Towards the Application of Multi-view Learning in Quality of Experience Collaborative Modelling

MohammadAli Hamidi, Simone Porcu, Alessandro Floris, and Luigi Atzori
DIEE, University of Cagliari, 09123 Cagliari, Italy
CNIT, University of Cagliari, 09123 Cagliari, Italy
{mohammadali.hamidi, simone.porcu, alessandro.floris84, l.atzori}@unica.it

***Abstract*—Multi-view (MV) learning is a machine learning technique for improving generalization efficiency by learning from different feature subsets derived from multiple sources. We believe this approach can help in Quality of Experience (QoE) modelling by integrating knowledge from different datasets generated by subjective tests conducted for the same or similar applications considering different QoE Influence Factors (IFs). To investigate this subject, in this paper, we present the experiments conducted starting from a complete dataset related to Web browsing sessions that has been artificially divided into two distinct subsets (views). The proposed MV learning approach implements a data fusion technique to integrate extracted features from different views into a unified feature space. To achieve a complete experiment on the entire problem space, all possible combinations of IFs (features) in two distinct partial views (PVs) are considered and trained in the MV approach; the full view (FV) approach, which utilizes the complete dataset, is also considered for performance comparison. Experimental results show the QoE estimation performance achieved by the MV (0.69) is comparable with that of the FV (0.72), although the 2 single views were used for training in the MV case. Moreover, the performance enhancement achieved by the MV compared with the PV is most noticeable when a lower number of features is used to train the models.**

***Index Terms*—Multi-view learning, Data fusion, Deep learning, Quality of experience, Subjective dataset.**

I. Introduction

In today's digital landscape, user-perceived quality plays a crucial role in determining the success of Web-based services. This is often measured with metrics of Quality of Experience (QoE), which refers to the quality perceived by the users of multimedia services and that is defined as *the degree of delight or annoyance of the user of an application or service* [1]. Objective QoE models adopted by Internet Service Providers (ISPs) and Over-the-top (OTT) application providers are primarily utilized to estimate the quality as a function of measurable network- (e.g., network delay, packet loss, throughput) and application-related (e.g., playout buffering, multimedia quality) QoE influence factors (IFs), respectively [2]. These are also used for root cause analysis to find inefficiency and issues and improve overall user satisfaction.

To build the above-mentioned models, there is a need to conduct subjective tests to collect the opinions of human users on several network and service configurations. Notwithstanding the significant efforts that have been devoted to this activity, there is still a strong need for additional subjective test-based experiments (which are costly and time-consuming) as the available datasets do not cover all the scenarios of interest, which also change over time. Additionally, very often a limited number of IFs has been considered in each experiment, making it hard to define a model that is somehow extendable (i.e., that can also be applied to other similar scenarios); it also happens that when the same scenarios have been considered by not synchronized and uncoordinated experiments, the defined parameters are too different so that the generated datasets cannot be merged to build a single model, e.g., these are not usable to train a single Neural Network (NN).

Artificial Intelligence (AI) research tried to find a solution to this problem. One promising approach to integrate the information from different datasets is the Multi-view (MV) learning approach [3], which considers multiple datasets or "views" to predict a precise target. By taking advantage of multiple views, the MV allows for capturing different aspects of the views' data and producing more accurate and reliable predictions, making it a highly promising technique for various applications in machine learning and data analysis. Based on these considerations, we believe that by integrating datasets generated from different subjective tests, an MV approach can produce more robust and efficient QoE predictors. Moreover, MV learning preserves data privacy by integrating the information from the NN hidden layers trained on the separated datasets. Thus, there is no need to share data between diverse entities, but each model can potentially enhance its prediction accuracy by integrating information learned from other predictors trained on different IFs.

In this paper, we investigate the application of MV learning in QoE modelling. In particular, we consider 3 different approaches, namely, full view (FV), partial view (PV), and MV. To evaluate the performance, we have done extensive experiments starting from a single complete dataset related to Web browsing sessions [4]; this has been artificially divided into two distinct subsets (views) to simulate the case of

This work has been partially supported by the European Union under the Italian National Recovery and Resilience Plan (NRRP) of NextGenerationEU, "Sustainable Mobility Center" (Centro Nazionale per la Mobilità Sostenibile, CNMS, CN_00000023), by the NRRP - M4C1 - Inv. 3.4 and M4C1 - Inv. 4.1, Ministerial Decree no. 351/2022, by the PON "Ricerca e Innovazione" 2014-2020 (PON R&I) "Azione IV.4 Dottorati e contratti di ricerca su tematiche dell'innovazione" assigned with D.M. 1062 on 10.08.2021, and by the Italian Ministry of Enterprises and Made in Italy (MIMIT) within the 5G technology support program, on axis 1 "House of Emerging Technologies" (CTE), Project Name "Cagliari Digital Lab" (ID: G27F22000040008).

