LIVELINET: A Multimodal Deep Recurrent Neural Network to Predict Liveliness in Educational Videos

Arjun Sharma
Xerox Research Centre India
Arjun.Sharma@xerox.com

Arijit Biswas
Xerox Research Centre India
Arijit.Biswas@xerox.com

Ankit Gandhi
Xerox Research Centre India
Ankit.Gandhi@xerox.com

Sonal Patil
Xerox Research Centre India
Sonal.Patil@xerox.com

Om Deshmukh
Xerox Research Centre India
Om.Deshmukh@xerox.com

ABSTRACT

Online educational videos have emerged as one of the most popular modes of learning in the recent years. Studies have shown that liveliness is highly correlated to engagement in educational videos. While previous work has focused on feature engineering to estimate liveliness and that too using only the acoustic information, in this paper we propose a technique called LIVELINET that combines audio and visual information to predict liveliness. First, a convolutional neural network is used to predict the visual setup, which in turn identifies the modalities (visual and/or audio) to be used for liveliness prediction. Second, we propose a novel method that uses multimodal deep recurrent neural networks to automatically estimate if an educational video is lively or not. On the StyleX dataset of 450 one-minute long educational video snippets, our approach shows an relative improvement of 7.6% and 1.9% compared to a multimodal baseline and a deep network baseline using only the audio information respectively.

Keywords

Liveliness, Educational Videos, Recurrent Neural Network, Deep Learning, LSTM, Engagement, Multimodal Analysis.

1. INTRODUCTION

The amount of freely available online educational videos has grown significantly over the last decade. Several recent studies [1, 2, 3] have demonstrated that when educational videos are not engaging, students tend to lose interest in the course content. This has led to recent research activity in speaking style analysis of educational videos. Authors in [4] used crowd-sourced descriptors of 100 video clips to identify various speaking-style dimensions such as liveliness, speaking rate, clarity, formality etc. that drive student engagement and demonstrated that liveliness plays the most significant role in video engagement. Using a set of acoustic features and LASSO regression, the authors also developed automatic methods to predict liveliness and speaking rate. The Authors in [5] analyze the prosodic variables in a corpus of eighteen oral presentations made by students of Technical English, all of whom were native speakers of Swedish. They found out that high pitch variation in speech is highly correlated with liveliness. Arsikere et al. [6] built a large scale educational video corpus called StyleX for engagement analysis and provided initial insights into the effect of various speaking-style dimensions on learner engagement. They also found out that liveliness is the most influential dimension in making a video engaging. In this paper, we propose a novel multimodal approach called LIVELINET that uses deep convolutional neural networks and deep recurrent neural networks to automatically identify if an educational video is lively or not.

A learner can typically perceive or judge the liveliness[1] of an educational video both through the visual and the auditory senses. A lecturer usually makes a video lively by using several visual actions such as hand movement, interactions with other objects (board/tablet/slides) and audio actions such as modulating voice intensity, varying speaking rate etc. In the proposed approach, both visual and audio information from an educational video are combined to automatically predict the liveliness of the video. Note that a given lecture can also be perceived as lively based on the contextual information (e.g., a historic anecdote) that the lecturer may intersperse within the technical content. We however don't address this dimension of liveliness in this work [2].

This paper is novel in three important aspects. First, the proposed approach is the first of its kind that combines audio and visual information to predict the liveliness in a video. Second, a convolutional neural network (CNN) is used to estimate the setup (e.g., lecturer sitting, standing, writing on a board etc.) of a video. Third, Long Short Term Memory (LSTM) based recurrent neural networks are trained to classify the liveliness of a video based on audio and visual features. The CNN output determines which of the audio and/or visual LSTM output should be combined for the liveliness prediction.

We observe that there is a lot of variation in what is being displayed in an educational video, e.g., slide/board, lecturer, both slide/board and lecturer, multiple video streams showing lecturer and slide etc.. These different visual setups usually indicate to what degree the audio and the visual information should be combined for predicting liveliness. For example, when the video feed only displays the slide or the board, the visual features do not play a critical role in determining liveliness. However, when the video is focussed on

[1] defined as “full of life and energy/active/animated" in dictionary

[2] Note that the human labelers who provided the ground truth for our database [6] were explicitly asked to ignore this aspect while rating the videos

the lecturer, the hand gestures, body postures, body movements etc. become critical, i.e., the visual component plays a significant role in making a video lively. Hence, we first identify the setup of a video using a CNN based classifier. Next, depending on the setup, we either use both audio and visual information or use only the audio information from a video for training/testing of the LSTM networks. We train two separate LSTM based classifiers, one each for audio and visual modalities, which take a temporal sequence of audio/visual features from a video clip as input and predict if the clip is lively or not. Finally, audio/visual features from a test video clip are forward-propagated through these LSTMs and their outputs are combined to obtain the final liveliness label.

We perform experiments on the StyleX dataset [6], and compare our approach with baselines that are based on visual, audio and combined audio-visual features. The proposed approach shows relative improvement of 7.6% and 1.9% with respect to a multimodal baseline and a deep network baseline using only the audio modality respectively.

2. RELATED WORK

In this section, we discuss the relevant prior art in deep learning and multimodal public speaking analysis in videos.

