Cognitive Video Streaming

D. PASUPULETI
P. MANNARU
B. BALASINGAM
M. BAUM
K. PATTIPATI
P. WILLETT
C. LINTZ
G. COMMEAU
F. DORIGO
J. FAHRNY

Video-on-demand (VoD) streaming services are becoming increasingly popular due to their flexibility in allowing users to access their favorite video content anytime and anywhere from a wide range of access devices, such as smart phones, computers and TV. The content providers rely on highly satisfied subscribers for revenue generation and there have been significant efforts in developing approaches to "estimate" the quality of experience (QoE) of VoD subscribers. However, a key issue is that QoE can be difficult to measure directly from residential and mobile user interactions with content. Hence, appropriate proxies need to be found for QoE, via the streaming metrics (the QoS metrics) that are largely based on initial startup time, buffering delays, average bit rate and average throughput and other relevant factors such as the video content and user behavior and other external factors. The ultimate objective of the content provider is to elevate the QoE of all the subscribers at the cost of minimal network resources, such as hardware resources and bandwidth.

In this paper, first, we propose a *cognitive video streaming* strategy in order to ensure the QoE of subscribers, while utilizing minimal network resources. The proposed cognitive video streaming architecture consists of an *estimation module*, a *prediction module*, and an *adaptation module*. Then, we demonstrate the prediction module of the cognitive video streaming architecture through a *play time prediction* tool. For this purpose, the applicability of different machine learning algorithms, such as the k-nearest neighbor, neural network regression, and survival models are experimented with; then, we develop an approach to identify the most relevant factors that contributed to the prediction. The proposed approaches are tested on dataset provided by Comcast Cable.

Manuscript received June 10, 2016; revised June 24, 2016; released for publication August 10, 2016.

Refereeing of this contribution was handled by Benjamin Slocumb.

Authors' addresses: D. Pasupuleti, P. Mannaru, B. Balasingam, M. Baum, K. Pattipati, and P. Willett are with the Department of Electrical and Computer Engineering, University of Connecticut, Storrs, CT, 06269, USA (E-mail: {devaki, pujitha.mannaru, bala, mbaum, krishna, willett}@engr.uconn.edu). C. Lintz, G. Commeau, F. Dorigo, and J. Fahrny are with Comcast Corporation, USA (E-mail: {Christopher_Lintz, Gabriel_Commeau, francesco_dorigo, Jim_Fahrny @cable.comcast.com).

Dr. Balasingam is the corresponding author.

Some initial works have been published in [40] and [41].

I. INTRODUCTION

Major advances in wireless communication and consumer electronics of the past decade have disrupted the traditional ways in which people used to consume video programs. In a traditional setting (see Figure 1), a viewer has to "tune-in" to a TV station via cable, satellite or on-air receiver in order to watch or record his/her favorite program. Today, with internet and wireless broadband connectivity, there are several options for a viewer to watch his/her favorite programs at the time of his/her convenience using a device of his/her choice (see Figure 2), such as a smart phone, tablet, computer or TV. As a result, the video distribution strategy also has gone through major changes.

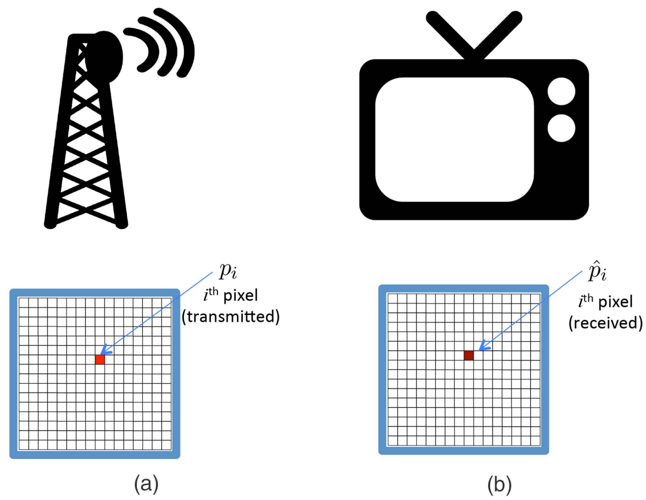


Fig. 1. **Traditional video transmission and reception.** Traditional QoS metrics try to quantify viewers' perception using objective metrics computed based on transmitted and received frame sequences. (a) Video transmission. (b) Video reception.

A brief description of each of the blocks in Figure 2 is given below:

- *Content.* Content can be divided into online streaming, i.e., regular TV programs, and recorded programs that are delivered as video-on-demand (VoD), the focus of this paper. In VoD, a viewer browses through the lists of available videos and selects one to play. Unlike online streaming, VoD offers the capability to pause and resume videos at any time.
- *Delivery service.* Delivery service providers, such as cable networks, bring the videos to the viewers. Usually, the viewer has to be a subscriber to the delivery service provider in order to get access to the content.
- *Viewer.* The viewer accesses the videos using devices, such as smart phones, tablets, TV and Computer. Each viewing device may have different connectivity and bandwidth. Depending on the access device (portable or desktop), the characteristics of the viewer might be different as well. For example, a viewer may be willing to tolerate intermittent buffering events and

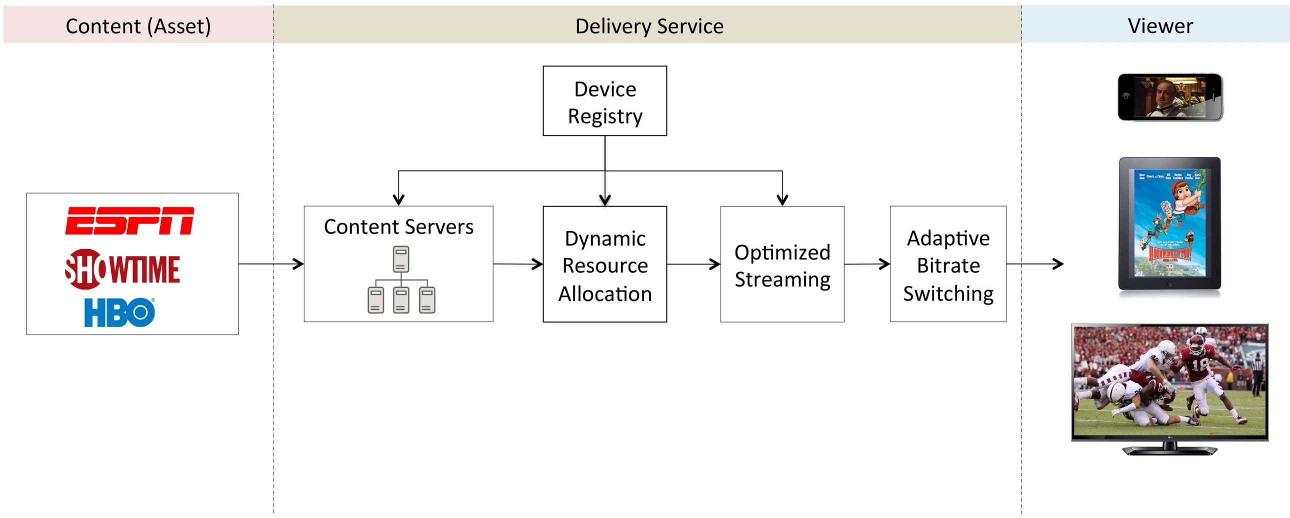


Fig. 2. **Description of a video-on-demand (VoD) system.** Unlike traditional video transmission systems, the viewers have the option of choosing from a large amount of video content or to select watching online video streaming.

longer startup times in a smart phone, while exhibiting lesser tolerance towards similar events in a TV.

- *Content servers.* Content servers respond to the VoD requests and stream videos to the viewers. Based on the popularity of particular videos, content servers adjust content delivery priorities in order to provide good QoS to the viewers.
- *Dynamic resource allocation.* Content service providers respond to rapidly increasing/decreasing demands to particular videos, anticipated and unexpected, such as major sports events and unexpected world events, by dynamically adjusting the streaming capacity of videos.
- *Optimized streaming.* Optimized streaming algorithms aim to deliver high quality videos at reduced cost (bandwidth) to the viewer. This is achieved by efficiently compressing subsequent video frames. Some other constraints include the power and memory requirements of the video player at the viewing devices.
- *Device registry.* An important challenge in maintaining superior quality of online video streaming is the increasing number of different types of devices available to viewers in order to play videos. Each of these devices has different hardware and software capabilities. Knowing the exact capabilities of a particular device is important in optimizing the video streaming.
- *View logs.* These represent feedback data from the video players to the content delivery service providers. The feedback contains data, such as bit rate, buffering information and media-failed events that are useful in assessing the quality of experience of the viewer.
- *Adaptive bitrate switching.* In mobile video devices, the available bandwidth can vary depending on the location of the receiver. For example, moving the device (e.g., moving between different parts of a house, traveling in a vehicle, walking through a mall, etc.), can result in varying download bandwidths at the device. The video streaming algorithms respond to this by adjusting the bit-rate of the content.

The quality of user experience has been a concern in both traditional and the emerging content delivery systems. In the traditional video broadcasting scenario, the issue of video quality arises due to video transmission and processing manifested in the form of noise, jitter, shape transformation, and so on. Traditional QoS assessment schemes focused on quantifying the perception of the viewers on videos with varying types and degrees of video transmission distortions; such distortions are generally defined as the QoS metrics, such as peak signal to noise ratio (PSNR,[50]), video quality metric (VQM,[42]), moving picture quality metric (MPQM,[48]), structural similarity index (SSIM,[51]), and noise quality measure (NQM,[14]). The viewers' perceptions as a result of varying QoS are obtained through subjective methods and quantified usually as a mean opinion score (MOS,[23]). The MOS, scaled between 0 and 5, represents the perceptual quality of the video; very pleasant and clear viewing experience will result in MOS of 5 and an intolerable video will have an MOS of 0. When poor QoS is detected in some areas, the broadcasters must find ways to increase the signal to noise ratio to the affected area; this can be achieved by increasing the power of existing transmitters or by installing additional transmitters (or repeaters) in the affected area. The MOS scheme in traditional TV broadcasting enjoys wide acceptance (see [20]).