datasets generated by different entities. The FV approach is implemented with an NN trained with the entire dataset, while for the PV approach, we have trained the same NN with a single view only. Finally, the proposed MV approach implements a data fusion technique to merge extracted features from the two views into a unified feature space. To achieve a complete experiment on the entire problem space, all possible combinations of IFs (features) in two distinct views were considered and trained in the MV and PV approaches. The results show that the MV approach achieves QoE estimation performance comparable with that of the FV, even if only the 2 single views were used for training in the MV case. In particular, the enhancement of QoE estimation accuracy provided by MV compared with PV is most noticeable when a lower number of features is used to train the NN.

The paper is structured as follows. Section II discusses the related work in this area. Section III presents the proposed approach, whereas in Section IV we describe the implementation details. In Section V, we discuss the obtained results, and, finally, Section VI concludes the paper.

II. Related Work

The MV learning approach is utilized for several research activities focused on multimedia processing. In [5], an MV multimodal transformer for image captioning is proposed. The NN structure is composed of two branches, which take the image and the caption text as the input, respectively. The two branches are then merged using a transformer NN that executes the image captioning. The proposed MV approach outperforms the state-of-the-art results. The multi-view multi-class disease classifier developed in [6] utilizes different types of voice-related features (Mel-Frequency-Cepstral-Coefficients (MFCC), Log Mel-filter bank coefficients (logFBANK), and Spectral Subband centroids) in a two-phase multi-class classification module. This approach enhances the model's effectiveness in predicting the presence of disease based on voice samples. In [7], the MV FER (Facial Expression Recognition) approach called OCA-MTL (orthogonal channel attention-based multi-task learning) is presented, which learns view-independent facial expression features using a Siamese CNN (convolutional neural network). The proposed model has two parallel paths for learning facial expression features from both frontal and non-frontal viewpoints. Each NN takes as the input different poses of the face turning the face from $-90°$ to $+90°$. This approach achieved a mean accuracy of 88.4% among 6 different predicted emotions, outperforming the state-of-the-art. The Noise-aware Incomplete Multi-view Learning Networks (NIM-Nets) framework proposed in [8] leverages incomplete data from various views to generate a shared representation that is both consistent and informative, while also being able to handle noise effectively.

The aforementioned studies demonstrate that exploring alternative methods, such as MV learning, that rely on different data views could potentially lead NN-based models to more accurate and efficient predictions. Despite some AI-based solutions focused on the fusion of data representations originating from different datasets have barely been investigated for the definition of QoE models, such as federated learning [9]–[11] and distributed deep learning [12], none of the state-of-the-art research studies has directly investigated the application of MV learning in QoE modelling.

Hence, further research is needed to determine if MV approaches are feasible and effective for QoE prediction and to explore their potential benefits compared to other approaches. Thus, in this paper, we focus on the application of the MV approach in QoE modelling to understand whether it can contribute to an improvement in the estimation performance of QoE models.

III. Proposed approach

The proposed research concerns the application of MV learning in QoE modelling. QoE predictors (or models) are objective models that describe the relationship between QoE IFs and QoE, i.e., they take some monitored QoE IFs as the input and predict the user's perceived QoE as the output. However, different QoE models are often *incompatible* (i.e., they do not work well on datasets dissimilar than those used to define them) because: i) the QoE depends on many IFs of different nature, such as network-based IFs (e.g., network delay, packet loss, throughput) measured by ISPs and application-based QoE IFs (e.g., playout buffering, multimedia quality) collected by OTT application providers [1]; ii) different sets of IFs or different ranges of variation for the same IFs have been considered to build different QoE models. This means that each model achieves the best estimation performance when its inputs are the same IFs (and varying in the same/similar range) used to build the model. The reason is that different IFs and different IFs' levels have a different impact on the user's perceived QoE. Above all, the ground-truth data used to build an objective model is collected through a controlled subjective assessment.

As a practical example, let's consider an ISP and an OTT monitoring the QoE of the same video streaming service. The ISP has its model that estimates the QoE of the video streaming service as a function of the network delay. Likewise, the OTT has its model that estimates the QoE of the video streaming service as a function of the playout buffering. Since the QoE depends on both network- and application-related IFs, these two models do not achieve a good estimation accuracy as they are missing important data for training their models. At the same time, they are not willing to share the collected data. Thus, our research question is: would it be possible to share some kind of *model information* between these two models (those modelling the delay-QoE and playout buffering-QoE relationship) to enhance their QoE estimation performance, i.e., to make them capable of estimating the QoE as a function of both the network delay and playout buffering without the need to share data between ISP and OTT and to conduct additional subjective tests where the two sets of IFs are considered?