Deep Learning: Recently deep neural networks have been extensively used in computer vision, natural language processing and speech processing. LSTM [7], a Recurrent Neural Network (RNN) [8] architecture, has been extremely successful in temporal modelling and classification tasks such as handwriting recognition [9], action recognition [10], image and video captioning [11, 12, 13], speech recognition [14, 15] and machine translation [16]. CNNs have also been successfully used in many practical computer vision tasks such as image classification [17], action recognition [18], object detection [19, 20], semantic segmentation [21], object tracking [22] etc.. In this work, we use CNNs for visual setup classification and LSTMs for the temporal modelling of audio/visual features.

Multimodal Public Speaking Analysis: Due to the recent development of advanced sensor technologies, there has been significant progress in the analysis of public speaking scenarios. The proposed methods usually employ use of multiple modalities such as microphone, RGB camera, depth sensor, kinect sensor, Google glasses, body wearables, etc. and analyse the vocal behaviour, body language, attention, eye contact, facial expression of the speakers along with the engagement of the audiences [23, 24, 25, 26]. Gan et al. [23] proposed baseline methods to do the quantification of several above mentioned parameters by analysing the multi-sensor data. Nguyen et al. [24] and Echeverria et al. [25] used kinect sensors to recognize the bodily expressions, body posture, eye contact of the speaker and thereby, providing feedback to the speaker. Chen et al. [26] presented an automatic scoring model by using basic features for the assessment of public speaking skills. It must be noted that all these works rely significantly on the sensor data captured during the presentation for their prediction task and hence, they are not applicable to educational videos that are available online. Moreover, all these approaches use shallow and hand-crafted audio features along with the sensor data. On the contrary, our proposed method uses deep learning based automatic feature extraction method for both audio and visual modalities from the video, and predicts the liveliness.

To the best of authors' knowledge, this is the first approach that uses a deep multimodal approach for educational video analysis.

3. PROPOSED APPROACH

In this section, we describe the details of the proposed approach. We begin with the description of how a given video is modeled as a sequence of temporal events, followed by the visual setup classification algorithm. Next, we provide the details of the audio and visual feature extraction. Finally, the details of the proposed multimodal method for liveliness prediction is described. The pipeline of the proposed approach is shown in Figure 1. The input to the system is a fixed length video segment of 10 seconds during both training and testing (referred to as 10-second clips throughout the paper). For any educational video of arbitrary length, 10-second clips are extracted with 50% overlap between the adjacent clips and the overall video liveliness label is determined based on the majority voting. In Section 5.1 we provide further details regarding extraction of these 10-second clips from the Stylex dataset.

3.1 Video Temporal Sequencing

Each 10-second clip is modeled as a temporal sequence of smaller chunks. If the total number of chunks in a 10-second clip is T, then $\{v_1, v_2, ..., v_t, ..., v_T\}$ and $\{a_1, a_2, ..., a_t, ..., a_T\}$ represent the temporal sequence of visual and audio features corresponding to each 10-second clip respectively. Note that, v_t (Section 3.3) and a_t (Section 3.4) are input to the visual and audio LSTM at time instant t.

3.2 Visual Setup Classification

One of our objectives is to automatically determine if both audio and visual information are required for liveliness prediction. If a video displays only slide/board, the visual features are less likely to contribute to the liveliness. However, if the camera displays that the lecturer is in a sitting/standing posture or is interacting with the content, the visual features could significantly contribute to the video liveliness. Hence, we collect a training dataset and train a CNN to automatically estimate the setup of a video. We describe the definition of the labels, the data collection procedure and the details of the CNN training in the next three subsections.

3.2.1 Video Setup Label Definition

We define five different categories which cover almost all of the visual setups usually found in educational videos.

- **Content:** This category includes the scenarios where the video feed mainly displays the content such as a blackboard or a slide or a paper. Frames, where the hand of the lecturer and/or pens or pointers are also visible, are included in this category. However, the video clips belonging to this category should not include any portion of the lecturer's face. Since the lecturer is not visible in this case, only the audio modality will be used for liveliness prediction.
- **Person Walking/Standing:** In this scenario, the content such as blackboard/slide are not visible. However, the lecturer walks around or remain in a standing posture. The lecturer's face and upper body parts (hand/shoulder) should be visible. Both audio and visual modality are used to predict liveliness in this case.
- **Person Sitting:** The content is not visible and the camera should focus only on the lecturer in a sitting posture. Both audio and visual modalities are considered for liveliness prediction.
- **Content & Person:** This includes all the scenarios where the upper body of the lecturer and the content both are visible. Frames, where the lecturer points to the slide/board or writes something on the board, are included in this category. Here also both the modalities are used for liveliness.