In VoD, the QoS factors are different from those in traditional video; some widely used QoS factors are based on startup time, buffering and transmission bitrate. The startup time is defined as the time between the initial video request (such as clicking on the play button on a web interface) and the time of playing the first video frame on the screen. Higher startup time can cause the viewer to abandon the video [26]. There are several factors affecting the startup time; connection bandwidth of the viewer, capability of the video distribution server, and network delays are a few of them. In order to reduce the startup time, the player "buffers" a portion of the video before it starts and the rest of the video is continuously buffered while the video is still playing. Buffering is supposed to happen in the background while the video is playing; however, similar to startup time, non ideal streaming conditions cause the player to pause and wait for the data to be buffered. It is reported in [26] that buffering delays negatively impact the likelihood of a viewer's return to the content provider. Adaptive bitrate switching [28] allows content providers to reduce the startup and buffering delays by adaptively switching the frame quality of the video based on the bandwidth and other hardware capability of the video player. The higher the bandwidth and processing capabilities of the player, the higher the bit-rate and quality of the video; the bitrate serves as a QoS factor. High average bitrate over a certain period of time indicates that the rendering quality was high and vice versa; frequent bitrate switching with high variation indicates poor quality of experience due to volatile bandwidth. Analysis of viewer responses to the startup time, buffering and bitrate related QoS factors are reported in [15]. The adaptive bitrate streaming technique has been widely adopted by many existing content providers; in [39] and [24], a general overview of the widely adopted HTTP adaptive streaming (HAS) protocol is provided.

Adaptive video streaming itself is challenging and diverse approaches have been published in the literature [45]. Most of the adaptive streaming strategies recommend adapting the bitrate based on buffering events [17]. Other than adaptive streaming, there are several suggestions in the literature to enhance a specific aspect of QoE; in [4], an approach is suggested to enhance the accessibility in shared video forums; [5] suggests exploiting the knowledge that concurrent viewers are viewing a specific content and using peer-to-peer (P2P) strategies to offload some of the workload of the content servers; an approach for client side server selection is presented in [29]; in [44], the QoE is modeled based on a packet loss model; in [49], the QoE is modeled in terms of the QoS factors such as loss, delay and jitter; and [11] talks about providing good quality video, while being aware of the bandwidth quota of the user.

Current adaptive streaming and other approaches developed to enhance QoE are designed to "react" to the QoS factors (that are largely based on startup time, buffer level and average bitrate) from the viewer's device. This does not guarantee that the quality of experience (QoE) of the viewer will be improved as a result. For example, the decision to downgrade the bitrate (i.e, the quality of the video) as a result of buffering delay may not be appreciated by all viewers; to make things worse, the same viewer might have varying preferences depending on circumstances such as the time of day. Further, there is explosive growth in the internet traffic caused by videos delivered by content delivery networks; this trend is expected to continue as more and more viewers turn from traditional TV to VoD [1]. Expanding the network infrastructure is costly and time consuming; a QoE based adaptive streaming will help ease some of the strain on the network by increasing the bitrate only when it is likely to advance the QoE of the viewer. In other words, a better and futuristic adaptive streaming technique has to be "proactive" rather than reactive.

The first step in QoE-based adaptive video streaming is to come up with accurate methods of estimating the QoE of the viewer. Taking cues from the widely adopted MOS in traditional TV, some initial attempts were made in [36] to estimate the MOS in response to the QoS factors of VoD. However, unlike traditional video, the MOS obtained through a limited experiment is unable to represent the viewers' perception in a wide ranging VoD scenario. It is found that the viewers react differently to the same video content with the same QoS factor; viewers seemed to tolerate QoS deficiencies in live video compared to non-live content [7]; viewers from well connected devices (those with better connection bandwidth) are found to be less tolerant compared to their low-bandwidth counterparts.

A VoD viewer has millions and millions of videos to choose from. Instead of traditional TV, there are devices of convenience (with trade offs) for a particular time of day; video in a smart phone might come with too many buffering events and blurry images compared to a TV; however, its portability is appealing to a certain viewer during day-time; the same viewer might prefer to continue the same video using TV during the evening. For content providers, the objective has become one of attracting and retaining subscribers by providing superior quality of experience. Due to the nature of VoD consumption, it is impossible to capture the QoE in terms of a single metric, such as MOS. Hence the MOS, which is subjectively estimated using a particular viewing scenario, is not adequate to quantify viewers' QoE [10].

Recently, there have been attempts to estimate QoE from user data; these approaches are generally termed "passive," "online" or "indirect" approaches of estimating QoE. In [6], [7], it was suggested to create a predictive model of viewer engagement (such as total play time, number of visits and probability of return) based on the observed QoS factors. A machine learning framework to estimate the QoE in mobile applications was proposed in [3]; this approach requires training data form past "good QoE" and "poor QoE" instances. Table I gives a comparative summary of existing QoS literature corresponding to traditional video transmissions and QoE metrics corresponding to VoD and internet video.

The existing approaches focus heavily on modeling the QoE as related to the QoS factors only. However, even though the QoE is significantly influenced by the

TABLE I
Summary of QoE Approaches in Traditional TV and VoD

	Traditional Video	VoD
QoS factors	• PSNR—Peak Signal to Noise Ratio [50] • VQM—Video Quality Metric [42] • MPQM—Moving Pictures Quality Metric [48] • SSIM—Structural Similarity Index [51] • NQM—Noise Quality Measure [14]	• Startup time [15] • Buffering time [43] • Buffering count [43] • Buffering ratio [15] • Rate of buffering events [15] • Normalized re-buffer delay [25] • Average bit rate [15] • Average throughput [39] • Frames per second (FPS) [15] • Failures [25]
User satisfaction metrics (alternately, viewer behavior metrics [25])	• Mean opinion score (MOS) [23]	• MOS [36] • Number of views [15], • Total play time [15], • Session duration ratio [43], • Abandonment [25], • Engagement [25], • Repeat viewers [25]
Related standards	• For cable TV (2004) [20] • For standard television (2004) [18] • For multimedia applications (2008) [22] • Relative to reduced bandwidth reference (2008) [21] • Television (2002) [19] • Multimedia (2008) [23]	• DASH [24] • 3GP-DASH [2]

QoS factors, there could be other factors that wield influence on the QoE of the viewers. For example, considering the vast amount of video content to choose from, the viewers' QoE can be be influenced by the type of content being accessed. Further, for a fixed video content, QoE varies significantly by demography, based on age, gender, ethnic background, and language. In addition, seasonal factors, such as the time of day, day of week and season of year, also might influence the QoE of the user towards a particular video content. Finally, there could be many other exogenous factors, such as important local/national/world events, that might contribute to the QoE of a particular viewer.

In the next Section, we describe our proposed cognitive video streaming strategy [40], which considers all the above factors in devising a video streaming strategy. It must be noted that there are no direct comparisons, because the proposed cognitive video streaming architecture is new and the proposed idea of using predicted play time as a surrogate of QoE is also new. However, the three prediction approaches (based on neural networks, survival models and k-nearest neighbor regression) that we discuss in Section IV have some comparisons. For example, [6] uses naive Bayes decision tree and regression methods to predict user engagement from quality metrics and in [12] survival models were used for remaining time prediction.

II. COGNITIVE VIDEO STREAMING

A block diagram of the proposed cognitive video streaming approach is shown in Figure 3. It is comprised of three fundamental modules: an *estimation module*, a *prediction module* and an *adaptation module*. The framework is designed in such a way that each module is able to function with some basic functionalities (sub-modules); as more sub-modules are added, the effectiveness of the module and the integrated system is expected to improve. Next, we describe each module in the proposed solution framework.

A. Prediction Module

The nature of completion of a particular video changes from viewer to viewer; some videos are abandoned in the process of "browsing"; some videos are terminated by the viewer because of lengthy buffering and other QoS issues; and some videos are "temporarily" abandoned to be resumed later. Once a viewer starts playing a video, the *remaining play time* of that video is a useful piece of information to the content provider in order to ensure adequate QoE to the viewer. For example, the knowledge of the remaining play time can be used to allocate server bandwidth to the user; it can be used to devise a more appropriate adaptive bitrate switching scheme; and the prior knowledge that a video is possibly terminated by the viewer can be used to recommend more appropriate videos in the first place. At the network level, the predicted play time of each viewing session is useful for managing network traffic.

In addition to QoS, there are several other factors determining the play time ratio (PTR) which is the ratio of the completed time to the actual length of the video ($\text{PTR} \in [0,1]$ is useful to compare the played times of two videos of different length.) However, it

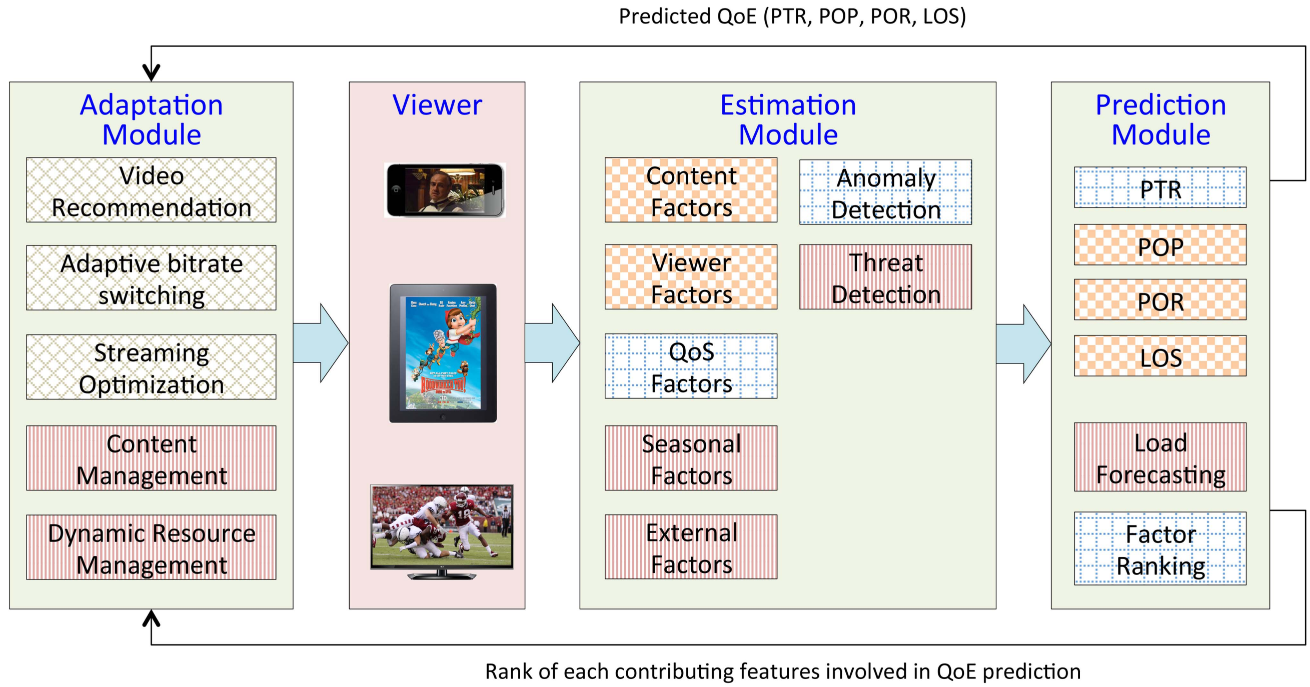


Fig. 3. **Proposed Cognitive Video Streaming Architecture.**

was reported that shorter videos tend to have higher PTR compared to longer videos [26]; hence, PTR gives better comparison for videos of comparable length. QoS factors such as buffering negatively affect the PTR in well-connected devices. All the relevant factors must be included in order to accurately predict the play time of a video. We divide the factors affecting the PTR into five categories: content-related, viewer-related, QoS-related, seasonal and external. Each factor contains several features affecting the play time; in Table II, we have provided some examples.