The literature offers a multitude of QoE models that can be potentially reused and improved. In particular, we con-

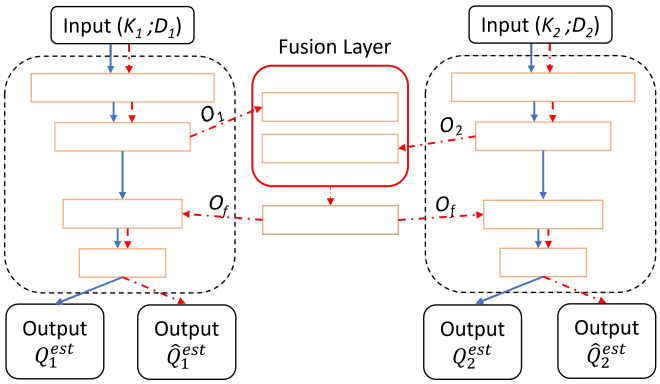

Fig. 1. PV approach: the blue solid lines indicate the $Q_x(K_x; D_x)$ models, whose outputs are Q_x^{est} $(x = 1, 2)$. MV approach: the red dashed lines indicate the $\hat{Q}_x(K_x; D_x, O_f)$ models, which are trained with the support of the fusion layer output O_f, and whose outputs are $\hat{Q}_x^{est}$ $(x = 1, 2)$; O_x is the output of one of the hidden layers of $\hat{Q}_x$ models $(x = 1, 2)$.

sider MV learning techniques to propose an approach that can potentially enable the implementation of QoE estimation models based on data originally collected by different *entities*. With entity, we refer to an organization (e.g., ISP and OTT, a research group) conducting QoE assessment studies and defining QoE models for Web-based multimedia applications. With MV techniques, it is possible to integrate knowledge from multiple datasets or "views" to predict a precise target. By taking advantage of multiple views, MV-based predictors can capture different aspects of the data and produce more accurate and reliable predictions. For these reasons, the MV learning approach fosters the reuse and integration of subjective datasets collected by different entities, aiming to develop enhanced QoE prediction models. Moreover, MV learning preserves data privacy by integrating the information from the NN hidden layers trained on the separated datasets. Thus, there is no need to share data between diverse entities (entities are often unwilling to share collected data), but each QoE model can potentially enhance its prediction performance by integrating information learned from other predictors trained on different views.

We define K_i as the vector encoding all the IFs which are considered by entity i and that are used to predict the user quality through model $Q_i(K_i; D_i)$:

$$Q_i^{est} = Q_i(K_i; D_i); 1 \leq Q_i^{est} \leq 5, \quad (1)$$

where we assume that the quality model has been created to estimate the QoE using the 5-level Absolute Category Rating (ACR) scale [13] and through appropriate training from the dataset D_i. The objective is to build a more accurate quality model $\hat{Q}_i(K_i; D_i, O_f)$ through the integration of *information* embedded in the fusion layer output O_f generated thanks to the Q_j models trained on datasets D_j, for $j = \{1, \ldots, J\}$ and $j \neq i$, where J is the number of entities that collaborate to create a joint model.

Let's consider 2 models $Q_1(K_1; D_1)$ and $Q_2(K_2; D_2)$ depicted in Fig. 1, which estimate the quality for the same service as a function of K_1 and K_2 IFs, respectively, so that:

$$Q_1^{est} = Q_1(K_1; D_1), \quad (2)$$

$$Q_2^{est} = Q_2(K_2; D_2). \quad (3)$$

We aim to enhance the prediction performance of these models by sharing information from their learning function:

$$\hat{Q}_1^{est} = \hat{Q}_1(K_1; D_1, O_f); 1 \leq \hat{Q}_1^{est} \leq 5, \quad (4)$$

$$\hat{Q}_2^{est} = \hat{Q}_2(K_2; D_2, O_f); 1 \leq \hat{Q}_2^{est} \leq 5, \quad (5)$$

where $\hat{Q}_x(K_x; D_x, O_f)$ is the prediction model that predicts from K_x and has been trained with dataset D_x and the fusion layer output O_f, with $x = \{1, 2\}$.