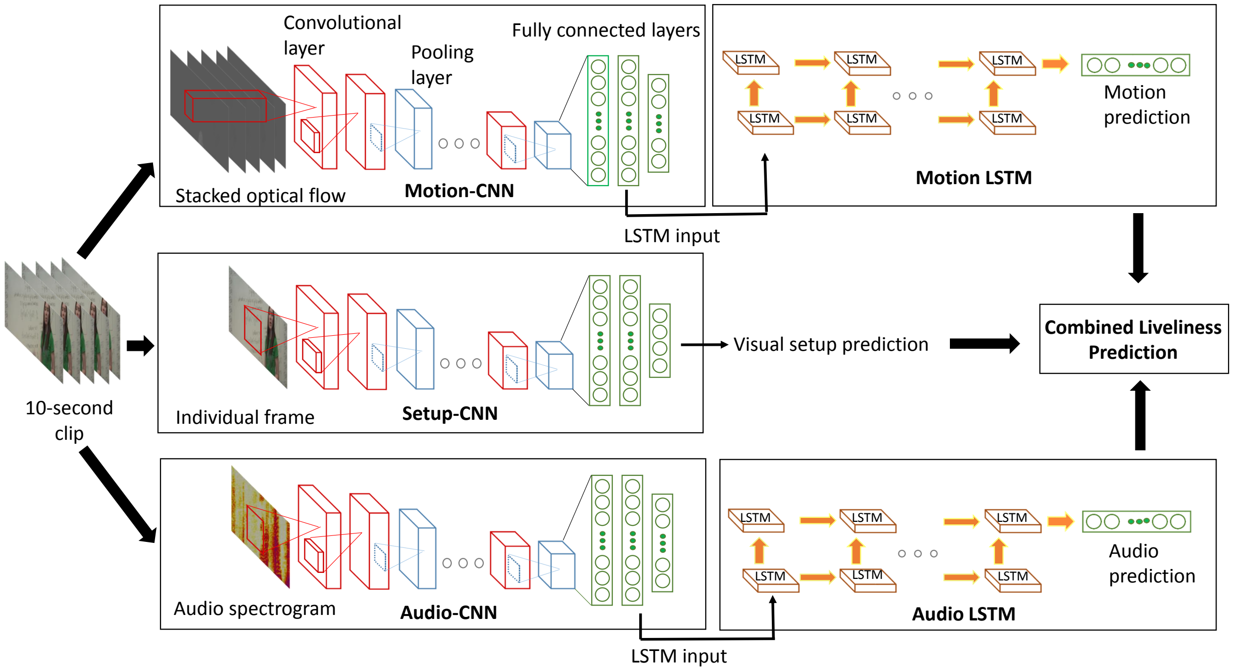

Figure 1: The overall pipeline of the proposed approach LIVELINET. The input to the system is a 10-second clip and output is the liveliness prediction label.

- **Miscellaneous:** This category includes all other scenarios which are not covered in the above four categories, e.g., two different video feeds for professor and content, students are also visible, multiple people (laboratory setups) are visible in the scene etc.. Since the frames from this category have significant intra-class variation and noise, we use only the audio information for liveliness prediction.

Some example frames from the above five categories are shown in Figure 2. The intra-class variation clearly shows the inherent difficulty of the setup classification task.

3.2.2 Label Collection

We used the StyleX dataset [6] for the liveliness prediction task. Although the liveliness labels were available along with the videos, video setup labels were not available. So we collect these additional labels using Amazon Mechanical Turk. We asked the Mturkers to look at the 10-second clips from StyleX and choose one of the five labels defined above. Each video clip is shown to three MTurk labellers and we assign the labels where at least two of the three labellers agreed. Although in most of the clips, all frames belong to only one of the above five categories, there were some 10-second clips (around 5%) where frames from more than one categories were present. In those cases, labellers were asked to provide the label based on the label of the majority of frames.

3.2.3 CNN for Label Classification

We used a CNN architecture to classify the setup of a 10-second clip. During training phase, all the frames belonging to a 10-second clip are used as the samples for the corresponding clip category. For this task, we use the same CNN architecture as used in [17]. In [17], the authors proposed a novel neural network model called Alexnet which improved the state-of-the-art imagenet classification [27] accuracy by a significant margin. Researchers in the computer vision community have often used the Alexnet architecture for other kinds of computer vision applications [28, 29]. Deep neural networks usually have millions of parameters. If the available training data for a particular classification task is not large enough, then training a deep neural network from scratch might lead to over fitting. Hence, it is a common practice to use a CNN which is already pre-trained for a related task and fine-tune only the top few layers of the network for the actual classification task.

We fine-tune the final three fully connected layers (fc6, fc7, fc8) of Alexnet for visual setup classification. First, we remove the 1000 node final layer fc8 (used to classify 1000 classes form imagenet [17]) from the network and add a layer with only five nodes because our objective is to classify each frame into one of the five setup categories. Since, the weights of this layer are learned from scratch we begin with a higher learning rate of 0.01 (same as Alexnet). We also fine tune the previous two fully connected layers (fc6 and fc7). However, their weights are not learned from scratch. We use a learning rate of 0.001 for these layers while performing the gradient descent with the setup classification training data. Once the Alexnet has been fine-tuned a new frame can be forward propagated through this network to find the classification label. For a test 10-second clip, we determine the setup label for each frame individually and assign the majority label to the full clip. We refer to this CNN as Setup-CNN.

