Sapienza
Università di Roma

Engagement Detection in e-learning Environments

Department of Information Engineering, Computer Science and Statistics
Corso di Laurea Magistrale in Data Science

Candidate
Onur Copur
ID number 1891194

Thesis Advisor
Prof. Simone Scardapane

Co-Advisor
Dr. Jürgen Slowack

Academic Year 2020/2021

Thesis defended on 25 October 2021
in front of a Board of Examiners composed by:

Prof. Tardella Luca (chairman)
Prof. Brutti Pierpaolo
Prof. Cianfrani Antonio
Prof. Crespi Mattia
Prof. Marzano Frank Silvio
Prof. Petti Manuela
Prof. Scardapane Simone

Engagement Detection in e-learning Environments
Master's thesis. Sapienza – University of Rome

This thesis has been typeset by LaTeX and the Sapthesis class.

Author's email: onurcopur12@gmail.com

Abstract

In this master thesis, the relation between facial expressions/body pose and the subject's engagement level is investigated in e-learning environments. We propose an end-to-end deep learning-based system that detects the engagement level of the subject given the video of the subject while watching educative material. The three main components of the model are feature extraction, feature aggregation, and sequence modeling. The proposed model achieved state-of-the-art results in two publicly available datasets. In addition to that, the integrated gradients method is used to analyze feature importance and results showed that head pose and eye gaze-related features are the most effective facial expressions/body pose features in engagement detection. Finally, the performance of the model is tested on unlabeled videos collected by the authors and observed that the model is also able to generate reasonable predictions and distinguish different levels of engagement on videos from outer sources.

Contents

Chapter 1

Introduction

1.1 Problem Definition

With the rise of deep learning and Artificial intelligence during the previous decade, AI tools become more and more involved in our daily lives. Especially the outstanding success of deep learning methods in computer vision improved the performance in many different tasks such as emotion recognition, object detection, video action recognition, visual tracking, etc. This motivated many researchers around the world to contribute and improve this area of research and today, deep learning is the most popular method not only in computer vision but also in many fields like natural language processing, robotics, medicine, etc. However, the industrial applications of deep learning-based methods are still not at a desirable level due to high computational costs and the lack of available training data. The potential of deep learning-based methods was discovered nearly 10 years ago and most of the firms around the world started to create partnerships with universities and attempt to industrialize the use of deep learning methods. Thus, the COVID-19 outbreak last year gave another boost to the technological improvements in many different sectors. The online working and learning environments became essential in our lives and tools used for these purposes developed rapidly in a single year.

Recognition of user interaction becomes highly important in a digital environment. Applications need to be “aware” of the user’s presence when delivering information[50]. For this reason, automatic analysis of non-verbal communication becomes crucial in online environments. Some applications in this context includes engagement detection in online learning [30, 26, 19], social role detection and engagement detection in meetings [68, 24], multi-model emotion detection [32, 55, 56], bodily expressed emotion understanding [38, 23] etc.

To this end, this thesis aims to explore and extend the state-of-the-art in non-verbal / body language analysis in collaboration with Barco [1], a company offering a wide range of products and solutions related to visualization such as health care solutions, projectors, LED displays, and video walls. One of the most innovative projects in Barco is the virtual classroom. Barco’s virtual classroom is a cloud-based SaaS solution where a teacher is in a specifically equipped room while students connect over the Internet. This is interesting in various education scenarios - such as corporate training and higher education - where students may be located in different places around the world. In the figure below, you can see two different use cases of the virtual classroom, one is for completely online scenario and the other one is

Figure 1.1. (a) Barco's virtual classroom all participants are attending online. (b) Blended learning, some participants are in-class some are attending online [1]

blended where there are both in-class and online participants.

Barco's virtual classroom is already providing many technological features to improve the learning experience and increase efficiency in corporate meetings but still, non-verbal communication is not as easy as direct interaction, especially in the online setting. To provide feedback to teachers, students, companies, and employees both in e-learning and corporate meeting scenarios, it is crucial to detect and analyze the social signals received from participants. Social signals can be defined as, communicative or informative signals which provide information about social facts.[10] While Social signal processing (SSP) is the computing domain aimed at modeling, analysis, and synthesis of social signals in human-human and human-machine interactions [10]. Here, we are interested in some sub-task of social signal processing such as detecting and analyzing the engagement level of participants which can give beneficial feedback in e-learning environments. Online learners participate in various educational activities including reading, writing, watching video tutorials, online exams, and online meetings. During the participation in these educational activities, they show various engagement levels, such as boredom, frustration, delight, neutral, confusion, and learning gain. To provide feedback to both instructors and students online educators need to detect their online learners' engagement status precisely and efficiently. In e-learning environments, students are not speaking most of the time. For this reason, the engagement detection model should extract valuable information from only visual input. This makes the problem non-trivial and subjective because people can perceive different engagement levels from the same input video.

In this thesis, we propose an end-to-end deep learning-based system that detects the engagement level of the subject in an e-learning environment. The input of the system is the video of a subject recorded while watching an educative material and the output of the system is the engagement level of the subject for each sub clip of the input video. The system consists of 3 main parts which are feature extraction, feature aggregation and engagement prediction with sequence models. First the input video is passed to *OpenFace* [8] and *OpenPose* [11] which are tools for facial behaviour analysis and body pose estimation. With the help of these tools, features like head pose, eye gaze, facial action units, and upper body key points are extracted in frame level. After that, the extracted features are aggregated with pre-defined aggregation functions for additional feature extraction. Finally, the features are fed into sequence models to exploit the temporal aspect of the input video segment and to make an estimation on the engagement level. We believe that deep learning-based sequence models such as Long Short Term Memory (LSTM) [29] and Gated Recurrent Unit (GRU) [14] will be a good fit for this multi-feature sequence classification/regression task. To train our neural network models, we used the only two publicly available datasets which are *Daisee* [26] and *Engagement in the wild* [34]. Both datasets are formed from video sequences of subjects watching educational material and for each video snippet, there is an engagement level label from $\{0, 1, 2, 3\}$ where 0 indicates very low engagement and 3 indicates very high engagement. Both datasets are challenging in terms of few samples and imbalanced labels which makes the learning procedure harder and deep learning models become more prone to over-fitting. However, still, the *state of the art results* (SOTA) are accomplished by deep learning-based methods so this motivates us to push the limits of deep learning-based methods. After training our sequence models with Daisee and Engagement in the wild datasets, we observed that our system can reach the new SOTA performance in both datasets. For engagement in the wild validation dataset with a mean square error (MSE) of 0.05011, whereas the previous best MSE was 0.05997. For the Daisee dataset, our model reached an accuracy of 64.42% whereas the previous best accuracy was 63.9%. In addition to that we used the integrated gradients method [48] to analyse the most effective features for the engagement detection task. We found out that head pose and eye gaze related features are most important for the proposed model. Finally, we also tested our system on a small group of individuals to observe the expected performance of the product when it will be deployed. The results show that the proposed model is able to generate reasonable predictions and distinguish between different levels of engagement.

The rest of this thesis is organized as follows;

In Chapter 2 We will introduce the background information needed to create a better understanding of this work. Namely, we will describe theoretical aspects of the methods and tools we used in our pipeline. Then, we will present the current literature on engagement detection in e-learning environments. First, we will give details about the two datasets *Daisee* [26] and *Engagement in the wild* [34] we used for training our model. After, we will present SOTA approaches and common pipelines for the engagement detection task.

In Chapter 3 We will give more details about Daisee and Engagements in the wild datasets and make an exploratory data analysis for both datasets. Moreover, we will investigate the extracted features with OpenPose and OpenFace. Finally, we will create simple baseline models for both datasets.

In Chapter 4 We will describe our model architecture in detail.

In Chapter 5 We will present our training procedures for both engagement classification with the Daisee dataset and engagement regression with Engagements in the wild dataset. After that, the experimental results will be reported with new SOTA results and feature importance will be shown with interpretability methods. Finally, we will show some examples of how the model is performing in real life and evaluate its applicability in a demo product.

In Chapter 6 the project will be discussed with its strong and weak sides and some possible future road maps will be mentioned.

Chapter 2

Background and Related Work

In this chapter, we will introduce some background knowledge to create a better understanding of the solution methods to the engagement detection problem. First, we will introduce OpenFace [8] and OpenPose [11], as feature extraction methods from videos. After that, we will describe the feature aggregation methods and *tsfresh* [15], a tool to extract characteristics from time series. Then, we will describe the Sequence models used to exploit temporal information in videos. Moreover we will describe the integrated gradients method [48] used for model interpretability and feature importance. Finally, we will mention current literature in this domain with publicly available datasets and SOTA models.

2.1 Feature Extraction

Feature extraction is an important phase in the predictive model pipeline for any machine learning task. We all know that the quality of the features dramatically affects the model performance. In social signal processing and human behavior analysis, Facial expressions and body pose give many clues about the emotional state of an individual. This is also true for engagement detection in e-learning environments. In the works [63, 30], authors use open source tools OpenFace [8] and OpenPose [11] to extract features many different features such as face landmarks, eye gaze, facial action units, head pose, and full-body pose.

2.1.1 OpenFace

OpenFace is the first open-source tool capable of facial landmark detection, head pose estimation, facial action unit recognition, and eye-gaze estimation. This tool is developed by combining methods that are generating state-of-the-art results for all the tasks above. By using this tool, we are able to extract 720 facial features related to eye gaze, head pose, facial landmarks, rigid and non-rigid face shape, facial action units, and histogram of oriented gradients (HOG). In figure 2.1, you can see a frame from the daisee dataset before and after feature extraction with OpenFace. In the rest of this part, we will focus on algorithms that are used in the OpenFace tool and how the feature extraction pipeline is constructed.

OpenFace Pipeline

OpenFace pipeline consists of 4 stages which are;

- Facial landmark detection and tracking

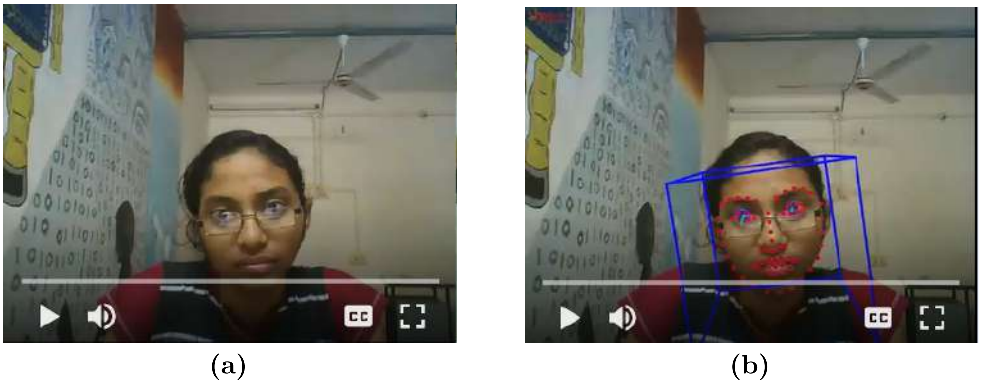


Figure 2.1. a is a frame from the Daisee dataset. (b) is the same frame with OpenFace features

- Head pose estimation
- Eye pose estimation
- Facial expression recognition

OpenFace uses Convolutional Experts Constrained Local Model (CE-CLM) [66] for **facial landmark detection and tracking**. This algorithm consists of 2 parts. Response map computation using Convolutional Experts Network (CEN) and shape parameter update using a Point Distribution Model. The CEN network estimates each landmark individually and independently of the position of other landmarks in the forward pass. While updating the parameters, the position of all landmarks is updated jointly penalizing misaligned landmarks and irregular shapes using a Point Distribution Model (PDM). Finally, the local appearance variations are modeled by patch experts. In figure 2.2, you can see the overview of CE-CLM pipeline. The correction Network is used for dataset-specific corrections for CE-CLM and the adjustment network and the adjustment network is for mapping 64 facial points to 84 point space.

In the OpenFace implementation of the CE-CLM model, some adjustments are made for speeding up optimization and to allow real-time performance. First, they retrained the patch experts (which is a deep neural net with approximately 180000 parameters) by using a deep neural network with half-size (approximately 90000 parameters. Second, they introduce the idea of *smart multiple hypothesis*. In the original work, CE-CLM uses multiple initialization hypotheses (11 in total) at different orientations to deal with hard images such as profile faces and occlusion. During fitting, the model with the best-converged likelihood is selected. This is crucial for e-learning environments because people generally touch their faces when watching lectures and this creates occlusions for facial landmark detection. However, this also slows down the approach. To speed up, an early hypothesis termination is applied based on the current model likelihood in the OpenFace application. Finally, in the original work, the response maps for each facial landmark are calculated in a dense grid around the current landmark estimate. Here the authors used a sparse grid instead of a dense grid to speed up. More details for the performance improvement adjustments and implementation details are presented in [8].

The head pose estimation can be easily done by using the same CE-CLM model used for facial landmark detection. This is possible because CE-CLM uses a 3D

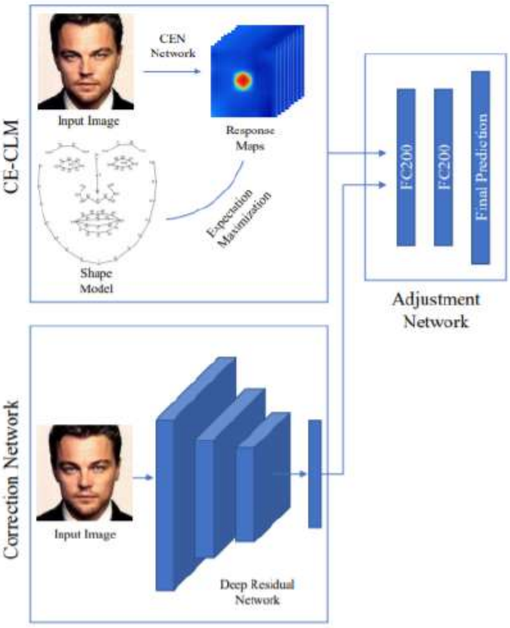


Figure 2.2. This is an image from a text that uses color to teach music [8].

representation of facial landmarks and projects them to the image. In this way head pose can be estimated by solving the n point in perspective problem with detected facial landmarks [28].
Next, the **eye gaze estimation** is needed to be done. For this task, a Constrained Local Neural Field (CLNF) landmark detector is used to detect eyelids, iris, and the pupil [60]. This model [7] is originally designed for face landmark detection but for detecting eyelids, iris, and the pupil, the model is trained on SynthesEyes dataset [60]. The detected pupil and eye location are used to calculate the gaze vector for each eye.
Finally, **the facial expression recognition** is done by using the SVM-based model suggested in [6]. In the figure 2.3 the architecture for detecting facial action units is presented. The model takes the input image with detected facial landmarks and then computes the HOG for the aligned and masked face. After that, a PCA dimensionality reduction is applied to HOG features and geometric features are extracted from the original image. Finally, all extracted features are fed into an SVM classifier for the Action unit classification task. The Labels are presented in the figure 2.4

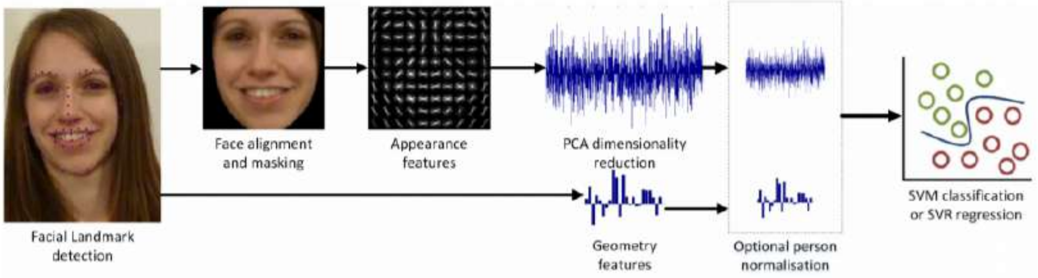
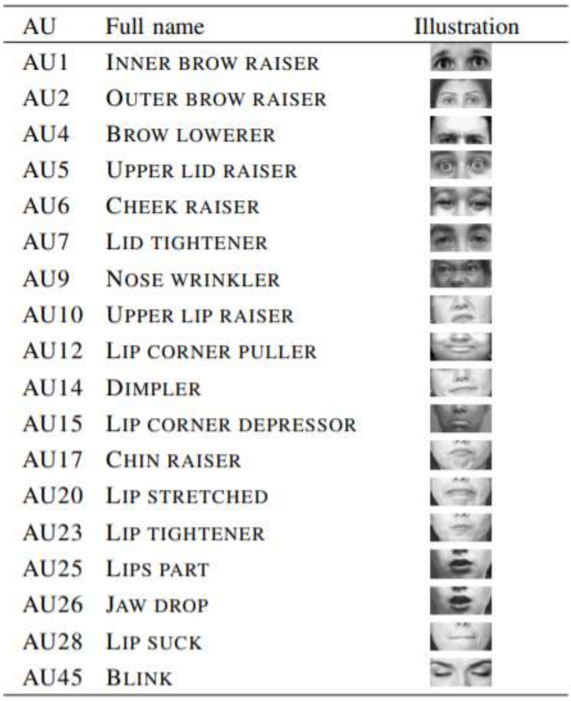

Facial Landmark detection
Face alignment and masking
Appearance features
PCA dimensionality reduction
Geometry features
Optional person normalisation
SVM classification or SVR regression

Figure 2.3. The model architecture of Facial Action Unit recognition[8].



AU	Full name	Illustration
AU1	Inner brow raiser	
AU2	Outer brow raiser	
AU4	Brow lowerer	
AU5	Upper lid raiser	
AU6	Cheek raiser	
AU7	Lid tightener	
AU9	Nose wrinkler	
AU10	Upper lip raiser	
AU12	Lip corner puller	
AU14	Dimpler	
AU15	Lip corner depressor	
AU17	Chin raiser	
AU20	Lip stretched	
AU23	Lip tightener	
AU25	Lips part	
AU26	Jaw drop	
AU28	Lip suck	
AU45	Blink	

Figure 2.4. Description and illustration of the f Facial Action Unit [8]

2.1.2 OpenPose

OpenPose is the first real-time multi-person system to jointly detect human body, hand, facial, and foot key points (in total 135 keypoints) on single images. In figure 2.5, you can see some snapshots from the Emotiw dataset after pose detection with OpenPose. The model takes a colored image as input and outputs the 2D locations of anatomical key points for each person in the image.

First, a CNN predicts a set of 2D confidence maps of body part locations and a set of 2D vector fields of part affinity fields (PAFs), which encode the degree of association between parts. Then, the confidence maps and the PAFs are parsed by greedy inference to output the 2D keypoints for all people in the image. In figure 2.6, you can see the complete pipeline of the system.

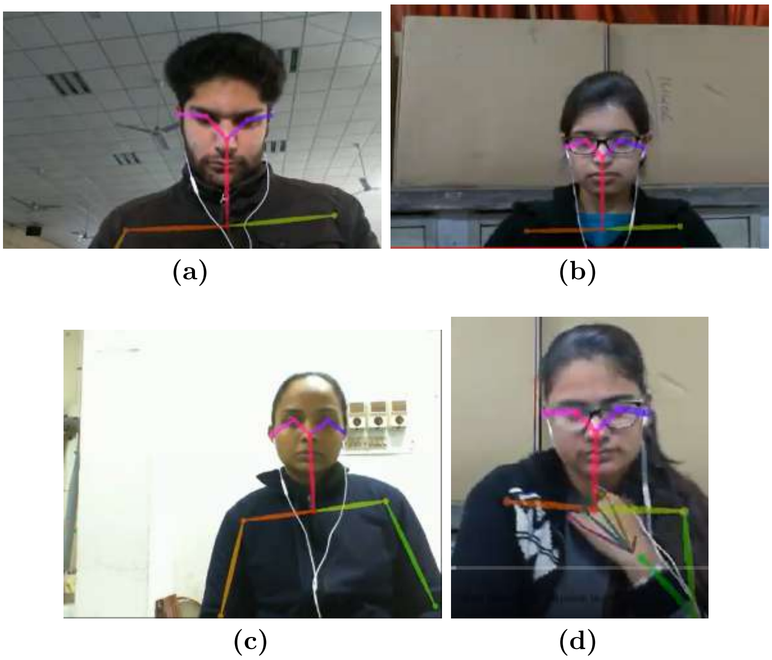
In figure 2.7, you can see the CNN architecture generating the Confidence Maps and PAFs. The first block shown in blue predicts the Affinity fields and then the detection of confidence maps with the second block shown in beige. The Input **F** of the first block is the feature vectors of images generated by the first 10 layers of a fine-tuned VGG-19 [47]. The set of Part Affinity Fields are a function of VGG features at time t and PAFs at time $t-1$. The function is shown by ϕ^t which refers to the CNN block in blue.

$$L^t = \phi^t(\mathbf{F}, L^{t-1})\ \forall t\ 2 \le t \le T_p$$

T_p is the total number of PAFs predictions. After T_p iterations, the process is repeated for the confidence maps detection, starting in the most updated PAF prediction,

$$S^{T_p} = \rho^t(\mathbf{F},\mathbf{L}^{T_p}), \forall t = T_p$$

$$S^t = \rho^t(\mathbf{F},\mathbf{L}^{T_p}, \mathbf{S}^{t-1})\forall T_p \le t \le T_p + T_c$$



(a)

(b)

(c)

(d)

Figure 2.5. different snaphots from Engagement in the wild dataset with OpenPose pose detection.

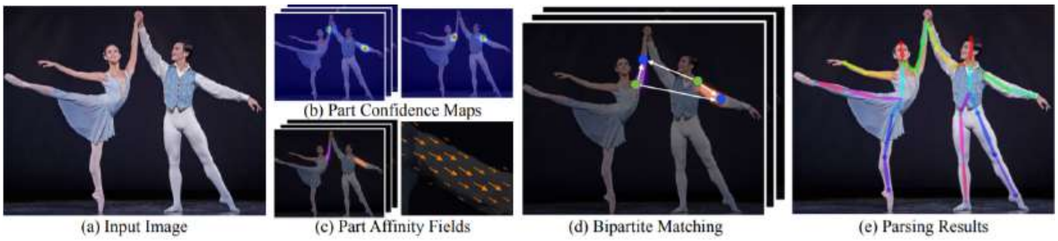


Figure 2.6. the overall pipeline of the OpenPose pose estimation model[11]

where ρ^t refers to o the CNNs for inference at Stage t and T_c to the number of confidence map stages. For both stages, the loss function between predictions and ground truth is the L_2 loss.