To better illustrate the implementation and the performance assessment of the proposed MV-based prediction model and the alternative ones, herein we summarize the considered different approaches:

1) Partial view (PV): each $Q_i(K_i; D_i)$ model is trained on the dataset D_i, which includes the values of the K_i IFs and the corresponding QoE. This represents the case where each entity builds its model whose output is Q_i^{est}.
2) Multi-View (MV): each $\hat{Q}_x(K_x; D_x, O_f)$ model is trained on the dataset D_x (including the values of the K_x IFs and the corresponding QoE) and with the support of the fusion layer output O_f. This is the proposed model whose output is $\hat{Q}_x^{est}$.
3) Full view (FV): a single QoE model, $Q(K; D)$ is trained on the dataset D, which includes all the $K = \cup_i K_i$ IFs and the corresponding QoE. This is the case where the different entities share the data and is introduced here for comparison purposes (it is very uncommon this happens).

IV. Implementation Details

In Section IV-A, we first present the considered dataset [4], which includes the subjective QoE of Web sessions influenced by 9 different IFs. We then based on Fully Connected Deep Neural Networks (FC-DNNs) to define the QoE models for the 3 approaches, as described in Section IV-B. Finally, Section IV-C presents the implementation of the 3 approaches.

A. Dataset

The used dataset [4] comprises quality ratings collected from 135 users who explicitly rated 3,400 Web browsing sessions (with diverse page sizes, number of objects, and loading times) using the ACR scale ranging from 1 (Bad) to 5 (Excellent). The authors of the dataset calculated the Pearson correlation coefficient (PCC) between the collected features of the Web sessions and the corresponding QoE scores. Among all the features, the nine features that achieved the highest PCC scores (> 0.7) include: (1) the time taken to load the Document Object Model (DOM), (2) the time taken to load the last visible image or other multimedia objects (Approximate Above-The-Fold, AATF), (3) the time taken to trigger the onLoad event (Page Load Time, PLT), (4-5) two ByteIndex (BI) metrics, (6-7) two ObjectIndex (OI) metrics, and (8-9) two ImageIndex (II) metrics. Being these the IFs with the

Fig. 2. The architecture of FC-DNN1. The input dataset X is D_1 for PV_1, D_2 for PV_2, and D for FV.

highest PCC score, they have been selected in our analysis to create the D dataset composed of $d = 9$ elements, i.e., the vectors of selected IFs.

We artificially divided D into 2 different datasets (two views), D_1 and D_2, each including a subset of the 9 IFs, so that $D_1, D_2 \subset D$ with $D_1 \cap D_2 = \emptyset$. The aim is to simulate datasets of IFs measured by two different entities. d_1 and d_2 are the number of elements of D_1 and D_2, respectively, so that $d_1 + d_2 = d$.

B. Neural Networks

We based on FC-DNNs to implement the QoE models for the 3 approaches because their fully connected layers are adept at generating a high-order feature representation, which can be conveniently separated into distinct classes [14]. FC-DNNs consist of layers that are fully connected (dense) between each other, i.e., each neuron in one layer can communicate with the neurons in the next layer. Hence, FC-DNNs can process the data by applying a series of matrix multiplications and non-linear activation functions to learn a complex mapping between the input features and the target variables.

We implemented 2 FC-DNNs: FC-DNN1 and FC-DNN2. The first takes as the input an entire dataset X of x features (QoE IFs) and outputs the predicted QoE score. The FC-DNN1, shown in Fig. 2, is composed of 8 hidden dense layers with 800, 600, 500, 400, 256, 128, 64, and 32 hidden neural units, respectively, followed by the output layer with 5 output neurons activated by the SoftMax activation function. Each hidden dense layer of the FC-DNN1 is activated by the Rectified Linear Unit (ReLU) activation function with an L2 kernel regularization function (regularization factor set to 0.01) to prevent over-fitting and a normalization layer.

Thus, we define the FC-DNN1 as composed of:

$$d_{nu}(.) \coloneqq nl(ReLU(.)), \tag{6}$$

$$\begin{aligned} Res = SoftMax(d_{nu_8}(...(d_{nu_2}(d_{nu_1}(Input))))), \\ \text{with } nu = [800, 600, 500, 400, 256, 128, 64, 32], \end{aligned} \tag{7}$$

where $Input$ is the input vector of features and Res is the prediction result from the SoftMax activation function executed on the recursive application of the function $d_{nu}(.)$ for each neuron unit nu. The normalization layer nl normalizes the activation of the neurons across each batch sample.