3.3 Visual Feature Extraction

In this section, we describe the details of the visual features used for predicting the liveliness of a video clip. The visual modality is

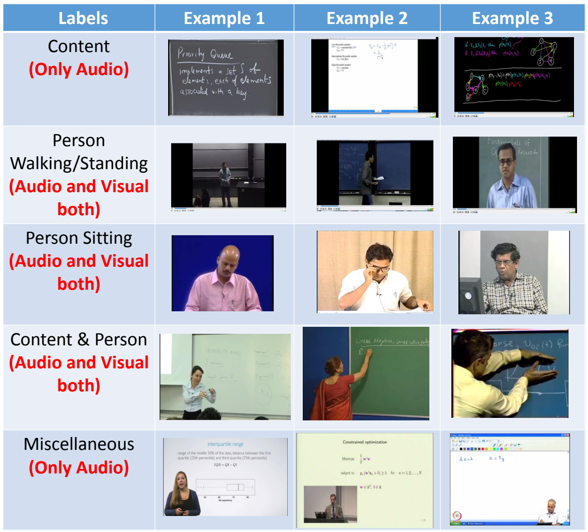

Figure 2: Example frames from different visual setup categories. We also point out the modalities which are used for liveliness in each of these setups.

used to capture the movement of the lecturer. We used a state-of-the-art deep CNN architecture to represent the visual information in the form of motion across the frames. Unlike the CNN model used in Section 3.2.3 (where input to the model was an RGB image comprising of 3 channels), the input to the CNN model in this section is formed by stacking horizontal and vertical optical flow images from 10 consecutive frames of a video clip. We refer to this CNN model as Motion-CNN in the subsequent sections of the paper.

For the Motion-CNN, we fine-tuned the VGG-16 temporal-net trained on UCF-101 [30] action dataset. The final fully connected layers (fc6, fc7, and fc8) of VGG-16 are fine-tuned with respect to the liveliness labels of the videos. The activations of the fc7 layer are extracted as the visual representation of the stacked optical flows which were provided as the input to the model. Given a 10-second clip, we generate a feature representation v_t (Section 3.1) from the corresponding 10 frame optical flow stack. We provide v_t as an input to LSTM module at time t to create a single visual representation for the full 10-second clip (Section 5.2).

Implementation Details: We use the GPU implementation of TVL1 optical flow algorithm [31]. We stack the optical flows in a 10-frame window of a video clip to receive a 20-channel optical flow image as an input (one horizontal channel and one vertical channel for each frame pair) to the Motion-CNN model. In Motion-CNN model, we also change the number of neurons in fc7 layer from 4096 to 512 before finetuning the model to get a lower dimensional representation of the 10 frame optical flow stack. We adopt a dropout ratio of 0.8 and set the initial learning rate to 0.001 for fc6, and to 0.01 for fc7 and fc8 layers. The learning rate is reduced by a factor of 10 after every 3000 iterations.

3.4 Audio Feature Extraction

We extract the audio feature a_t (Section 3.1) using a convolutional neural network. For each t, we find a corresponding one second long audio signal from the 10-second clip. We apply the Short-Time Fourier Transformation to convert each one second 1-d audio signal into a 2-D image (namely log-compressed mel-spectrograms with 128 components) with the horizontal axis and vertical axis being time-scale and frequency-scale respectively. The CNN features are extracted from these spectrogram images and used as inputs to the LSTM. We finetune the final three layers of Alexnet [17] to learn the spectrogram CNN features. We change the number of nodes in fc7 to 512 and use the fc7 representation corresponding to each spectrogram image as input to the LSTMs. The fine tuned Alexnet for the spectrogram feature extraction is referred as Audio-CNN. Learning rate and dropout parameters are chosen same as mentioned in Section 3.3.

3.5 Long Short Term Memory Networks

The Motion-CNN (Section 3.3) and the audio-CNN (Section 3.4) model only the short-term local motion and audio patterns in the video respectively. We further employ LSTMs to capture long-term temporal patterns/dependencies in the video. LSTMs map the arbitrary length sequential information of input data to output labels with multiple hidden units. Each of the units has built-in memory cell which controls the in-flow, out-flow, and accumulation of information over time with the help of several non-linear gate units. We provide a detailed description of LSTM networks below.

RNNs [8] are a special class of artificial neural networks, where cyclic connections are also allowed. These connections allow the networks to maintain a memory of the previous inputs, making them suitable for modeling sequential data. Given an input sequence $\mathbf{x}$ of length T, the fixed length hidden state or memory of an RNN $\mathbf{h}$ is given by

$$h_t = g(x_t, h_{t-1}) \quad t = 1, \ldots, T \tag{1}$$

We use $h_0 = 0$ in this work. Multiple such hidden layers can be stacked on top of each other, with x_t in equation 1 replaced with the activation at time t of the previous hidden layer, to obtain a 'deep' recurrent neural network. The output of the RNN at time t is computed using the state of the last hidden layer at t as

$$y_t = \theta(W_{yh} h_t^n + b_y) \tag{2}$$

where θ is a non-linear operation such as sigmoid or hyperbolic tangent for binary classification or softmax for multiclass classification, b_y is the bias term for the output layer and n is the number of hidden layers in the architecture. The output of the RNN at desired time steps can then be used to compute the error and the network weights updated based on the gradients computed using Back-propagation Through Time (BPTT). In simple RNNs, the function g is computed as a linear transformation of the input and previous hidden state, followed by an element wise non-linearity.

