict a user's WebRTC video communication QoE based on the current wireless network measurement results. The first model can be used by a user to estimate video QoE before she initializes a video call, and the second model is for a system to adjust its servicing strategy in real-time during a video call. Experimental result demonstrated that the models are accurate, with F1 scores above 70%. Moreover, our analysis results and models clearly show that the current WebRTC implementation's QoE problem is mainly due to volatility of RTT. Our QoE evaluation method, analysis results, and prediction models provide valuable insights for wireless WebRTC video communication system design.

Acknowledgment

This work was supported by ZTE Corporation and National Science Foundation of China under grant nos. 61271199 and 61301082.

References

[1] W. Sun and X. Qin, "End-to-end delay analysis of wechat video call service in live dc-hspa+ network," in *Wireless Communications and Signal Processing (WCSP), 2014 Sixth International Conference on*. IEEE, 2014, pp. 1–5.

[2] L. De Cicco, G. Carlucci, and S. Mascolo, "Experimental investigation of the google congestion control for real-time flows," in *Proceedings of the 2013 ACM SIGCOMM workshop on Future human-centric multimedia networking*. ACM, 2013, pp. 21–26.

[3] C. Kilinc and K. Andersson, "A congestion avoidance mechanism for webrtc interactive video sessions in lte networks," *Wireless personal communications*, vol. 77, no. 4, pp. 2417–2443, 2014.

[4] S. Chen, C.-Y. Chu, S.-L. Yeh, H.-H. Chu, and P. Huang, "Modeling the qoe of rate changes in skype/silk voip calls," *IEEE/ACM Transactions on Networking (TON)*, vol. 22, no. 6, pp. 1781–1793, 2014.

[5] X. Zhang, Y. Xu, H. Hu, Y. Liu, Z. Guo, and Y. Wang, "Modeling and analysis of skype video calls: Rate control and video quality," *Multimedia, IEEE Transactions on*, vol. 15, no. 6, pp. 1446–1457, 2013.

[6] 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 *Proceedings of the ACM SIGCOMM 2013 conference on SIGCOMM*. ACM, 2013, pp. 339–350.

[7] F. Dobrian, V. Sekar, A. Awan, I. Stoica, D. Joseph, A. Ganjam, J. Zhan, and H. Zhang, "Understanding the impact of video quality on user engagement," *ACM SIGCOMM Computer Communication Review*, vol. 41, no. 4, pp. 362–373, 2011.

[8] C. Yu, Y. Xu, B. Liu, and Y. Liu, "Can you see me now? a measurement study of mobile video calls," in *INFOCOM, 2014 Proceedings IEEE*. IEEE, 2014, pp. 1456–1464.

[9] A. PescapÃÍ, A. Dainotti, and A. Botta, "On internet measurement research," in *1st Newsletter of the IEEE Technical Committee on Communication Systems Integration and Modeling*, 2006.

[10] V. Aggarwal, E. Halepovic, J. Pang, S. Venkataraman, and H. Yan, "Prometheus: Toward quality-of-experience estimation for mobile apps from passive network measurements," in *Proceedings of the 15th Workshop on Mobile Computing Systems and Applications*. ACM, 2014, p. 18.

[11] J. Howard. (2012, Feb.) The two most important algorithms in predictive modeling today. [Online]. Available: http://strataconf.com/strata2012/public/schedule/detail/22658