Considering all the relevant factors/features helps in accurately predicting the PTR of a particular video session. This also allows us to investigate the features that are significant to PTR prediction. It must be noted that the dominant factor affecting play time will be different from one viewer to the next. Identifying these factors (even after knowing that a particular video has been terminated) will help in devising individualized remedies.

Similar to PTR, there are other user engagement metrics that are indicative of the QoE of a viewer:

- ***Probability of return (POR)*** tells if the viewer will return to a previously abandoned video. Returning to the same video indicates the importance of that video to the viewer. Hence, POR combined with PTR forms a stronger indicator of the QoE.
- ***Probability of re-play (POP)*** tells if the viewer will re-play a previously completed video. The difference between POR and POP is that the former is the (probability of) return to an abandoned video and the latter is the (probability of) return to a previously watched video.

TABLE II
Factors Affecting Play-Time Prediction and Sample Features in Each Factor

Factor	Features
Content	popularity, age, length, match to viewer's preference
Viewer	age, gender, ethnic background, language
QoS	startup time, buffering, average bitrate, throughput
Seasonal	time of day, day of week, season of year
External	important local/national/world events

- ***Average length of scrubbing (LOS)*** tells how long a particular video will be "scrubbed," i.e., rewound or forwarded. Scrubbing is the process of moving the player to a different point in the video. For example, most of the viewers might try to scrub past a commercial segment (due to this reason, many video players nowadays disable the scrubbing option during commercial breaks). Apart from commercial breaks, abnormal scrubbing behavior might strongly correlate to the QoE, hence LOS is another effective indicator of QoE.

Later in the paper, we are demonstrating only the PTR prediction. The same algorithms can be used for other three metrics, however, POR, POP and LOS are not computed due to some features missing in the analyzed data.

Developing the ability to understand and predict all the user engagement metrics will help in developing an adaptive streaming method that is responsive to the QoE of the individual viewer (instead of just the QoS factor of a viewer's device). Another important system variable is load; indeed, ***load forecasting*** algorithms will be useful in dynamic resource allocation. In [41], we experimented with Neural networks [30], [38], Nearest neighbor classifiers [35], and Survival modeling [13] techniques in developing a PTR prediction tool. The remainder of this paper is dedicated to PTR prediction. This will be useful in developing the proposed system and the concomitant user-centered QoE prediction models.

B. Estimation Module

The objective of the estimation module is to infer and provide all the features required by the predictive module. First, the estimation module performs the following to prepare the data for training.

- ***Anomaly detection:*** It is desired to avoid using data containing anomalous events for training. Anomaly detection [8] is also important for accurate feature extraction, security threat detection and QoE monitoring.
- ***Threat detection:*** Threats are unauthorized usage of content such as accessing unauthorized videos (by sharing login credentials or through other means). Threats are more difficult to detect than anomalies because what constitutes a threat depends on the circumstance. In the VoD domain, threat is an unauthorized usage of content by the subscribers and non-subscribers getting access to content that are not intended to be accesses. Such unauthorized usage is not conducive to the sustained operation of the content provider. The most effective threat detection combines informative features from both anomaly-based and signature-based approaches; understanding of normal (and possibly abnormal) signatures is crucial to devising an effective threat detection strategy.

C. Adaptation Module

The adaptation model consists of the following important sub-modules:

- ***Video recommendation:*** Video recommendation is an indirect way of improving the QoE of a viewer. Significant attention has been given in the past decade in developing recommendation algorithms. Our proposed methodology will benefit from such recommendation algorithms.
- ***Adaptive bitrate switching:*** Adaptive bitrate switching strategy helps in achieving uninterrupted play of the video regardless of fluctuating bandwidth (mostly on the user's side).
- ***Streaming optimization:*** Streaming optimization aims to achieve the most economic usage of bandwidth.
- ***Content management:*** Content management is required to respond to uneven and unexpected demand of particular video content at particular times.
- ***Dynamic resource management:*** Dynamic resource allocation [16] helps in optimizing the resources, such as server bandwidth and content, in a way that a guaranteed QoE can be maintained across all (of the tens of millions of) subscribers.

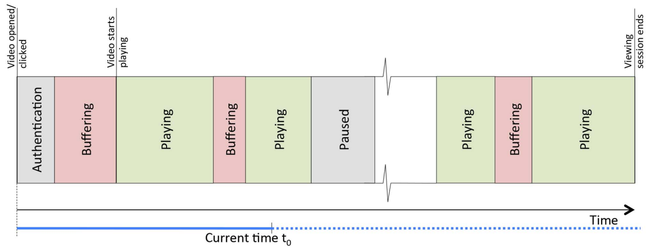


Fig. 4. **Typical video viewing session.** The purpose of the play time prediction tool (PPT) is to estimate the remaining playtime at the current point in time t_0.

III. PLAYTIME PREDICTION TOOL (PPT)

In this section, we provide a detailed description of the play time prediction tool [41] of the cognitive video streaming architecture.

Figure 4 shows a typical sequence of events in a viewing session. The session starts when the viewer requests a video. The request may go through an authentication process for non-public videos and then the video starts buffering into the local player. The amount of video being buffered (before the first video frame starts playing) depends on factors, such as the player or the bandwidth. Once a certain portion of the video buffer is filled, the video starts playing in the local player. If the streaming rate is poor, the video player might be forced to temporarily stop playing the video due to an empty buffer. As soon as the buffer is filled again, playing resumes. Nowadays, most streaming protocols use adaptive bitrate switching—meaning the bitrate is adapted dynamically in order to get the best possible video quality for the current bandwidth. The viewing session ends when the entire video is finished playing or when the viewer actively closes that video.

Functionality of the Playtime Prediction Tool (PPT)

In this work, we aim at developing an online *playtime prediction tool (PPT)* that estimates the remaining playtime in a viewing session, see Figure 4. Technically, the tool may run on either the client side or the server side.

To the best of our knowledge, there is no work yet on an online prediction of the session playtime based on an ongoing session. The most similar work [15] aims at developing methods for predicting the playtime of completed sessions.

The PPT presented in this work is the first step in creating a tool that forecasts the entire set of events in a session.

Data used for PPT

In order to perform playtime prediction, the tool exploits protocol data reported by the video player. Typically, this data contains high-level information about the video session such as in Figure 4. Content related features, e.g., the popularity of the video, also play an important role. A detailed description of the features used in this work will be given in Section V.

Methods

We demonstrate several supervised machine learning approaches for play time prediction. These approaches use previously logged protocol data for training. The proposed play time predictor can be set up for specific users, particular VoD assets, or a group of users.

Benefits

The PPT is of high value to the content provider. First and foremost, it allows the content provider to react before the session is terminated. For example, the content provider can enact counter measures to increase the service quality or recommend alternate content. Even if the PPT predicts a long playtime, the content provider in general could decrease the quality of service to a minimum acceptable level.

Second, the learned playtime prediction model encodes important information about the viewer behavior (of the entire population or even a specific viewer). For example, it is possible to perform a diagnosis that gives the most relevant features that influence the playtime. Also, a playtime prediction model allows for detecting a change in user behavior, and this potentially is of interest when threat detection is the goal.

Last but not least, playtime is a very strong indicator of the QoE. Intuitively, if the QoE is bad, the playtime will be low, too. And if the playtime is long, the QoE cannot be that bad. Hence, a model for the playtime will always be a significant part of a QoE model. In this sense, content providers are interested in increasing the playtime, i.e., the user engagement.

All told, the PPT had a substantial impact on improving the overall QoE of video streaming.

IV. METHODS FOR PLAY-TIME PREDICTION

In this section, we introduce several approaches for playtime prediction at a single specific time t_0.[1]

A. Linear Regression-based Prediction

A simple prediction model of playtime might be a linear combination of the observed features:

$$y_i = \sum_{n=0}^{N_x} k_n x_{i,n} \tag{1}$$

where $x_{i,n}$ is the nth observed feature corresponding to the ith viewing session, and y_i is the playtime. The parameter $\mathbf{k} = [k_0, k_1, k_2, \ldots, k_{N_x}]$ can be estimated by collecting the observation pairs $\{y_i, \mathbf{x}_i\}$ where $\mathbf{x}_i = [1, x_{i,1}, x_{i,1}, \ldots, x_{i,N_x}]^T$ for $i = 1, \ldots, M$, i.e.,

$$\hat{\mathbf{k}} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y} \tag{2}$$

where $\mathbf{y} = [y_1, y_2, \ldots, y_M]^T$ and $\mathbf{X} = [\mathbf{x}_1^T, \mathbf{x}_2^T, \ldots, \mathbf{x}_M^T]^T$.

For a given observed feature $\mathbf{x}_j = [1, x_{j,1}, x_{j,1}, \ldots, x_{j,N_x}]^T$, the predicted playtime is given as

$$\hat{y}_j = \mathbf{x}_j^T \hat{\mathbf{k}} \tag{3}$$

The linear prediction is useful as a comparison against other nonlinear approaches described later.