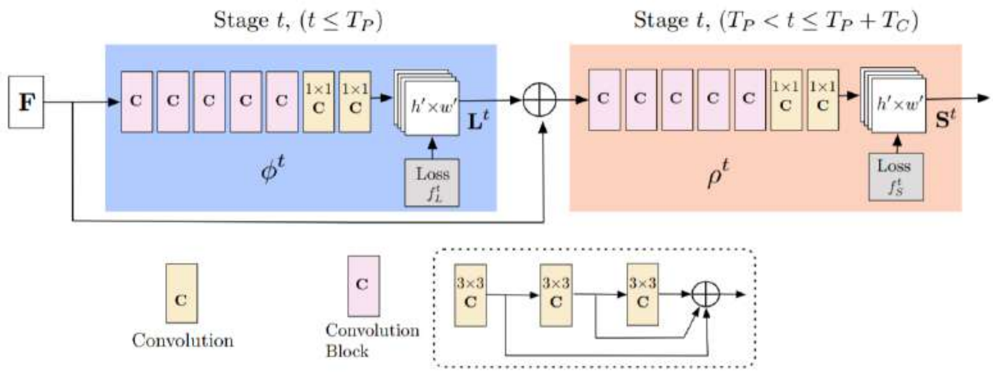


Figure 2.7. CNN architecture for Confidence Map and Part Affinity Field prediction [11]

After detecting the body parts, the next problem is combining them and ensuring that the combined parts belong to the same person. Although in Daisee and Engagement in the wild datasets there is only a single subject in the sceen mos of the time and this problem is not in our scope, OpenPose is a strong tool when it comes to estimate poses of multiple people. PAFs preserve both location and orientation information across the region of support of the limb. For each pixel in the area belonging to a particular limb, a 2D vector encodes the direction that points from one part of the limb to the other. Each type of limb has a corresponding PAF joining its two associated body parts.

2.2 Feature Aggregation

OpenFace and OpenPose provide much information about facial expressions and body pose at the frame level. After extracting this useful information, one can use consider this information as different time series changing throughout the video and conduct multiple time-series analyses. The feature aggregation step is important because it can exploit different characteristics of the time series. For example in e-learning scenarios, if the subject is looking at different places, this can be a sign of low engagement. The variance of the eye gaze vector in a window can give information about the engagement level for a specific part of the video. Based on this idea, we did experiments on our datasets with different aggregation functions such as mean, maximum, minimum, etc. for different window lengths. While doing these experiments, we used tsfresh [15], an efficient, scalable feature extraction algorithm for time series. Tsfresh provides many different feature extraction functions. Among these, we considered simple statistical aggregation functions and frequency domain functions. The simple statistical aggregation functions are mean, maximum, minimum, variance standard deviation, and length. The frequency-domain functions are the spectral centroid (mean) and variance of the absolute Fourier transform spectrum and the Fourier coefficients of the one-dimensional discrete Fourier Transform. Since the Fourier transform extracts information about the cyclic patterns in time series, we believe that it can provide useful information for the aggregated parts of the feature sequences.

2.2.1 Fourier Coefficients

Calculates the fourier coefficients of the one-dimensional discrete Fourier Transform for real input by fast fourier transformation algorithm.

$$A_k = \sum_{m=0}^{n-1} a_m \exp\{-2\pi i \frac{mk}{n}\}, \qquad k = 0, \ldots, n-1.$$

2.2.2 Spectral Centroid

It is calculated as the weighted mean of the frequencies present in the signal, determined using a Fourier transform, with their magnitudes as the weights.

2.3 Sequence Models

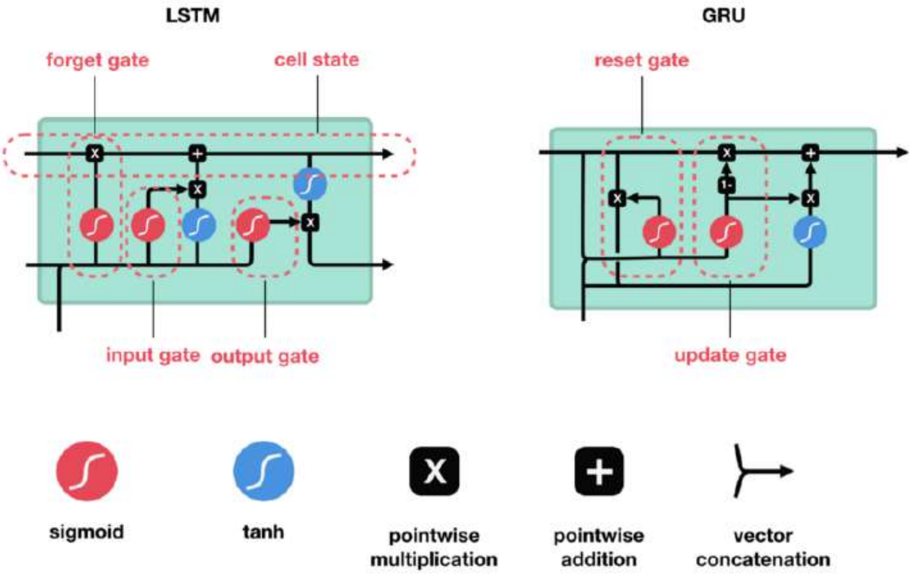


Figure 2.8. Inner mechanism of LSTM (left) and GRU (right) cells. [44]

2.3.1 LSTM and GRU

In figure 2.8, you can see the illustration of LSTM and GRU cells. LSTM and GRU models are frequently used to exploit the temporal aspect of video data [65, 57, 33]. LSTM and GRU are a type of Recurrent Neural Networks (RNN). Vanilla RNNs suffer from vanishing gradients so they are not capable of storing information from the past when the input sequence is long. In other words, vanilla RNN suffers from short-term memory. However, LSTM and GRU models have inner operations called gates that are regulating the flow of information, and only the relevant information from the input sequence is kept in the memory. For this reason, these two sequence models are candidate models for the Engagement detection task. Since a change in engagement, level depends on facial expression and body pose clues at specific times in a video sequence, LSTM and GRU models can capture these important moments and make engagement level decisions for short and long-duration videos.

2.4 Model Interpretability

2.4.1 Integrated Gradients

Integrated gradients represents the integral of gradients with respect to inputs along the path from a given baseline to input. [48].

$$IntegratedGrads_i(x) ::= (x_i - x_i') \times \int_{\alpha=0}^{1} \frac{\partial F(x' + \alpha \times (x - x'))}{\partial x_i} d\alpha$$

In the formula above, the function F represents the neural network. More specifically in the engagement detection task, the F function is the sequence model. x_i represents the i^{th} feature of the input and x_i' represents a baseline for the i^{th} feature. For image networks, the baseline could be the black image, while for text models it could be the zero embedding vector. In this engagement detection task, the features are sequential facial and body pose features. So for each feature such as eye gaze and head pose, the baseline would be the average value of the feature sequence. The Integrated gradients method considers a straight line path from the baseline to the input and computes the gradients at all points along the path. Integrated gradients are obtained by cumulating these gradients.

2.5 State-of-the-art on Engagement detection

2.5.1 Datasets for e-learning Environments

Daisee and Engagement in the wild

For e-learning environments, the datasets are mostly created to detect the engagement level of students [19]. DAISEE dataset [26] differs from the others in this sense. DAiSEE is a multi-label video classification dataset comprising of 9068 video snippets captured from 112 users for recognizing the user's affective states. Each video snippet is 10 seconds long and for each snippet, there are 4 different affective states labels as {Boredom, Confusion, Engagement, Frustration } and each label has a score as { Very Low, Low, High, Very High}. Many papers used this dataset and tried to improve the performance [30, 50, 18, 37, 39, 45]. This gives us the chance to compare our model with other applications.

Engagement in the wild is the sub-task in the Emotion Recognition in the Wild (EmotiW 2018) challenge [20]. The dataset is pretty similar to Daisee.The annotations are made by a team of 5 and the engagement of each subject is scored from $0 - 3$, the same as in the Daisee dataset. The distribution of videos in the dataset is as follows 9 videos belong to level 0, 53 for level 1, 82 for level 2, and 50 for level 3. Further details about the datasets as well as an in depth analysis will be provided in Chapter 3.

Other Datasets not publicly available

The Affective database for e-learning and classroom [3] is a new effective database using the students' facial expressions, hand gestures, and body postures. The labels for this dataset are { happiness, sadness, surprise, fear, disgust, anger, engaged, sleepy, boredom, frustrated, confuse }. Unlike other e-learning datasets, this set also includes hand gestures and body postures which are relatively more important in in-class lectures and corporate meeting scenarios. In addition to that, the additional annotations give the chance to use multi-modal architectures which is not possible for most of the e-learning datasets. Besides these advantages, this dataset also has similar limitations to DAISEE like lack of ethical diversity, unbalanced distribution of gender, and crowd-sourced annotations. Also, this fairly new dataset is not available to the public yet, and there are not many studies using this dataset. Another very recently realised dataset is Student Engagement Dataset [17]. This is a dataset of college students solving math problems on the educational platform MathSpring.org with a front facing camera collecting visual feedback of student gestures. The video dataset is annotated to indicate whether students' attention at specific frames is engaged or wandering. Unlike other dataset, this dataset contains samples with ethnic diversity. However the dataset contains only 19 students and the annotation is made crowd-sourced.

2.5.2 Engagement Detection Models

One of the first attempts to investigate the relationships between facial features, conversational cues, and emotional expressions with engagement detection is presented in [21]. Later on in [59, 25] authors used the Facial Action Coding System (FACS) which is a measure of discrete emotions with facial muscle movements, and point out the relation between specific engagement labels and Facial actions. Also in the work [59], authors show that automated engagement detectors perform with comparable accuracy to humans. All these works were using classical machine learning classifiers such as Gentleboost and SVM. After the deep learning revolution, more works with deep learning methods emerged. For example in [9] authors compared the performance of an LSTM based method with SVM and KNN methods. Unlike previous works, they considered engagement detection tasks both as classification and regression problems. They also used many non-verbal features such as Facial landmarks, FACS, Mean and median average optical flow velocities and average direction vector, Head size and pose, Facial geometric features. Moreover in [18], authors proposed a Local Directional Pattern (LDP) to extract person-independent edge features for the different facial expressions and Kernel Principal Component Analysis (KPCA) to capture the nonlinear correlations among the extracted features. After that, a deep belief network is trained to make classifications. The experiments are conducted on two-class and three-class classifications tasks and DAISEE dataset [26] is used.

DAISEE [26] dataset is pretty popular for e-learning engagement detection tasks and there are many works other works using this dataset. For example in [30], authors propose a model: Deep Engagement Recognition Network (DERN) which combines temporal convolution, bidirectional LSTM, and attention mechanism to identify the degree of engagement based on the features captured by OpenFace [8]. Moreover in [37] a model called the Deep Facial Spatiotemporal Network (DFSTN) is proposed. This model contains a pre-trained SE-ResNet-50 (SENet), which is used for extracting facial spatial features, and an LSTM with global attention for sequence modeling. The attention mechanism yields the discriminative attentional hidden state with the LSTM, which improves the experimental results effectively. Finally in works [50, 39, 45] authors used relatively simple models which are based on Convolutional Neural Networks (CNN) and Residual Networks (ResNet)[27]. All the works above consider the engagement detection problem as a multi-class classification problem and to the best of our knowledge, the most recent and state of the art performance for the DAISEE dataset is achieved by Abedi et al. [2] where they used a Resnet50[27] model to create a one-dimensional embedding vector in frame level and then used a Temporal Convolutional Network (TCN) [4] for temporal analysis of the video frames. They achieved an accuracy of 63.9% in the 4-class classification engagement task and they also reported their confusion matrix, unlike the previous works. Even though this is the SOTA model for the Daisee dataset, the confusion matrix shows that the model is good at predicting the high level of engagement labels correctly but still very bad at detecting the low level of engagement labels. This may be due to the highly imbalanced number of samples between low level and high level of engagement labels but if the model is not able o detect the low level of engagement values, its practical use is highly questionable.

Another very popular dataset for engagement detection in e-learning environments is Engagement in the wild dataset [34]. This dataset is used in Emotiw challenges made yearly as mentioned before and this contributed to an increase in the number of academic papers using this dataset. Unlike works using the Daisee dataset, in this challenge the engagement detection problem is considered as a regression problem which is more suitable to practical implementation due to the subjective nature of the perceived engagement. For example, the work by Yang et al. [63], presents the winner approach for Emotiw 2018 challenge. Here they used OpenFace [8] tool for extracting eye gaze and head pose features and for body posture, they used OpenPose [11]. Moreover, they also extract facial descriptor vector and body action features vector by using a pre-trained C3D network [53]. After extracting features from 4 different sources, 4 different LSTM units are trained separately and then the regression results are fused. The results showed that the best performance is achieved with 2 LSTM units and OpenFace features. Another successful model for Emotiw 2018 challenge is proposed by Niu et al. [40]. In this work, the authors created a 117-dimensional feature vector composed of eye gaze action units and head pose features extracted by OpenFace [8]. After that, they passed these feature vectors to 3 Gated Recurrent Units (GRU) and fused the 3 outputs with mean pooling. OpenFace [8] features continued to be used also in the work by Thomas et al [51]. Here authors used the same set of features with [40] but used a Temporal Convolutional Network (TCN) for the regression task. One of the best performance in terms of MSE 0.0611 is achieved by Wu et al. [61] with the model shown below in figure 2.9. This is a common pipeline also used in [63, 51, 40] and which we also took inspiration from. In this model first sequences of overlapping frames are extracted by a sliding window approach. Then features like eye gaze, head pose, etc. for each frame in a window are aggregated with classical statistics like minimum, maximum, and variance. After that, the resulting sequence of aggregated features is passed to a sequence model for engagement regression.

To the best of our knowledge, the only work using both Daisee and Engagement in the wild together is by Liao et al.[37]. In this work, the authors trained their model on the Daisee dataset with both regression and classification losses and then tested it on the Emotiw dataset. For both datasets, the performance is far from SOTA but for the first time, they used a Grad-CAM to visualize the gradients of

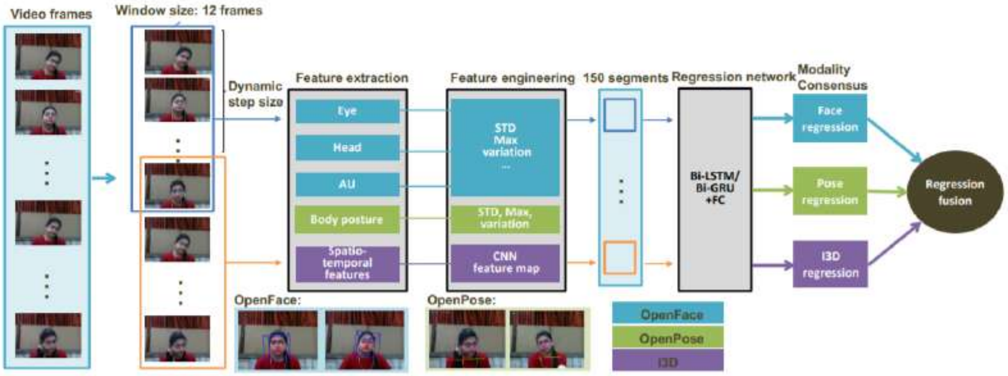


Figure 2.9. Model architecture by Wu et al. [61]

pixels in consecutive frames.

To sum up, we can say that features extracted from various tools such as OpenFace and OpenPose are frequently used in the engagement detection task. The general approach is extracting features with these tools and then using a sequential model which can capture the temporal relations between video frames. Also using both datasets for training the models is a possible direction to go which is not yet tried in the literature.

Chapter 3

Datasets

In this chapter, the DAIEE and Engagement in the wild datasets will be described and analyzed in detail. More specifically, First, the data collection procedures will be described. After that, the annotation techniques and reliability of the labels will be discussed and analyzed. Finally, the relationship between engagement level and facial features will be analyzed with an unsupervised manner and baseline predictor models will be suggested for both datasets.

3.1 Daisee

In this section, we will summarize the main characteristics of the DAISEE dataset to create a better understanding of the main challenges in this problem. First, we will analyze the annotations and then OpenFace features. For the whole analysis in this section, a clean version of the training dataset will be used.

DAiSEE is a multi-label video classification dataset comprising of 9068 video snippets captured from 112 users for recognizing the user's affective states. The subjects in the videos are a group of Asian students between the age of 18-30 with 32 females and 80 males. There are 6 different locations where the videos are recorded such as dorm rooms, crowded lab spaces, library, etc, and 3 different illumination settings (light, dark and neutral). To simulate the e-learning environment, a custom application was created that presented a subject with 2 different videos (20 minutes total in length), one educational and one recreational to capture both focused and relaxed settings, which allow natural variations in users' engagement levels. Each video snippet is 10 seconds long and for each snippet, there are 4 different affective states labels as {Boredom, Confusion, Engagement, Frustration } and each label has a score as { Very Low, Low, High, Very High}. The labels are crowd annotated from 10 different annotators. To remove the unreliable annotators and their annotations, a weighted Cohen's κ (score between 0-1) with quadratic weights is given to each annotator and any annotator whose agreement is less than 0.5 is removed. Then the remaining annotations(which varies from 4-10 for each video snippet) are aggregated using Dawid-Skene vote algorithm [16].
In figures from 3.1 to 3.4, you can see the faces of students with varying levels of engagement, boredom, confusion, and frustration. By looking at the samples, it is not hard to say that facial features like eye gaze, head pose, or facial action units will be good features for this classification task.

However, there are some drawbacks to this dataset. First of all the dataset contains only visual information of Indian students. There is no cultural diversity

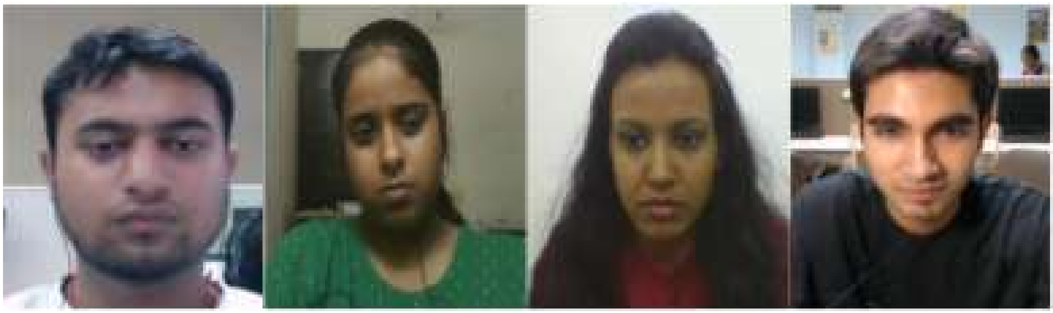


Figure 3.1. Samples from daisee dataset with engagement level varying from very low (left) to very high right [26].

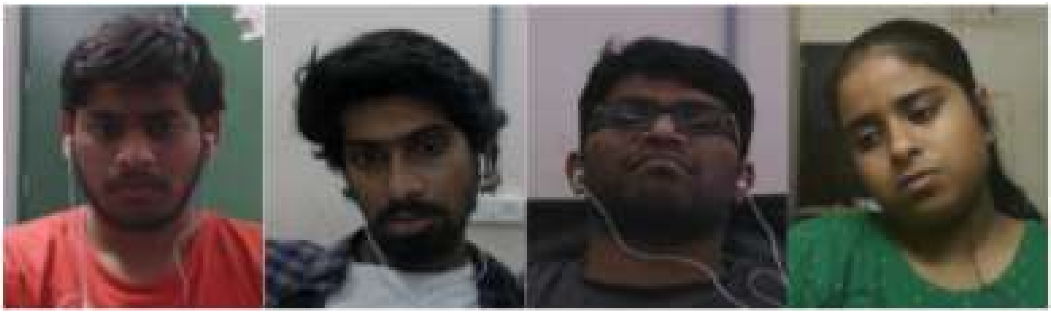


Figure 3.2. Samples from daisee dataset with boredom level varying from very low (left) to very high right [26].

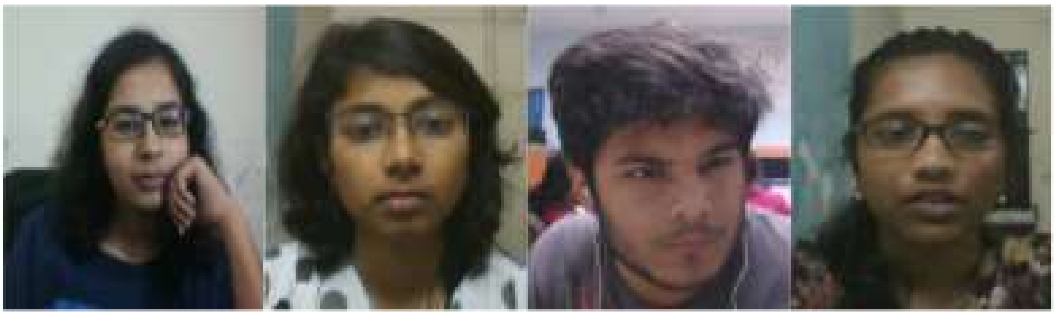


Figure 3.3. Samples from daisee dataset with confusion level varying from very low (left) to very high right [26].

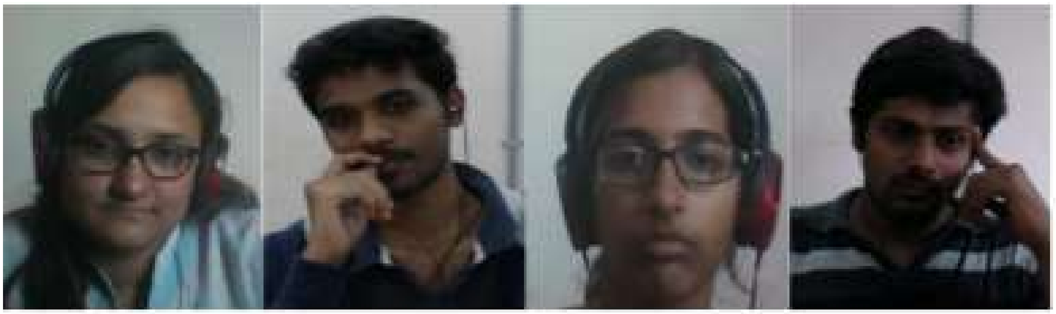


Figure 3.4. Samples from daisee dataset with frustration level varying from very low (left) to very high right [26].

and this can be a problem when a model trained with this dataset is used for European students for example. Also, the number of male and female students is highly unbalanced and this may cause a lack of accuracy for female students. Another limitation of this dataset is the ambiguity in labeling the frames with appropriate engagement levels. In crowdsourcing, ambiguity in labeling frequently occurs due to not having a clear guideline for mapping facial indicators to different affective states or engagement levels of the online learners.