The normalization layer function is defined as follows:

$$nl = \frac{ReLU(activations) - \text{mean}(ReLU(activations))}{\sqrt{\text{var}(ReLU(activations))}}. \tag{8}$$

The FC-DNN2, shown in Fig. 3, is a two-branch NN where each branch takes one of the datasets D_1 and D_2. Each input is analysed by 5 hidden dense layers with a different number of neural units, which are activated by the ReLU activation function with an L2 kernel regularization function (regularization factor set to 0.01) to prevent over-fitting and a normalization layer. The number of neural units of each dense layer is defined as: $nu = [800, 600, 500, 400, 256]$. Thus, the application of the sequential dense layers is defined as:

$$O_{bn} = d_{nu_5}(...(d_{nu_2}(d_{nu_1}(Input)))), \tag{9}$$

where $Input$ is the input vector of features, O_{bn} is the result of the application of the sequence of dense layers function d with nu neuron units, and $bn = \{1, 2\}$ is the branch index.

Then, the Fusion layer takes as the input O_1 and O_2 that are fused into a single feature space using a concatenation function that merges the features along the x-axis. The Fusion layer implements the following equation:

$$O_f = du_{256}(O_1 \oplus O_2), \tag{10}$$

where O_f is the output of the application of the dense layer du_{256} with 256 neuron units on the concatenation of O_1 and O_2. The Fusion layer output, O_f, has a size of 256 and is shared back to each last hidden dense layer ($nu = 256$) of the two branches. Therefore, each NN branch follows applying a sequence of d_{nu} functions with $nu = [128, 64, 32]$ defined as:

$$LastO_{bn} = d_{nu_3}(d_{nu_2}(d_{nu_1}(O_f))), \tag{11}$$

where $LastO_{bn}$ is the result of applying the sequence of dense layers function d with nu neuron units, and $bn = \{1, 2\}$ is the branch index. $LastO_1$ and $LastO_2$ are the input of an Output layer composed of a dense layer with 5 neurons activated by a SoftMax activation function.

Finally, each branch output is defined as follows:

$$\hat{Q}_{bn}^{est} = SoftMax(LastO_{bn}), \tag{12}$$

where $\hat{Q}_{bn}^{est}$ is the QoE prediction of each branch and $bn = \{1, 2\}$ is the branch index.

Concerning both the FC-DNNs, the number of hidden layers and neurons was defined based on the experiments to achieve the best estimation performance. We set different numbers of hidden layers and neurons before the fusion layer, but five hidden layers and 256 neurons for the last hidden dense layer provided the most accurate information to find patterns within the two input layers for this particular study. Both the FC-DNNs were trained using the categorical cross-entropy loss function and the Adam optimization method, with the patience value of the early stop function set to 20 and the maximum number of epochs set to 3000. The datasets were divided with a $70\%/30\%$ splitting rate for the training and validation sets, respectively, along with the k-fold cross-validation ($k = 5$).

C. Approaches

We used the dataset described in Section IV-A as the data source D, which includes $d = 9$ IFs. We created two different views of the dataset, D_1 and D_2, each including a subset of

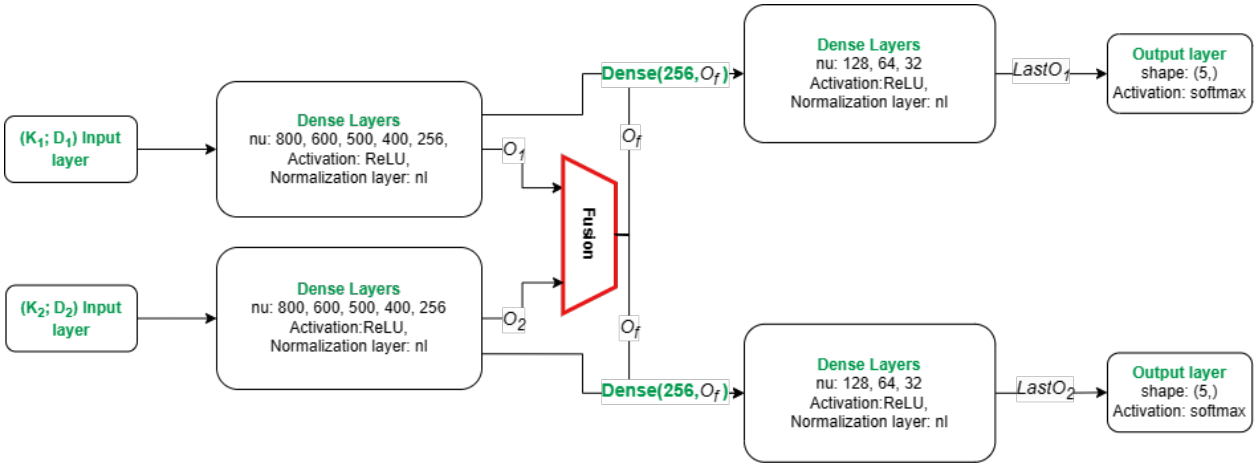

Fig. 3. The architecture of FC-DNN2.

the 9 IFs. The number of combinations of IFs in subsets D_1 and D_2 generated from dataset D of IFs is given by:

$$N_{d_1,d_2} = \frac{d!}{d_1!d_2!}. \tag{13}$$

Thus, the number of feature combinations is the following:

- $d_1 = 8$ and $d_2 = 1$: 9 combinations.
- $d_1 = 7$ and $d_2 = 2$: 36 combinations.
- $d_1 = 6$ and $d_2 = 3$: 84 combinations.
- $d_1 = 5$ and $d_2 = 4$: 126 combinations.

