DeSVQ: Deep Learning Based Streaming Video QoE Estimation

Conference Paper · January 2022

DOI: 10.1145/3491003.3491023

3 authors, including:

Monalisa Ghosh
International Institute of Information Technology, Bhubaneswar
11 PUBLICATIONS **40** CITATIONS
SEE PROFILE

Chetna Singhal
Indian Institute of Technology Kharagpur
58 PUBLICATIONS **382** CITATIONS
SEE PROFILE

DeSVQ: Deep Learning Based Streaming Video QoE Estimation

Monalisa Ghosh
monalisa11@iitkgp.ac.in
IIT Kharagpur, India

Rushikesh Wayal
wayalhrushi@gmail.com
IIT Kharagpur, India

Chetna Singhal
chetna@ece.iitkgp.ac.in
IIT Kharagpur, India

ABSTRACT

The quality-of-experience (QoE) is a notable subjective quality metric to assess the efficiency of multimedia streaming services. Video streaming is a segment seeing notable growth in the past decade. It is affected by a mixed interplay of quality switching, compression, and buffering events that induce distortions that vary with time. Accurate streaming QoE prediction can help in adapting the content transmission to improve end viewers experience. In this work, we propose DeSVQ, a deep learning approach that uses an integrated framework consisting of Convolutional Neural Network (CNN) and Long Short Term Memory (LSTM) networks that combine multiple feature processing stages, each of them effectively capturing the complex dependencies underlying the QoE prediction process. In one stage, the features are extracted per frame from the distorted videos by the CNN that are mapped sequentially to QoE scores by the LSTM network. In another stage, the LSTM network explores the temporal dependencies using the objective (numerical) features. The output from both the stages are linearly combined and fed to the decision trees. A cross-validation framework is used for evaluation. Our proposed model is shown to perform better QoE prediction over the existing approaches.

KEYWORDS

Convolutional Neural Network (CNN), Dynamic Adaptive Streaming over HTTP (DASH), Long Short Term Memory (LSTM), Quality-of-Experience (QoE), Video streaming

ACM Reference Format:
Monalisa Ghosh, Rushikesh Wayal, and Chetna Singhal. 2022. DeSVQ: Deep Learning Based Streaming Video QoE Estimation. In *23rd International Conference on Distributed Computing and Networking (ICDCN 2022), January 4–7, 2022, Delhi, AA, India.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3491003.3491023

1 INTRODUCTION

In recent years, there has been a global surge in usage of multimedia streaming applications. The key dominating players in the current global market for such applications include Akamai Technologies, Apple Inc, Netflix, Amazon Web services, and Hulu. Rapid advancement of mobile networks and extensive use of smartphones and smart portable devices with higher processing power provides smooth multimedia content access to the users [15]. Mobile data traffic has increased tremendously and is still expected to increase manifolds in the coming years. Cisco [7] forecasts, by 2022, videos will occupy 82% and 79% of total internet and mobile traffic, respectively. The reports in [24] predict that 90% of net 5G traffic will be videos. Although the network capacity has enlarged, still it falls short of streaming greater content and meeting the video quality demands. The dynamic network conditions adversely affect the streaming quality; thus, leading to distortion.

The end users perceptual expectations are continuously rising in hope of better quality. In [16], a survey investigated Quality-of-Experience (QoE) to be the most preferable user option for video delivery services over other categories like nature of content, ease-of-use, timing, portability, and sharing. Studies in [19] reports huge revenue losses to delivery networks due to poor streaming quality. Therefore, network operators aim to provide better streaming quality and optimize the viewing experience.

Dynamic Adaptive Streaming over HTTP (DASH), HTTP Live Streaming (HLS), HTTP Dynamic Streaming (HDS), and Microsoft Smooth Streaming (MSS) enable video rate adaptation by offering video segments in different bitrates [8]. In turn, this provides quality variations. There exists a manifest file at the server that stores each segment's media information that is forwarded to the client. The client adaptively selects the appropriate streams based on conditions such as buffer occupancy, playback rate, and instantaneous throughput of the network.

In streaming session, the QoE has to be evaluated continuously. The evaluation is challenging as there exists non-linearity among several factors influencing QoE, such as, video quality, frequent bitrate adaptation, and rebuffering events. Apart from spatial distortions, streaming video quality contains complex temporal dependencies. The unpredictable network conditions lead to buffer depletion due to delayed arrival of the video packets, resulting in stalling/rebuffering events. These events affect the viewing experience and users tend to quit the streaming session below a certain tolerance level. DASH relies on rate-adaptation strategy that causes the video quality to continually vary with time.

In this paper, we have proposed a combined learning framework consisting of Convolutional Neural Network (CNN) and Long Short Term Memory (LSTM) networks. The CNN is very effective in extracting the high-level spatio-temporal features of the distorted (obtained after streaming) videos. LSTM network traps the non-linearities and temporal reliance involved in the QoE variation. We have taken a multi-stage feature processing approach, wherein, the high-level spatio-temporal features are processed in the first stage and the low-level features explored by the Video Quality Assessment (VQA) metrics are processed in the second stage. The output of both the stages are combined by a linear layer, which in turn is fed to the decision trees. The complete framework cumulatively maps the features from the videos to continuous quality scores.