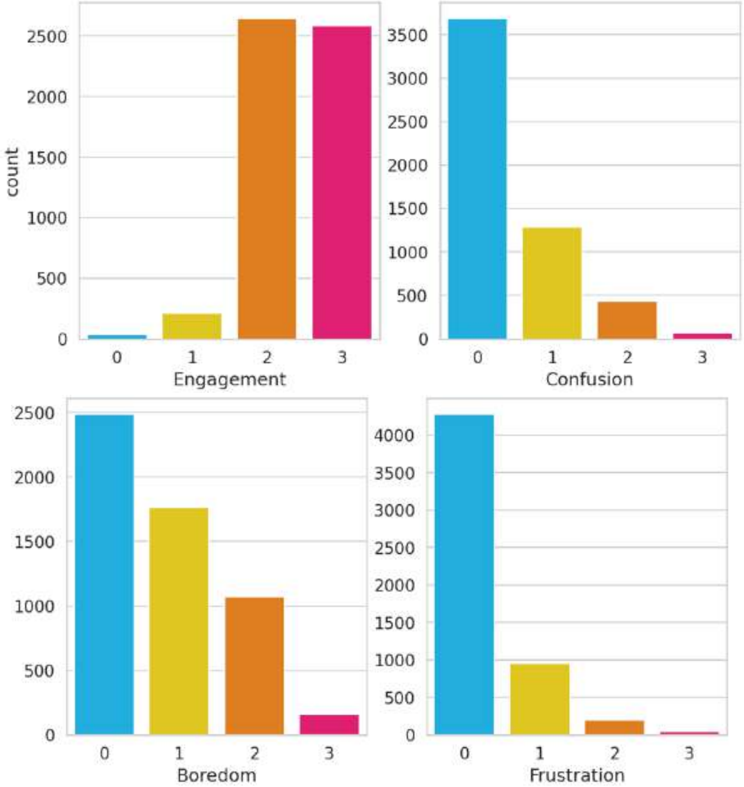


Figure 3.5. The histogram of scores for each label category in Daisee dataset

3.1.1 Annotations

As mentioned before, the DAISEE dataset has 4 labels per video clip which are, *Engagement, Boredom, Confusion, and Frustration.* In the training dataset, there are 70 students and 5478 clips each 10 seconds long. The minimum, maximum and average number of clips per student is $1, 142, 78$ respectively. Bellow, you can see the other summary statistics about the number of clips per student.

- mean: 78
- std: 36
- min: 1
- 25%: 47
- 50%: 86

- 75%: 109
- max: 142

During the recording process of the dataset, the students are recorded for approximately 20 minutes which refers to 120 clips per student. In the appendix, you can see figure 6.1, which shows the engagement levels of 22 students having more than 100 clips (more than 16 minutes of recording) as a single time series. Intuitively, one would expect that the engagement of a student will start high and decrease slowly because of concentration loss, tiredness, etc. Maybe 16-23 minutes is a short period of time to observe tiredness or concentration loss but still, we would expect a smooth graph. However for all labels *Engagement, Boredom, Confusion, Frustration* in figures 6.1 to 3.4, we observe label score changes very frequently almost in all students. The main reason behind this can be the crowdsource annotation of video clips. Since the annotators are only seeing the 10 second long clips, they do not have prior knowledge about the previous clips of the same recording. Moreover, we can also claim that annotators are not very good at distinguishing engagement levels that are one step ahead of each other. For instance, in the graphs of Students 1, 10, and 18 in figure 6.1, the instant movements between engagement levels 2 and 3 are clearly observable. The reason for that is because annotators are not able to distinguish whether the engagement level is 2 or 3 and the majority vote for these clips is probably not better than a random guess between scores 2 and 3. This behavior of annotators is pretty natural since these engagement level labels are very subjective especially when deciding on the labels that are one step away from each other. Unfortunately, the resulting time series of labels for each student is not a good reflection of the reality since the label scores are changing very frequently in most of cases.

For each clip in the dataset, there are scores from $0 - 3$ for all four labels. As you can see from the histograms in figure 3.5, the score labels are highly imbalanced for all four categories. Especially for Engagement label, there are so few samples from level 0 and level 1 and this makes very hard to correctly classify low level engagement scores as seen in [2, 37].By only looking at the histograms, it is not hard to see the correlation between labels.Engagement level is negatively correlated with all the other labels as show in figure 3.6. The highest absolute correlation is between Engagement and Boredom labels with a score of -0.42. This allows us to train our model with Boredom labels and then fine-tune with engagement labels. By this way we can propose alternative solution approach to handle miss classification of low level engagement because the boredom labels are more balanced compared to engagement labels.

3.1.2 Survey

Since the reliability of the labels are questionable, we decided to conduct a survey to measure the human performance on the Daisee dataset. To do that, we selected 60 random samples from the training dataset. Number of samples for each engagement label 0,1,2,3 are 16,14,17,13 respectively. The survey is created using google survey. Participants watch the each video with duration 10 seconds and then label the engagement level with one of the labels from set $\{0, 1, 2, 3\}$. In total 15 participants joined the survey and the final labels are created by majority voting all the votes from participants. All the participants are European and their age vary between 22 to 45. In figure 3.7, the classification statistics and confusion matrix are present. Even the number of participants attended to survey is very low, the accuracy of the

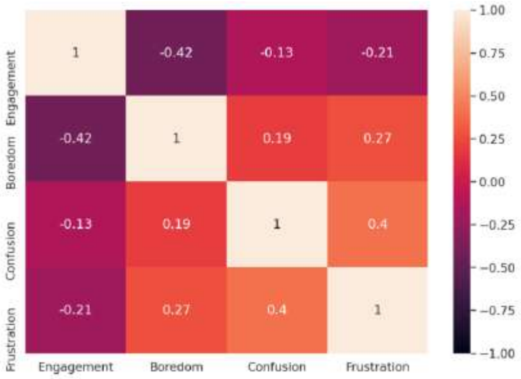
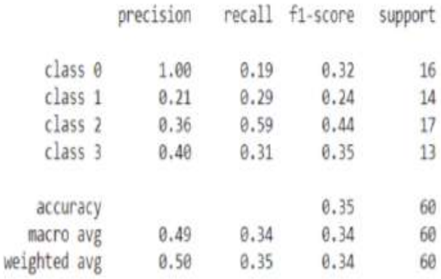


Figure 3.6. The heatmap of Pearson correlation scores for each pair of label categories

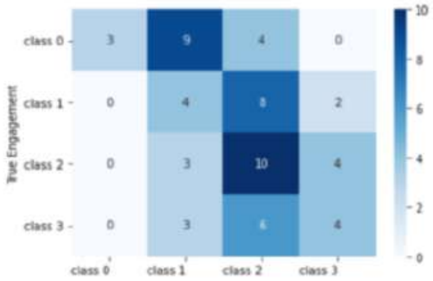


```
              precision    recall  f1-score   support

     class 0       1.00      0.19      0.32        16
     class 1       0.21      0.29      0.24        14
     class 2       0.36      0.59      0.44        17
     class 3       0.40      0.31      0.35        13

    accuracy                           0.35        60
   macro avg       0.49      0.34      0.34        60
weighted avg       0.50      0.35      0.34        60
```

(a)



(b)

Figure 3.7. (a) Survey statistics (b) Survey confusion matrix

majority votes are only 35%. This shows that the perceived engagement levels are very different and it is very hard to create a consensus on labels. We can say that since all the students in the videos are south Asian, participants of the survey are not good at judging the engagement levels. There also some inferences we can make by looking at figure 3.7. First, as seen in b, people are hesitant to put extreme labels (0 and 3) to video clips, especially label 0. Second, participants predict the engagement level 2 with highest f1 score and engagement level 1 with the lowest f1 score. Third, classification task is not suitable for engagement detection. The confusion matrix shows that participants labeled majority of zero engagement videos as one and one and three engagement videos as two. However, accuracy score is not considering the distance between classes and human predictions are over penalized in this case.

3.1.3 Analysis on Eye Gaze and Head Pose related Features

Since the main challenge is the small number of samples, increasing the dimension of the feature space may result in loss of generalization ability. To handle this problem, we can use some dimensionality reduction techniques such as Principle Component Analysis (PCA) [42].

In 3.7, we showed that the annotations are not reflecting the reality for all samples. So, trusting the supervising ability of the labels may result in conflicting results in real-life applications. To better observe the separability of our feature space, we first aggregated sequential information in each video clip and then applied PCA to reduce dimension. In figure 3.8, you can see the overall pipeline for this process. First, eye gaze and head pose related features are extracted from each video snippet resulting in a feature matrix with size $30x12$ where 30 is the number of frames and 12 is the number of head pose and eye gaze features. Then for each feature, we calculated the following 10 statics to aggregate frame level sequential information; *{mean, variance, standard deviation, minimum, maximum, mean and variance of the absolute fourier transform spectrum and top 3 fourier coefficients of the one-dimensional discrete Fourier Transform.}*[15]. The resulting $12 \times 10 = 120$ dimensional samples are concatenated for n different video snippets and the resulting matrix $M_{n\times 120}$ is fed into PCA for dimensionality reduction.

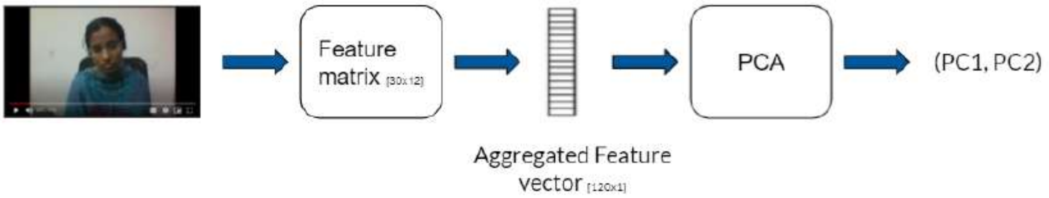
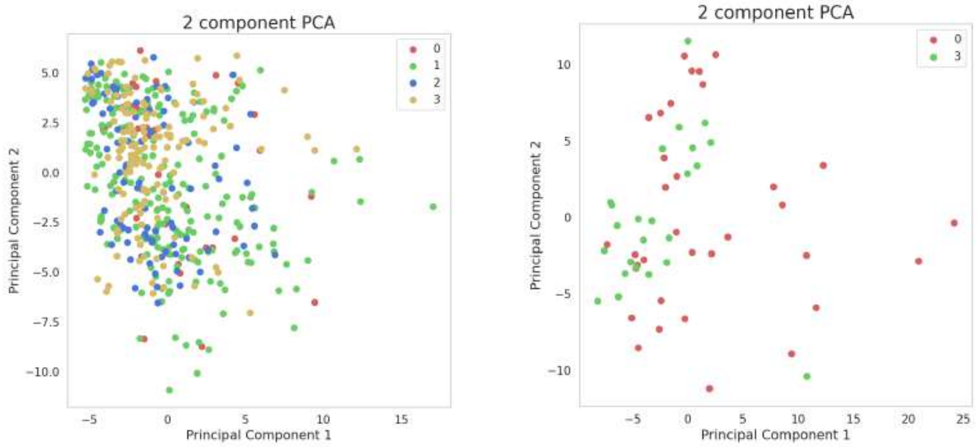


Figure 3.8. The pipeline for extracting principle components from each video snippet.(from left to right) First, eye gaze and head pose feature extraction. Second, aggregation of feature sequences with standard statistics, and frequency domain properties. Third, dimension reduction with PCA.

In figure 3.9, you can see the principle component graphs for 2 different subsets of the training dataset. Since the dataset is highly imbalanced, In (a) we took all samples with labels 0 and 1 and as oppose to that selected random samples with labels 2 and 3. The resulting subset contains 34, 214, 97, 146 samples from all labels $0 - 3$ respectively. As seen, there is no clear separation between data points in 2D. This is not surprising because these two principle components are representing only 39% of the total variance. In order to see the effectiveness of eye gaze and head pose features, we decided to simplify the problem and introduce the subset used in (b). Here we took all 34 zero-labeled samples in the training set and then randomly select 29 samples with label three. Since we are considering only the two extreme label sets, eye gaze and head pose features should be more effective in spotting the difference between labels and the feature space should be more separable. In (b), we observe a similar behavior to our expectations. Even though the samples are not clearly separated in the 2D principal component plot (since they are representing 49% of the total variance), we can say that the inner variance of samples with high engagement is lower compared to the samples with low engagement and high engagement samples are more accumulated in the are where $PC2 \geq -5$ *and* $PC1 \leq 3$. On the other hand samples with low engagement are spread around the feature space.

3.1.4 SVM Baseline

With the pipeline shown in figure 3.8, we can use a Support Vector Machine (SVM) [13], to create a baseline model. After extracting features with OpenFace, we used tsfresh [15] package to compute feature aggregations. Then the PCA output and SVM models are created with scikit-learn [43] package. In the table 3.1.4, you can see the accuracy scores for the different number of PCA components.



(a) (b)

Figure 3.9. (a) Principle component plot for 491 video clips sampled from the training dataset containing samples from all 4 labels. (b) Principle component plot for 63 video clips sampled from the training dataset containing samples from only 0 and 3 labels.

# components	accuracy
2	0.4958
5	0.5101
10	0.5288
20	0.5243

3.2 Engagement in the wild

In this section, we will summarize the main characteristics of Engagement in the wild dataset. Similar to 3.1, first the annotations and then OpenFace features will be analyzed.

Engagement in the wild is the sub-task in the Emotion Recognition in the Wild (EmotiW 2018) challenge [20]. The dataset is pretty similar to Daisee. First, the subject is recorded while watching the stimulus video around 5 minutes long. Then the subject is given 15 seconds to talk about the video and asked whether it was engaging or if they find it interesting or any comments/suggestions on how the video could have been made more engaging? The content of the selected videos is, *Learn the Korean Language in 5 minutes*, *a pictorial video (Tips to learn faster)* and *How to write a research paper*. The dataset has 78 subjects (25 female and 53 male) in total. The age range of the subjects is 19-27 years. A total of 195 videos are collected, each approximately 5 minutes long. The dataset is collected in the unconstrained environment i.e. at different locations such as computer labs, hostel rooms, open ground, etc. [34]. The videos are captured via a Skype video call and it was made sure that the subject was not disturbed by the Skype recording. The annotations are made by a team of 5 and the engagement of each subject is scored from $0-3$, the same as in the Daisee dataset. The distribution of videos in the dataset is as follows 9 videos belong to level 0, 53 for level 1, 82 for level 2, and 50 for level 3. The annotator reliability is assessed with a weighted Cohen's κ coefficient similar to the Daisee dataset and label votes of any annotator with coefficient 0.4 is removed. After that, the labels are averaged and rounded off to the nearest integer to give a ground truth engagement rating to the video.

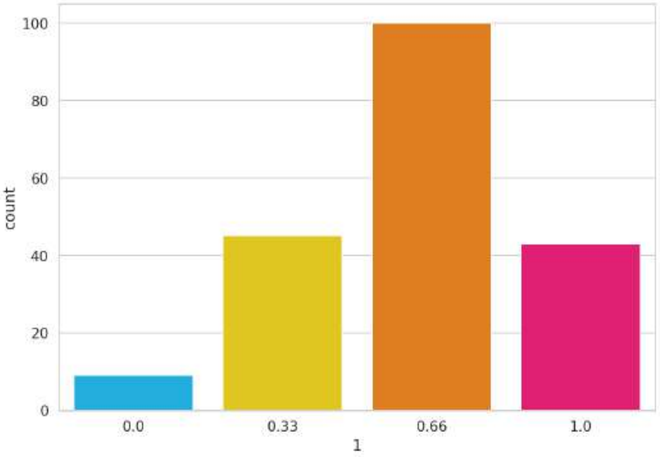


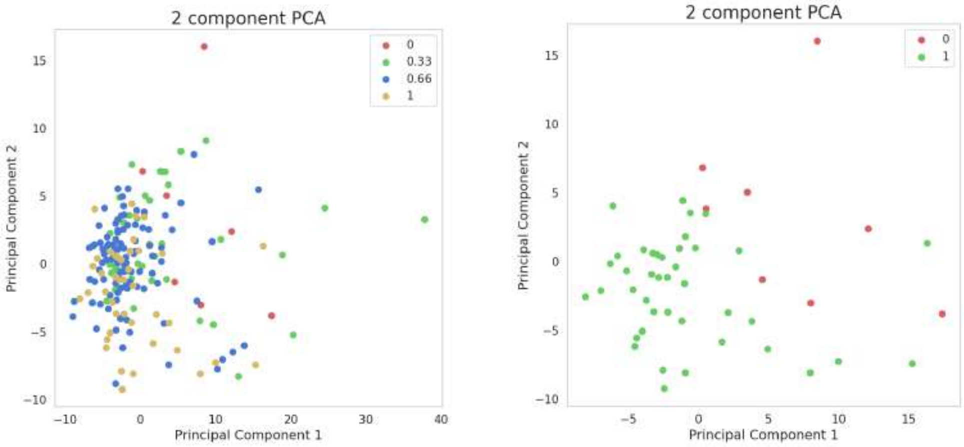
Figure 3.10. The histogram showing engagement level distribution for Engagement in the wild dataset.

3.2.1 Annotations

Engagement in the wild dataset is smaller in sample size compared to DAISEE. It includes 147 training and 48 validation videos. Each video is around 5 minutes and like the DAISEE dataset, some subjects have more than one video. While most of the subjects have only one video, there are 18 subjects with 5 or more videos. As you can see in appendix figure 6.5, most of the students show low engagement through the end of the video recording. The recording times vary from 25 minutes to 45 minutes with an engagement label per 5 minutes. Since it is easier to judge engagement for 5 minutes rather than 10 seconds and longer recordings allow more clear observations of change in engagement levels, we can say that Engagement in the wild dataset is more reliable compared to Daisee. Moreover, the samples are more balanced. In figure 3.10, you can see the histogram showing engagement label distribution for the whole dataset.

3.2.2 Analysis on Eye Gaze and Head Pose related Features

The OpenFace features and aggregation statistics used for Engagement in the wild dataset are the same as described for the DAISEE Dataset. With these features, we used the same pipeline in figure 3.8 to extract principal components as seen in figure 3.11. Similar to figure 3.9, in (a) samples belonging to different engagement classes are not visually separable in 2D principal component space. However, in (b) the samples belonging to classes 0 and 1 are almost clearly separable. As seen, the samples belonging to class 1 are more likely to have lower values of both principal components and the ones with class 1 are more likely to have the higher values of both principal components. When compared to the DAISEE experiment, it looks like the plot in (b) has a more clear separation. The reason for that can be long-term engagement detection is an easier problem and OpenFace features are more effective when feature sequences are longer.



(a) (b)

Figure 3.11. (a) Principle component plot for 195 video clips (combination of train and test sets of Engagement in the wild dataset) containing samples from all 4 labels. (b) Principle component plot for 62 video clips sampled from the combination of train and test sets of Engagement in the wild dataset containing samples from only 0 and 1 labels.

3.2.3 Linear Regression Baseline

Similar to what we have done with DAISEE dataset classification with SVM, now we will show the performance of PCA features for the different number of components. This time we will use a Linear regression model since Engagement in the wild dataset is used for regression tasks. In the table 3.2.3, you can see the MSE errors for the test set of Engagement in the wild dataset. The lowest achieved MSE is 0.0958 which can be a simple baseline to compare with deep learning methods.

# components	MSE
2	0.1064
5	0.1036
20	0.0981
40	0.0958
45	0.0970

Although Engagement in the wild and Daisee datasets have many similarities they are also different in many aspects. First of all the duration of the input videos are very different from each other. In Daisee there is an engagement label for each 10-second video snippet while in Engagement in the wild there is a single label for each 5-minute video. This is an important difference because the features contributing to long-term engagement detection will be different than features contributing to short-term engagement detection. Secondly, there is no conversation with the subject after watching the content in the Daisee dataset. This makes Engagement in the wild annotations more reliable because the subject's own opinion is also considered. On the other hand, the Daisee dataset has more variety of annotations and in this way, we are able to learn more about the affective state of subjects. Moreover, in Daisee, subjects are recorded for longer times which allows us the observe the changes in affective states more easily. This two dataset has their own advantages and disadvantages in many aspects. A cross-dataset approach would be an interesting analysis since these datasets are potentially suitable to be used together.

Chapter 4

Model Design

In this chapter, the proposed deep learning model will be introduced. The main motivation in the design of this model is to show that deep learning-based methods significantly perform better than the naive baselines introduced in chapter 3 and improve the state-of-the-art results mentioned in chapter 2. The model is inspired from [34] and improved with various aggregation functions and training techniques.

4.1 Model Architecture

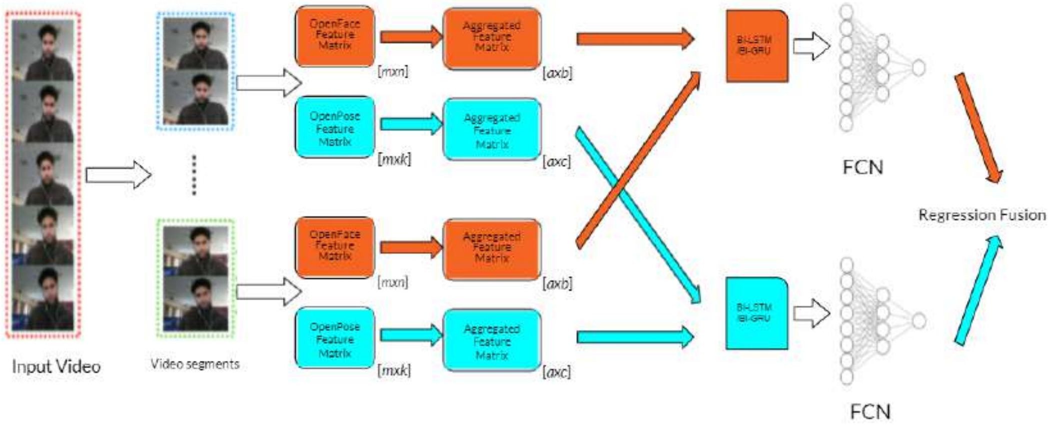


Figure 4.1. The General Architecture of the Model.