$$g(x_t, h_{t-1}) = \theta(W_{hx} x_t + W_{hh} h_{t-1} + b_h) \tag{3}$$

Such simple RNNs, however, suffer from the vanishing and exploding gradient problem [7]. To address this issue, a novel form of recurrent neural networks called the Long Short Term Memory (LSTM) networks were introduced in [7]. The key difference between simple RNNs and LSTMs is in the computation of g, which is done in the latter using a memory block. An LSTM memory block consists of a memory cell c and three multiplicative gates which regulate the state of the cell - forget gate f, input gate i and output gate o. The memory cell encodes the knowledge of the inputs that have been observed up to that time step. The forget gate controls whether the old information should be retained or forgotten. The input gate regulates whether new information should be added to the cell state while the output gate controls which parts of the new cell state to output. The equations for the gates and cell updates at time t are as follows:

$$i_t = \sigma(W_{ix}x_t + W_{ih}h_{t-1} + b_i) \tag{4}$$

$$f_t = \sigma(W_{fx}x_t + W_{fh}h_{t-1} + b_f) \tag{5}$$

$$o_t = \sigma(W_{ox}x_t + W_{oh}h_{t-1} + b_o) \tag{6}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \phi(W_{cx}x_t + W_{ch}h_{t-1} + b_c) \tag{7}$$

$$h_t = o_t \odot c_t \tag{8}$$

where $\odot$ is the element-wise multiplication operation, σ and ϕ are, respectively, the sigmoid and hyperbolic tangent functions, and h_t is the output of the memory block. Like simple RNNs, LSTM networks can be made deep by stacking memory blocks. The output layer of the LSTM network can then be computed using equation 2. We refer the reader to [7] for more technical details on LSTMs. The details of the architecture used in this work are described in section 5.2

3.6 Multimodal LSTM for liveliness classification

In the proposed approach, LSTMs are used to learn the discriminative visual and audio feature representations for liveliness. The estimates from audio and visual LSTMs are combined to estimate the overall liveliness of videos. For setup categories 'Person Walking/Standing', 'Person Sitting' and 'Content & Person' setup, both the modalities are used for liveliness prediction. For the remaining videos from 'Content' and 'Miscellaneous' categories, only the audio LSTM representation is used to determine the liveliness label. The details of the proposed approach are described below:

- **Visual-LSTM:** A multi-layer LSTM network is trained to learn the discriminative visual features for liveliness. The number of layers and the number of nodes in each layer in the LSTM network are determined based on a validation dataset. The input to the network at each time step t is a 512 dimensional visual feature extracted as described in 3.3.
- **Audio-LSTM:** The approach for training an audio LSTM is similar to that for training the visual LSTM. The only difference is that the visual features are replaced by the audio features as described in 3.4.
- **Multimodal-LSTM:** Once we learn the discriminative audio and visual LSTMs, the next step is to combine their predictions to determine the final liveliness. The visual and audio features from each 10-second clip are now forward-propagated through the visual-LSTM and audio-LSTM respectively. Once the features corresponding to all the time-steps of a clip have been forward-propagated, the liveliness prediction from each of these LSTM networks are obtained. If the setup corresponding to a clip requires combining audio and visual modality information, we assign the clip a positive liveliness label if any one of the visual-LSTM or Audio-LSTM network predicts the label of the clip as positive. Otherwise, the audio-LSTM label is used as the final label for the 10-second clip.

The proposed multimodal pipeline for liveliness prediction is called **LIVELINET** and will be referred as that from now on.

4. BASELINE DETAILS

In this section, we describe several baselines which do not use any deep neural network for feature extraction or classification. However, these methods have demonstrated state-of-the-art accuracy in many video/audio classification applications. We wanted to evaluate how good these "shallow" methods perform on the liveliness prediction task.

4.1 Visual Baseline

The visual baseline consists of training a SVM classifier on state-of-the-art trajectory features aggregated into local descriptors. Improved Dense Trajectories (IDT) [32] have been shown to achieve state of the art results on a variety of action recognition benchmark datasets. Visual feature points on the visual frames are densely sampled and tracked across subsequent frames to obtain dense trajectories. Once the IDTs are computed, VLAD (Vector of Locally Aggregated Descriptors) encoding [33] is used to obtain a compact representation of the video. We set the number of clusters for VLAD encoding at 30 and obtain a 11880-dimensional representation for each video. SVM classifier with RBF kernel is used for the classification. We compare this visual baseline against the proposed approach.

4.2 Audio Baselines

We compare LIVELINET with two different audio baselines; the first one uses bag of audio words and the second one uses Hidden Markov Models (HMM). The audio features are computed at a frame rate of 10 ms. The features are computed using the open source audio feature extraction software OpenSMILE [34]. Motivated by the findings in [35] and [36], where the authors show superior performance on various paralingustic challenges, our frame-level features consist of (a) loudness, defined as normalized intensity raised to a power of 0.3, (b) 12 Mel Frequency Cepstral Coefficients (MFCCs) along with the log energy ($MFCC_0$) and their first and second order delta values to capture the spectral variation, and (c) voicing related features such as the fundamental frequency (F0), voicing probability, harmonic noise ratio and zero crossing rate. (Intensity and fundamental frequency features have been found to be beneficial in liveliness classification in [4] also.) Authors in [36] refer to these frame-level features as Low Level Descriptors (LLD) and provide a set of 21 functionals based on quartile and percentile to generate chunk level features. We use all of these LLDs and the functionals for the audio feature extraction. For every one second audio signal (obtained using the same method as described in Section 3.4), these frame-level features are concatenated to form a $(44 * 100 = 4400)$ dimensional feature vector. The dimensionality of the chunk-level audio feature is further reduced to 400 by performing a PCA across all the chunks in the training data.