The main contributions of our work are summarized as follows-

ICDCN 2022, January 4–7, 2022, Delhi, AA, India

ACM ISBN 978-1-4503-9560-1/22/01…$15.00
https://doi.org/10.1145/3491003.3491023

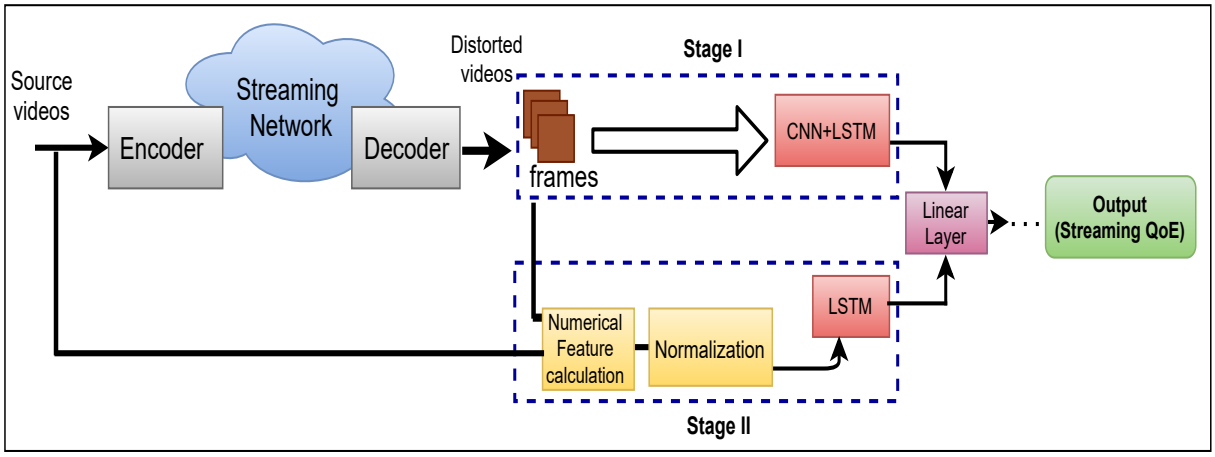

Fig. 1: Flowchart of proposed framework for predicting streaming QoE

(i) We have modelled an integrated CNN with LSTM framework that performs multi-stage processing of features for predicting streaming video QoE.
(ii) In one stage, we extract the high level spatio-temporal features of the streamed videos using the CNN that is sequentially fed to the LSTM, whereas, in other stage, the low-level features explored by VQA metrics are processed by the LSTM network.
(iii) Our integrated framework outperforms their individual counterparts by 2.3-10% and 0.7-6.45% in terms of accuracy on the experimented datasets.

Rest of the paper is structured as follows. Section 2 reviews the related works. Section 3 discusses our proposed learning framework, DeSVQ, for estimating the time-varying QoE. Section 4 presents the experimental settings and performance evaluation of our proposed DeSVQ framework. Section 5 concludes the paper.

2 RELATED WORKS

With the evolving multimedia streaming applications, a lot of efforts is directed towards accessing and improving the viewing experience. Goring *et al.* [17] developed a pixel based No-Reference (NR) to Full-Reference (FR) as well as hybrid video quality assessment models that include client obtainable meta-data. The hybrid models still uses basic details of the distorted video e.g., video codec, bitrate, framerate, and resolution. Robitza *et al.* [25] estimated the streaming quality under the influence of loading time, stalling, and customer engagement using ITU-T P.1203 model. For P.1203, Model 0 was used that requires codec, bitrate, resolution, and framerate information. Its application is restricted due to the use of distinct codec and simple inputs. In [26], the authors first deduce the formation of basic video quality dimensions to frame the entire quality from a new method, called Direct Scaling. Then QoE prediction model is formulated based on the obtained quality dimensions. However, they have not considered the streamed videos. These discussed works predict only the overall video quality.

Studies in [20, 21, 28, 30] analyze the factors affecting user experience in a streaming session and observe that when the video quality is not upto the acceptable level, users tend to quit. [30] estimate the average viewing percentage using public details, such as video subject matter, context, and channel, without considering any user respond. [20] predict the percentage of viewers watching a particular video with respect to time. They analyze and identify user disengagement due to several quality related components, low coding quality, and stalls.

Studies in [31], [5] depict the increasing attraction in implementation of Deep Neural Network (DNN) for video quality estimation. Zadtootaghaj *et al.* [31] estimate the overall quality of gaming and non-gaming streaming videos using a deep learning (DL) approach. They used CNN (DenseNet) by training it on an objective metric inorder to acquire the video artifacts. Then they pool the temporal information (TI) of the videos and frame-level estimations using Random Forest (RF) and predict the quality. In [5], two no-reference (NR) machine learning (ML) based approach using neural network (NN) and support vector Regression (SVR) are taken for estimating the quality of gaming video streaming. It is limited by use of basic feature representations i.e., bitrate, temporal information, and resolution. Lekharu *et al.* [22] have used a Deep Neural Network based model which picks the suitable bit-rate that can lead to overall QoE maximization of a user.

Works in [2, 6, 9–11, 13] show that continuous-time QoE estimation has received a lot of attention. In [6], the authors used Hammerstein Weiner (HW) model, but had limitations in evaluating only the rate adaptive videos. Ghadiyaram *et al.* [13] also used muti-stage and multi-learner HW model that considers association between stalling occurrences. [11] modelled their framework using support vector machine (SVM) in playback and exponential model in rebuffering state. Bampis *et al.* [2] proposed a kind of recurrent neural network models using video quality assessment metrics, rebuffering and memory related inputs. While Eswara *et al.* [10] used only LSTM network and Duc *et al.* [9] used only CNN using almost the same set of inputs as in [2]. In contrast to the aforementioned works, our work uses different set of inputs (numerical features) and mainly focuses on the feature extraction from the distorted video frames using an integrated framework consisting of CNN and LSTM networks.