The total number of feature combinations is 255.

The 3 approaches were implemented as follows:

1) PV: for PV_1 the FC-DNN1 was trained with the D_1 dataset including d_1 IFs. For PV_2 the FC-DNN1 was trained with the D_2 dataset including d_2 IFs.
2) MV: the FC-DNN2 was trained with the D_1 and D_2 datasets, which are the input of the 2 branches.
3) FV: the FC-DNN1 was trained with the complete dataset D including all the $d = 9$ IFs.

To achieve a complete experiment on the entire feature space, all possible combinations of IFs (features) have been considered and trained for the PV and MV approaches.

V. Results

In this section, we present the experimental results achieved with the 3 approaches (FV, PV, and MV) for the validation phase in terms of mean accuracy, precision, recall, and F1-score of QoE prediction scores. Table I reports the mean QoE estimation accuracy of the 3 approaches for different combinations and sizes of D_1 and D_2. This means that these accuracy values are the mean of:

- $5\times9 = 45$ values for $d_1 = 8$ and $d_2 = 1$.
- $5\times36 = 180$ values for $d_1 = 7$ and $d_2 = 2$.
- $5\times84 = 420$ values for $d_1 = 6$ and $d_2 = 3$.
- $5\times126 = 630$ values for $d_1 = 5$ and $d_2 = 4$.
- $5\times1 = 5$ values for $d = 9$.

Each number of combinations is multiplied by 5 as we computed a 5-fold training/validation process for each combination. These results show that the FV achieves the greatest accuracy (0.72), i.e., training the single NN (FC-DNN1) with the entire dataset D achieved the most accurate QoE estimation performance. However, the proposed MV approach achieves QoE estimation accuracy results that are slightly lower (0.69 for both MV_1 and MV_2 when $d_1 = 5$ and $d_2 = 4$) but comparable with that achieved by FV. In particular, it must be considered that the MV results are achieved by training the branches of FC-DNN2 with a subset of features than the entire dataset. Moreover, no data is exchanged between the 2 branches, but only information from the NN hidden dense layers is shared with the Fusion layer, which allows for the preservation of the privacy of the 2 data owner entities. The results of Table I also show the enhancement of QoE estimation accuracy provided by the MV approach compared with the PV approach, which is most noticeable when a lower number of features is used to train the NN. For instance, the mean accuracy achieved by PV_2 is increased by 0.47, 0.36, 0.24, and 0.12 for MV_2 when 1, 2, 3, and 4 features were used to train the NN, respectively. This demonstrates that the knowledge integration applied by the Fusion layer of FC-DNN2 enables improved QoE estimation performance.

Table II reports the mean QoE estimation performance of the MV and PV approaches for the combination of features that provided the highest mean QoE estimation accuracy, i.e., $d_1 = 5$ and $d_2 = 4$, with $D_1 = \{IF_2, IF_6, IF_7, IF_8, IF_9\}$ and $D_2 = \{IF_1, IF_3, IF_4, IF_5\}$. The FV performance was computed on the entire dataset, but we also reported these values in this table for comparison with PV and MV approaches. The results show that not only the number of features of the two datasets D_1 and D_2 is important, but also their combination, i.e., how the features are divided between the two datasets can be relevant, although to a lesser extent than the number of features for each dataset. With this specific combination of features for D_1 and D_2, the MV approach achieved overall comparable performance with the FV approach in terms of all

TABLE I
MEAN QOE ESTIMATION ACCURACY OF THE FV, MV, AND PV APPROACHES FOR DIFFERENT COMBINATIONS AND SIZES OF D_1 AND D_2.

Input features	MV_1	MV_2	PV_1	PV_2	FV
$d_1 = 8$, $d_2 = 1$	0.68	0.67	0.69	0.20	-
$d_1 = 7$, $d_2 = 2$	0.68	0.68	0.68	0.32	-
$d_1 = 6$, $d_2 = 3$	0.68	0.68	0.65	0.44	-
$d_1 = 5$, $d_2 = 4$	**0.69**	**0.69**	0.68	0.57	-
$d = 9$	-	-	-	-	**0.72**

TABLE II
MEAN QOE ESTIMATION PERFORMANCE OF THE FV, MV, AND PV APPROACHES FOR THE BEST COMBINATION OF FEATURES WHEN $d_1 = 5$ AND $d_2 = 4$, I.E., $D_1 = \{IF_2, IF_6, IF_7, IF_8, IF_9\}$ AND $D_2 = \{IF_1, IF_3, IF_4, IF_5\}$. M-AVG IS THE MACRO AVERAGE AMONG THE 5 ACR SCORES.