The audio features from all the one second audio signals in the training videos are clustered into 256 clusters. A nearest neighbour cluster centre is found for each of these audio features. We then create a 256-dimensional histogram for each clip based on these nearest neighbour assignments. This approach, known as the bag-of-words model is popular in computer vision and natural language processing, and is beginning to be extended to the audio domain in the form of bag-of-audio-words (BoAW) (e.g., [37]). A SVM classifier with RBF kernel is trained on this BoAW representation.

As a second baseline, two 3-state HMMs, one each for the positive and the negative class, are trained using the sequence of audio features computed on these one second audio signals. Only left-to-right state transitions are permitted with a potential skip from the first state to the third state. Each state is modeled as 16-mixture Gaussian Mixture Model. The 44 frame-level LLD are the inputs to the HMM framework. The Scilearn implementation of HMM is used.

4.3 Multimodal baseline

For combining the audio and video modalities we employ a classifier stacking approach. Stacking involves learning an algorithm to combine the predictions of other classifiers. We first train two SVM classifiers on audio and video features separately. The features and kernels used here are the same as the individual audio and visual baselines described earlier. Subsequently, another SVM classifier (with RBF kernel) is trained on the predictions of the audio and video classifiers to make the final prediction. We compare this baseline against the proposed multimodal classifier.

5. EXPERIMENTAL RESULTS

In this section, we provide the details of the experimental results. First, we describe the StyleX dataset followed by the details of the proposed LSTM network architecture and setup classification results. Next, we provide the liveliness classification results using the proposed multimodal deep neural network method. Finally, we perform some preliminary quality analysis of the lively/not-lively videos.

5.1 Dataset

We use the StyleX dataset proposed in [6] for our experiments. StyleX comprises of 450 one-minute video snippets featuring 50 different instructors, 10 major topics in engineering and various accents of spoken English. Each video was annotated by multiple annotators for liveliness. The scores from all annotators (in the range $0 - 100$, where 0 implies least lively and 100 implies most lively) corresponding to a particular video were averaged to obtain the mean liveliness score. The bimodal distribution of the mean liveliness scores were analyzed to estimate the threshold for binary label assignment (lively and not-lively). All videos with liveliness score above the threshold were assigned to the positive class whereas the remaining videos were assigned to the negative class. At a threshold of 54, we have 52% videos in the negative class (Thus, a simple majority-class classifier would lead to 52% classification accuracy). Out of the 450 StyleX videos, we randomly choose 60% for training, 20% for validation and 20% for testing while ensuring a proportional representation of both the classes in each subset. Since the proposed method takes 10-second clips as input during training and testing, we further split each one-minute video into 10-second clips bookended by silence, with a 50% overlap across adjacent clips. Each of these 10-second clips are assigned the same label as the actual one-minute videos and are treated as independent training instances. Likewise, during test, the 10-second clips are extracted from one-minute videos. The label is predicted for each 10-second clip and the label of the one-minute video is determined based on the majority vote.

5.2 LSTM Architecture Details

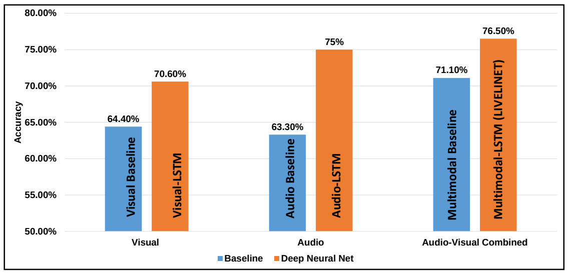

Figure 3: A comparison of results obtained from our proposed Multimodal-LSTM (LIVELINET) approach and the baselines.

The parameters of the proposed visual-LSTM and audio-LSTM were selected using the validation set. The learning rate was initialized to 10^{-4} and decayed after every epoch. Dropout rate of 0.2 was used for the activations of the last hidden layer. We tried nine different combinations for the number of hidden layers (1, 2, 3) and number of units in each layer (128, 256, 512), for both visual and audio modalities. Visual-LSTM with 2 layers and 256 hidden units and audio-LSTM with 2 layers and 256 hidden units led to the optimal performance on the validation set.

5.3 Setup Classification

In this section, we report the visual setup classification results obtained using the framework proposed in Section 3.2. As discussed in Section 5.1, the number of video clips used is 2700 for the training phase and 900 each for the validation and testing phase (all clips are approximately 10 seconds long). The network is trained with all the frames ($\sim$ 300K) extracted from the training video clips. At the time of testing, a label is predicted for each of the frame in a 10-second clip and their majority is taken as the label of the full clip. We evaluate 5-way classification accuracy of the video clips into different visual setups. Our proposed CNN architecture achieves a classification accuracy of 86.08% for this task. However, we notice that for the task of liveliness prediction, we only require the classification of video clips into two different classes - (a) clips requiring only audio modality, and (b) clips requiring both audio and video modality for liveliness prediction. For this task of binary classification ('Content or Miscellaneous' v/s 'Person Walking/Standing or Person Sitting or Content & Person'), our system achieves an accuracy of 93.74%. Based on the visual setup label of a clip, we use either both audio/visual or only audio modality for liveliness prediction.