B. K-Nearest Neighbor Method

In the k-nearest neighbor approach, the target and feature pairs $\{\mathbf{y}, \mathbf{X}\}$ are kept as training-data. Given the observed feature $\mathbf{x}_j$, first, the following distance metric is computed

$$d_{i,j} = \mathcal{D}(\mathbf{x}_i, \mathbf{x}_j) \tag{4}$$

where $\mathcal{D}(\mathbf{x}_i, \mathbf{x}_j)$ is a distance measure between the arguments $\mathbf{x}_i$ and $\mathbf{x}_j$. Let $\mathbf{y}^k$ correspond to the play time of the first k of the smallest distance measures. Now, $\hat{y}_j$ is obtained in two different ways: (i) mean of $\mathbf{y}^k$, (ii) median of $\mathbf{y}^k$. The median is robust to anomalies and outliers.

C. Survival Models

Survival modeling has found wide application in a number of areas, including medicine [13] and equipment failure analysis [27]. Survival modeling was employed to derive a QoE metric in [12]. In this section, we briefly describe how survival models can be used for playtime prediction.

Let ξ be the time of termination of a particular video. The probability density function of ξ can be written as

$$P_\xi(t) \stackrel{\Delta}{=} f(t) \tag{5}$$

where $f(t)$ is also known as the *survival density function*.

The cumulative probability distribution function of ξ

$$F(t) = P(\xi \le t) = \int_0^t f(u)du \tag{6}$$

is the fraction of the videos terminated at time t. The remaining (still playing) portion of videos is given by

$$R(t) = P(\xi > t) = 1 - F(t) \tag{7}$$

where $R(t)$ is also known as the *reliability.*

Given that a video has survived until time t, it is often of interest to know the probability that it will be terminated in the next moment, i.e.,

$$h(t) = f(t \mid \xi > t) = \frac{f(t)}{R(t)} \tag{8}$$

denotes the instantaneous risk or *hazard rate* of the system. Let us rewrite (8) as

$$h(t) = \frac{f(t)}{1 - F(t)} = \frac{F'(t)}{1 - F(t)} = -\frac{R'(t)}{R(t)} \tag{9}$$

[1]Hence, we can omit t_0 in the notation used in the remainder of this paper.

Integrating both sides of (9)

$$-\int_0^t h(u)du = \ln R(t) \tag{10}$$

Hence,

$$R(t) = \exp\{-H(t)\} \tag{11}$$

where $H(t) = \int_0^t h(u)du$ is the *cumulative hazard function.*

Using (7) and (11)

$$1 - F(t) = \exp\{-H(t)\}$$

$$f(t) = h(t)\exp\{-H(t)\} \tag{12}$$

So far it has been assumed that $f(t)$ (and hence $R(t)$ and $h(t)$) are all functions of time only. However, all of these functions are dependent on features $\mathbf{x} = \{\mathbf{x}_i\}$, or *covariates*. The *proportional hazard function*, proposed by Cox [13], suggests to separate the time-dependent and feature-dependent hazards as follows:

$$h(t,\mathbf{x}) = \lambda(t)\exp\{\mathbf{b}^T\mathbf{x}\} \tag{13}$$

where $\lambda(t)$ is the baseline time-dependent hazard function, x_i is the covariate, and b_i is the coefficient corresponding to the ith covariate, x_i.

Now, (11) and (12) are rewritten as

$$f(t) = \lambda(t)\exp\{\mathbf{b}^T\mathbf{x} - \Lambda(t)e^{\mathbf{b}^T\mathbf{x}}\} \tag{14}$$

$$R(t) = \exp\{-\Lambda(t)e^{\mathbf{b}^T\mathbf{x}}\} \tag{15}$$

where $\Lambda(t) = \int_0^t \lambda(u)du$. Cox suggested that the the model parameters $\mathbf{b}$ can be estimated independent of $\lambda(t)$ by maximizing the partial likelihoods. Once $\mathbf{b}$ is estimated, there are several approaches in the literature to model and estimate (the parameters of) $\lambda(t)$.

Once the parameters are estimated, the remaining play time at time u can be computed as

$$\hat{y}_j(u) = \frac{\int_u^\infty (t-u)f_j(t)dt}{R_j(u)} \tag{16}$$

where $f_j(t)$ and $R_j(u)$ are obtained by substituting $\mathbf{x}_j$ for $\mathbf{x}$ in (14) and (15), respectively, and u is the time elapsed.

An advantage of the survival model-based approaches described above is that the playtime prediction can be updated as the video progresses. In this paper, we assume $\lambda(t) = \lambda$.

D. Neural Networks

The playtime can be modeled as a function of the observed features using artificial neural networks (e.g., multi-layer perceptrons)

$$y_i = f(\mathbf{x}_i, \{w_{l,k}\}_{l=1,k=1}^{N_L,N_h}) \tag{17}$$

where $w_{l,k}$ are different weights and N_L is the number of layers and N_h is the number of hidden nodes. Given a set of (past) training data $\mathbf{y}, \mathbf{X}$, there are several approaches to learn the weights [37]. A trained neural network can be used to predict the playtime for a given feature set $\mathbf{x}_j$.

Neural Network predictor was implemented by the use of the built in neural network function in Matlab™. The number of neurons and the number of hidden layers are selected to be the ones to give the highest prediction accuracy metrics with the training data. For the particular example described in Section V, a multi-layer perceptron model was selected with three hidden layers each having six neurons.

V. SIMULATION STUDIES

In this section, we evaluate the proposed approaches using data from 8808 viewing sessions. In order to avoid any confounding effects, all these 8808 viewing sessions are selected from the same type of video; in particular, all these videos are selected to be the episodes of "The Simpsons." Further, all these videos were viewed on the same day. We focus on the first 8 minutes as we try to understand early quitters due to the low streaming quality. A portion of these sessions is randomly selected and denoted as the "learning" dataset, and the rest is kept for testing. Each feature in the testing data is used for predicting its playtime. This procedure is repeated for 10 Monte-Carlo runs.

Our work is based on a dataset from the VoD streaming service *Xfinity On Demand* from *Comcast*. The available data was logged by the video players and consists of a sequence of events that come with time stamps, device ids, and further information. Specifically, we use the following logged events from each user. For each of these events, the starting and ending times are available.

- *Opening:* Indicates that a new viewing session is opened by the user.
- *Playing:* Video starts playing.
- *Buffering:* The player starts buffering; the video doesn't play until a certain amount of data is buffered. Further, the buffering event can occur while a video is playing.
- *Paused:* The pause event occurs when the user presses the pause button.
- *Closing:* Video may stop playing either due to the user ending the session or when the end of the video is reached.
- *Bitrate switched:* This event occurs whenever the streaming bitrate changes.

We define a viewing session as the events between the opening and closing events at a particular device. Based upon the above described events, we determine the following session features that potentially affect the playtime and the QoE.

A. Data Analysis and Visualization

The following features are used in our current analysis:

1) Number of buffering events (f_1)

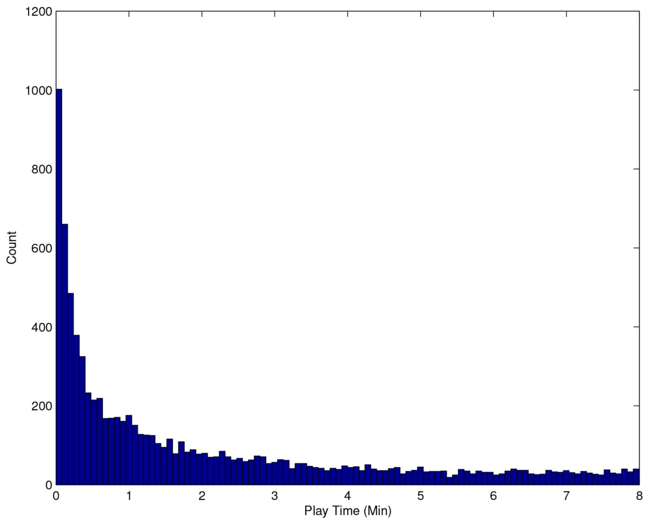


Fig. 5. **Histogram of playtime.**

2) Number of paused events (f_2)
3) Inter buffering time (f_3): The average time (in seconds) between two buffering events.
4) Startup time (f_4): The time it takes from when the user hits the play button to the time the video starts playing on the screen.
5) Average bit rate (f_5): The average bit rate is measured in Mega bits per second (Mbps).
6) Buffering ratio (f_6): the relation between the total buffering time and the total play time of a video. The buffering ratio negatively affects the QoE.

Figure 5 shows the histogram of playtime for all the 8808 viewing sessions. The play time distribution suggests an exponential decay in this case. Figure 6 shows the histograms of the corresponding features. It can be seen that the majority of the video sessions had up to two buffering and paused events each. The startup time is approximately 4 seconds for the majority of the videos. The peaks around the 1.8 Mbps and 4.2 Mbps indicate the presence of standard video and high definition video, respectively.

B. Performance Metrics

In this section, we use the algorithms introduced in Section IV for playtime prediction and assess their performance. Due to lack of knowledge on the statistical