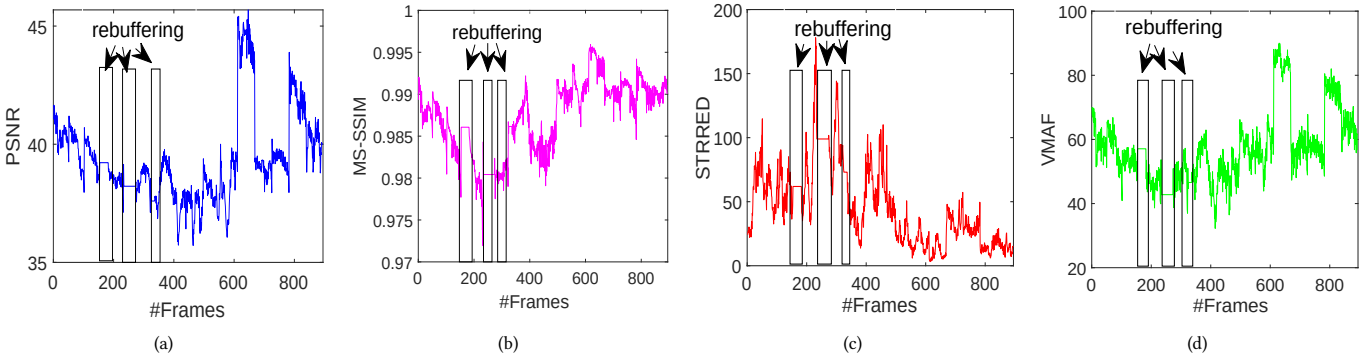

Fig. 2: Variation of different numerical features (a) PSNR (b) MS-SSIM (c) STRRED and (d) VMAF with respect to frame index for the distorted video sample shown in Fig. 3(a) from LIVE NFLX II

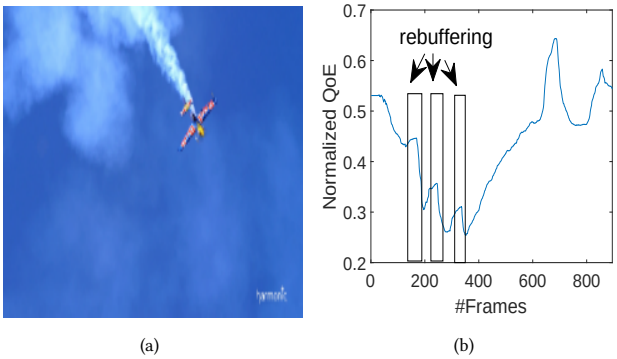

Fig. 3: (a) Snapshot of the distorted video sample from LIVE NFLX II (b) Video quality variation for the distorted video shown in Fig. 3(a)

3 PROPOSED LEARNING FRAMEWORK FOR TIME VARYING QOE: DeSVQ

In this section, we discuss the details of the input features and our proposed QoE prediction architecture, DeSVQ.

We have designed a Deep Learning (DL) framework that accurately predicts the continuous QoE of the streaming videos. Fig. 1 contains flowchart depicting the goal of our complete work. An encoder is used for encoding the source videos. The source videos cover a variety of contents. The source videos, on passing through the streaming networks get distorted. The distorted videos are obtained at the output of the decoder. In our work, we have obtained the distorted videos from the publicly accessible datasets, discussed in Section 4.1. We have used the frames from these distorted videos to compute the numerical features that are normalized and given as input in second stage to our DL framework. We have divided our DL framework into two stages. Each stage is concerned with its specific feature processing work. Stage I contains CNN and LSTM network whose output is linearly combined with the output of another LSTM network from Stage II. The challenges in applying DL is the requirement of large amount of data. In our DeSVQ framework, CNNs extract the features directly from the distorted videos whereas manual (numerical) features are more resilient to overfitting when there is not so much data. So in a way our dual structure allows balancing this aspect. We have discussed our DL architecture in details in Section 3.3. Finally, we obtain the predicted streaming video quality on a per-frame basis.

3.1 Video quality assessment

As per the European Union (EU) Qualinet Community [18], QoE is defined as *the degree of delight or annoyance of the user of an application or service.* Video quality serves to a degree as a proxy of QoE. The video quality scores (QoE) facilitate to understand the deterioration in quality compared to the original video. Our evaluation focuses on video quality estimation which is an important determinant of overall video experience. The subjective scores obtained by subjective quality assessment method are provided in terms of Mean Opinion Scores /Difference-MOS (DMOS)/Z-scores. In LIVE NFLX II[3], the video quality scores are given as *z-scores.*

In the process of obtaining subjective scores, any observer may have preferences for particular reference videos. This aspect is noted by computing the difference scores between distorted video and its reference video. Difference scores, $S_{jkl} = r_{jk_{ref}l} - r_{jkl}$, where r_{jkl} is the score given by observer j to test video k in session l and k_{ref} is the corresponding reference video. The Z-scores per session, z_{jkl} is computed from the difference score. Provided that the z-scores are in the range of [0,1], then the different quality levels indicating the viewing experience can be classified into different categories from bad to excellent, as given in Table 1. The corresponding MOS are also listed there. $z_{jkl} = (S_{jkl} - \mu_{jl})/\sigma_{jl}$ where $\mu_{jl} = \frac{1}{T_{jl}} \sum_{k=1}^{T_{jl}} S_{jkl}$ and $\sigma_{jl} = \sqrt{\frac{1}{T_{jl}-1} \sum_{k=1}^{T_{jl}} (S_{jkl} - \mu_{jl})^2}$. T_{jl} denotes the number of distorted videos perceived by observer j in l^{th} session . Each distorted video from the database is seen by the observer in only one of the sessions. For all distorted videos, T available in the database, a matrix $\{z_{jk}\}$ is computed combining Z-scores obtained from sessions undertaken where k varies from 1 to T. $\{z_{jk}\}$ depicts Z-score given by j^{th} observer to k^{th} test video.