5.4 Liveliness Classification

In this section, we present the performance of proposed multimodal deep neural network for liveliness prediction. Figure 3 depicts the results of our experiments. We obtain an accuracy of 70.6% with the Visual-LSTM, an absolute improvement of 6.2% over the visual baseline. The two audio baselines of HMM and BoAW methods lead to an accuracy of 60% and 63.3%, respectively. The Audio-LSTM setup leads to 75.0% accuracy, an increase of 11.7% over the best audio baseline. The proposed Multimodal-LSTM method (LIVELINET) achieves an accuracy of 76.5% compared to 71.1% obtained using the audio-visual baseline, an absolute improvement of 5.4% (relative improvement of 7.6%). We are also relatively 1.9% better than using only the audio-LSTM. The boost in accuracy when using both the modalities indicates that the information available from audio and visual modalities are complimentary and the proposed approach exploits it optimally.

5.5 Qualitative Analysis

We also perform qualitative analysis of the videos that are predicted lively/not-lively by LIVELINET. Our goal is to determine the general visual and audio patterns that make a video lively. These is the preliminary analysis of exemplar lively and exemplar non-lively lectures. We continue to perform a more systematic and in-depth qualitative analysis to understand two aspects: (a) patterns that the proposed classifier identifies as representative of lively and of not-lively, and (b) general audio-visual patterns that may have influenced the human labelers in assigning the 'lively or non-lively' label . One of the current directions for extending this work is to understand pedagogically-proven best practices of teaching and codify that knowledge in the form of features to be extracted and fed to the classifier. Some example frames from lively and not-lively videos as predicted by LIVELINET are shown in Figure 4. Some of our initial finding are: (a) Lecturers who alternate between making eye contact with the audience and looking at the content are perceived as more lively. (b) Similarly, voice modulations and moving around in the classroom (as opposed to sitting in place) and specific visual references (like pointing to written content) to synchronize with the spoken content seem to positively influence perceived liveliness.

6. CONCLUSION

We propose a novel method called LIVELINET that combines visual and audio information in a deep learning framework to predict liveliness in an educational video. First, we use a CNN architecture to determine the overall visual style of an educational video. Next, audio and visual LSTM deep neural networks are combined to estimate if a video is lively or not-lively. We performed experiments on the StyleX dataset and demonstrated significant improvement compared to the state-of-the-art methods. Future directions include incorporating text-based features for a content-based liveliness scoring. We also note that LIVELINET is going to be part of our e-learning platform TutorSpace.

References

[1] P. J Guo and K. Reinecke. Demographic differences in how students navigate through moocs. In *LAS*. ACM, 2014.

[2] J. Kim, P. J Guo, D. T Seaton, P. Mitros, K. Z Gajos, and R. C Miller. Understanding in-video dropouts and interaction peaks inonline lecture videos. In *LAS*. ACM, 2014.

[3] S S. Krishnan and R. K Sitaraman. Video stream quality impacts viewer behavior: inferring causality using quasi-experimental designs. *Networking, IEEE/ACM Transactions on*, 2013.

[4] S. Mariooryad, A. Kannan, D. Hakkani-Tur, and E. Shriberg. Automatic characterization of speaking styles in educational videos. In *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*. IEEE, 2014.

[5] Rebecca Hincks. Measures and perceptions of liveliness in student oral presentation speech: A proposal for an automatic feedback mechanism. *System*, 33(4):575–591, 2005.

[6] H. Arsikere, S. Patil, R. Kumar, K. Shrivastava, and O. Deshmukh. Stylex: A corpus of educational videos for research on speaking styles and their impact on engagement and learning. In *INTERSPEECH*, 2015.

[7] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 1997.

[8] R. J Williams and D. Zipser. A learning algorithm for continually running fully recurrent neural networks. *Neural computation*, 1989.

[9] A. Graves and J. Schmidhuber. Offline handwriting recognition with multidimensional recurrent neural networks. In *NIPS*, 2009.

[10] M. Baccouche, F. Mamalet, C. Wolf, C. Garcia, and A. Baskurt. Sequential deep learning for human action recognition. In *Human Behavior Understanding*. Springer, 2011.

[11] X. Chen and Lawrence Z. C. Mind's eye: A recurrent visual representation for image caption generation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

[12] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan. Show and tell: A neural image caption generator. In *CVPR*, 2015.

[13] L. Yao, A. Torabi, K. Cho, N. Ballas, C. Pal, H. Larochelle, and A. Courville. Describing videos by exploiting temporal structure. *stat*, 2015.

[14] A. Graves, A. Mohamed, and G. Hinton. Speech recognition with deep recurrent neural networks. In *ICASSP*. IEEE, 2013.

[15] A. Graves and N. Jaitly. Towards end-to-end speech recognition with recurrent neural networks. In *ICML*, 2014.

[16] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*, 2014.

[17] A. Krizhevsky, I. Sutskever, and G. E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, 2012.