In figure 4.1, you can see the overall architecture of the model. This architecture is a backbone model we will use for all the experiments. There can be slight differences in some cases. For instance, the regression fusion step will be 4- class classification majority vote for the Daisee dataset. Also, we will evaluate OpenPose and OpenFace based models individually without making any fusion. This architecture visualizes the fundamental steps of the pipeline.

$$OpenFace(X^i_{hwcm}) = Y^i_{m\times n} \tag{4.1}$$

$$Aggregation(Y^i_{m\times n}) = Z^i_{a\times b} \tag{4.2}$$

$$LSTM(Z^i_{a\times b}) = T^i_v \tag{4.3}$$

$$MLP(T_v) = O^i \tag{4.4}$$

First the input videos with l number of frames are divided into video segments with a window size of m and with j overlapping frames where $1 \leq m \leq l, m \in \mathbb{Z}$ and $0 \leq j \leq m-1, j \in \mathbb{Z}$. Second, the video segments are passed to OpenFace and OpenPose tools for frame-level feature extraction. OpenFace and OpenPose generates n and $m$different features respectively for all m frames. The resulting matrices with shapes $m \times n$ and $m \times k$ are aggregated by using a subset of the following functions, *{mean, variance, standard deviation, minimum, maximum, length, mean, and variance of the absolute Fourier transform spectrum and top 3 Fourier coefficients of the one-dimensional discrete Fourier Transform.}*[15] and an aggregation frame size of z where $z \leq m$. The aggregation process generates matrices with new shapes such that $a \leq m$, $b \geq n$ and $c \geq k$. Third, the aggregation matrices are fed into Bidirectional LSTM and Bidirectional GRU units for sequence modeling. Finally, a fully connected network is used for regression and classification tasks and the predictions are fused with weighted averaging for regression and majority voting for classification. The equations from 4.1 to 4.4 shows the same flow in functional form.

4.2 Feature Extraction

4.2.1 OpenFace

OpenFace provides many different facial features as described in chapter 2. However, only some of these features are related to the engagement level of a subject. In order to narrow down the feature space, we will only consider 29 features as done in [30, 63, 40, 51, 61] which are related to eye gaze, head pose, head rotation, and facial action units. The eye gaze-related features are, *gaze_0_x, gaze_0_y, gaze_0_z* which are eye gaze direction vectors in world coordinates for the left eye and *gaze_1_x, gaze_1_y, gaze_1_z* for the right eye in the image. The head pose related features are *pose_Tx, pose_Ty, pose_Tz* representing the location of the head with respect to the camera in millimeters (positive Z is away from the camera). *pose_Rx, pose_Ry, pose_Rz* indicates the rotation of the head in radians around x,y,z axes. This can be seen as pitch (Rx), yaw (Ry), and roll (Rz). The rotation is in world coordinates with the camera being the origin. Finally, we will use the following 17 facial action unit intensities varying in the range $0-5$ listed in 2.4 named as, *AU01_r, AU02_r, AU04_r, AU05_r, AU06_r, AU07_r, AU09_r, AU10_r, AU12_r, AU14_r, AU15_r, AU17_r, AU20_r, AU23_r, AU25_r, AU26_r, AU45_r*. For each clip, we have an input matrix with size $m \times n$ which indicates the number of image frames captured from the video and the feature values mentioned above for each image frame.

4.3 Feature Aggregation

The common approach described in chapter 2 and also shown in figure 2.9 is aggregating information from multiple frames selected by a sliding window approach. The most common aggregation statistics are minimum, maximum, and variance/standard deviation of the signal in a window. Including more complex hand-crafted features can add valuable information but each additional hand-crafted feature increases the feature dimension as much as the number of standard features it is applied. Different from previous works, in addition to standard statistics like we also considered the spectral centroid (mean) and variance of the absolute Fourier transform spectrum and the Fourier coefficients of the one-dimensional discrete Fourier Transform of video snippet. In the experiments, it is observed that best performances are achieved by using only common aggregation statistics such as minimum, maximum, and variance. In equations 4.5 to 4.10, you can see the aggregation operation in functional form. More specifically, a frame size of z is selected and each feature of consecutive z frames are aggregated with *minimum, maximum and variance* functions and concatenated in-frame direction. Finally, the resulting matrices from 3 functions concatenated in feature direction and the resulting matrix $Z^i_{a \times b}$ with a frames and b features is achieved for video snipped i.

$$\oplus_{j=0}^{a}(Min(Y^{ij}_{z \times n})) = \alpha^i_{a \times n} \; \forall j = 0, ..., a \; and \; a = \frac{m}{z} \tag{4.5}$$

$$\oplus_{j=0}^{a}(Max(Y^{ij}_{z \times n})) = \beta^i_{a \times n} \; \forall j = 0, ..., a \; and \; a = \frac{m}{z} \tag{4.6}$$

$$\oplus_{j=0}^{a}(Var(Y^{ij}_{z \times n})) = \gamma^i_{a \times n} \; \forall j = 0, ..., a \; and \; a = \frac{m}{z} \tag{4.7}$$

$$\oplus_{j=0}^{a}(Std(Y^{ij}_{z \times n})) = \tau^i_{a \times n} \; \forall j = 0, ..., a \; and \; a = \frac{m}{z} \tag{4.8}$$

$$\oplus_{j=0}^{a}(Mean(Y^{ij}_{z \times n})) = \theta^i_{a \times n} \; \forall j = 0, ..., a \; and \; a = \frac{m}{z} \tag{4.9}$$

$$\alpha^i_{a \times n} \oplus \beta^i_{a \times n} \oplus \gamma^i_{a \times n} \oplus \tau^i_{a \times n} \oplus \theta^i_{a \times n} = Z^i_{a \times b} \tag{4.10}$$

4.4 Sequence Model

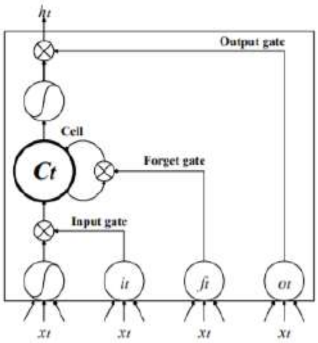


Figure 4.2. Long-Short Term Memory cell [64]

4.4.1 BI-LSTM

For the sequence model, we used a bi-directional LSTM model following with an MLP for classification and regression tasks. In figure 4.2, you can see the illustration of an LSTM cell. Given an input video sequence $Z = (z_1, ..., z_a)$ and $z_a \in \mathbb{R}$, the hidden sate for each frame is calculated as shown in equations from 4.11 to 4.15. The σ refers to sigmoid activation function and $i, f, o \ and \ c$ are respectively the input gate, forget gate, output gate and cell activation vectors, all of which are the same size as the hidden vector h.

$$
\begin{aligned}
i_t &= \sigma(W_{zi}x_t + W_{hi}h_{t-1}) + W_{ci}c_{t-1} + b_i) && (4.11)\\
f_t &= \sigma(W_{zf}x_t + W_{hf}h_{t-1}) + W_{cf}c_{t-1} + b_f) && (4.12)\\
c_t &= f_t ct - 1 + i_t tanh(W_{zc}x_t + W_{hc}h_{t-1} + b_c) && (4.13)\\
o_t &= \sigma(W_{zo}x_t + W_{ho}h_{t-1}) + W_{co}c_{t-1} + b_o) && (4.14)\\
h_t &= o_t tanh(c_t) && (4.15)
\end{aligned}
$$

The Bi-directional LSTM model with MLP is shown in figure 4.3. The Figure shows two unfolded LSTM layers representing forward and backward sequence flow for T time steps. For each video snippet, the hidden state vectors from beginning to end and end to beginning are computed and then concatenated. The resulting vector is then passed to an MLP for classification or regression of engagement for each video snippet.

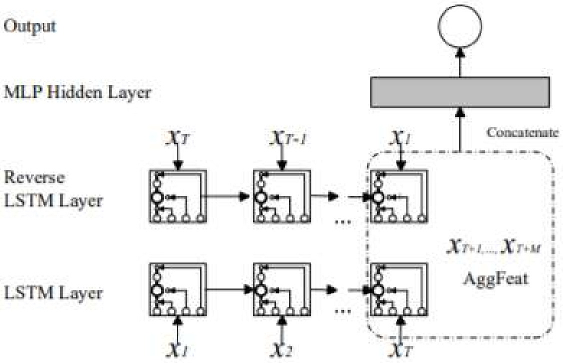


Figure 4.3. Bi-LSTM model with MLP [64]

We believe that the proposed architecture is a good fit for this problem for many reasons. First, the usage of OpenFace and OpenPose features. Many works in the literature are considering pre-trained convolutional neural networks such as ResNet [27], VGG [47] and Inception [49] for frame-level feature extraction. These networks are trained on large-scale datasets like ImageNet [36] which is a dataset containing images from various domains. However, OpenPose and OpenFace use many different pre-trained networks that are trained on datasets for specific tasks such as gaze estimation, body pose estimation, and facial action recognition. In other words, OpenFace and OpenPose provide more tailored information for the engagement detection task. Second, feature aggregation. According to the best of our knowledge, none of the works using the DAISEE dataset consider feature aggregation [30, 50, 18, 37, 39, 45]. Feature aggregation requires manual selection of aggregation functions and they can be called hand-crafted features. Using hand-crafted features is against the spirit of using deep neural networks since one of the most fundamental benefits of deep learning is automatic feature extraction and selection. However, as described in chapter 3, the datasets are considerably small and imbalanced. These obstacles make it very hard to design a system that can automatically learn the aggregation functions. For this reason, providing an effective set of aggregation functions increases the performance of the sequence models in the training process. Finally, the use of sequence models is fundamental to model the temporal aspect of the video data. The use of sequence models are very common in most the works in this domain.

Chapter 5

Experiments and Results

The main motivation of this thesis is to extract and analyze facial features to show the relationship between those features and the engagement level of the subjects. Until now, the model pipeline and datasets for the engagement detection task are described. In this chapter, we will describe the training procedure, experiments, and results for both Daisee and Engagement in the wild datasets. Both datasets come with advantages and challenges. In order to achieve the best performance on both datasets, different training approaches were applied and experiments were reported under different scenarios. For the DAISEE dataset, the main challenge is the imbalanced samples as mentioned in chapter 3. To deal with this, frame aggregation and a 2-step training approach are proposed different from previous works. The proposed model is also tested on the survey dataset described in 3 and machine performance is compared with human performance. For Engagement in the wild dataset, the existing architecture in [61] simplified and tuned. In addition to that, a novel training procedure is proposed with triplet loss [5]. Moreover, the effect of facial features is analyzed and compared. The contribution of each facial feature is computed with integrated gradients as described in chapter 2. Finally, the best-performing model is tested on real-life scenarios.

5.1 Daisee Dataset

5.1.1 Training

In chapter 3, we described the challenges of Daisee dataset in detail. In this section we will describe 3 different training strategies which are;

- Training without frame aggregation.
- Training with frame aggregation.
- Fine-tuning best model with pre-trained boredom weights.

When training without aggregation, we will pass feature sequences directly to Bi-LSTM units without any manipulation. For all the strategies mentioned above, only OpenFace features described in section 4.2.1 are used.

Training Parameters

The training parameters are selected with hyperparameter tuning on the model without frame aggregation. After finding the best parameter values in the search space with the grid search method, same parameter values are used for models with frame aggregation. The parameter values are;

- LSTM parameters:

  number of hidden units: 256

  number of layers: 2

- MLP parameters:

  num neurons 1st layer: 128

  num neurons 2nd layer: 32

  num neurons 3rd layer: 4

- Training Parameters:

  Batch size: 64

  Learning rate: 0.0005

  Number of epochs: 30

  Dropout probability : 0.2

5.1.2 Results

In table 5.1, the 5-fold cross-validation accuracy results of four different inputs are presented. To compare with the previous work, we combined the train and validation set proposed by the dataset providers. The best performance is achieved by aggregating 10 consecutive frames and calculating the following statistics *mean, variance, maximum, minimum and standard deviation* for each feature sequence. Like training parameters, the aggregation functions are also considered as hyperparameters and selected according to their contribution to the accuracy. In table 5.2, you can see the recall and f1 scores of low engagement level samples for four different inputs. By looking at the table we can say that class 0 recall and f1 scores are increasing until aggregation with 10 consecutive frames. Moreover, for class 1 recall and f1 scores, they increase as the aggregated frame number increases. So we can claim that information coming from aggregated functions helps identification of low-level engagement level samples even there are so few in numbers.

Input type	average cv acc	best cv acc	best cv loss
no aggregation	0.5980	0. 6343	0.7827
aggregation num frames =5	0.6182	0. 6519	0.7415
aggregation num frames =10	**0.6266**	0. 6674	0.7013
aggregation num frames = 15	0.6182	0. 6512	0.7155

Table 5.1. accuracy and cross entropy losses for no aggregation and aggregation inputs.

As suggested in section 3.1.1, Now we will train the model with boredom labels and then fine-tune with engagement labels. First, the dataset used in previous experiments (table 5.1) was randomly divided into two. The first part of the dataset is used with 5-fold cross-validation to train with boredom labels. The input feature sequences are aggregated per 10 frames since this is the best performing input format as shown in 5.1. The same hyperparameters are used to train with boredom labels.

Then for fine-tuning with engagement labels, the second part of the dataset is used again with 5-fold cross-validation. Moreover, the learning rate is reduced to 0.0001. In this way, the average 5-fold cross-validation score is increased to 0.6442, which is better than the current SOTA by [4]. This suggested model is called *Bi-lstm 2stage*. In figure 5.1 you can see 5-fold cross-validation scores of various models on the DAISEE dataset. The figure shows highest accuracy is achieved by the proposed Bi-LSTM 2stage method. According to the best of our knowledge, it is the only method using other labels of the DAISEE dataset.

In figure 5.2, The train and validation loss curves are shown for all 5 folds off cross-validation. The steps on the x-axis represent each gradient update so the number of steps per epoch is the number of samples dived by the batch size. In this case, there are around 86 steps per epoch. The training error for all folds is decreasing and the model stats to over-fit around 9-10 epochs for all folds. The validation curve looks similar almost in all folds except the last one shown by the pink curve. For that fold, the cross-entropy loss is higher than the others.

Results on test set

There are very few works in literature providing the performance of the model on the test set. However, it is crucial to compare test set results to confirm the models are not over-fitting to the validation dataset. In figure 5.3, you can see confusion matrices for four different methods. The models with the highest accuracy are achieved by the Resnet-TCN and Resnet LSTM models. However, these models are only making high-level engagement predictions. This is counterintuitive because distinguishing between high and low levels of engagement is easier compared to distinguishing between two high-level engagement labels. Moreover, in chapter 3, we showed that there is no clear separation between samples having both high or both low-level engagement labels. On the other hand, the proposed Bi-LSTM 2 stage model and Resnet-TCN with weighted sampling and weighted loss have lower accuracy's but they also make low-level engagement predictions.

5.1.3 Results on Survey Data

In chapter 3, we showed that human performance on a subset of the DAISEE dataset is only 35% by providing the results of our survey. In figure 5.4, you can see the performance of Bi-LSTM 2 stage model on survey data. The accuracy of the model is only 28% which is slightly better than a random guess. Moreover, there are very few low engagement predictions. These results clearly show that even though the Bi-LSTM 2 stage model is able to achieve the highest average cross-validation accuracy on training, It fails dramatically on survey data. In fact, one shouldn't expect the model to perform better than human performance so the achieved low accuracy can be considered reasonable but the imbalanced predictions show the

Input type	class0 recall	class0 f1	class1 recall	class1 f1
no aggregation	0	0	0	0
aggregation frames 5	0.07	0. 13	0	0
aggregation frames 10	0.23	0.33	0.01	0.01
aggregation frames 15	0.14	0.24	0.03	0.05

Table 5.2. recall and f1 scores of class 0 and class 1 samples for no aggregation and aggregation inputs.

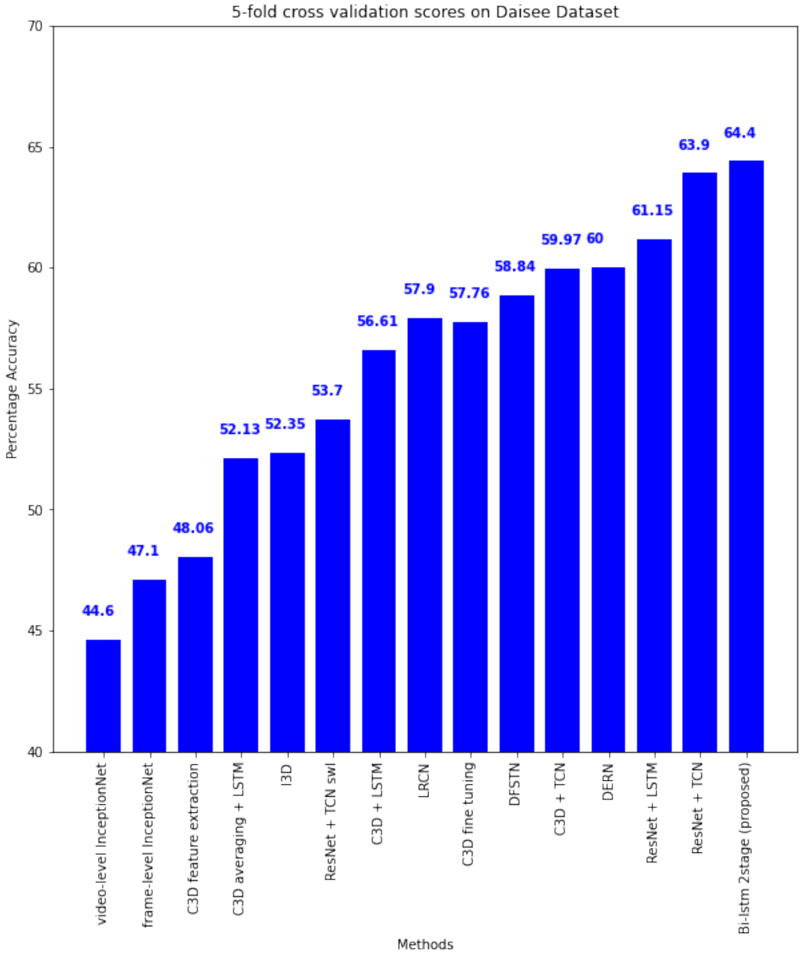
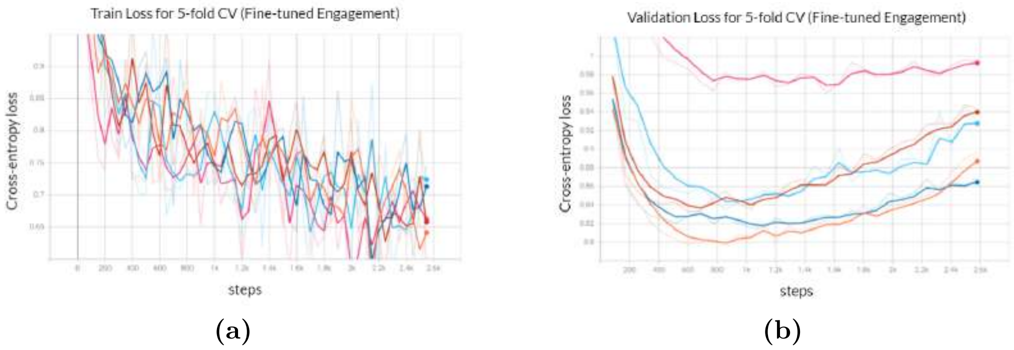
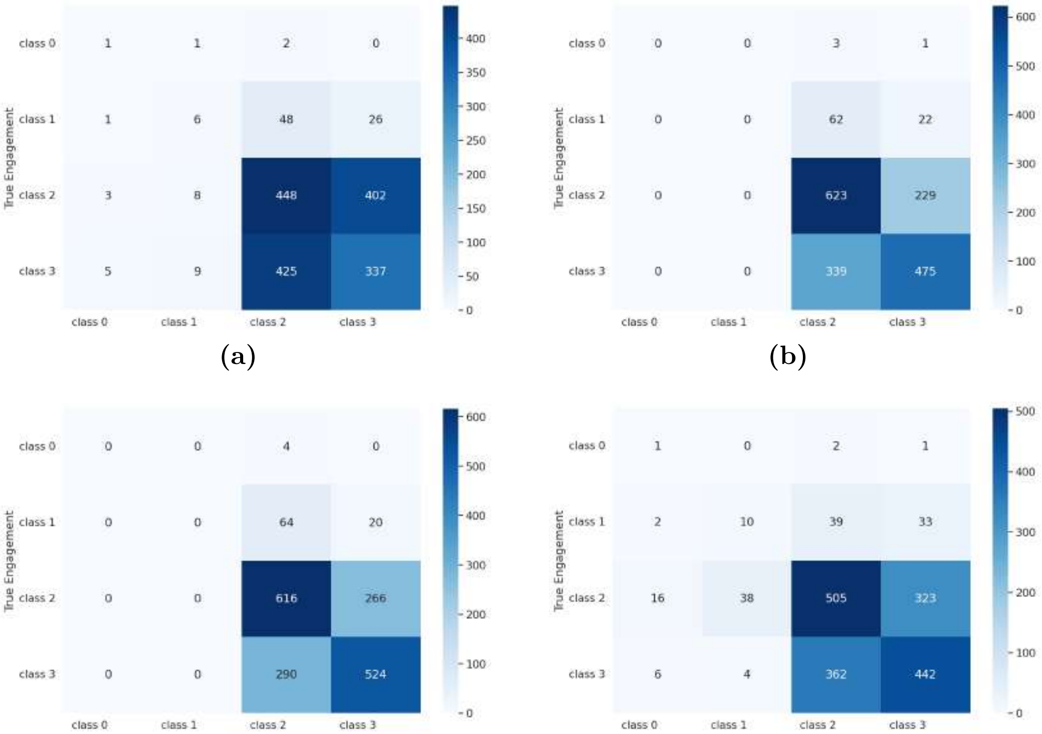


Figure 5.1. The performances of models in literature Bi-lstm 2stage is the proposed model. video-level InceptionNet [26], frame-level InceptionNet [26], C3D feature extraction [26], C3D averaging + LSTM [41],I3D [67], ResNet + TCN with sampling and weighted loss [4], C3D + LSTM [41], LRCN [22], C3D fine tuning [54], DFSTN [37], C3D + TCN [4], DERN [31], ResNet + LSTM [4], ResNet + TCN [4], Bi-lstm 2stage (proposed)



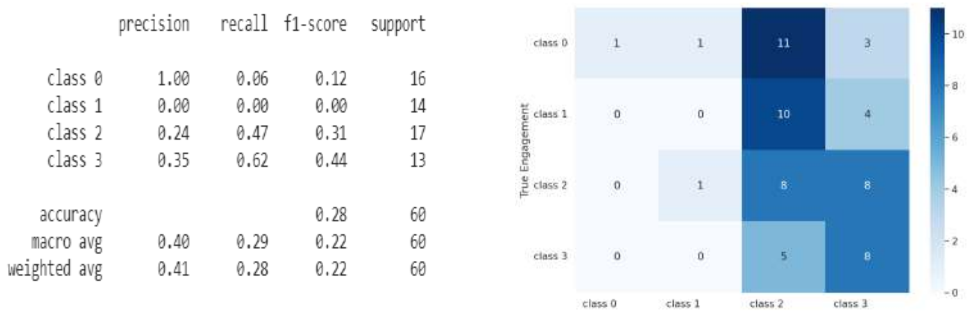
(a) (b)

Figure 5.2. (a) Training loss curves of engagement fine-tuning for all 5 five folds with 10 frame aggregation (b) Validation loss curves of engagement fine-tuning for all 5 five folds of cross validation with 10 frame aggregation



(a) (b) (c) (d)

Figure 5.3. (a) Bi-lstm 2 stage confusion matrix of test set. Accuracy= 47% (b) Resnet-LSTM [4] confusion matrix of test set. Accuracy=61% (c) Resnet-TCN [4] confusion matrix of test set. Accuracy=63% (d) Resnet-TCN with weighted sampling and weighted loss [4] confusion matrix of test set. Accuracy=53%



```
              precision    recall  f1-score   support

     class 0       1.00      0.06      0.12        16
     class 1       0.00      0.00      0.00        14
     class 2       0.24      0.47      0.31        17
     class 3       0.35      0.62      0.44        13

    accuracy                           0.28        60
   macro avg       0.40      0.29      0.22        60
weighted avg       0.41      0.28      0.22        60
```

(a) (b)

Figure 5.4. (a) classification statistics of Bi-lstm 2 stage model on survey data (b) Confusion matrix of Bi-lstm 2 stage model on survey data.

model fails the learn the relation between facial features and engagement level but only fits the noise in the data.