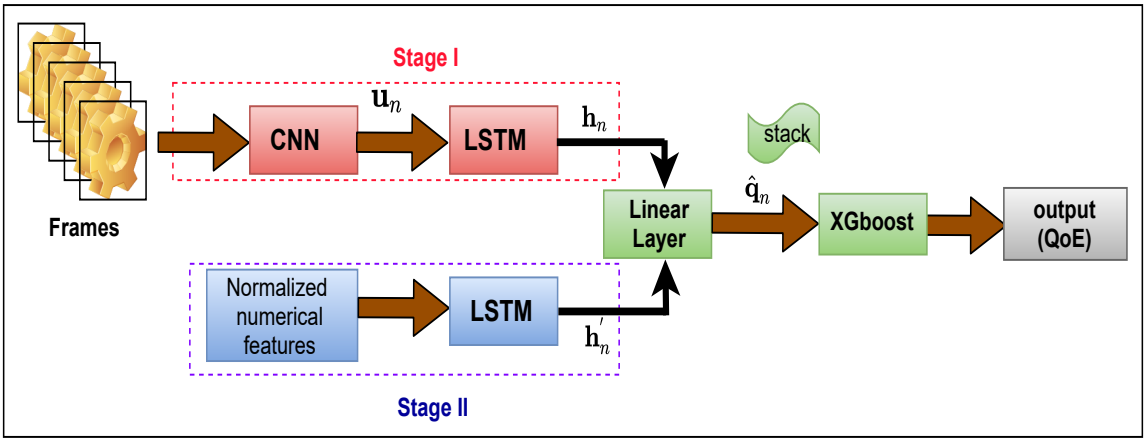

Fig. 4: Proposed DeSVQ architecture for QoE prediction

Table 1: The video quality (Z-scores) corresponding to different quality levels

MOS	Z-scores range	Quality level (QoE)
1	0	Bad
2	(0.0-0.25]	Poor
3	(0.25-0.5]	Fair
4	(0.5-0.75]	Good
5	(0.75-1.0]	Excellent

Table 2: Correlation between Objective metrics and z-scores computed per-frame basis

Objective metrics	SROCC
PSNR	0.5247
MS-SSIM	0.6702
STRRED	-0.5495
VMAF	0.6038

3.2 Input Features

In the second stage, we have used numerical features (objective metrics) as input to the LSTM network. The objective VQA metrics are meant to derive the distinctive features which can represent the spatiotemporal regularities of the videos that are distorted. We have used selective numerical features, i.e., Peak-Signal-to-Noise-Ratio (PSNR), Multi-Scale Structural Similarity Index MS-SSIM [29], Spatio-Temporal Reduced Reference Entropic Differencing (STRRED) [27], and Video Multi-method Assessment Fusion (VMAF) [23] as input to the LSTM network (in the second stage) that is very effective in modelling the temporal dependencies. Fig. 2 shows the variation of numerical features (i.e., PSNR, MS-SSIM, STRRED and VMAF) of the distorted video sample, shown in Fig. 3(a) from LIVE NFLX II [4]. The numerical features are in different ranges (shown in Fig. 2). So, we have normalized the numerical features in order to bring them to a common range before they are provided as input to the LSTM network in the second stage. It can be seen (from Fig. 2) that the numerical feature variations are somewhat correlated with the video quality variations shown in Fig. 3(b). In Table 2 the correlation coefficients are calculated per frame between the numerical feature and z-score for the video sample in Fig. 3(a). The normalized QoE score in Fig. 3(b) is obtained from the z-scores. In Table 2, MS-SSIM achieves highest correlation, while STRRED shows a negative correlation coefficient. Incase of rebuffered frames, we have repeated the same value of VQA metric till the last rebuffered frame index, until the next playback frame. However, it is known that the performance of VQA metrics varies with regards to the distortion type/category. This is because the features represented by VQA metrics are yet low-level.

Due to the recognized learning ability of the deep networks, in particular CNN, it becomes possible to analyze the high-level spatio-temporal features from the distorted videos. So, in the first stage, we have used distorted video frames as input to the CNN.

3.3 Learning Framework, DeSVQ architecture

We have developed a learning framework, DeSVQ. Its architecture is shown in Fig. 4 that consists of frame-level CNN (specifically VGG16) sequence features fed to the LSTM network in the first stage. In the second stage, the numerical features represented by the VQA metrics are processed by another LSTM network. The numerical representative features carry their own significance owing to the correlation they possess with the QoE scores, as shown in Table 2. A linear layer is used to combine the predicted output from both the stages.