[18] Zuxuan Wu, Xi Wang, Yu-Gang Jiang, Hao Ye, and Xiangyang Xue. Modeling spatial-temporal clues in a hybrid deep learning framework for video classification. In *Proceedings of the 23rd Annual ACM Conference on Multimedia Conference*, pages 461–470. ACM, 2015.

[19] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2015.

[20] Saurabh Gupta, Ross Girshick, Pablo Arbeláez, and Jitendra Malik. Learning rich features from RGB-D images for object detection and segmentation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2014.

[21] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. *CVPR*, 2015.

[22] Seunghoon Hong, Tackgeun You, Suha Kwak, and Bohyung Han. Online tracking by learning discriminative saliency map with convolutional neural network. *CoRR*, 2015.

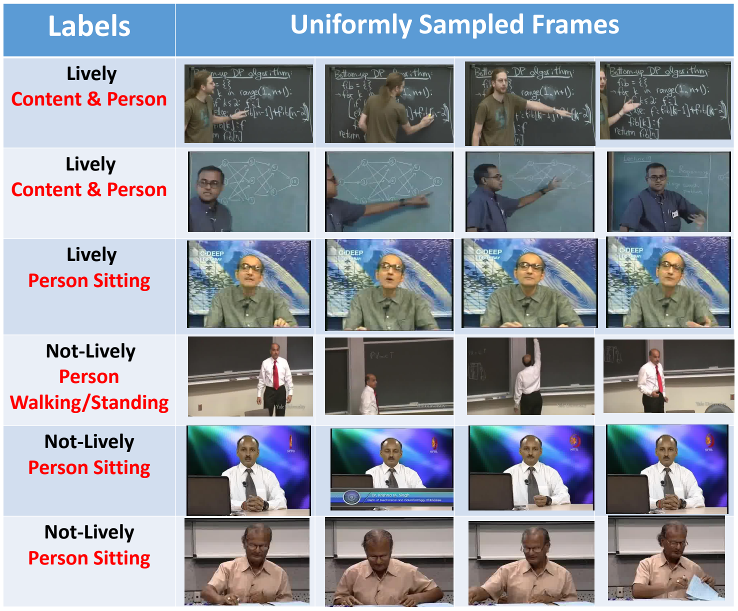

Figure 4: Some example frames from videos predicted as lively and not-lively by our proposed method LIVELINET. The setup labels predicted by the proposed Setup-CNN approach are also shown.

[23] Tian Gan, Yongkang Wong, Bappaditya Mandal, Vijay Chandrasekhar, and Mohan S. Kankanhalli. Multi-sensor self-quantification of presentations. In *Proceedings of the 23rd ACM International Conference on Multimedia*, 2015.

[24] A. T. Nguyen, W. Chen, and M. Rauterberg. Online feedback system for public speakers. In *E-Learning, E-Management and E-Services (IS3e), 2012 IEEE Symposium on*, 2012.

[25] Vanessa Echeverría, Allan Avendaño, Katherine Chiluiza, Aníbal Vásquez, and Xavier Ochoa. Presentation skills estimation based on video and kinect data analysis. In *Proceedings of the 2014 ACM Workshop on Multimodal Learning Analytics Workshop and Grand Challenge*, MLA '14, 2014.

[26] Lei Chen, Gary Feng, Jilliam Joe, Chee Wee Leong, Christopher Kitchen, and Chong Min Lee. Towards automated assessment of public speaking skills using multimodal cues. In *Proceedings of the 16th International Conference on Multimodal Interaction*, 2014.

[27] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015.

[28] Jeffrey Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, and Trevor Darrell. Long-term recurrent convolutional networks for visual recognition and description. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2625–2634, 2015.

[29] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3431–3440, 2015.

[30] Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. In *NIPS*, pages 568–576, 2014.

[31] C. Zach, T. Pock, and H. Bischof. A duality based approach for realtime tv-l1 optical flow. In *Proceedings of the 29th DAGM Conference on Pattern Recognition*, 2007.

[32] H. Wang and C. Schmid. Action recognition with improved trajectories. In *ICCV*. IEEE, 2013.

[33] H. Jégou, M. Douze, C. Schmid, and P. Pérez. Aggregating local descriptors into a compact image representation. In *CVPR*. IEEE, 2010.

[34] Florian Eyben, Martin Wöllmer, and Björn Schuller. Opensmile: The munich versatile and fast open-source audio feature extractor. In *Proceedings of the International Conference on Multimedia*, MM '10, pages 1459–1462, New York, NY, USA, 2010. ACM.

[35] Björn Schuller, Stefan Steidl, Anton Batliner, Felix Burkhardt, Laurence Devillers, Christian A Müller, and Shrikanth S Narayanan. The interspeech 2010 paralinguistic challenge. 2010.

[36] Björn Schuller, Stefan Steidl, Anton Batliner, Alessandro Vinciarelli, Klaus Scherer, Fabien Ringeval, Mohamed Chetouani, Felix Weninger, Florian Eyben, Erik Marchi, et al. The interspeech 2013 computational paralinguistics challenge: social signals, conflict, emotion, autism. 2013.

[37] Stephanie Pancoast and Murat Akbacak. Bag-of-audio-words approach for multimedia event classification.