5.2 Engagement in the wild

In the previous section, we show that the proposed Bi-LSTM 2 stage model is able to achieve the state-of-the-art result on a 5-fold cross-validation dataset but fails to predict low-level engagement classes on the survey dataset. In order to create a more robust model that can detect both low and high levels of engagement levels, a similar model is trained on Engagement in the wild dataset. Engagement in the wild dataset has some advantages compared to the DAISEE dataset. First, the labels are more balanced compared to the DAISEE dataset. Second, engagement detection is considered a regression problem, and MSE loss is used in training. MSE is more suitable than categorical cross-entropy loss to engagement detection tasks since it considers the distance between engagement labels. Finally, the engagement in the wild dataset, the video duration per label is around 5 minutes. This is very long compared to the DAISEE dataset which is only 10 seconds. This makes the labels more reliable since longer video includes more facial expression clues indicating engagement level. To this end, the proposed model and training procedure for the engagement regression task on engagement in the wild dataset will be present in this section. Moreover as a novel approach, we will introduce a triplet loss to measure the similarity between input videos.

5.2.1 MSE Loss Experiments

Training

The model is similar to the one used for the DAISEE dataset but some parameters are different. Since Engagement in the wild dataset videos are much longer and duration is variable, we set a fixed sequence length of 150 and then aggregated the different number of frames for each video. The sequence lengths,100,150, and 200 are tried to be in line with Daisee experiments and observed that 150 perform the best. Since the number of aggregated frames varies from sample to sample, we also considered this as an aggregation function and used *Maximum, Minimum, Variance, Frame number* as aggregation functions. Also, the number of LSTM hidden units is increased to 512, since now there are more parameters with the increasing number of steps. Finally, the batch size is reduced to 8, because of the small sample size and the number of epochs is increased to 350.

Results

In the table, 5.3, you can see the performance of the model under different combinations of feature spaces. The best performance is achieved by using all the features except head rotation.

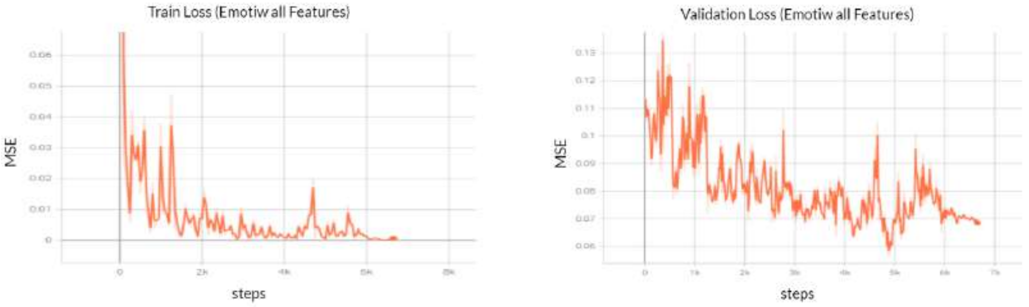
Eye Gaze	Head Pose	Head Rotation	Action Units	MSE Score
✓	✗	✗	✗	0.09135
✗	✓	✗	✗	0.06872
✓	✗	✓	✗	0.09412
✓	✓	✗	✗	0.05814
✓	✓	✓	✗	0.06516
✓	✓	✗	✓	**0.05395**
✓	✓	✓	✓	0.05411

Table 5.3. Model performance under different combination of feature spaces

It is possible to make some claims according to these results. First, the decrease in MSE score when eye gaze and head pose features are considered together. As seen from the first and third row of the table, the MSE score increased when head rotation features are considered together with eye gaze features. The same behavior is also visible in the fourth and fifth rows of the table. Eye gaze and head pose features perform better than Eye gaze, head pose, and head rotation features. Also in the last two rows, we see that all features without head rotation have slightly higher MSE score compared to all features together. These results may indicate a correlation between eye gaze and head rotation features. This makes intuitive sense because when a subject rotates his/her head, the eye gaze will also change so the head rotation information can be captured from the eye gaze vector. However, eye gaze information is more valuable since the subject can look away without rotating the head but only by moving the eyes. Second, the head pose itself performs well and dramatically improves performance compared to cases not including the head pose. This shows head pose is an important feature to detect engagement level.

In figure 5.5, you can see the training and validation MSE losses for 350 epochs (approximately 7000 steps). The curves are pretty sharp compared to the DAISEE experiments because of the small batch size. The training loss converges to 0 very fast but we don't observe an increase in validatin loss for a very long time. In fact, it reaches MSE 0.05395 around 250 epochs.

In figure 5.6, you can see the predictions and ground truth labels for the test set sample of engagement in the wild dataset. The test videos are grouped according to their ground truth labels for visual convenience. The green points show the ground truth labels of the samples and the red points show the predicted engagement level for the corresponding video sample. The yellow line is the average engagement level of the predictions in each group. For each group, we see that the average predictions are different. This shows that the model is able to make predictions for all four levels of engagement. However, for engagement level 0, the predictions are higher than expected and in fact closer to engagement level 0.33. Similar behavior is also visible for engagement level 1 and in this case, the predictions are below expected and closer



(a) (b)

Figure 5.5. (a) Training loss curve of model with all features for Engagement in the wild dataset. (b) Validation loss curve of model with all features for Engagement in the wild dataset

to engagement level 0.66. This shows that the proposed model still suffers from detecting extreme labels. The model performs the best with samples having a ground truth engagement level 0.66. Here we see that the average predicted engagement is very close to 0.66. However, there are 5 samples out of range $0.4 - 0.8$ which can be considered as outliers. Overall we can say that the model is able to distinguish between four engagement levels but with some variability. The performance of the proposed model is compared with other studies on literature in figure 5.7. we can see that the proposed model performs better than the current state of the art with a margin on the test set provided by engagement in the wild dataset.

5.2.2 MSE & Triplet Loss Experiments

Now, we will combine the MSE loss with Triplet loss. Triplet loss is a loss function where a baseline (anchor) sample is compared with a positive and negative sample. The distance between anchor and positive sample is maximized and the distance between anchor and negative is minimized. In equation 5.1, the triplet loss function is defined. Where N is the batch size; d is the euclidean distance and $margin$ is a non negative margin representing the minimum difference between the positive and negative distances that is required for the loss to be 0.

$$\ell(a,p,n) = L = \{l_1, \ldots, l_N\}^\top, \quad l_i = \max\{d(a_i,p_i) - d(a_i,n_i) + \mathrm{margin}, 0\} \tag{5.1}$$

Training

In figure 5.8, the training procedure with triplet loss is illustrated. Each sample in a batch is considered as anchor input and for each anchor, a positive and a negative sample is randomly selected. In order to distinguish between low-level and high-level engagement samples, engagement level samples $0 - 0.33$ and $0.66 - 1$ are grouped together. Depending on the label of the anchor sample, a positive and a negative sample are selected randomly from these two groups. After that, the triplet loss is calculated from the hidden states of the last layer of Bi-LSTM units. Finally, it is summed with the MSE loss of anchor sample to form the new loss.

We believe that introducing triplet loss is suitable to improve the performance for some reasons. First of all, by grouping low-level and high-level engagement level

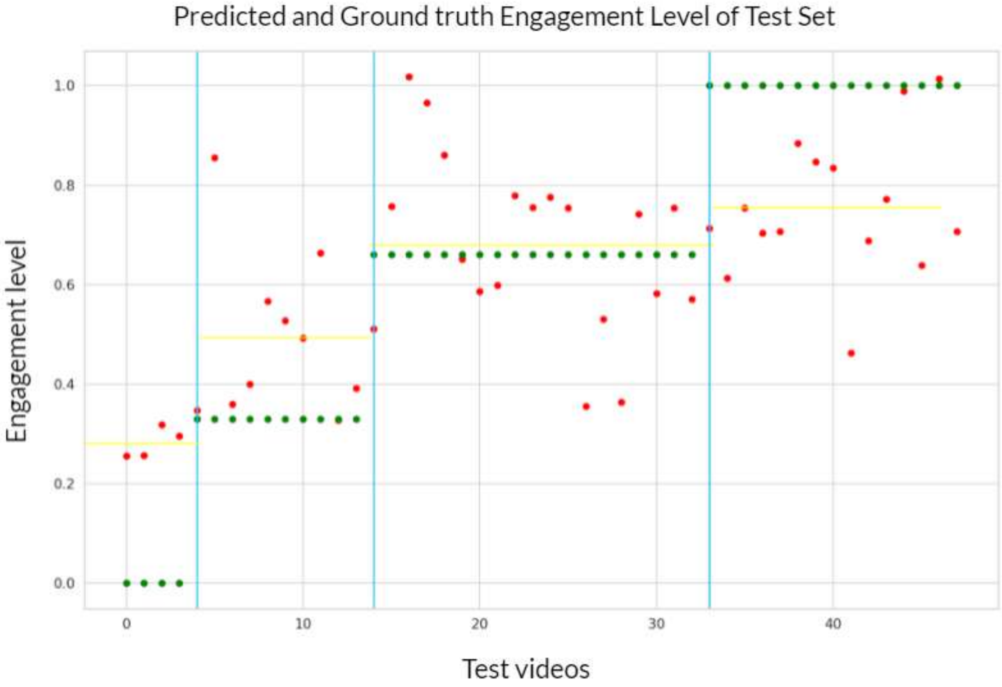


Figure 5.6. The predicted (red) and ground truth (green) engagement labels of Engagement in the wild dataset. The yellow line represents the average score of predicted label scores for each label class.

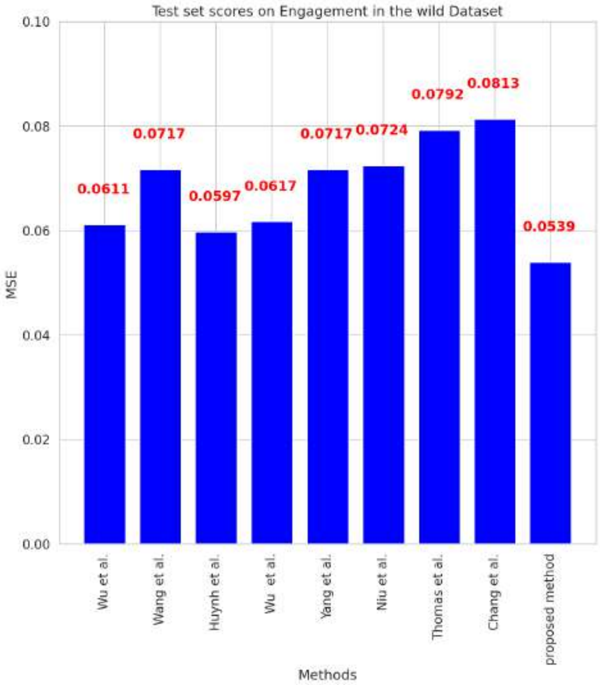


Figure 5.7. The performances of models in literature. *Wu et al.* [61] score: 0.061110, *Wang et al.* [58] score: 0.0717, *Huynh et al.* [52] score: 0.0597, *Wu et al.* [62] score: 0.061740, *Yang et al.* [63] score: 0.0717, *Niu et al.* [40] score: 0.0724, *Thomas et al.* [51] score: 0.0792, *Chang et al.* [12] score: 0.0813, *proposed method* score: 0.0539

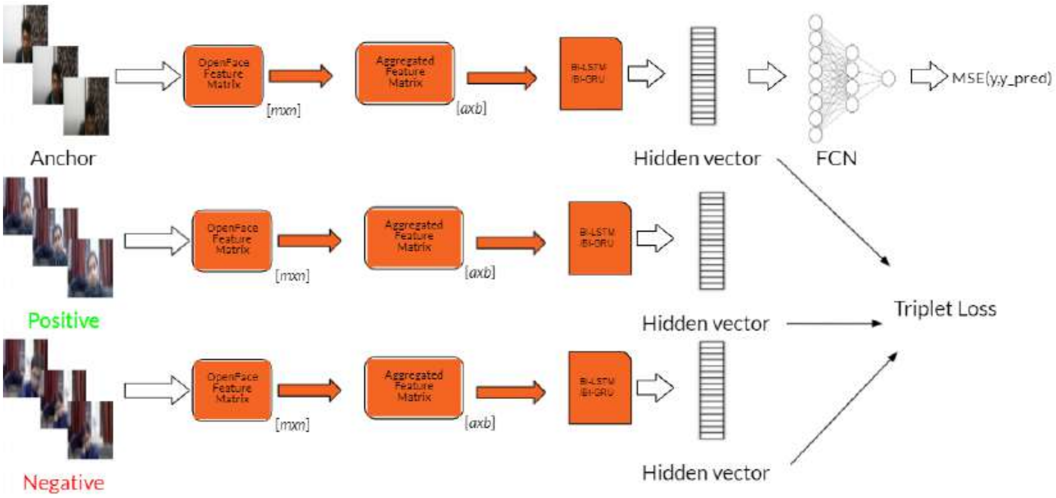


Figure 5.8. The architecture for Triplet Loss training.

samples together and reducing the number of classes, we believe we created more reliable classes. Moreover, by introducing the triplet loss, the problem became a multi-task learning problem. This will introduce an additional regularization and hopefully improve the over-fitting problem due to very small sample size. By this way, we better modeled the similaritydissimilarity of samples with triplet loss and still able to capture 4 different levels of engagement with regression loss.

Results

After experimenting with different subset of features as in table, 5.3, the model with MSE + triplet loss improved on the state-of-the art and achieved an MSE of 0.05011 on the test set by using all the features shown in table 5.3. In figure 5.9, you can see the predictions and ground truth labels for the test set sample of engagement in the wild dataset. Compared with 5.6, the within variance of predictions are lower. However, the predictions for engagement level 0 is pretty high, the reason for that is these samples are considered in the same group with engagement level 0.33 samples. Overal, we can say that introducing triplet loss reduced the within variance of predictions while preserving hierarchy between labels.

5.2.3 Feature Importance

In table 5.3, we showed that some features are more important than others for engagement detection with some comparative experiments. Now we will use the Integrated gradients technique [48] introduced in chapter 2 to better interpret the proposed model. The integral of integrated gradients can be efficiently approximated via a summation. Simply, sum the gradients at points occurring at sufficiently small intervals along the straight-line path from the baseline x' to the input x.

$$IntegratedGrads_i(x) ::= (x_i - x'_i) \times \sum_{k=1}^{m} \frac{\partial F(x' + \frac{k}{m} \times (x - x'))}{\partial x_i} \times \frac{1}{m}$$

The summation above is computed with captum library [35]. In figure 5.10, the sum of gradients on the path from a zero baseline to a zero-labeled sample

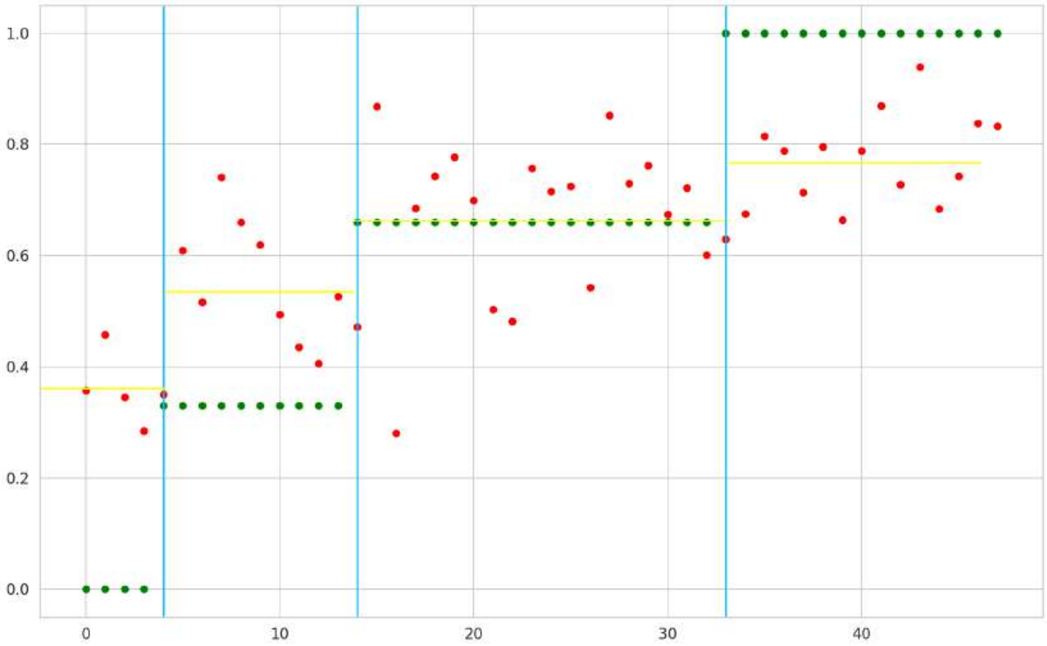


Figure 5.9. Test set results of triplet loss training. MSE score:0.05011. The predicted (red) and ground truth (green) engagement labels of Engagement in the wild dataset. The yellow line represents the average score of predicted label scores for each label class.

from the test set is shown. The model evaluated here is only trained with gaze and head pose-related features and maximum, minimum, and variance aggregation functions. For the sake of simplicity, only the last 10 values of the 150-element sequence are illustrated. By looking at the values in the figure, we can say that feature values in the most recent time steps have more importance compared to feature values in previous time steps. Secondly, we can say that head pose-related features and variance of eye gaze features are the ones most effective on engagement level detection.

In [48], authors mention the importance of choosing the baseline and used zero input baseline model for other sequence modeling tasks such as machine translation and question classification. Besides the zero baselines, we also computed the integrated gradients with average values of the feature sequences in training data as a baseline. However, there is no significant difference in the importance of features when the baseline is changed.

Importance	**Eng 0**	**Eng 0.33**	**Eng 0.66**	**Eng 1**
1	pose_Tz_var	pose_Tz_var	pose_Tz_var	pose_Tz_var
2	pose_Tz_min	pose_Tz_max	pose_Tx_var	pose_Tz_max
3	pose_Tx_max	pose_Tx_max	pose_Tx_max	pose_Tx_max
4	pose_Tx_min	pose_Tx_min	pose_Tx_min	pose_Tx_min
5	gaze_0_x_var	gaze_0_x_var	gaze_0_z_var	gaze_1_x_var

Table 5.4. 5 most important features for all engagement label groups for the best performing model.

For the best performing model, the integrated gradients are calculated for all test samples and averaged for all samples in the same engagement level. In table 5.4,

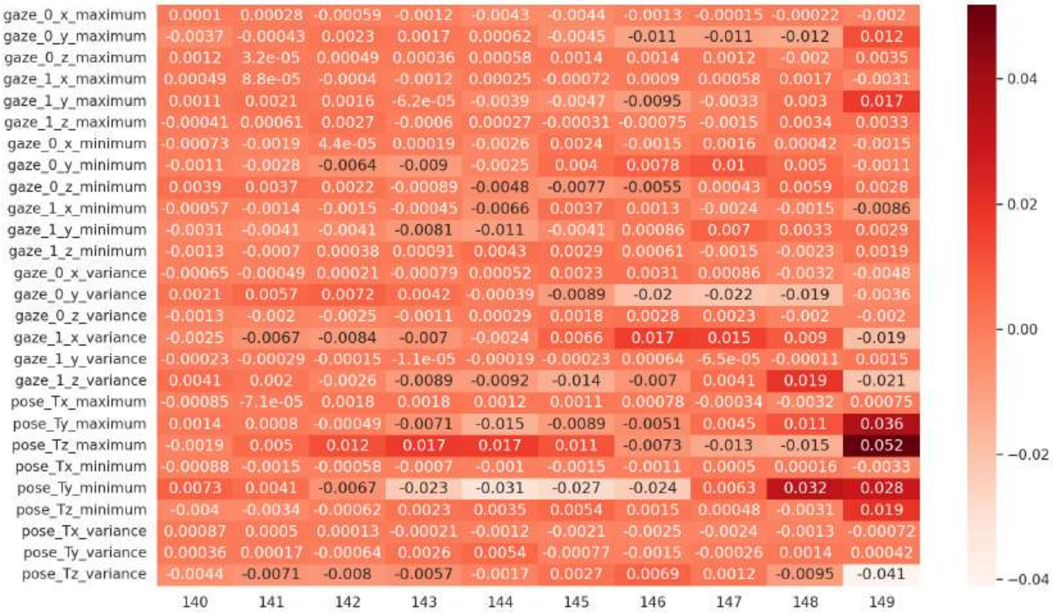


Figure 5.10. The sum of gradients on the path from a zero baseline to a zero labeled sample from the test set.

you can see the top 5 important features for all engagement levels. To calculate the most important features, the absolute value of features for all time steps are summed and then sorted in descending order. In the table, we see that head pose-related features are the top 3 features for all engagement level samples. This indicates that the head pose is the most important feature for engagement level detection. Thus, for all engagement levels, the variance of pose_Tz is the most important feature. This feature refers to the distance between the head and the screen. Moreover, for all engagement levels, the head pose features are followed by the variance of eye gaze-related features. These results make intuitive sense and are also in line with results in table 5.3.

5.2.4 Real Life Performance

The best-proposed model is also tested on real-life engagement detection tasks. The input videos are divided into 10-second snippets as samples to predict the engagement level. Then each sample is passed to OpenFace for extraction of eye gaze, head pose, and facial action unit extraction. After that, the extracted features are aggregated with *minimum, maximum, variance, and length* aggregation functions. For each sample, approximately 80 frames are aggregated with 8 overlapping frames by using a sliding window approach. Finally, the feature matrix is passed to pre-trained Bi-LSTM-MLP unit for engagement regression.