We denote the distorted video samples affected by streaming artifacts as $\mathbf{V}$. Let $\mathbf{V}$ has m frames i.e., $\mathbf{V} = \{v_1, v_2, v_3, \ldots, v_m\}$. These m frames are passed via a stack of convolutional layers, which consists of filters that stride through the frames horizontally and vertically to perform the convolutional products. Some convolutional layers are followed by max-pooling layers that finds the maximum amidst the output of the filters. Finally, three fully connected layers are followed by a softmax layer. We denote the output of CNN as

$$\mathbf{u}_n = conv(\mathbf{V}) \tag{1}$$

where, $conv(.)$ is used to represent entire process of the CNN. The following network in the first stage is the LSTM network that is beneficial in mapping the obtained CNN output, $\mathbf{u}_n$ in a sequential manner by involving memory units. In brief, LSTM network contains multiple LSTM units in different layers, where, each unit is composed of the following- input gate ($\mathbf{i}_n$), forget gate ($\mathbf{f}_n$), memory cell ($\mathbf{c}_n$), output gate ($\mathbf{o}_n$), and hidden state ($\mathbf{h}_n$).

$$\mathbf{i}_n = \sigma(\mathbf{W}_{iu}\mathbf{u}_n + \mathbf{W}_{ih}\mathbf{h}_{n-1} + \mathbf{W}_{ic}\mathbf{c}_{n-1} + \mathbf{b}_i) \quad (2)$$

$$\mathbf{f}_n = \sigma(\mathbf{W}_{fu}\mathbf{u}_n + \mathbf{W}_{fh}\mathbf{h}_{n-1} + \mathbf{W}_{fc}\mathbf{c}_{n-1} + \mathbf{b}_f) \quad (3)$$

$$\mathbf{c}_n = \mathbf{f}_t \odot \mathbf{c}_{n-1} + \mathbf{i}_n \odot \phi(\mathbf{W}_{cu}\mathbf{u}_n + \mathbf{W}_{ch}\mathbf{h}_{n-1} + \mathbf{b}_c) \quad (4)$$

$$\mathbf{o}_n = \sigma(\mathbf{W}_{ou}\mathbf{u}_n + \mathbf{W}_{oh}\mathbf{h}_{n-1} + \mathbf{W}_{oc}\mathbf{c}_n + b_o) \quad (5)$$

$$\mathbf{h}_n = \mathbf{o}_n \odot \phi(\mathbf{c}_n) \quad (6)$$

where, n, $\mathbf{W}$, $\mathbf{b}$, $\sigma(.)$, $\phi(.)$ and $\odot$ represents the frame instant, weight matrices, biases, sigmoid function, activation function, and element-wise operation respectively. The updation of each LSTM unit is given as:

$$\mathbf{h}_n = LSTM(\mathbf{h}_{n-1}, \mathbf{u}_n, \boldsymbol{\theta}) \quad (7)$$

where, $LSTM(.)$ consists of equations (2)-(6) and $\boldsymbol{\theta}$ contains all the LSTM network parameters.

The LSTM attached to CNN part (in stage I) is responsible for finding relations between different video frames and forcing CNN to learn features via back-propagation, while the one with numerical features (in stage II) also finds the temporal relation between the frames, but in a restricted way covering only the numerical features which are provided as input.

The updation of each LSTM unit in the second stage is given as:

$$\mathbf{h}'_n = LSTM(\mathbf{h'}_{n-1}, \mathbf{x}_n, \boldsymbol{\theta}') \quad (8)$$

where, $\mathbf{x}_n \in \mathbb{R}^n$ is the set of numerical features, and $\boldsymbol{\theta}'$ contains all the parameters of LSTM network in the second stage.

The resulting output from both the stages are combined by a linear layer given as:

$$\hat{\mathbf{q}}_n = \mathbf{h}_n + \mathbf{h}'_n \quad (9)$$

We observed that the predicted output from the linear layer was not able to cope up well with the deviations present in the target video quality scores. In order to map to the high variance of target output scores, we use decision tree based eXtreme Gradient Boosting (XGBoost) algorithm as a stacking regressor. The XGBoost algorithm contains several decision trees, where each tree is formed by applying gradient descent method, and optimization is carried out by minimizing the loss (our objective) function between actual and estimated QoE at each frame index. Thus, we obtain the predicted continuous streaming video quality.

4 EXPERIMENTAL SETTINGS AND RESULTS

In this section, we discuss the databases used for simulations, learning model settings, procedure and experimental results.

4.1 Database Information

To evaluate the performance of our DeSVQ architecture, we validated the model on different datasets. We have experimented on three publicly accessible databases i.e., LIVE Netflix I [4], LIVE NFLX II [3], and Mobile stall II [14]. There are 14 reference and 112 distorted videos with QoE scores collected from 55 viewers in LIVE Netflix I, where the videos have frame rate of 24, 25 and 30 fps and resolution of 1920p. The videos are H.264 compressed, contain rebuffering events, including combinations of both. The LIVE NFLX II has 15 HD source and 140 distorted videos of 1090p resolution. It includes several design aspects, like, bitrate adaptation, video content and network conditions. The Mobile stall II contains 174 streaming videos collected on mobile gadgets that includes 26 various stalling patterns. Patterns include time period, location and frequency of stall happenings.