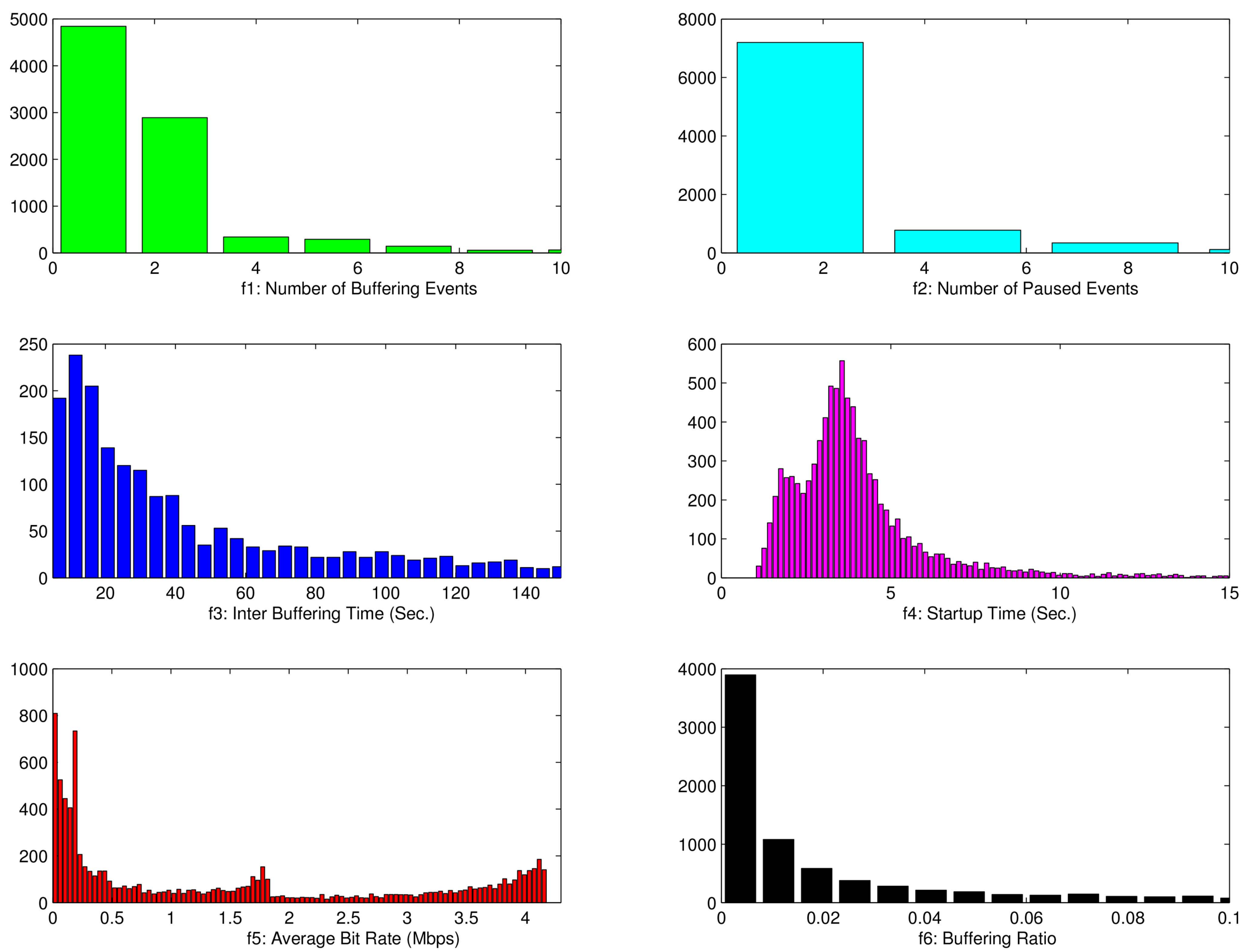


Fig. 6. **Histogram of features.**

properties of playtime,we suggest using several surrogate metrics for assessing playtime. The following four metrics were considered.

1) Normalized Mean-Squared Error (NMSE): This metric gives insight on the error in playtime prediction and is given by

$$\mathrm{NMSE} = \frac{1}{M}\sum_{i=1}^{M}\left(\frac{y_i - \hat{y}_i}{y_i}\right)^2 \tag{18}$$

2) R^2 Fit: The coefficient of determination, R^2, gives insight into how well the data points fit the statistical model used to predict playtime. A value of $R^2 = 1$ indicates perfect fit, and smaller the R^2, the poorer is the fit.

$$R^2 = 1 - \frac{\sum_{i=1}^{M}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{M}(y_i - \bar{y})^2} \tag{19}$$

where $\bar{y} = (1/M)\sum_{i=1}^{M} y_i$.

3) Ratio of Predicted and True Playtime Greater than r: The playtime is a quantity that can generally vary anywhere from less than 1 minute to several hours. A prediction error of 1 min is significant if the actual play time is 5 min; however, it is not so significant if the actual play time is 2 hours. The NMSE captures this through normalization; however, the following metric captures this error in a different light.

$$\mathrm{RG}(r) = \frac{\#\left\{\dfrac{\hat{y}_i}{y_i} > r\right\}}{M} \tag{20}$$

where $\#\{\cdot\}$ denotes the number of times the argument is true.

4) Ratio of Predicted and True Playtime Less than $1/r$: Similar to $\mathrm{RG}(r)$, the following metric captures the instances when the prediction was significantly smaller than the true value of playtime.

$$\mathrm{RL}(1/r) = \frac{\#\left\{\dfrac{\hat{y}_i}{y_i} < \dfrac{1}{r}\right\}}{M} \tag{21}$$

C. Feature Selection

With N features, there are $2^N - 1$ possible subsets of features. Although it might be thought that more is better, in machine learning, one can be subject to the "curse of dimensionality": extra features that are uninformative actually hurt prediction performance by "fitting to the noise." In Figures 7, 8, 9 and 10, we show the performance(s) plotted against binary representation of feature combinations, from 1 to $2^N - 1$. Each time, half the dataset is randomly selected and used for learning and the playtime is predicted using the rest of the data. This procedure is repeated for 10 Monte-Carlo runs (This is called a 10×2 cross validation.) and the median of each of the metrics is plotted in Figures 7–10. There are six subplots in each of Figures 7–10, showing the results of different playtime prediction approaches: Survival modeling, k-nearest neighbor (mean), k-nearest neighbor (median), LS, neural networks and random. In "random" approach, we randomly select a playtime from the training dataset.

TABLE III
Performance Metrics

Feature Combination ID	f1	f2	f3	f5	f6	f7	R2	NMSE	RG(2)	RL(0.5)
61	x		x	x	x	x	**0.74479**	1.1028	0.19743	0.074251
55	x	x	x		x	x	0.73473	1.1427	0.22604	0.073569
63	x	x	x	x	x	x	0.71984	1.1738	0.24353	0.068233
57	x			x	x	x	0.71637	1.1788	0.25386	0.059378
59	x	x		x	x	x	0.7162	1.1766	0.25079	0.065054
49	x				x	x	0.71531	1.1683	0.2584	0.066417
61	x		x	x	x	x	0.74479	**1.1028**	0.19743	0.074251
55	x	x	x		x	x	0.73473	1.1427	0.22604	0.073569
49	x				x	x	0.71531	1.1683	0.2584	0.066417
63	x	x	x	x	x	x	0.71984	1.1738	0.24353	0.068233
59	x	x		x	x	x	0.7162	1.1766	0.25079	0.065054
57	x			x	x	x	0.71637	1.1788	0.25386	0.059378
61	x		x	x	x	x	0.74479	1.1028	**0.19743**	0.074251
37	x		x			x	0.69295	1.2289	0.2072	0.070391
36			x			x	0.54402	1.4849	0.21946	0.067666
55	x	x	x		x	x	0.73473	1.1427	0.22604	0.073569
45	x		x	x		x	0.65841	1.2883	0.23399	0.073683
63	x	x	x	x	x	x	0.71984	1.1738	0.24353	0.068233
41	x			x		x	0.65943	1.2865	0.28213	**0.043824**
53	x		x		x	x	0.71386	1.1796	0.271	0.057788
57	x			x	x	x	0.71637	1.1788	0.25386	0.059378
39	x	x	x			x	0.67346	1.2644	0.25318	0.060513
59	x	x		x	x	x	0.7162	1.1766	0.25079	0.065054
33	x					x	0.65471	1.2997	0.31903	0.065622
Total Features	23	10	14	14	18	24				

TABLE IV
Feature Ranking Based on Borda Count

Feature ID	Rank: R2	Rank: NMSE	Rank: RG(2)	Rank: RL(0.5)	Borda Count	Borda Rank
61	1	1	1	13	40	1
55	2	2	4	11	37	2
57	4	6	9	3	34	3
59	5	5	7	5	34	3
63	3	4	6	9	34	3
49	6	3	10	7	30	4
53	7	7	11	2	29	5
37	8	8	2	10	28	6
39	9	9	8	4	26	7
41	10	10	12	1	23	8
36	13	13	3	8	19	9
45	11	11	5	12	17	10
33	12	12	13	6	13	11

Next we select just one playtime prediction approach shown in Figures 7–10 and try to select the best feature set (out of $2^N - 1$) for online prediction. We select the neural networks approach for this evaluation. The objective of feature selection is to find the features that gives the best result across all performance metrics defined in Section V-B.

Table III shows the first six feature sets ranked according to each of the performance metrics: R^2, NMSE, RG(2) and RL(0.5). For example, the features corresponding to the binary number 61, i.e, NRB, IBT, STT, BR and BUR, give the best performance according to R^2, NMSE and RG(2), whereas the features corresponding to the binary number 41, i.e., NRB, STT, and BUR, give the best performance according to RL(0.5).

We employ a method known as *Borda count* [9] in order to select the best feature subset based on all four evaluation metrics. For each feature ID (binary number) in Table III, the Borda count gives a point based on the ranking of that ID using each of the four evaluation metrics. Then, the feature ID having the most Borda points is selected as the best feature set in terms of all four evaluation metrics. Table IV summarizes the Borda count procedure in selecting the best feature subset. For this particular example, the feature subset with ID 61 is ranked first, while the one with ID 41 is ranked 8th.