In figures 5.11 to 5.14, you can see snapshots from different input videos representing different levels of engagement.5.11 includes images sampled from video snippets with engagement levels varying from 0.82 to 0.98. These images represent the highest engagement level class. Top 2 images have engagement levels 0.903 and 0.982 and the bottom 2 images have engagement levels 0.854 and 0.828. In all four samples, subjects are staying steady and directly looking at the camera. The reason for the engagement difference gap between top and bottom images can be the distance between the head and the screen. As seen, subjects in top images are more close to the screen but bottom images are more distant. This can be considered as a reasonable engagement level assignment since being closer to the screen and looking straight to the screen may indicate higher engagement.

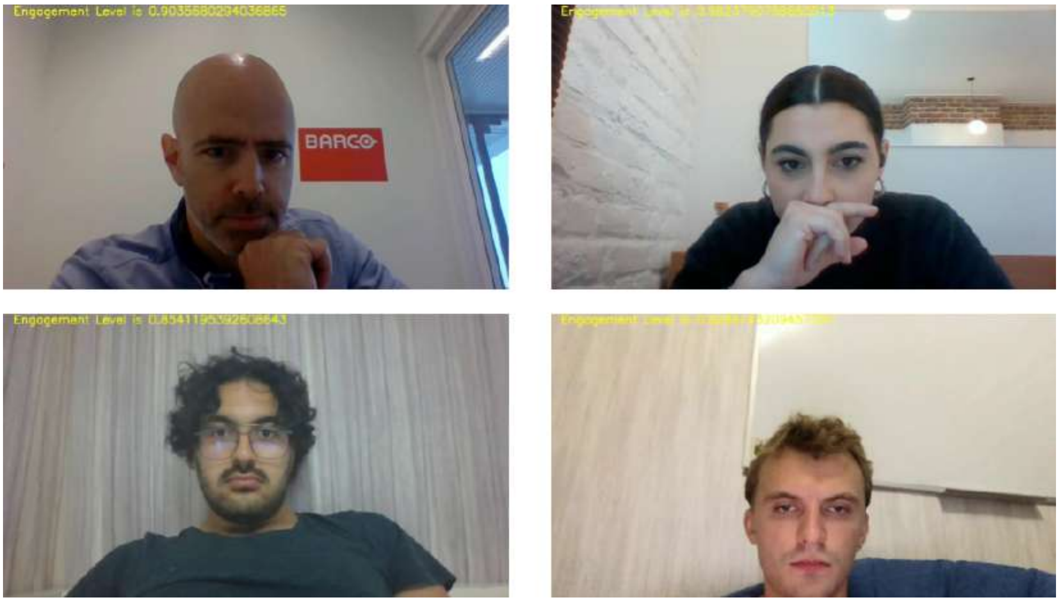

Engagement Level is 0.9035580294036865
Engagement Level is 0.8541195392608643

Figure 5.11. images sampled from video snippets with very high engagement predictions.

In 5.12, you can see sample images sampled from video snippets with engagement levels varying from 0.60 to 0.68. In the left top image, you can see that the subject's face is occluded with some hand movement but the eyes are still looking to the screen. In the other 3 images, subjects are all looking straight to the screen but maybe with confusion expression, especially for the bottom right image. However, it is not trivial to judge the engagement by looking only at these images in this case. These facial expressions generally occur when the engagement level of the subject is changing from low to high or high to low. So it makes more intuitive sense to judge these scenes with the frames before and after. In the appendix, you can see consecutive images representing the change of engagement levels.

In 5.13, you can see sample images sampled from video snippets with engagement levels varying from 0.39 to 0.48. These samples represent a low engagement class. As seen from the figure, none of the subjects are looking at the screen. except for the top right image, the other subjects look some other place and their head is straight to the camera but they show profile to view. Here we can claim that eye gaze and head pose features are effective in low-level engagement level assignment. The images in the top left and bottom right are very similar. In fact, the head pose is very similar in both images but in one of them the subject is looking to the top and in one of them the subject is looking to the bottom. Here we see that model assigns lower to the subject looking up. In the top right image, The subject's head is straight to the screen but his eyes are closed and the engagement level is 0.436. Intuitively, one would expect a lower engagement level since the eyes are closed. However, the model gives more importance to head pose than eye gaze information so in these cases model may not be so accurate. In fact, 0.436 is not a very high

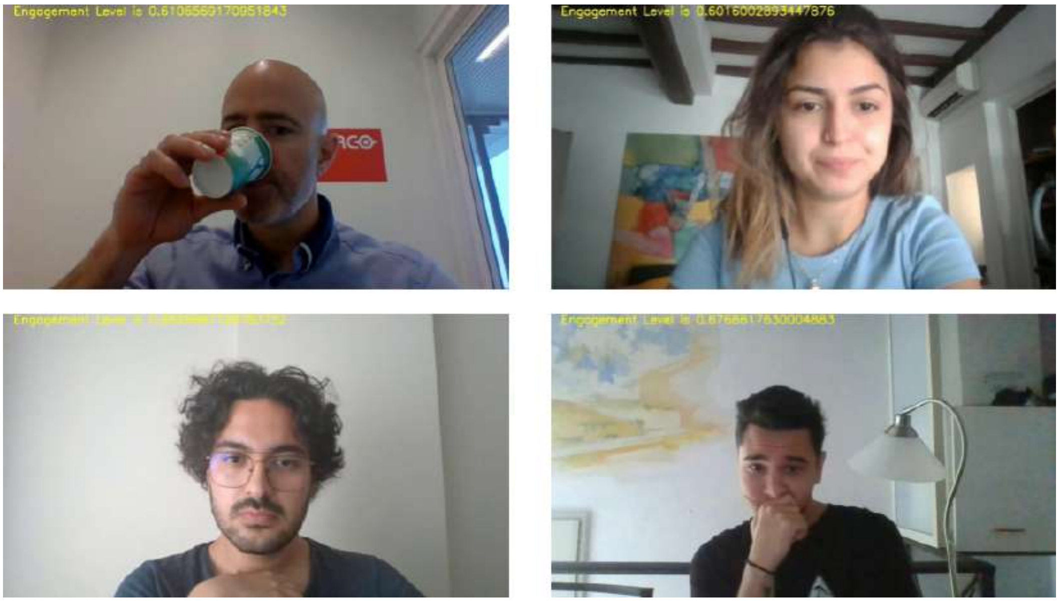

Engagement Level is 0.6106569170951843
Engagement Level is 0.6016002893447876
Engagement Level is
Engagement Level is 0.6766817630004883

Figure 5.12. images sampled from video snippets with high engagement predictions.

engagement level and the model is still able to reduce the engagement level with eye gaze information even though the head pose is straight. This is also a piece of evidence showing the model is considering other information besides head pose.

Finally, in 5.14, you can see sample images sampled from video snippets with engagement levels varying from 0.014 to 0.348. These samples represent the lowest engagement class. For the top right image, the subject's head is up and looking top. In this case, engagement level 0.348 is a reasonable prediction. However for the bottom left The subject is close to the screen, the head position is also not straight but the engagement level is only 0.277. One reason for that can be frequent head movement in frames before and after. However, by looking only at this image, it looks like the model assigned an engagement level lower than it should be. The images on the right are nice examples of the lowest possible engagement levels. In the top right, the subject is yawning and in the bottom right the subject is sleeping and the engagement level is almost zero. This shows that the model can detect the lowest possible engagement levels.

To sum up, we can say that the proposed model trained on Engagement in the wild dataset is also able to generate reasonable predictions in real-life scenarios. Although the predictions made by the model can have high variance, the model can distinguish between different levels of engagement and successfully predict very low, low, high, and very high engagement levels. While generating engagement level predictions, the model gives the highest importance to head pose-related features. After that, eye gaze-related features are also considered but there is a significant margin between the importance of head pose and eye gaze-related features. However, the eye gaze-related features are still relevant and they affect the predictions as shown in figure 5.13.

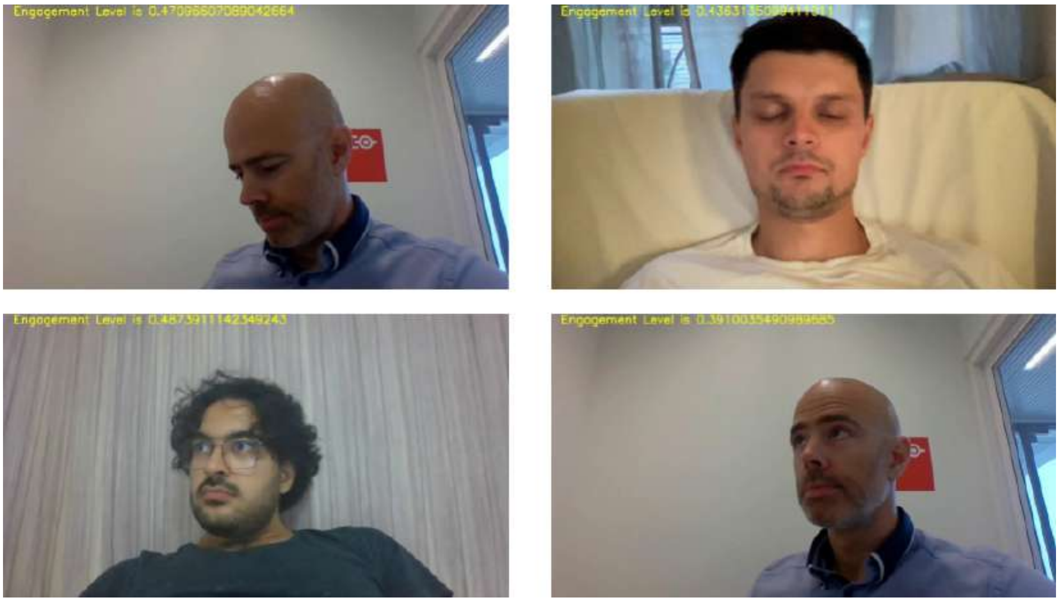

Engagement Level is 0.47096607089042664
Engagement Level is 0.4363135098411911
Engagement Level is 0.48739111423349243
Engagement Level is 0.39100354909989685

Figure 5.13. images sampled from video snippets with low engagement predictions.

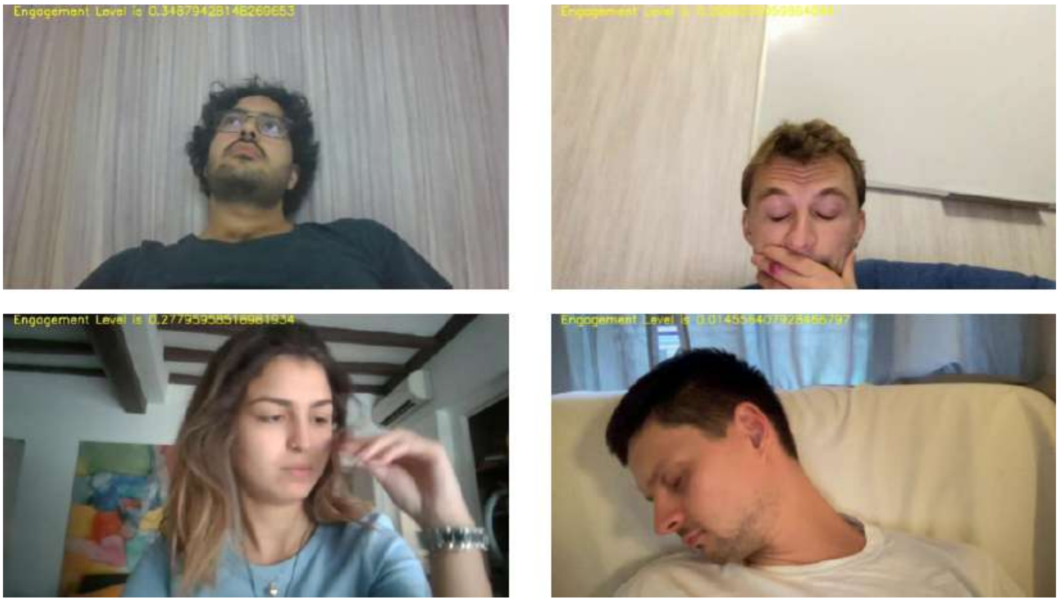

Engagement Level is 0.34879428146260653
Engagement Level is
Engagement Level is 0.27795958516981934
Engagement Level is 0.014556407928866797

Figure 5.14. images sampled from video snippets with very low engagement predictions.

Chapter 6

Conclusion, Evaluation and Future Work

6.1 Conclusion

In this master thesis, the relation between facial expressions/body pose and the subject's engagement level is investigated in e-learning environments. Moreover, a predictive model is proposed for engagement level detection. The proposed model takes an input of a video snippet recorded while a subject is watching an educative material and outputs the engagement level for the corresponding part of the video. The model achieved state-of-the-art results in two publicly available datasets which are Daisee [26] and Engagement in the wild [34]. the proposed model first extracts the facial features with OpenFace [8] and aggregates these feature values in time dimensions with statistical aggregation functions to extract more features. Then the resulting sequence of features is modeled using recurrent neural networks to exploit the temporal aspects of the video data and an engagement prediction is generated for both classification and regression tasks.

For Daisee dataset[26], the main challenge is the imbalanced number of samples that makes it very hard to predict low-level engagement labels. In order to overcome this, the proposed architecture is trained in two stages. First, the model has trained one boredom label which is inversely correlated with engagement and considerably more balanced. After that, the model is fine-tuned on engagement labels and achieved an accuracy score of 64.4% on 5-fold cross-validation training. However, the model still fails to predict low-level engagement classes on test and survey datasets.

In order to create a predictor that can distinguish between high and low levels of engagement, the proposed model is trained on Engagement in the wild [34] dataset. Engagement in the wild dataset has more balanced samples and more reliable labels since the video duration per label is longer than the DAISEE dataset. The proposed model achieved an MSE score of 0.0539 on the test set of Engagement in the wild dataset and the results showed that the model is able to distinguish between four levels of engagement. In addition to that, the integrated gradients method is used to analyze feature importance and results showed that head pose and eye gaze related features are most important for the proposed model.

Finally, the proposed model trained on Engagement in the wild dataset is tested on unlabeled videos collected by the authors. The results show that the proposed model is also able to generate reasonable predictions and distinguish different levels of engagement on videos from outer sources.

6.2 Evaluation

In this work, we propose an end-to-end deep learning-based system that detects the engagement level of the subject in an e-learning environment. Experiments showed that the model is able to distinguish between different levels of engagement. However, there are some limitations regarding the training data and proposed model. The Engagement in the wild dataset is very small and contains only around 150 videos. For this reason, the model overfits very quickly for large batch sizes, and for smaller batch sizes the training proceeds very unstable. The second limitation is the lack of ethnic diversity in the dataset. The dataset is collected from only South Asian students. Although we see that this was not a big problem during the real-life tests. But this is also because the model is only considering head pose and eye gaze features. Since the model is making very simple decisions only with 2 types of features, ethnic diversity is not causing a problem in this scenario. Finally, the reliability of the labels is always a question since engagement level is subjective and it can change from annotator to annotator. The proposed model is only able to learn simple clues that can indicate engagement level but in a real e-learning environment, the engagement level does not only depend on the head pose and eye gaze features. For example, the subject will be taking notes so he/she will be always looking at the notebook and screen which will increase the eye gaze variance and predicted as low engagement by the proposed model.

6.3 Future Work

There are some possible directions to extend and improve this work. The first thing would be designing a training procedure that will use both Daisee [26] and Engagement in the wild [34] datasets together. Since these datasets are similar to each other, one can be used to overcome the challenges on the other. Second, the feature aggregation part of the model requires manual selection of aggregation functions and these functions can change when training on different data. To overcome this, aggregation functions can be replaced with convolutional layers so that they can be learned from data as in [33]. Finally, a self-supervised learning-based method can be used to cluster images/videos belonging to the same engagement category as in [46]. In this way, the reliability problem of engagement level labels can be avoided.

Bibliography

[1] Barco. https://www.barco.com/en/. Accessed: 2021-04-27.

[2] Abedi, A. and Khan, S. S. Improving state-of-the-art in detecting student engagement with resnet and tcn hybrid network. *arXiv preprint arXiv:2104.10122*, (2021).

[3] Ashwin, T. and Guddeti, R. M. R. Affective database for e-learning and classroom environments using indian students' faces, hand gestures and body postures. *Future Generation Computer Systems*, **108** (2020), 334.

[4] Bai, S., Kolter, J. Z., and Koltun, V. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *arXiv preprint arXiv:1803.01271*, (2018).

[5] Balntas, V., Riba, E., Ponsa, D., and Mikolajczyk, K. Learning local feature descriptors with triplets and shallow convolutional neural networks. In *Bmvc*, vol. 1, p. 3 (2016).

[6] Baltrušaitis, T., Mahmoud, M., and Robinson, P. Cross-dataset learning and person-specific normalisation for automatic action unit detection. In *2015 11th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*, vol. 6, pp. 1–6. IEEE (2015).

[7] Baltrusaitis, T., Robinson, P., and Morency, L.-P. Constrained local neural fields for robust facial landmark detection in the wild. In *Proceedings of the IEEE international conference on computer vision workshops*, pp. 354–361 (2013).

[8] Baltrusaitis, T., Zadeh, A., Lim, Y. C., and Morency, L.-P. Openface 2.0: Facial behavior analysis toolkit. In *2018 13th IEEE International Conference on Automatic Face Gesture Recognition (FG 2018)*, pp. 59–66 (2018). doi:10.1109/FG.2018.00019.

[9] Booth, B. M., Ali, A. M., Narayanan, S. S., Bennett, I., and Farag, A. A. Toward active and unobtrusive engagement assessment of distance learners. In *2017 Seventh International Conference on Affective Computing and Intelligent Interaction (ACII)*, pp. 470–476 (2017). doi:10.1109/ACII.2017.8273641.

[10] Burgoon, J. K., Magnenat-Thalmann, N., Pantic, M., and Vinciarelli, A. *Social signal processing*. Cambridge University Press (2017).

[11] Cao, Z., Hidalgo Martinez, G., Simon, T., Wei, S., and Sheikh, Y. A. Openpose: Realtime multi-person 2d pose estimation using part affinity fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, (2019).

[12] Chang, C., Zhang, C., Chen, L., and Liu, Y. An ensemble model using face and body tracking for engagement detection. In *Proceedings of the 20th ACM International Conference on Multimodal Interaction*, pp. 616–622 (2018).

[13] Chang, C.-C. and Lin, C.-J. Libsvm: a library for support vector machines. *ACM transactions on intelligent systems and technology (TIST)*, **2** (2011), 1.

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

[15] Christ, M., Kempa-Liehr, A. W., and Feindt, M. Distributed and parallel time series feature extraction for industrial big data applications. *arXiv preprint arXiv:1610.07717*, (2016).

[16] Dawid, A. P. and Skene, A. M. Maximum likelihood estimation of observer error-rates using the em algorithm. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, **28** (1979), 20.

[17] Delgado, K., et al. Student engagement dataset. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*, pp. 3628–3636 (2021).

[18] Dewan, M. A. A., Lin, F., Wen, D., Murshed, M., and Uddin, Z. A deep learning approach to detecting engagement of online learners. In *2018 IEEE SmartWorld, Ubiquitous Intelligence Computing, Advanced Trusted Computing, Scalable Computing Communications, Cloud Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI)*, pp. 1895–1902 (2018). doi:10.1109/SmartWorld.2018.00318.

[19] Dewan, M. A. A., Murshed, M., and Lin, F. Engagement detection in online learning: a review. *Smart Learning Environments*, **6** (2019), 1.

[20] Dhall, A., Kaur, A., Goecke, R., and Gedeon, T. Emotiw 2018: Audio-video, student engagement and group-level affect prediction. In *Proceedings of the 20th ACM International Conference on Multimodal Interaction*, pp. 653–656 (2018).

[21] D'Mello, S. K., Craig, S. D., and Graesser, A. C. Multimethod assessment of affective experience and expression during deep learning. *International Journal of Learning Technology*, **4** (2009), 165.

[22] Donahue, J., Anne Hendricks, L., Guadarrama, S., Rohrbach, M., Venugopalan, S., Saenko, K., and Darrell, T. Long-term recurrent convolutional networks for visual recognition and description. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2625–2634 (2015).

[23] Filntisis, P. P., Efthymiou, N., Potamianos, G., and Maragos, P. Emotion understanding in videos through body, context, and visual-semantic embedding loss. In *European Conference on Computer Vision*, pp. 747–755. Springer (2020).

[24] Frank, M., Tofighi, G., Gu, H., and Fruchter, R. Engagement detection in meetings. *arXiv preprint arXiv:1608.08711*, (2016).

[25] Grafsgaard, J., Wiggins, J. B., Boyer, K. E., Wiebe, E. N., and Lester, J. Automatically recognizing facial expression: Predicting engagement and frustration. In *Educational Data Mining 2013* (2013).

[26] Gupta, A., D'Cunha, A., Awasthi, K., and Balasubramanian, V. Daisee: Towards user engagement recognition in the wild. *arXiv preprint arXiv:1609.01885*, (2016).

[27] He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778 (2016).

[28] Hesch, J. A. and Roumeliotis, S. I. A direct least-squares (dls) method for pnp. In *2011 International Conference on Computer Vision*, pp. 383–390. IEEE (2011).

[29] Hochreiter, S. and Schmidhuber, J. Long short-term memory. *Neural computation*, **9** (1997), 1735.

[30] Huang, T., Mei, Y., Zhang, H., Liu, S., and Yang, H. Fine-grained engagement recognition in online learning environment. In *2019 IEEE 9th International Conference on Electronics Information and Emergency Communication (ICEIEC)*, pp. 338–341 (2019). doi:10.1109/ICEIEC.2019.8784559.

[31] Huang, T., Mei, Y., Zhang, H., Liu, S., and Yang, H. Fine-grained engagement recognition in online learning environment. In *2019 IEEE 9th international conference on electronics information and emergency communication (ICEIEC)*, pp. 338–341. IEEE (2019).

[32] Kahou, S. E., et al. Emonets: Multimodal deep learning approaches for emotion recognition in video. *Journal on Multimodal User Interfaces*, **10** (2016), 99.

[33] Karim, F., Majumdar, S., Darabi, H., and Harford, S. Multivariate lstm-fcns for time series classification. *Neural Networks*, **116** (2019), 237.

[34] Kaur, A., Mustafa, A., Mehta, L., and Dhall, A. Prediction and localization of student engagement in the wild. In *2018 Digital Image Computing: Techniques and Applications (DICTA)*, pp. 1–8. IEEE (2018).