Table 3: Performance comparison of our proposed DeSVQ model on LIVE Netflix I with existing QoE models

QoE Models	LCC	SROCC	RMSE
Bampis model [2]	0.6741	0.5354	0.943
Eswara's model [10]	0.8085	0.7187	0.759
Duc's model [9]	0.8526	0.7680	0.486
Proposed DeSVQ	**0.8935**	**0.8908**	**0.327**

Table 4: Performance comparison of our proposed DeSVQ model on LIVE NFLX II database with existing QoE models

QoE Models	LCC	SROCC	RMSE
Bampis model [2]	0.7367	0.6783	0.789
Eswara's model [10]	0.8276	0.8087	0.645
Duc's model [9]	0.8355	0.8183	0.534
Proposed DeSVQ	**0.8894**	**0.8867**	**0.363**

Table 5: Performance comparison of our proposed DeSVQ model on Mobile Stall II with existing QoE models

QoE Models	LCC	SROCC	RMSE
Bampis model [2]	0.7668	0.7443	0.907
Eswara's model [10]	0.8783	0.8607	0.682
Duc's model [9]	0.8927	0.8864	0.427
Proposed DeSVQ	**0.8988**	**0.8936**	**0.352**

4.2 Learning Model Settings

The video samples in the dataset are available in .mp4 (in LIVE NFLX II database) and .yuv (in LIVE Netflix and Mobile Stall II databases) format. We transcode the .yuv video samples using ffmpeg [12]. Then, we have dissected the video samples frame by frame basis. In the first stage, these dissected frames are fed to the CNN. To make training simpler and reduce the number of parameters to tune, we have used transfer learning with VGG16 as a pre-trained model. The last layer of VGG16 was freed for training and rest layers were frozen to their initial weights. We pass current frame along with the past 15 frames to the CNN network. In Stage I, the LSTM network contains 4 hidden layers and 24 units in each layer. The output of CNN forms a sequence of length 16 as input for the LSTM network. In Stage II, the LSTM network contains 4 hidden layers and 16 units in each layer. In the second stage, the numerical features are fed as a sequence of length 16 to the LSTM network. The outputs of both stages is combined by a linear layer. The framework is also stacked with decision based XGBoost Algorithm to increase

(a) (b)

Fig. 5: Snapshot of the video samples from (a) LIVE Netflix I and (b) LIVE NFLX II

prediction power. Here, the selection of these model parameters (i. e., number of layers and units) are determined experimentally. Using the inputs (discussed in Section 3.2), the number of layers and units were varied over a range of values and it was noticed that the model did not perform significantly with further increment in model parameters.

4.3 Learning Procedure

We have used PyTorch (version 1.7.0) implementation of neural networks to make use of numerical features available as .mat files and for modeling the DL framework. The dataset is split into 80:20 ratio; and the train and validation indices are scrambled by using PyTorch's sampler. Custom function and dataset loaders are used to extract sequences of 16 frames (including current frame) and 16 values of each numerical feature. The model is trained on the 80 percent split and validated on the 20 percent split. This is further carried out in a cross validation way to validate on whole dataset. In the cross validation method, the validation is made on one of the folds, while training on the rest folds. The parameters such as learning rate, loss function etc are selected by performing grid search where each type of combination is performed to select the best performing one's. The cross validated values are obtained from models with different parameters. These values are then stacked into a pandas dataframe and the XGBoost is used as a stacking model to map the values to match the large variance. The parameters for the XGBoost are also trained by using grid search cv. For testing with XGBoost, the model is trained on all data points except one data point. Prediction is recorded on that one data point and this procedure is repeated for all data points.

4.4 Results

We report the performance measures of our proposed DeSVQ framework using the following three metrics [1]-(i) Spearman Rank Order Correlation Coefficient (SROCC) (ii) Linear Correlation Coefficient (LCC), and (iii) Root Mean Square Error (RMSE). Our framework is evaluated on the datasets described in Section 4.1. We set the model parameters as discussed in Section 4.2. We train and validate the model following the procedure mentioned in Section 4.3. We obtain results on the databases using our DeSVQ framework along with comparison with up-to-date methods.

The results obtained on LIVE Netflix I are shown in Table 3. Our DL model achieves SROCC and LCC of 0.8935 and 0.8908, respectively. The achieved LCC value is 10%, and 4.8% relatively

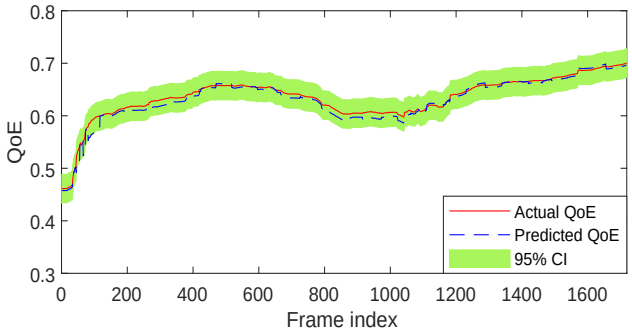

Fig. 6: Best DeSVQ prediction result on test video sample from LIVE Netflix I for the corresponding source video in Fig.5(a). Here, CI represents the confidence interval.

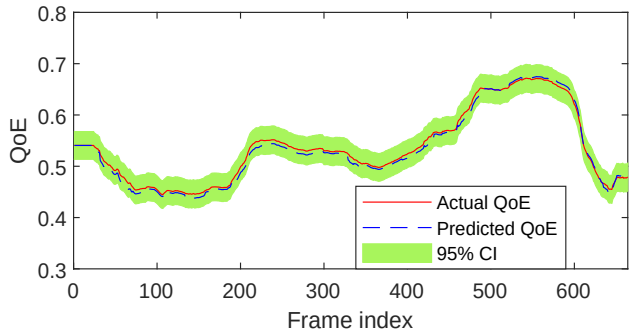

Fig. 7: Best DeSVQ prediction result on test video sample from LIVE NFLX II for the corresponding source video in Fig. 5(b)