Appr.	Metric	ACR scores					M-AVG
		1	2	3	4	5	
FV	**Mean Acc.**	0.72					
	Precision	0.91	0.76	0.65	0.57	0.67	0.71
	Recall	0.94	0.85	0.66	0.55	0.58	0.72
	F1-Score	0.92	0.80	0.65	0.56	0.62	0.71
PV_1	**Mean Acc.**	0.68					
	Precision	0.89	0.73	0.61	0.55	0.59	0.67
	Recall	0.89	0.78	0.57	0.49	0.65	0.68
	F1-Score	0.89	0.75	0.59	0.52	0.62	0.67
PV_2	**Mean Acc.**	0.57					
	Precision	0.74	0.61	0.53	0.44	0.51	0.57
	Recall	0.80	0.62	0.48	0.42	0.53	0.57
	F1-Score	0.77	0.62	0.50	0.43	0.52	0.57
MV_1	**Mean Acc.**	0.71					
	Precision	0.91	0.76	0.65	0.57	0.64	0.71
	Recall	0.93	0.81	0.65	0.54	0.64	0.71
	F1-Score	0.92	0.78	0.65	0.56	0.64	0.71
MV_2	**Mean Acc.**	0.70					
	Precision	0.91	0.78	0.64	0.57	0.61	0.70
	Recall	0.91	0.83	0.63	0.50	0.65	0.71
	F1-Score	0.91	0.80	0.64	0.53	0.63	0.70

metrics. Table II also shows the QoE estimation performance achieved for the single ACR scores. It can be seen that the models can better estimate lower ACR scores (i.e., 1 and 2) than medium and higher scores. This could be due to an unbalanced dataset reason because the medium to high QoE score values are more frequent and then more difficult to estimate for the QoE models. Moreover, it can be seen that the MV approach enhances QoE estimation performance, making the different performance of PV_1 and PV_2 (PV_1 achieved higher performance than PV_2 because it is trained with one more feature) become comparable between MV_1 and MV_2, as well as with FV.

Finally, Table III reports the mean QoE estimation performance of the MV and PV approaches when $d_1 = 8$ and $d_2 = 1$, with $D_1 = \{IF_1, IF_2, IF_3, IF_5, IF_6, IF_7, IF_8, IF_9\}$ and $D_2 = \{IF_4\}$. We have chosen this particular combination of features because it emphasizes the relevant contribution provided by the MV in enhancing the QoE estimation performance of PV_2 trained with the minimum number of features (just 1), which with MV_2 becomes performance comparable with that obtained by PV_1 (trained with 8 features). These results suggest that the FC-DNN implemented for the MV approach can be extended with more than 2 branches because

TABLE III
MEAN QOE ESTIMATION PERFORMANCE OF THE MV AND PV APPROACHES WHEN $d_1 = 8$ AND $d_2 = 1$, WITH $D_1 = \{IF_1, IF_2, IF_3, IF_5, IF_6, IF_7, IF_8, IF_9\}$ AND $D_2 = \{IF_4\}$. M-AVG IS THE MACRO AVERAGE AMONG THE 5 ACR SCORES.

Appr.	Metric	ACR scores					M-AVG
		1	2	3	4	5	
PV_1	**Mean Acc.**	0.70					
	Precision	0.90	0.74	0.65	0.57	0.61	0.69
	Recall	0.93	0.82	0.61	0.52	0.61	0.70
	F1-Score	0.91	0.78	0.63	0.54	0.61	0.70
PV_2	**Mean Acc.**	0.20					
	Precision	0.04	0.02	0.10	0.13	0.17	0.09
	Recall	0.20	0.09	0.49	0.11	0.11	0.20
	F1-Score	0.07	0.03	0.16	0.04	0.04	0.07
MV_1	**Mean Acc.**	0.71					
	Precision	0.89	0.78	0.61	0.60	0.62	0.70
	Recall	0.94	0.83	0.72	0.42	0.62	0.71
	F1-Score	0.92	0.81	0.66	0.50	0.62	0.70
MV_2	**Mean Acc.**	0.70					
	Precision	0.87	0.78	0.65	0.58	0.59	0.69
	Recall	0.92	0.81	0.63	0.47	0.66	0.70
	F1-Score	0.89	0.79	0.64	0.51	0.63	0.69

even the information provided by a few features (provided by other entities) can be important to achieve an enhanced collaborative model.