[35] Kokhlikyan, N., et al. Captum: A unified and generic model interpretability library for pytorch (2020). arXiv:2009.07896.

[36] Krizhevsky, A., Sutskever, I., and Hinton, G. E. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, **25** (2012), 1097.

[37] Liao, J., Liang, Y., and Pan, J. Deep facial spatiotemporal network for engagement prediction in online learning. *Applied Intelligence*, (2021), 1.

[38] Luo, Y., Ye, J., Adams, R. B., Li, J., Newman, M. G., and Wang, J. Z. Arbee: Towards automated recognition of bodily expression of emotion in the wild. *International Journal of Computer Vision*, **128** (2020), 1.

[39] Murshed, M., Dewan, M. A. A., Lin, F., and Wen, D. Engagement detection in e-learning environments using convolutional neural networks. In *2019 IEEE Intl Conf on Dependable, Autonomic and Secure Computing, Intl Conf on Pervasive Intelligence and Computing, Intl Conf on Cloud and Big Data Computing, Intl Conf on Cyber Science and Technology Congress (DASC/PiCom/CBDCom/CyberSciTech)*, pp. 80–86. IEEE (2019).

[40] Niu, X., Han, H., Zeng, J., Sun, X., Shan, S., Huang, Y., Yang, S., and Chen, X. Automatic engagement prediction with gap feature. In *Proceedings of the 20th ACM International Conference on Multimodal Interaction*, pp. 599–603 (2018).

[41] Parmar, P. and Tran Morris, B. Learning to score olympic events. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 20–28 (2017).

[42] Pearson, K. Liii. on lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin philosophical magazine and journal of science*, **2** (1901), 559.

[43] Pedregosa, F., et al. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, **12** (2011), 2825.

[44] Phi, M. Illustrated guide to lstm's and gru's: A step by step explanation (2018). Available from: https://learnedvector.medium.com/.

[45] Rao, K. P. and Rao, M. C. S. Recognition of learners' cognitive states using facial expressions in e-learning environments.

[46] Roy, S. and Etemad, A. Self-supervised contrastive learning of multi-view facial expressions. *arXiv preprint arXiv:2108.06723*, (2021).

[47] Simonyan, K. and Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, (2014).

[48] Sundararajan, M., Taly, A., and Yan, Q. Axiomatic attribution for deep networks. In *International Conference on Machine Learning*, pp. 3319–3328. PMLR (2017).

[49] Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., and Wojna, Z. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2818–2826 (2016).

[50] Thiruthuvanathan, M. M., Krishnan, B., and Rangaswamy, M. Engagement detection through facial emotional recognition using a shallow residual convolutional neural networks.

[51] Thomas, C., Nair, N., and Jayagopi, D. B. Predicting engagement intensity in the wild using temporal convolutional network. In *Proceedings of the 20th ACM International Conference on Multimodal Interaction*, pp. 604–610 (2018).

[52] Thong Huynh, V., Kim, S.-H., Lee, G.-S., and Yang, H.-J. Engagement intensity prediction withfacial behavior features. In *2019 International Conference on Multimodal Interaction*, pp. 567–571 (2019).

[53] Tran, D., Bourdev, L., Fergus, R., Torresani, L., and Paluri, M. Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)* (2015).

[54] Tran, D., Bourdev, L., Fergus, R., Torresani, L., and Paluri, M. Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pp. 4489–4497 (2015).

[55] Tripathi, S., Tripathi, S., and Beigi, H. Multi-modal emotion recognition on iemocap dataset using deep learning. *arXiv preprint arXiv:1804.05788*, (2018).

[56] Tzirakis, P., Trigeorgis, G., Nicolaou, M. A., Schuller, B. W., and Zafeiriou, S. End-to-end multimodal emotion recognition using deep neural networks. *IEEE Journal of Selected Topics in Signal Processing*, **11** (2017), 1301.

[57] Ullah, A., Ahmad, J., Muhammad, K., Sajjad, M., and Baik, S. W. Action recognition in video sequences using deep bi-directional lstm with cnn features. *IEEE Access*, **6** (2018), 1155. doi:10.1109/ACCESS.2017.2778011.

[58] Wang, K., Yang, J., Guo, D., Zhang, K., Peng, X., and Qiao, Y. Bootstrap model ensemble and rank loss for engagement intensity regression. In *2019 International Conference on Multimodal Interaction*, pp. 551–556 (2019).

[59] Whitehill, J., Serpell, Z., Lin, Y., Foster, A., and Movellan, J. R. The faces of engagement: Automatic recognition of student engagementfrom facial expressions. *IEEE Transactions on Affective Computing*, **5** (2014), 86. doi:10.1109/TAFFC.2014.2316163.

[60] Wood, E., Baltrusaitis, T., Zhang, X., Sugano, Y., Robinson, P., and Bulling, A. Rendering of eyes for eye-shape registration and gaze estimation. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 3756–3764 (2015).

[61] Wu, J., Yang, B., Wang, Y., and Hattori, G. Advanced multi-instance learning method with multi-features engineering and conservative optimization for engagement intensity prediction. In *Proceedings of the 2020 International Conference on Multimodal Interaction*, pp. 777–783 (2020).

[62] Wu, J., Zhou, Z., Wang, Y., Li, Y., Xu, X., and Uchida, Y. Multi-feature and multi-instance learning with anti-overfitting strategy for engagement intensity prediction. In *2019 International Conference on Multimodal Interaction*, pp. 582–588 (2019).

[63] Yang, J., Wang, K., Peng, X., and Qiao, Y. Deep recurrent multi-instance learning with spatio-temporal features for engagement intensity prediction. In *Proceedings of the 20th ACM international conference on multimodal interaction*, pp. 594–598 (2018).

[64] Yu, Z., Ramanarayanan, V., Suendermann-Oeft, D., Wang, X., Zechner, K., Chen, L., Tao, J., Ivanou, A., and Qian, Y. Using bidirectional lstm recurrent neural networks to learn high-level abstractions of sequential features for automated scoring of non-native spontaneous speech. In *2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, pp. 338–345. IEEE (2015).

[65] Yue-Hei Ng, J., Hausknecht, M., Vijayanarasimhan, S., Vinyals, O., Monga, R., and Toderici, G. Beyond short snippets: Deep networks for video classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2015).

[66] Zadeh, A., Chong Lim, Y., Baltrusaitis, T., and Morency, L.-P. Convolutional experts constrained local model for 3d facial landmark detection. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV) Workshops* (2017).

[67] Zhang, H., Xiao, X., Huang, T., Liu, S., Xia, Y., and Li, J. An novel end-to-end network for automatic student engagement recognition. In *2019 IEEE 9th International Conference on Electronics Information and Emergency Communication (ICEIEC)*, pp. 342–345. IEEE (2019).

[68] Zhang, L. and Radke, R. J. A multi-stream recurrent neural network for social role detection in multiparty interactions. *IEEE Journal of Selected Topics in Signal Processing*, **14** (2020), 554. doi:10.1109/JSTSP.2020.2992394.

Appendix

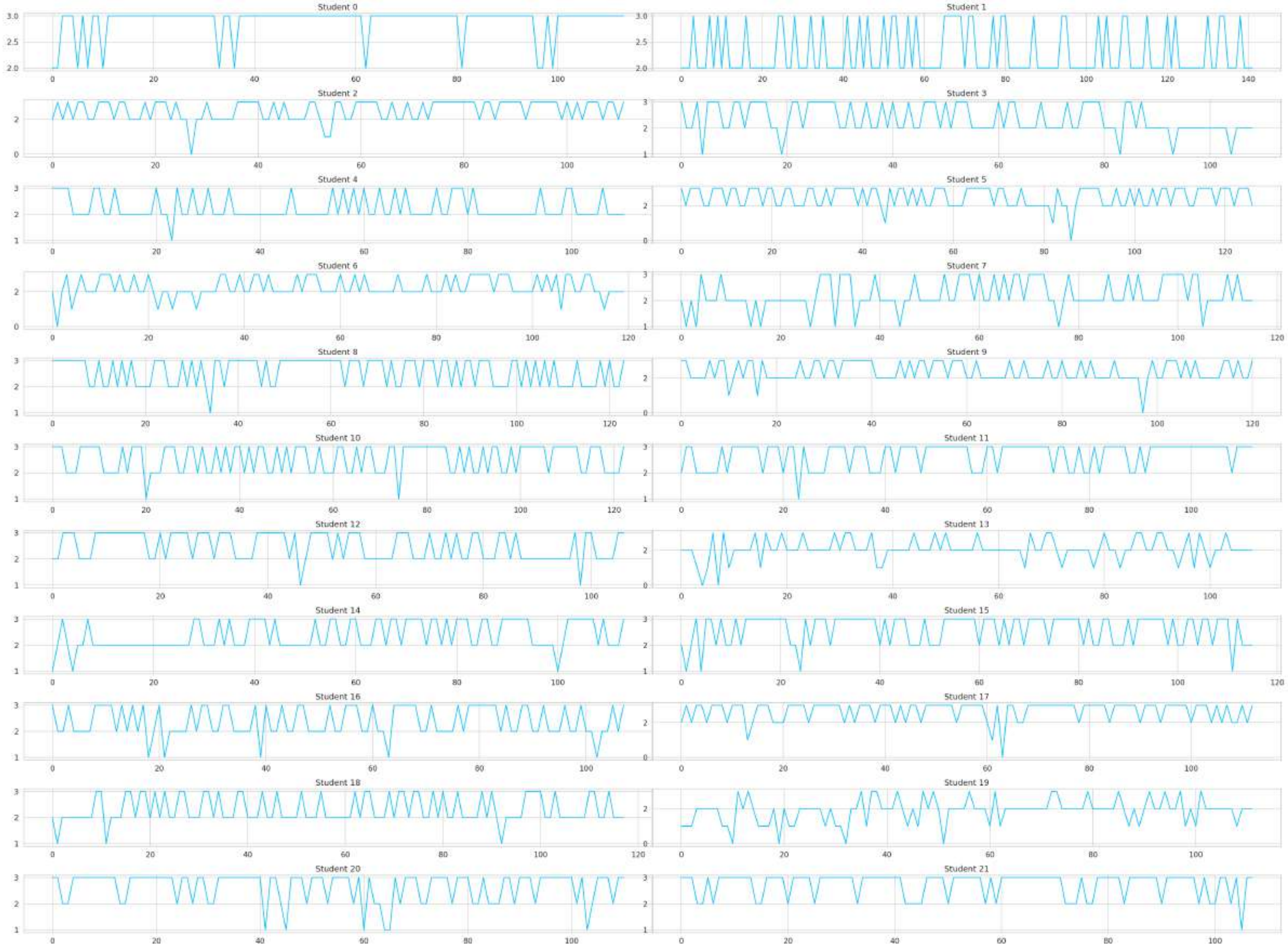


Figure 6.1. The engagement levels of 22 students selected from the training set with at least 100 clips per student. The clips are concatenated with respect to time.

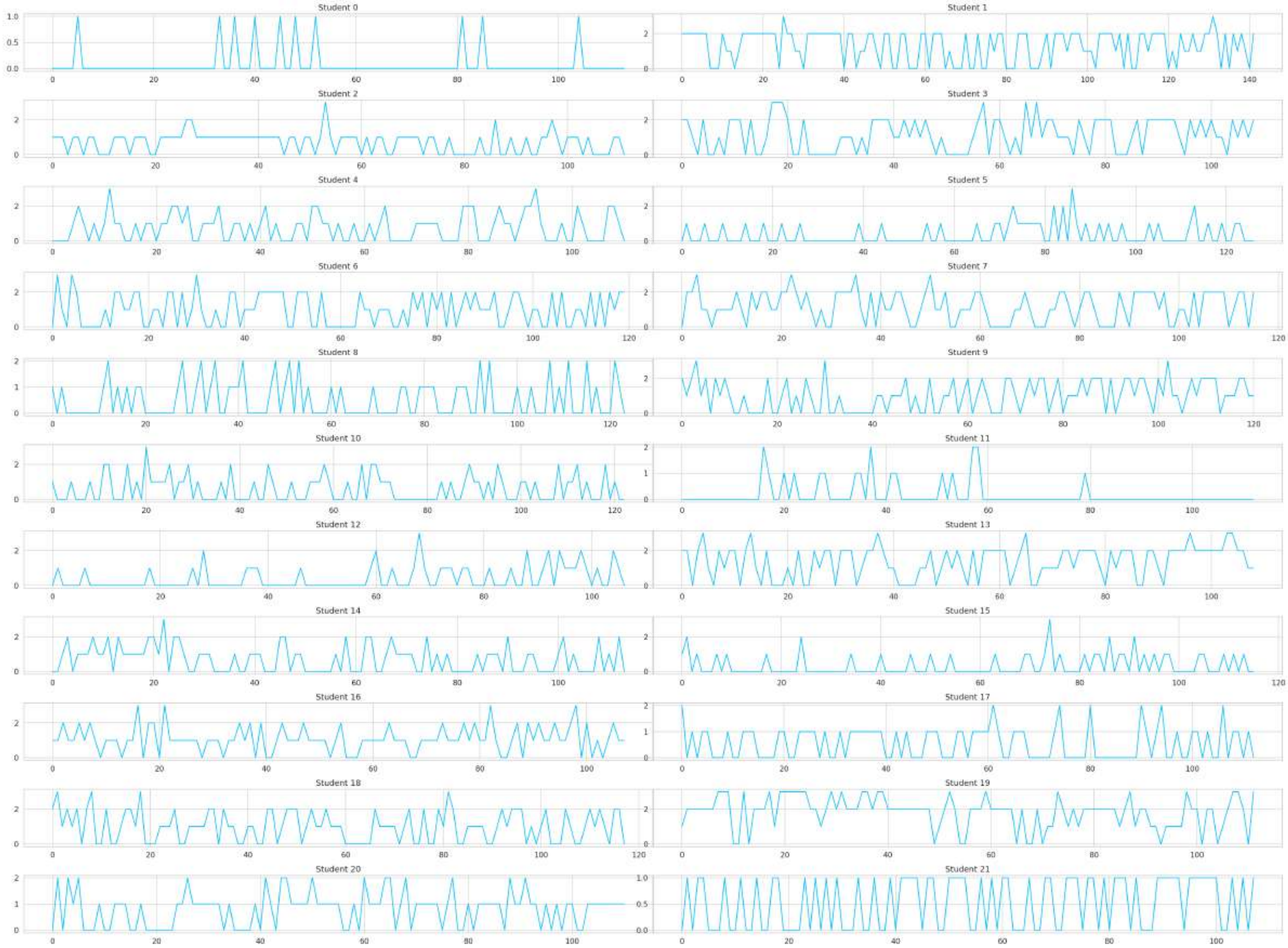


Figure 6.2. The boredom levels of 22 students selected from the training set. with at least 100 clips per student. The clips are concatenated with respect to time.

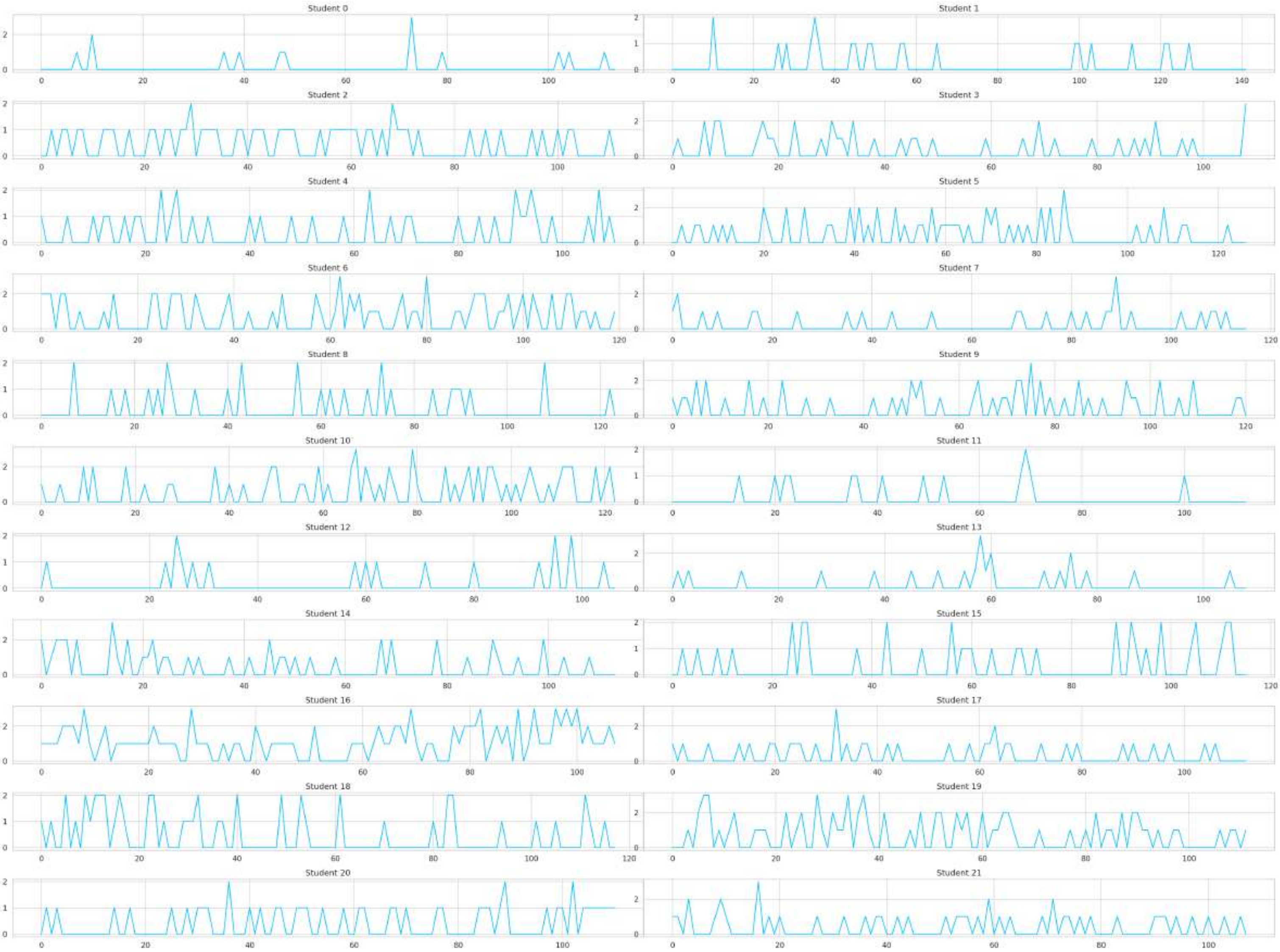


Figure 6.3. The confusion levels of 22 students selected from the training set with at least 100 clips per student. The clips are concatenated with respect to time.

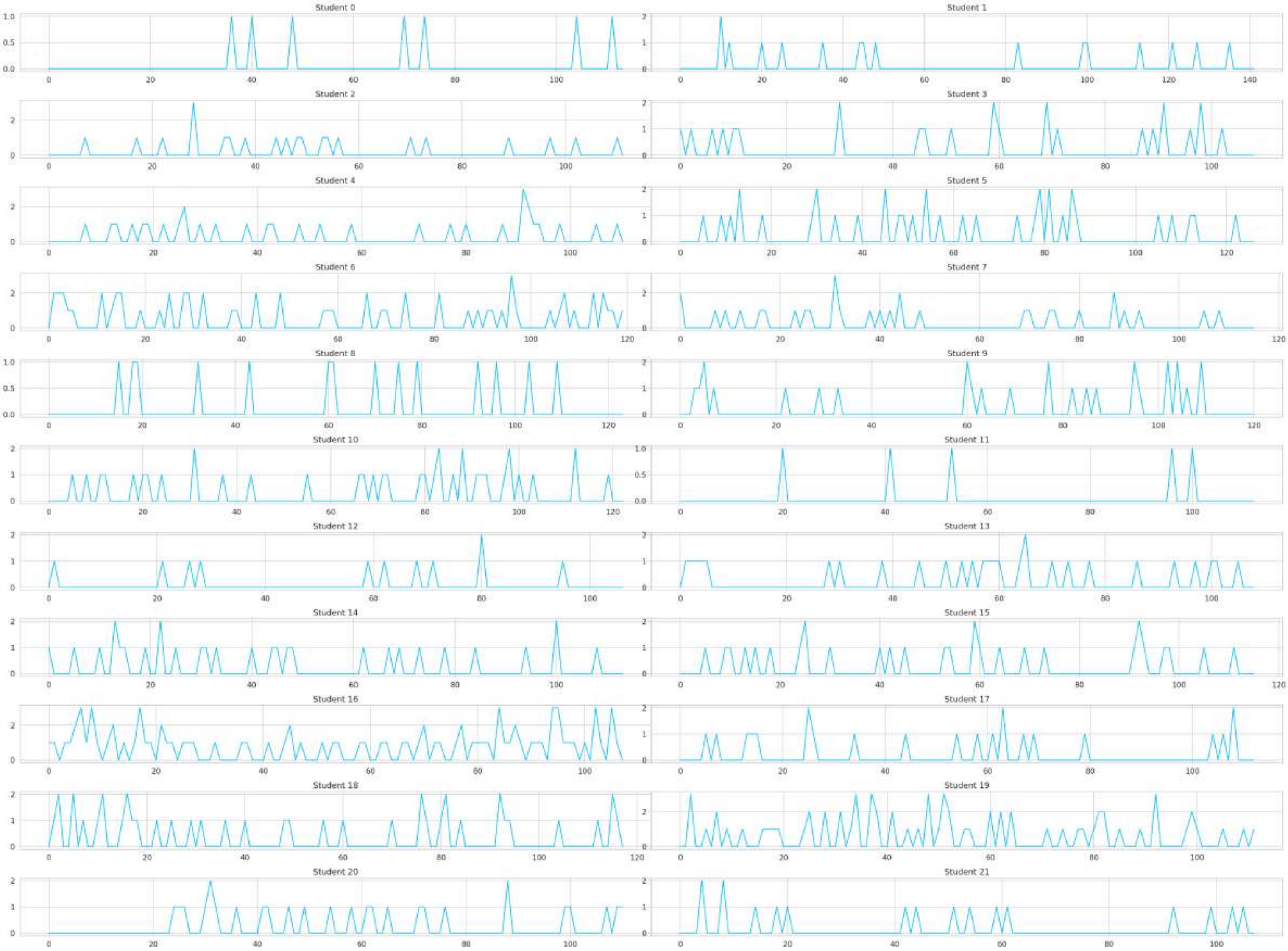


Figure 6.4. The frustration levels of 22 students selected from the training set with at least 100 clips per student. The clips are concatenated with respect to time.