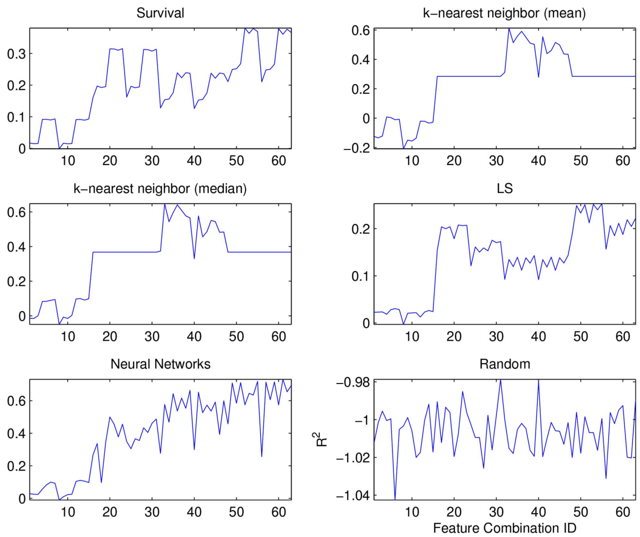


Fig. 7. **Median R^2 Fit.**

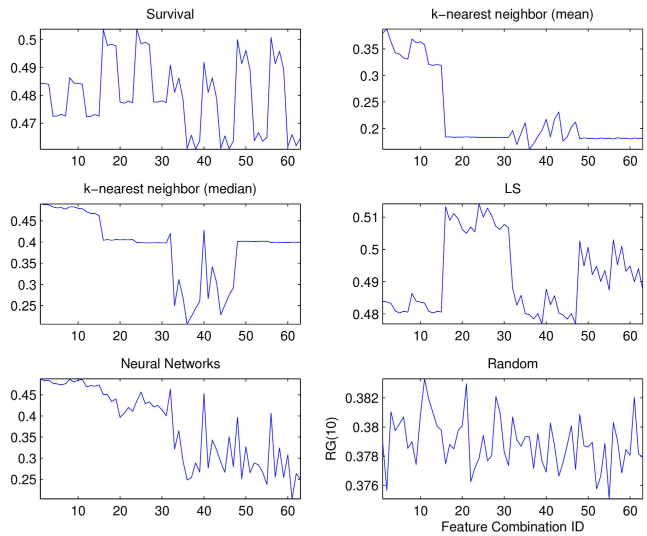


Fig. 9. **Median RG(r).**

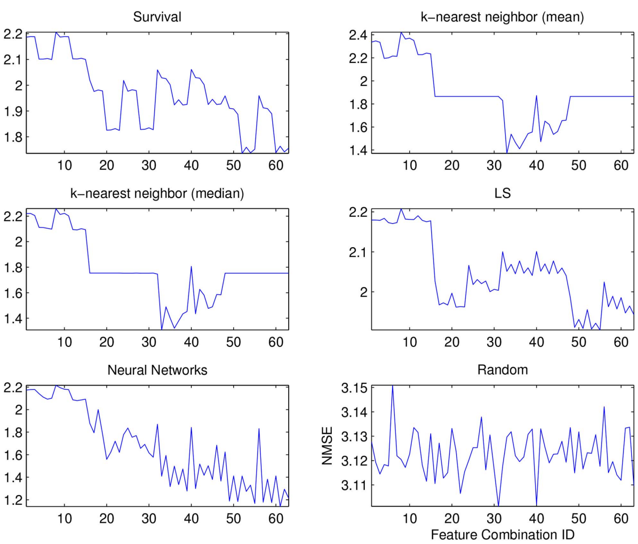


Fig. 8. **Median NMSE.**

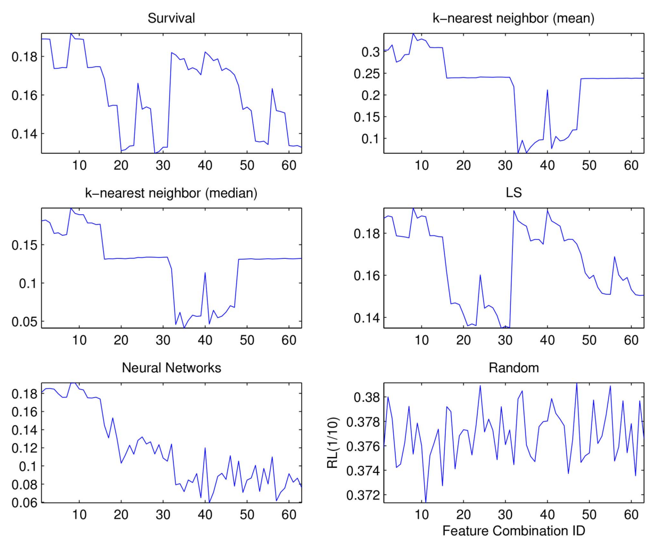


Fig. 10. **Median RL($1/r$).**

Hence, for neural network approach, the features NRB, IBT, STT, BR and BUR will be used for online playtime prediction. The features are selected in a similar fashion for the rest of the five playtime prediction methods.

D. Playtime Prediction Results

Assuming that the best features are selected offline based on the approach described in the previous section, in this section we show the online playtime prediction results of each approach.

Figure 11 shows a scatter plot of true vs. predicted play time. Each subplot corresponds to the prediction approach mentioned in the title. For each approach, the feature ID corresponding to the top Borda count is displayed in parenthesis as well. Ideally, the scatter plot should look like a line from the origin with gradient 1; the "thickness" of the scatter as well as "concentrations" at off-diagonal places indicate the error in predictions.

Figure 12 shows the predicted play time as an overlay plot of true and estimates; the blue line shows the actual play time and the red stars are the predicted ones.

Figure 13 shows the prediction errors as a histogram; a "thin" histogram implies good prediction and vice versa. Out of all the six approaches, the neural network approach yields the best prediction results followed by both of the k-nearest neighbor methods. The "random" approach is shown as a measure of comparison to the worst method; the random approach randomly picks a data from the training set as the predicted play time.

The playtime prediction results shown in this section demonstrate that the proposed approaches are promising, since they all perform better than the random prediction approach. However, our objective in this paper is not to develop a perfect play time prediction tool, rather to demonstrate a functioning PPT, which is a component of the proposed cognitive video streaming architecture.

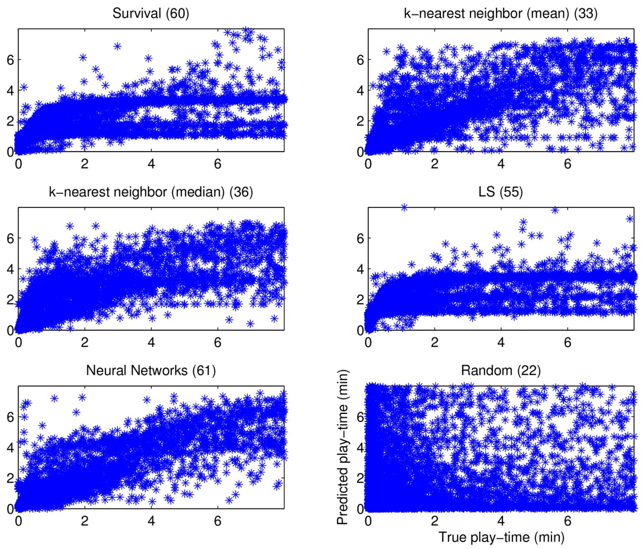


Fig. 11. **Scatter plot of true vs. predicted playtime.**

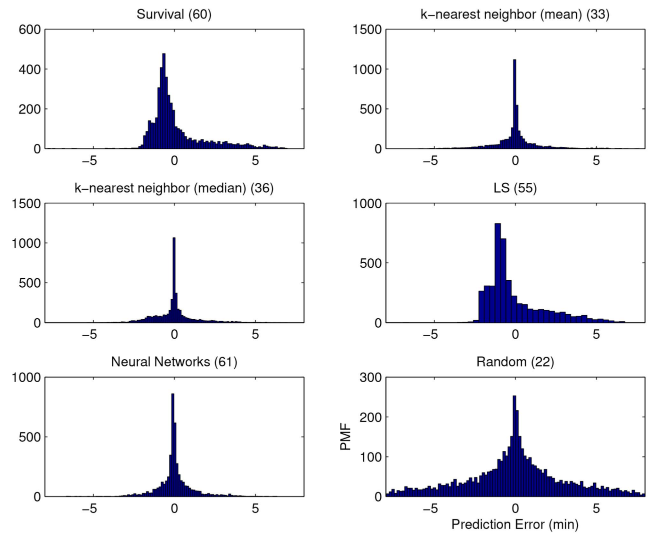


Fig. 13. **PMF of playtime-prediction error.**

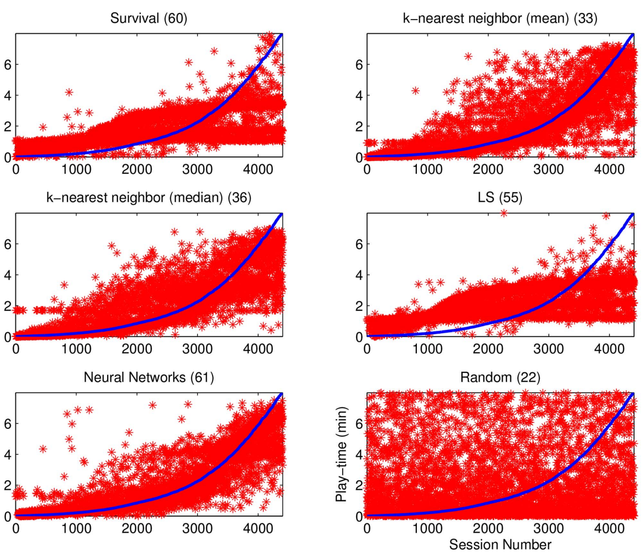


Fig. 12. **Overlay plot of true vs. predicted playtime.**

For more accurate playtime prediction results, the PPT has to be "fed" with more than the QoS related features, such as the content related features, the viewer related features and the external features. In this paper, we limited the discussion to outlining the technical details and demonstration of the PPT with just QoS related features.

VI. CONCLUSIONS AND DISCUSSION

The contribution of the paper is to describe a framework for modeling the variables that may affect the quality of experience (QoE) of video-on-demand (VoD) services, with the aim of maximizing QoE for subscribers of commercial media providers. However, QoE is difficult to measure based on the usage data that is available to the service provider: it is only indirectly inferable. Hence, one contribution of this paper is to formulate the problem and discuss the relevant literature, much of which appears in quite diverse journals. Implicit is that if QoE is well understood, its real-time prediction might be used to adapt the underlying content delivery strategy via adaptive bit-rate switching, streaming optimization, content management, dynamic resource management and video recommendations.

We proposed and discussed various QoE measures, such as *playtime, probability of return, probability of replay* and *average length of scrubbing*. We discussed approaches for the prediction of such QoE measures by supervised machine learning algorithms. Further, we described the type of features that can be computed from the subscriber data for use in the QoE prediction algorithms as predictive features. We categorized these features into *content-related features*, *viewer-related* features and *quality of service (QoS)-related* features. Then, we demonstrated playtime prediction through several supervised classification approaches using QoS related features that are collected from the subscribers of a popular VoD service.

The proposed cognitive video streaming architecture is suitable to future developments in the fast changing video consumption arena. For example, more accurate measure of the QoE can be obtained by making use of other relevant data. In [31], [32] and [34], we used the eye tracking data, such as pupil dilation and eye-gaze pattern in order to estimate the cognitive context of unmanned aerial system (UAS) operators, while they execute reconnaissance missions. However, existing VoD systems are not equipped to measure/collect eye tracking data. Considering the fact that most of the video playing devices (with the exception of TV) are equipped with a front facing camera, creating and exploiting eye tracking data for QoE estimation has a good chance of becoming a reality.

The availability of additional physiological measurements, such as heart rate, breathing rate and body temperature will pave the way for improved accuracy in QoE estimation. None of the existing video devices are equipped with the sensors to measure these physiological features. However, driven by the personal health monitoring devices (also known as fitness trackers, such as fitbit™), the (direct or indirect) availability of these physiological measurements for VoD services could become a reality in the future. If and when that happens, the QoE estimation can be done with increased confidence and the cognitive video streaming architecture will be able to cater to more advanced form of entertainment.