VI. CONCLUSION

In this paper, we applied the MV learning approach in QoE collaborative modelling. First, we artificially created two different views of a subjective dataset, which were the input of the FC-DNN2 implementing the MV learning approach. The MV approach achieved a mean accuracy of 0.69, which is comparable to that achieved by the FV approach (0.72), which was implemented by training the FC-DNN1 with the complete dataset. This result shows that the MV approach achieves competitive QoE estimation performance despite only a view of the dataset being used to train the MV_1 and MV_2 models. This demonstrates that the information provided by the fusion layer enhances the estimation performance achieved by the PV_1 and PV_2 models, which do not share any information. Moreover, no feature data is exchanged in the MV approach, but only information from the NN hidden dense layers, which allows for the preservation of the privacy of the 2 entities.

Therefore, this opens the application of the MV learning approach for the creation of novel QoE models based on the integration of different subjective datasets collected by different research groups for the same multimedia service. Future work is needed to further investigate the type of datasets that can be integrated using the proposed solution, including the kind of IFs and range of values. Moreover, alternative data fusion techniques can be considered to enhance the QoE estimation performance achieved by the proposed MV-based approach.

REFERENCES

[1] P. Le Callet, S. Möller, and A. Perkis. (2012) Qualinet White Paper on Definitions of Quality of Experience (2012). European Network on Quality of Experience in Multimedia Systems and Services (COST Action IC 1003), Lausanne, Switzerland, Version 1.2, March 2013.

[2] D. Tsolkas, E. Liotou, N. Passas, and L. Merakos, "A survey on parametric QoE estimation for popular services," *Journal of Network and Computer Applications*, vol. 77, pp. 1–17, 2017.
[3] X. Yan, S. Hu, Y. Mao, Y. Ye, and H. Yu, "Deep multi-view learning methods: A review," *Neurocomputing*, vol. 448, pp. 106–129, 2021.
[4] D. da Hora, A. Asrese, V. Christophides, R. Teixeira, and D. Rossi, "Narrowing the gap between QoS metrics and Web QoE using Above-the-fold metrics," in *Passive and Active Measurement (PAM)*, R. Beverly, G. Smaragdakis, and A. Feldmann, Eds. Springer International Publishing, 2018, pp. 31–43.
[5] J. Yu, J. Li, Z. Yu, and Q. Huang, "Multimodal Transformer With Multi-View Visual Representation for Image Captioning," *IEEE Trans on Circ. and Syst. for Video Tech.*, vol. 30, no. 12, pp. 4467–4480, 2020.
[6] S. Bhattacharjee and W. Xu, "VoiceLens: A multi-view multi-class disease classification model through daily-life speech data," *Smart Health*, vol. 23, p. 100233, 2022.
[7] J. Chen, L. Yang, L. Tan, and R. Xu, "Orthogonal channel attention-based multi-task learning for multi-view facial expression recognition," *Pattern Recognition*, vol. 129, p. 108753, 2022.
[8] Y. Qin, C. Qin, X. Zhang, D. Qi, and G. Feng, "NIM-Nets: Noise-Aware Incomplete Multi-View Learning Networks," *IEEE Transactions on Image Processing*, vol. 32, pp. 175–189, 2023.
[9] S. Ickin, K. Vandikas, and M. Fiedler, "Privacy Preserving QoE Modeling Using Collaborative Learning," in *Proc. of the 4th Internet-QoE Workshop on QoE-Based Analysis and Management of Data Communication Networks*. ACM, 2019, p. 13–18.
[10] Y. Gao, X. Wei, and L. Zhou, "Personalized QoE Improvement for Networking Video Service," *IEEE Journal on Selected Areas in Communications*, vol. 38, no. 10, pp. 2311–2323, 2020.
[11] S. Porcu, A. Floris, and L. Atzori, "CB-FL: Cluster-Based Federated Learning applied to Quality of Experience modelling," in *2022 16th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS)*, 2022, pp. 585–591.
[12] S. Ickin, M. Fiedler, and K. Vandikas, "QoE Modeling on Split Features with Distributed Deep Learning," *Network*, vol. 1, no. 2, pp. 165–190, 2021.
[13] ITU, "Methods for subjective determination of transmission quality." Recommendation ITU-T P.800, 1996.
[14] T. N. Sainath, O. Vinyals, A. Senior, and H. Sak, "Convolutional, Long Short-Term Memory, fully connected Deep Neural Networks," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015, pp. 4580–4584.