higher than the Eswara and Duc's model. Table 4 presents the results on LIVE NFLX II. Particularly, our proposed DeSVQ beats Bampis, Eswara and Duc's model on LCC by 20%, 7.4%, and 6.45% , respectively; and on SROCC by 30%, 9.6%, and 8.35% , respectively. Table 5 shows that our model attains LCC, SROCC and RMSE values of 0.8988, 0.8936, and 0.352, respectively. Here also, we see a relative gain of 0.7%, 0.81% interms of LCC and SROCC with respect to Duc's model. From the experimental results, we note that, our proposed DeSVQ performs better than the existing models. This maybe due to the fact that Bampis's model [2] uses only recurrent neural networks that are not very efficient in capturing the spatio-temporal artifacts. Eswara's model [10] uses only LSTM networks and Due's model [9] uses only CNN for predicting the video quality. The combined use of CNN and LSTM in our framework is very effective in capturing the spatio-temporal artifacts.

The above experimental results obtained on the databases show that the performance of our DeSVQ framework exceeds the up-to-date methods. Fig. 6 and 7 depict the best prediction obtained on the validation set for a video sample from LIVE Netflix I, and LIVE NFLX II, respectively for the corresponding source video samples in Fig. 5(a) and 5(b), respectively. Also, we can see that our predicted QoE value lies within 95% CI of the actual QoE. Pictorially, we can

see that our predicted QoE values for test video samples are able to track the actual QoE values very well.

5 CONCLUSIONS

In this paper, DeSVQ-an integrated DL (CNN and LSTM) framework is proposed for continuous streaming video quality prediction. The model undertakes multiple feature processing stages and demonstrates its superior performance in terms of prediction accuracy. DeSVQ attains highest LCC (0.8988), SROCC (0.8936), and least RMSE (0.363) on the Mobile Stall II dataset. Our integrated model achieves relatively 2.3-10% and 0.7-6.45 % LCC improvement with respect to their individual counterparts, Eswara and Duc's model, respectively on validated datasets. The experimental results show the robustness of our DeSVQ model. In future, we aim to improve our predictor model considering the temporal dynamics of network bandwidth and traces. We will be using such predicted QoE for improving the overall streaming system.

REFERENCES

[1] Jochen Antkowiak, TDF Jamal Baina, France Vittorio Baroncini, Noel Chateau, France FranceTelecom, Antonio Claudio França Pessoa, FUB Stephanie Colonnese, Italy Laura Contin, Jorge Caviedes, and France Philips. 2000. Final report from the video quality experts group on the validation of objective models of video quality assessment march 2000. (2000).

[2] C. G Bampis, Zhi Li, I. Katsavounidis, and A. C Bovik. 2018. Recurrent and dynamic models for predicting streaming video quality of experience. *IEEE Transactions on Image Processing* 27, 7 (2018), 3316–3331.

[3] C. G Bampis, Z. Li, I. Katsavounidis, T.-Yuan Huang, and Alan C Ekanadham, C.and Bovik. 2018. Towards perceptually optimized end-to-end adaptive video streaming. *arXiv preprint arXiv:1808.03898* (2018).

[4] C. G. Bampis, Z. Li, A. Krishna Moorthy, I. Katsavounidis, A. Aaron, and A. C. Bovik. 2017. Study of Temporal Effects on Subjective Video Quality of Experience. *IEEE Transactions on Image Processing* 26, 11 (2017), 5217–5231. https://doi.org/10.1109/TIP.2017.2729891

[5] N. Barman, E. Jammeh, S. A. Ghorashi, and M. G. Martini. 2019. No-Reference Video Quality Estimation Based on Machine Learning for Passive Gaming Video Streaming Applications. *IEEE Access* 7 (2019), 74511–74527. https://doi.org/10.1109/ACCESS.2019.2920477

[6] C. Chen, L. K. Choi, G. de Veciana, C. Caramanis, R. W. Heath, and A. C. Bovik. 2014. Modeling the Time—Varying Subjective Quality of HTTP Video Streams With Rate Adaptations. *IEEE Transactions on Image Processing* 23, 5 (2014), 2206–2221. https://doi.org/10.1109/TIP.2014.2312613

[7] Cisco. 2019. *Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2017–2022.*

[8] Z. Duanmu, K. Zeng, Kede M., Abdul R., and Zhou W. 2017. A Quality-of-Experience Index for Streaming Video. *IEEE Journal of Selected Topics in Signal Processing* 11, 1 (2017), 154–166. https://doi.org/10.1109/JSTSP.2016.2608329

[9] T. Nguyen Duc, C. Tran Minh, T. P. Xuan, and E. Kamioka. 2020. Convolutional Neural Networks for Continuous QoE Prediction in Video Streaming Services. *IEEE Access* 8 (2020), 116268–116278. https://doi.org/10.1109/ACCESS.2020.3004125

[10] N. Eswara, S. Ashique, A. Panchbhai, S. Chakraborty, H. P. Sethuram, K. Kuchi, A. Kumar, and S. S. Channappayya. 2020. Streaming Video QoE Modeling and Prediction: A Long Short-Term Memory Approach. *IEEE Transactions on Circuits and Systems for Video Technology* 30, 3 (2020), 661–673. https://doi.org/10.1109/TCSVT.2019.2895223

[11] N. Eswara, A. Manasa, K. anbampisrnnd Kommineni, S. Chakraborty, H. P. Sethuram, K. Kuchi, A. Kumar, and Sumohana S. Channappayya. 2018. A Continuous QoE Evaluation Framework for Video Streaming Over HTTP. *IEEE Transactions on Circuits and Systems for Video Technology* 28, 11 (2018), 3236–3250. https://doi.org/10.1109/TCSVT.2017.2742601

[12] Ffmpeg. [n.d.]. Open-source software. https://www.ffmpeg.org/. Online.