In general, QoE estimation is a part of the much larger human machine systems (HMS). An HMS is formed when a semi-autonomous system is operated by a human (or group of humans) operator(s); (i) a pilot flying an aircraft, (ii) a car driven by a human driver, (iii) a person working on a computer, and (iv) an unmanned aerial system (UAS) mission executed by a group of operators are all examples of HMS. It has been well understood that human performance becomes suboptimal when the workload is too high as well as when it is too low [46]. An important research challenge in the HMS domain is to create machines that are able to better understand human behavior so that the overall efficiency of the HMS can be improved through increased productivity and reduced safety risk [33]. An understanding of the physiological behavior of the human body can be combined with statistical machine learning theory in order to develop algorithms that are able to accurately predict the cognitive context (such as the difficulty of work, level of alertness, etc) experienced by humans. Also, a more generalized topic of context based information fusion [47] gives additional insights on this emerging research field.

The future of VoD system will look more similar to a HMS with one exception: the objective of all the other HMS is to perform a certain task with high efficiency, whereas the objective of the VoD system is to offer entertainment and pleasure to the human. A new ecosystem of services and applications are waiting to be developed around a successful VoD system. For example, doctors might prescribe certain videos as part of treatment plans; students might be asked to watch a certain video as part of a curriculum, all under the assumption that a dependable QoE estimation system (in general terms, a cognitive context detection system) is available. The immediate challenge of the information fusion community is to develop such systems.

ACKNOWLEDGEMENTS

The authors would like to thank the anonymous reviewers for constructive reviews of this work. This research was funded by Comcast Corporation and the U.S. Office of Naval Research (ONR) under contracts #PO 998604 and #N00014-16-1-2036. Peter Willett was supported by the Naval Postgraduate School via ONR contract #N00244-16-1-0017.

REFERENCES

[1] "Cisco visual networking index: Forforecast methodology 2013–2018." Cisco, Tech. Rep., June 2014.

[2] *Transparent End-to-End Packet Switched Streaming Service (PSS); Progressive Download and Dynamic Adaptive Streaming Over HTTP (3GP-DASH)*, 3GPP Std. TS 26.247 v10.1.0, June 2011.

[3] V. Aggarwal, E. Halepovic, J. Pang, S. Venkataraman, and H. Yan Prometheus: Toward quality-of-experience estimation for mobile apps from passive network measurements, *ACM HotMobile*, 2014.

[4] A. Anand, A. Balachandran, A. Akella, V. Sekar, and S. Seshan Enhancing video accessibility and availability using information-bound references, in *Proceedings of the ninth ACM conference on Emerging networking experiments and technologies*. ACM, 2013, pp. 345–356.

[5] A. Balachandran, V. Sekar, A. Akella, and S. Seshan Analyzing the potential benefits of CDN augmentation strategies for internet video workloads, in *Proceedings of the 2013 conference on Internet measurement conference*. ACM, 2013, pp. 43–56.

[6] A. Balachandran, V. Sekar, A. Akella, S. Seshan, I. Stoica, and H. Zhang A quest for an internet video quality-of-experience metric, in *Proceedings of the 11th ACM Workshop on Hot Topics in Networks*. ACM, 2012, pp. 97–102.

[7] ——— Developing a predictive model of quality of experience for internet video, in *ACM SIGCOMM Computer Communication Review*, vol. 43, no. 4. ACM, 2013, pp. 339–350.

[8] B. Balasingam, S. Sankavaram, K. Choi, D. M. Ayala, D. Sidoti, K. R. Pattipati, P. Willett, C. Lintz, G. Commeau, F. Dorigo, and J. Fahrny Online anomaly detection in big data, in *International Conference on Information Fusion*, 2014.

[9] D. Black Partial justification of the Borda count, *Public Choice*, vol. 28, no. 1, pp. 1–15, 1976.

[10] P. Brooks and B. Hestnes User measures of quality of experience: why being objective and quantitative is important, *IEEE Network*, vol. 24, no. 2, pp. 8–13, March 2010.

[11] J. Chen, A. Ghosh, J. Magutt, and M. Chiang QAVA: quota aware video adaptation, in *Proceedings of the 8th international conference on Emerging networking experiments and technologies*. ACM, 2012, pp. 121–132.

[12] K.-T. Chen, C.-Y. Huang, P. Huang, and C.-L. Lei Quantifying Skype user satisfaction, in *ACM SIGCOMM Computer Communication Review*, vol. 36, no. 4. ACM, 2006, pp. 399–410.

[13] D. R. Cox and D. Oakes *Analysis of survival data*. CRC Press, 1984, vol. 21.

[14] N. Damera-Venkata, T. D. Kite, W. S. Geisler, B. L. Evans, and A. C. Bovik
Image quality assessment based on a degradation model,
IEEE Transactions on Image Processing, vol. 9, no. 4, pp. 636–650, 2000.

[15] F. Dobrian, V. Sekar, A. Awan, I. Stoica, D. A. Joseph, A. Ganjam, J. Zhan, and H. Zhang
Understanding the impact of video quality on user engagement,
SIGCOMM-Computer Communication Review, vol. 41, no. 4, p. 362, 2011.

[16] X. Han, S. Mandal, K. Pattipati, D. Kleinman, and M. Mishra
An optimization-based distributed planning algorithm: A blackboard-based collaborative framework,
IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol. 44, no. 6, pp. 673–686, June 2014.

[17] T.-Y. Huang, R. Johari, and N. McKeown
Downton Abbey without the hiccups: Buffer-based rate adaptation for http video streaming,
in *Proceedings of the 2013 ACM SIGCOMM workshop on Future human-centric multimedia networking*. ACM, 2013, pp. 9–14.

[18] *Objective perceptual video quality measurement techniques for standard definition digital broadcast television in the presence of a full reference*,
International Telecommunication Union Std., ITU-R Recommendation BT.1683, 2004.

[19] *Methodology for the subjective assessment of the quality of television pictures*,
International Telecommunication Union Std., ITU-R Recommendation BT.500-11, 2002.

[20] *Objective perceptual video quality measurement techniques for digital cable television in the presence of a full reference*,
International Telecommunication Union Std., ITU-T Recommendation J.144, 2004.

[21] *Perceptual visual quality measurement techniques for multimedia services over digital cable television networks in the presence of a reduced bandwidth reference*,
International Telecommunication Union Std., ITU-T Recommendation J.246, 2008.

[22] *Objective perceptual multimedia video quality measurement in the presence of a full reference*,
International Telecommunication Union Std., ITU-T Recommendation J.247, 2008.

[23] *Subjective video quality assessment methods for multimedia applications*,
International Telecommunication Union Std., ITU-T Recommendation P.910, 2008.

[24] *Information Technology—Dynamic Adaptive Streaming Over HTTP (DASH)—Part 1: Media Presentation Description and Segment Formats*,
ISO Std. IEC DIS 23 009-1, Aug 2011.

[25] S. S. Krishnan and R. K. Sitaraman
Video stream quality impacts viewer behavior: inferring causality using quasi-experimental designs,
in *Proceedings of the 2012 ACM conference on Internet measurement conference*. ACM, 2012, pp. 211–224.

[26] S. Krishnan and R. Sitaraman
Video stream quality impacts viewer behavior: Inferring causality using quasi-experimental designs,
IEEE/ACM Transactions on Networking, vol. 21, no. 6, pp. 2001–2014, Dec 2013.

[27] W. C. Levy, D. Mozaffarian, D. T. Linker, S. C. Sutradhar, S. D. Anker, A. B. Cropp, I. Anand, A. Maggioni, P. Burton, M. D. Sullivan et al.
The seattle heart failure model prediction of survival in heart failure,
Circulation, vol. 113, no. 11, pp. 1424–1433, 2006.

[28] Z. Li, X. Zhu, J. Gahm, R. Pan, H. Hu, A. Begen, and D. Oran
Probe and adapt: Rate adaptation for HTTP video streaming at scale,
IEEE Journal on Selected Areas in Communications, vol. 32, no. 4, pp. 719–733, April 2014.

[29] C. Liu, R. K. Sitaraman, and D. Towsley
"Go-with-the-winner: Client-side server selection for content delivery,"
arXiv preprint arXiv:1401.0209, 2013.

[30] R. Lynch and P. Willett
Bayesian classification and feature reduction using uniform Dirichlet priors,
IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, vol. 33, no. 3, pp. 448–464, June 2003.

[31] P. Mannaru, B. Balasingam, K. Pattipati, C. Sibley, and J. Coyne
Cognitive context detection in UAS operators using gaze patterns,
in *SPIE Conferences on Defense, Security, and Sensing*, April 2016.

[32] ———
Cognitive context detection in UAS operators using pupillary measurements,
in *SPIE Conferences on Defense, Security, and Sensing*, April 2016.

[33] ———
Human-machine system improvement through cognitive context detection,
in *Annual Meeting of the Human Factors and Ergonomics Society*, Sept. 2016.

[34] ———
On the use of hidden Markov models for eye-gaze pattern modeling and classification,
in *SPIE Conferences on Defense, Security, and Sensing*, April 2016.

[35] S. Marano, V. Matta, and P. Willett
Nearest-neighbor distributed learning by ordered transmissions,
IEEE Transactions on Signal Processing, vol. 61, no. 21, pp. 5217–5230, Nov 2013.

[36] R. K. Mok, E. W. Chan, and R. K. Chang
Measuring the quality of experience of HTTP video streaming,
in *IFIP/IEEE International Symposium on Integrated Network Management*. IEEE, 2011, pp. 485–492.

[37] K. P. Murphy
Machine learning: a probabilistic perspective.
MIT press, 2012.

[38] C. Neukirchen, J. Rottland, D. Willett, and G. Rigoll
A continuous density interpretation of discrete HMM systems and MMI-neural networks,
IEEE Transactions on Speech and Audio Processing, vol. 9, no. 4, pp. 367–377, May 2001.

[39] O. Oyman and S. Singh
Quality of experience for HTTP adaptive streaming services,
IEEE Communications Magazine, vol. 50, no. 4, pp. 20–27, 2012.

[40] D. Pasupuleti, P. Mannaru, B. Balasingam, M. Baum, K. R. Pattipati, and P. Willett
Cognitive video streaming,
in *International Conference on Electrical, Electronics, Engineering Trends, Communication, Optimization and Sciences*, March 2015.

[41] D. Pasupuleti, P. Mannaru, B. Balasingam, M. Baum, K. R. Pattipati, P. Willett, C. Lintz, G. Commeau, F. Dorigo, and J. Fahrny
Online playtime prediction for cognitive video streaming, in *IEEE International Conference on Information Fusion*, July 2015.

[42] M. H. Pinson and S. Wolf
A new standardized method for objectively measuring video quality,
IEEE Transactions on Broadcasting, vol. 50, no. 3, pp. 312–322, 2004.

[43] F. Qiu and Y. Cui
A quantitative study of user satisfaction in online video streaming,
in *Consumer Communications and Networking Conference*. IEEE, 2011, pp. 410–414.

[44] J. Shaikh, M. Fiedler, T. Minhas, P. Arlos, and D. Collange
Passive methods for the assessment of user-perceived quality of delivery,
in *SNCNW*, 2011, p. 73.

[45] S.-H. Shen and A. Akella
An information-aware QoE-centric mobile video cache,
in *Proceedings of the 19th annual international conference on Mobile computing & networking*. ACM, 2013, pp. 401–412.

[46] C. Sibley, J. Coyne, and J. Morrison
Research considerations for managing future unmanned systems. 2015,
in *AAAI Spring Symposium on Foundations of Autonomy and Its (Cyber) Threats: From Individuals to Interdependence*. AAAI Press, 2015.

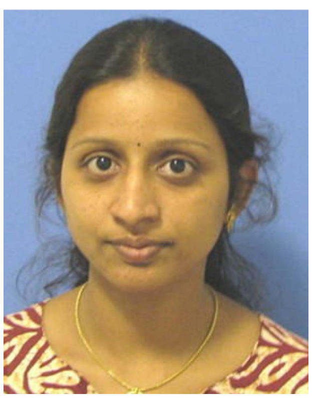
[47] L. Snidaro, J. García, and J. Llinas
Context-based information fusion: a survey and discussion,
Information Fusion, vol. 25, pp. 16–31, 2015.

[48] C. J. Van den Branden Lambrecht and O. Verscheure
Perceptual quality measure using a spatiotemporal model of the human visual system,
in *Electronic Imaging: Science & Technology*. International Society for Optics and Photonics, 1996, pp. 450–461.

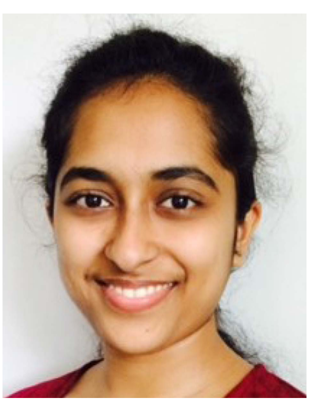
[49] M. Venkataraman and M. Chatterjee
Quantifying video-QoE degradations of internet links,
IEEE/ACM Transactions on Networking, vol. 20, no. 2, pp. 396–407, April 2012.

[50] Z. Wang and A. C. Bovik
Mean squared error: love it or leave it? a new look at signal fidelity measures,
IEEE Signal Processing Magazine, vol. 26, no. 1, pp. 98–117, 2009.

[51] Z. Wang, L. Lu, and A. C. Bovik
Video quality assessment based on structural distortion measurement,
Signal processing: Image communication, vol. 19, no. 2, pp. 121–132, 2004.



Devaki Pasupuleti received her M.S. in Electrical Engineering from University of Connecticut, Storrs in 2015. Since 2015, she has been a Software Engineer at Cisco Systems, San Jose. Her research interests are in Big Data, Machine learning, Networking Protocols, Software Defined Networking, distributed information fusion and their applications in automated systems. Devaki Pasupuleti has authored 2 conference papers in these areas.



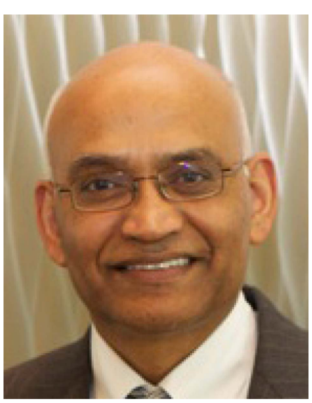
Pujitha Mannaru received the B.E. degree in Electronics and Communications Engineering from PES Institute of Technology—Bangalore South Campus, affiliated to Visvesvaraya Technological University, India, in 2013. She is currently pursuing her Ph.D. in Systems Engineering at the University of Connecticut, Storrs, USA. Her research interests are in the areas of applications of signal processing, machine learning, and psychophysiological measures to proactive decision support tools and human-machine systems.

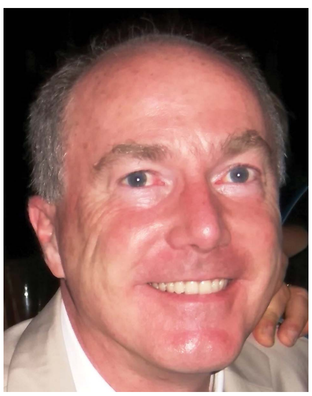
Balakumar Balasingam received his M.A.Sc. and Ph.D. in Electrical Engineering from McMaster University, Canada in 2004 and 2008, respectively. He held a postdoctoral position at the University of Ottawa from 2008 to 2010, and then a University Postdoctoral position at the University of Connecticut from 2010 to 2012. Since 2012, he has been an Assistant Research Professor in the Department of Electrical and Computer Engineering at the University of Connecticut. His research interests are in signal processing, machine learning, and distributed information fusion and their applications in cyber-physical systems, cyber-human systems and human-machine systems. Dr. Balasingam has authored over 50 research publications in these areas.

Marcus Baum is a Juniorprofessor (Assistant Professor) at the University of Goettingen, Germany. He received the Diploma degree in computer science from the University of Karlsruhe (TH), Germany, in 2007, and graduated as Dr.-Ing. (Doctor of Engineering) at the Karlsruhe Institute of Technology (KIT), Germany, in 2013. From 2013 to 2014, he was postdoc and assistant research professor at the University of Connecticut, CT, USA. His research interests are in the field of data fusion, estimation, and tracking. Marcus Baum is associate administrative editor of the *Journal of Advances in Information Fusion* (JAIF) and serves as local arrangement chair of the 19th International Conference on Information Fusion (FUSION 2016). He received the best student paper award at the FUSION 2011 conference.

Krishna Pattipati received the B. Tech. degree in electrical engineering with highest honors from the Indian Institute of Technology, Kharagpur, in 1975, and the M.S. and Ph.D. degrees in systems engineering from UConn, Storrs, in 1977 and 1980, respectively. He was with ALPHATECH, Inc., Burlington, MA from 1980 to 1986. He has been with the department of Electrical and Computer Engineering at UConn, where he is currently the Board of Trustees Distinguished Professor and the UTC Chair Professor in Systems Engineering. Dr. Pattipati's research activities are in the areas of proactive decision support, uncertainty quantification, smart manufacturing, autonomy, knowledge representation, and optimization-based learning and inference. A common theme among these applications is that they are characterized by a great deal of uncertainty, complexity, and computational intractability. He is a cofounder of Qualtech Systems, Inc., a firm specializing in advanced integrated diagnostics software tools (TEAMS, TEAMS-RT, TEAMS-RDS, TEAMATE), and serves on the board of Aptima, Inc.

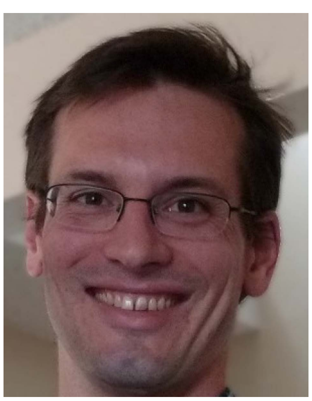
Dr. Pattipati was selected by the IEEE Systems, Man, and Cybernetics (SMC) Society as the Outstanding Young Engineer of 1984, and received the Centennial Key to the Future award. He has served as the Editor-in-Chief of the IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—PART B from 1998 to 2001, Vice-President for Technical Activities of the IEEE SMC Society from 1998 to 1999, and as Vice-President for Conferences and Meetings of the IEEE SMC Society from 2000 to 2001. He was co-recipient of the Andrew P. Sage Award for the Best SMC Transactions Paper for 1999, the Barry Carlton Award for the Best AES Transactions Paper for 2000, the 2002 and 2008 NASA Space Act Awards for "A Comprehensive Toolset for Model-based Health Monitoring and Diagnosis," and "Real-time Update of Fault-Test Dependencies of Dynamic Systems: A Comprehensive Toolset for Model-Based Health Monitoring and Diagnosticsâ, and the 2003 AAUP Research Excellence Award at UCONN. He also won the best technical paper awards at the 1985, 1990, 1994, 2002, 2004, 2005 and 2011 IEEE AUTOTEST Conferences, and at the 1997, 2004 Command and Control Conference. He is an elected Fellow of IEEE and of the Connecticut Academy of Science and Engineering.





Peter Willett (F'03) received his B.A.Sc. (engineering science) from the University of Toronto in 1982, and his Ph.D. degree from Princeton University in 1986. He has been a faculty member at the University of Connecticut ever since, and since 1998 has been a Professor. His primary areas of research have been statistical signal processing, detection, machine learning, data fusion and tracking. He also has interests in and has published in the areas of change/abnormality detection, optical pattern recognition, communications and industrial/security condition monitoring. He is editor-in-chief of IEEE Signal Processing Letters. He was editor-in-chief for IEEE Transactions on Aerospace and Electronic Systems (2006–2011), and was Vice President for Publications for AESS (2012–2014). He is a member of the IEEE AESS Board of Governors 2003–2009, 2011–2016. He was General Co-Chair for the 2006 IEEE/ISIF Fusion Conference in Florence, Italy, and again for both 2008 in Cologne, Germany and 2011 in Chicago IL. He was Program Co-Chair for the 2016 ISIF/IEEE Fusion Conference in Heidelberg, Germany, the 2003 IEEE Conference on Systems, Man & Cybernetics in Washington DC, and Program Co-Chair for the 1999 Fusion Conference in Sunnyvale CA.

Christopher Lintz is with Comcast Corporation.



Gabriel Commeau is a software engineer who has been working for Comcast Corporation for the last 5 years on BigData projects and real-time processing platforms. He is interested in cutting-edge technologies, especially in the domains of distributed systems, artificial intelligence and security. He currently provides technical leadership for an initiative to create an enterprise Stream Data Platform in order to manage the wide range of data streams generated by Comcast's systems.

Franchesco Dorigo is with Comcast Corporation.

Jim Fahrny is with Comcast Corporation.