[13] D. Ghadiyaram, J. Pan, and Alan C Bovik. 2018. Learning a continuous-time streaming video QoE model. *IEEE Transactions on Image Processing* 27, 5 (2018), 2257–2271.

[14] D. Ghadiyaram, J. Pan, and Alan C. Bovik. 2019. A Subjective and Objective Study of Stalling Events in Mobile Streaming Videos. *IEEE Transactions on Circuits and Systems for Video Technology* 29, 1 (2019), 183–197. https://doi.org/10.1109/TCSVT.2017.2768542

[15] Monalisa Ghosh, Anubhav, and Chetna Singhal. 2020. DA-TV: Dynamic Adaptive Television Broadcast for Mobile Users. In *2020 International Conference on COMmunication Systems NETworkS (COMSNETS)*. 137–143. https://doi.org/10.1109/COMSNETS48256.2020.9027462

[16] Cisco IBSG Youth Focus Group. [n.d.]. Cisco IBSG youth sur. http://www.cisco.com/c/dam/en_us/about/ac79/docs/ppt/Video_Disruption_SP_Strategies_IBSG.pdf. Online.

[17] S. Göring, R. R. Ramachandra Rao, B. Feiten, and Alexander Raake. 2021. Modular Framework and Instances of Pixel-Based Video Quality Models for UHD-1/4K. *IEEE Access* 9 (2021), 31842–31864. https://doi.org/10.1109/ACCESS.2021.3059932

[18] T. Hoßfeld, M. Hirth, Judith Redi, Filippo M., Pavel K., Babak N., Michael S., Bruno G., Sebastian E., and Christian K. 2014. Best Practices and Recommendations for Crowdsourced QoE-Lessons learned from the Qualinet Task Force" Crowdsourcing". (2014).

[19] Conviva Inc. [n.d.]. Viewer experience report. http://www.conviva.com/conviva-customer-survey-reports/ott-beyond-entertainment-csr/. Online.

[20] P. Lebreton and K. Y. 2020. Predicting user quitting ratio in adaptive bitrate video streaming. *IEEE Transactions on Multimedia* (2020), 1–1. https://doi.org/10.1109/TMM.2020.3044452

[21] P. Lebreton and K. Yamagishi. 2019. Study on user quitting rate for adaptive bitrate video streaming. In *2019 IEEE 21st International Workshop on Multimedia Signal Processing (MMSP)*. IEEE, 1–6.

[22] A. Lekharu, K Y Moulii, A. Sur, and A. Sarkar. 2020. Deep Learning based Prediction Model for Adaptive Video Streaming. In *2020 International Conference on COMmunication Systems NETworkS (COMSNETS)*. 152–159. https://doi.org/10.1109/COMSNETS48256.2020.9027383

[23] Z. Li, A. Aaron, I. Katsavounidis, A. Moorthy, and M. Manohara. 2016. Toward a practical perceptual video quality metric. *The Netflix Tech Blog* 6, 2 (2016).

[24] Jeremy Naydler. 2018. 5G: the final assault. *New View* (2018), 33–9.

[25] W. Robitza, A. M. Dethof, S. Göring, A. Raake, A. Beyer, and Tim Polzehl. 2020. Are You Still Watching? Streaming Video Quality and Engagement Assessment in the Crowd. In *2020 Twelfth International Conference on Quality of Multimedia Experience (QoMEX)*. 1–6. https://doi.org/10.1109/QoMEX48832.2020.9123148

[26] F. Schiffner and S. Moller. 2018. Direct Scaling amp; Quality Prediction for perceptual Video Quality Dimensions. In *2018 Tenth International Conference on Quality of Multimedia Experience (QoMEX)*. 1–3. https://doi.org/10.1109/QoMEX.2018.8463431

[27] R. Soundararajan and A. C Bovik. 2012. Video quality assessment by reduced reference spatio-temporal entropic differencing. *IEEE Transactions on Circuits and Systems for Video Technology* 23, 4 (2012), 684–694.

[28] X. Tan, Y. Guo, M. A Orgun, L. Xue, and Y. Chen. 2018. An Engagement Model Based on User Interest and QoS in Video Streaming Systems. *Wireless Communications and Mobile Computing* 2018 (2018).

[29] Z. Wang, E.P. Simoncelli, and A.C. Bovik. 2003. Multiscale structural similarity for image quality assessment. In *The Thrity-Seventh Asilomar Conference on Signals, Systems Computers, 2003*, Vol. 2. 1398–1402 Vol.2. https://doi.org/10.1109/ACSSC.2003.1292216

[30] S. Wu, M. Rizoiu, and L. Xie. 2018. Beyond views: Measuring and predicting engagement in online videos. In *Twelfth international AAAI conference on web and social media*.

[31] S. Zadtootaghaj, N. Barman, R. R. Ramachandra Rao, S. Göring, Maria G. Martini, A. Raake, and Sebastian Möller. 2020. DEMI: Deep Video Quality Estimation Model using Perceptual Video Quality Dimensions. In *2020 IEEE 22nd International Workshop on Multimedia Signal Processing (MMSP)*. 1–6. https://doi.org/10.1109/MMSP48831.2020.9287080