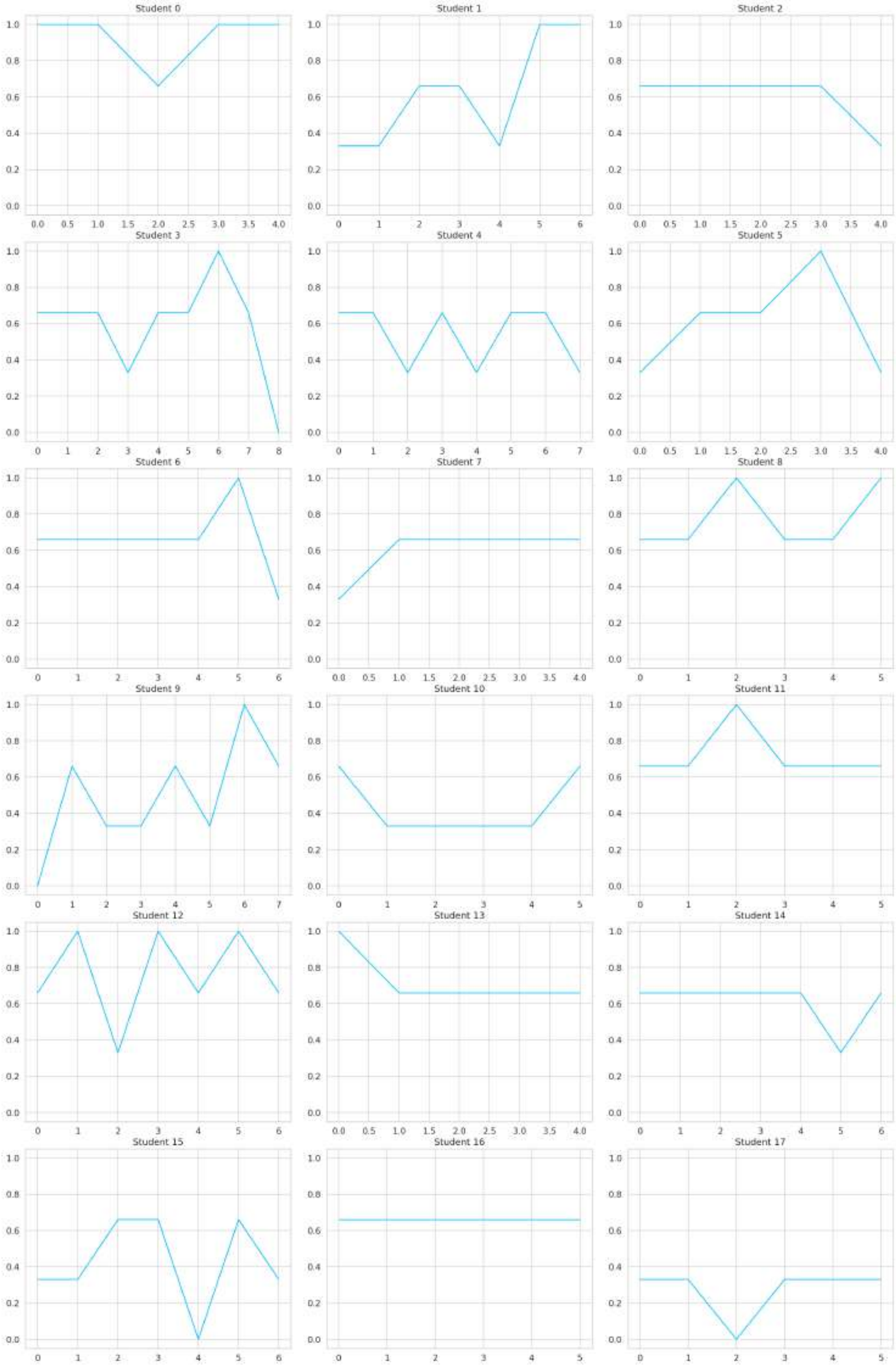
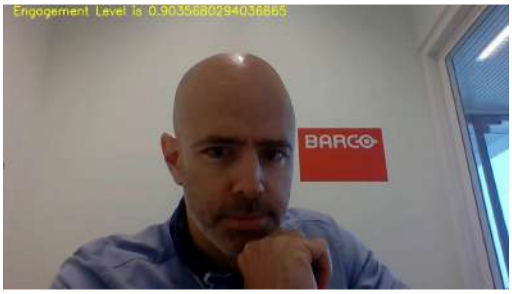
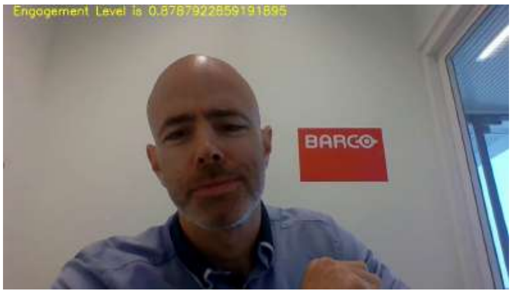
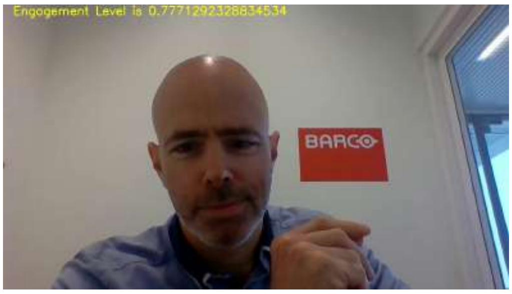
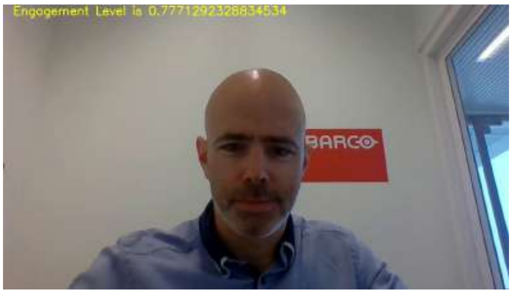


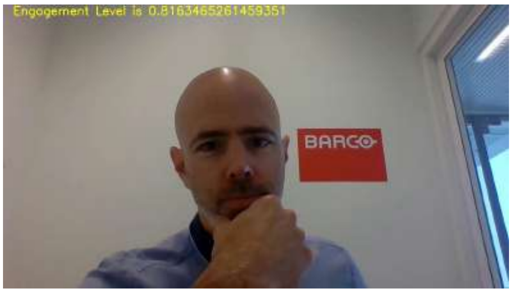
Figure 6.5. The engagement levels of 18 students selected from the training set with at least 5 clips per student. The clips are concatenated with respect to time.

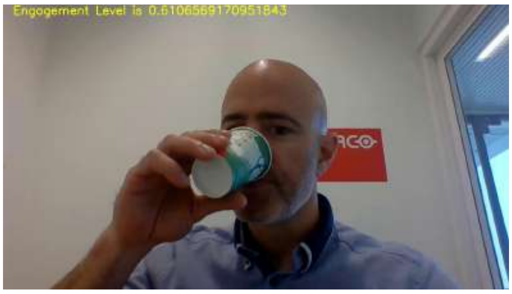

Engogement Level is 0.9035680294036865
BARCO

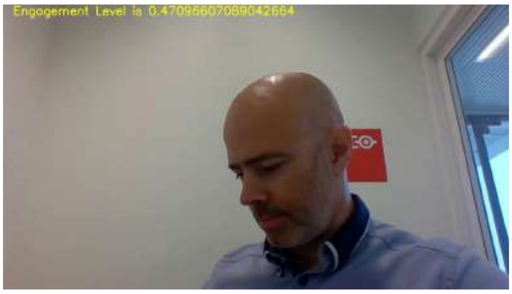

Engogement Level is 0.8787922859191895
BARCO

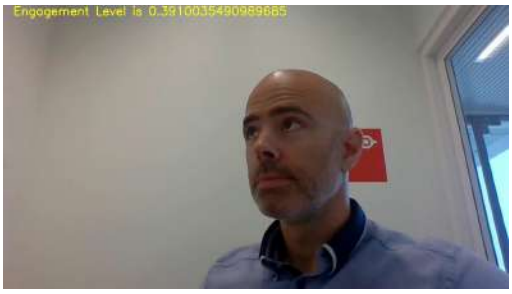

Engogement Level is 0.7771292328834534
BARCO


Engogement Level is 0.7771292328834534


Engogement Level is 0.8163465261459351
BARCO


Engogement Level is 0.6106569170951843


Engogement Level is 0.47096607089042664


Engogement Level is 0.3910035490989685

Engagement Level is 0.8541195392608643

Engagement Level is 0.4873911142349243

Engagement Level is 0.34879428148269653

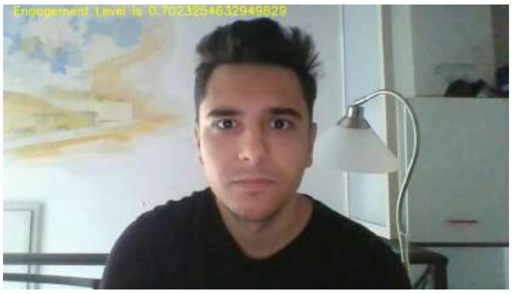
Engagement Level is 0.5265830159187317

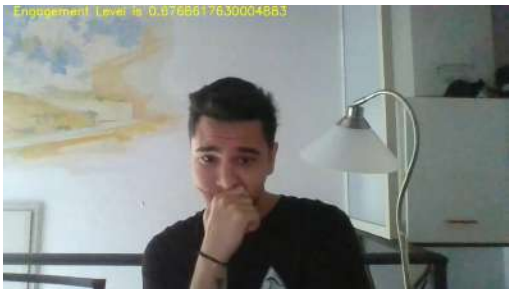
Engagement Level is 0.7551443481446312

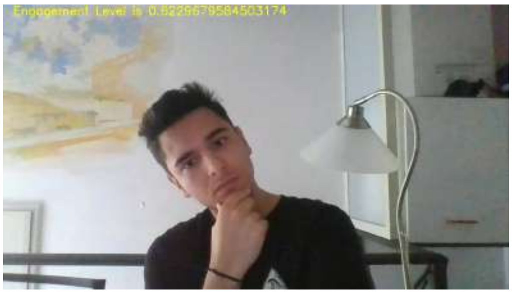
Engagement Level is 0.7027071026416667

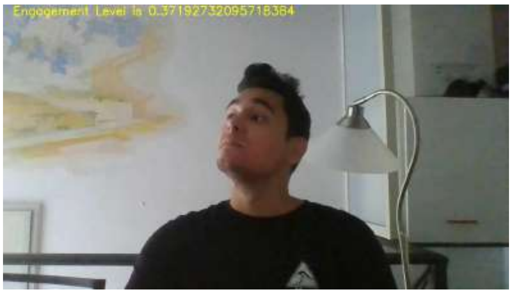
Engagement Level is 0.7609342776158548

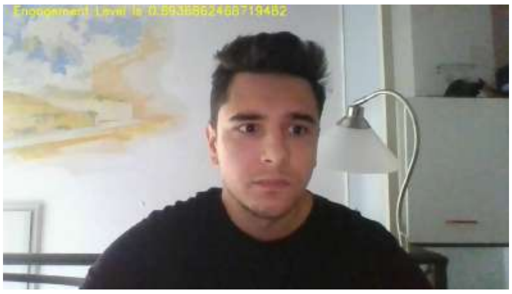
Engagement Level is 0.6669997100783752

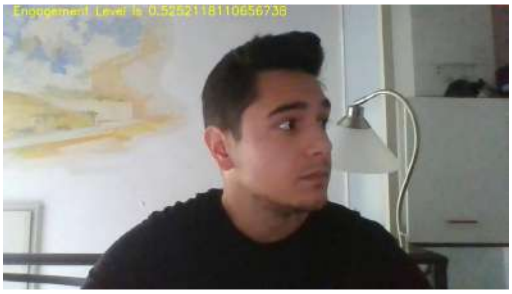

Engagement Level is 0.7023254632949829

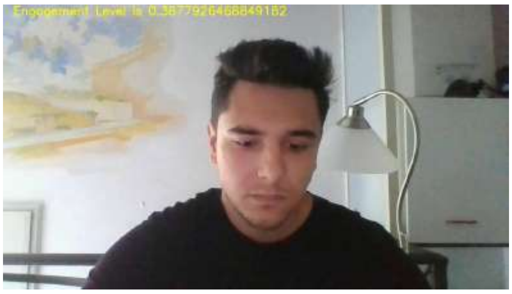

Engagement Level is 0.6766617630004883

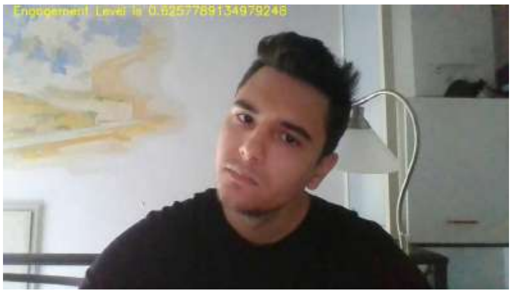

Engagement Level is 0.8229679584503174

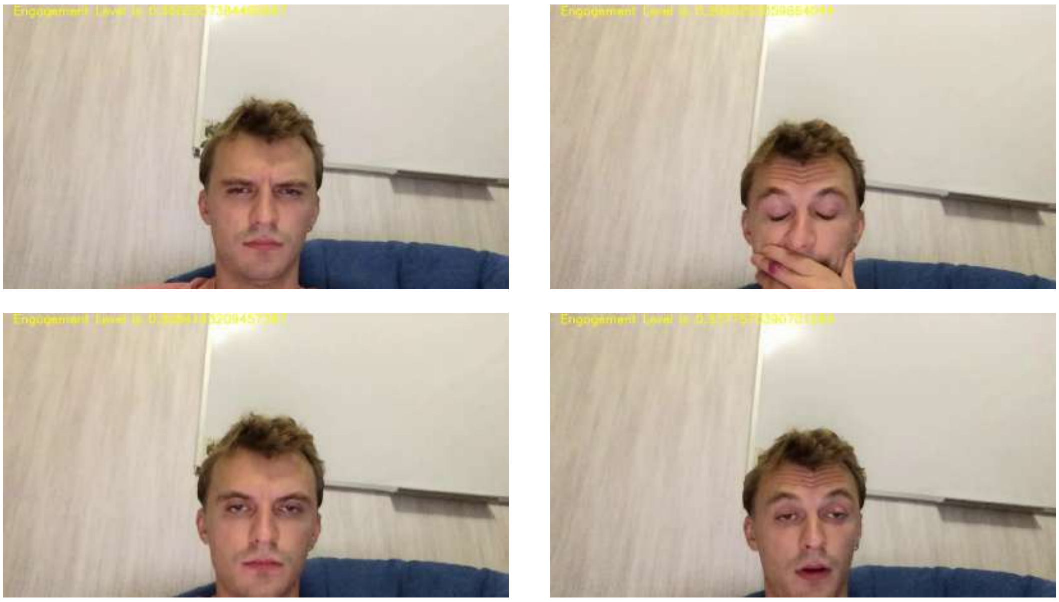

Engagement Level is 0.37192732095718384

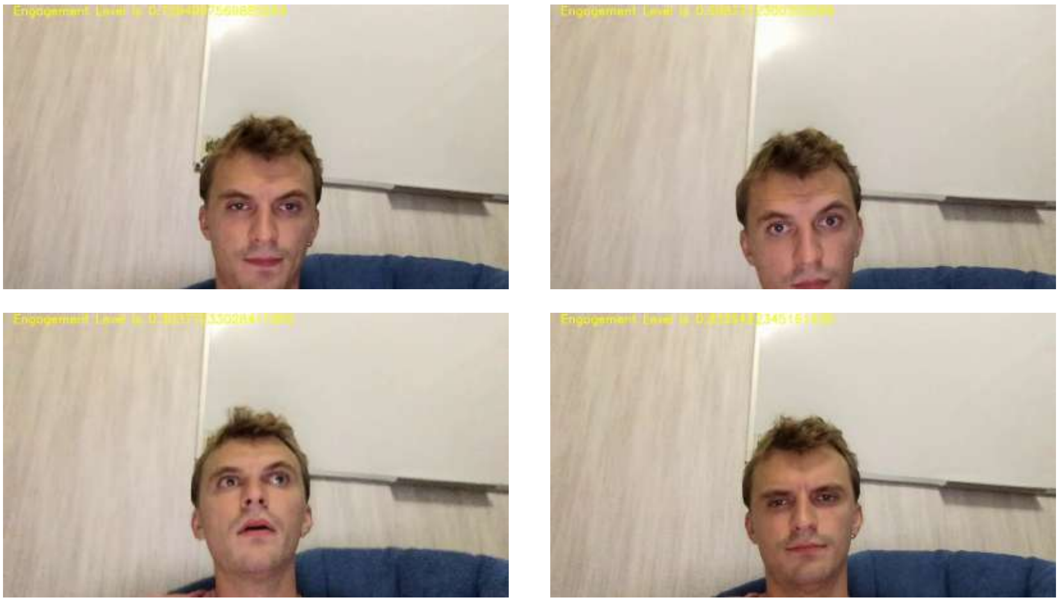

Engagement Level is 0.8936862468719482

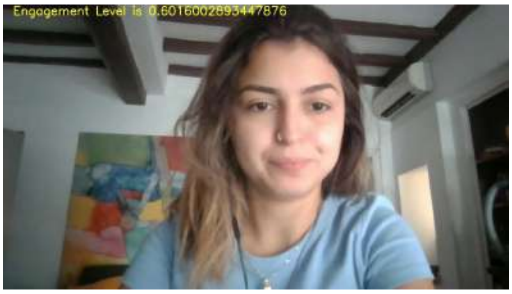

Engagement Level is 0.5252118110656738

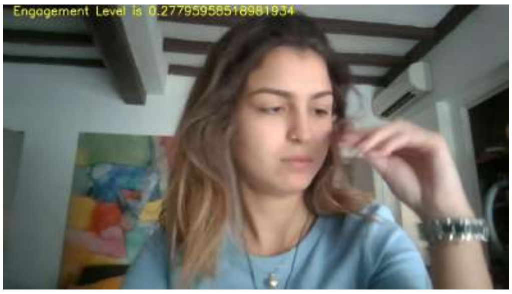

Engagement Level is 0.3677926466849182

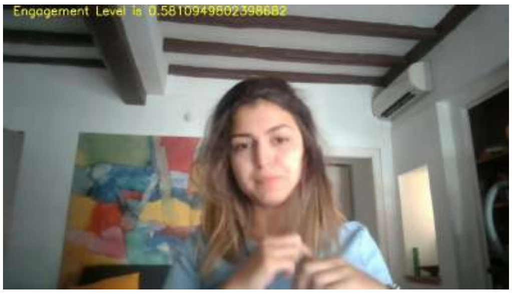

Engagement Level is 0.6257789134979248

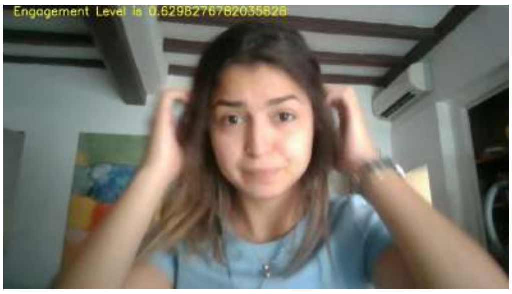

Engagement Level is:
Engagement Level is:
Engagement Level is:
Engagement Level is:

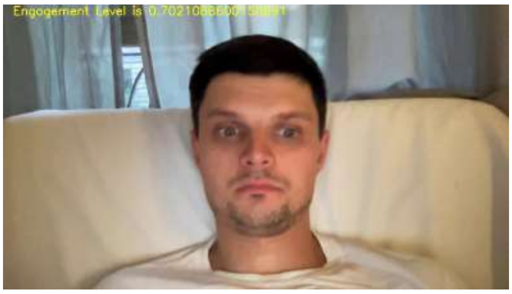

Engagement Level is:
Engagement Level is:
Engagement Level is:
Engagement Level is:

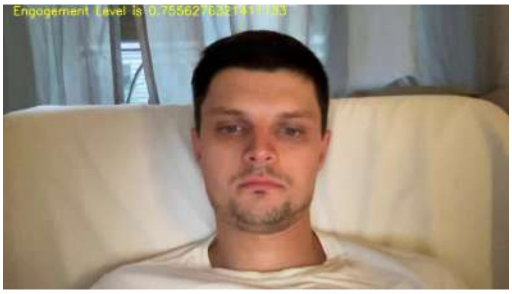

Engagement Level is 0.6016002893447876

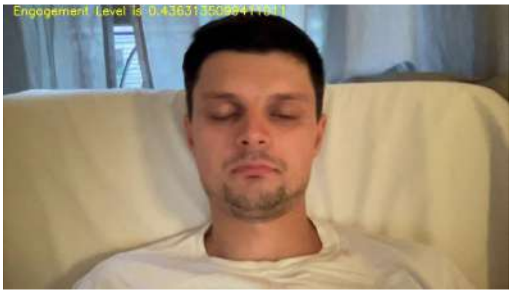

Engagement Level is 0.27795958518981934

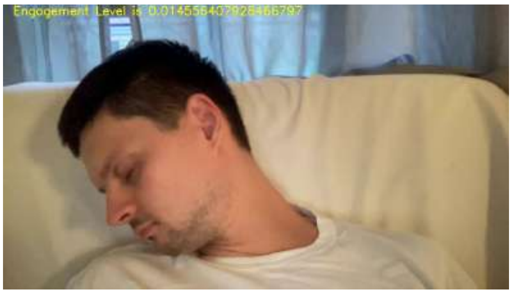

Engagement Level is 0.5810949802398682

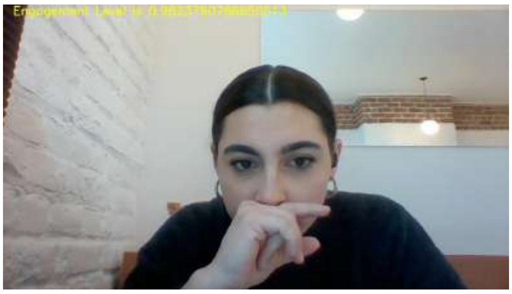

Engagement Level is 0.6298276782035828

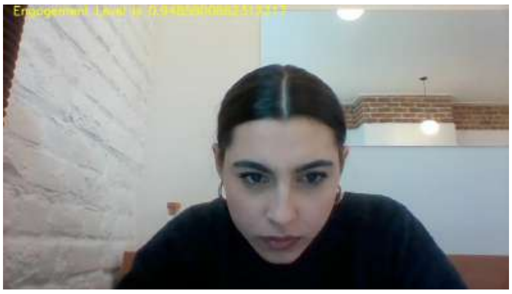

Engagement Level is 0.7021066600158891

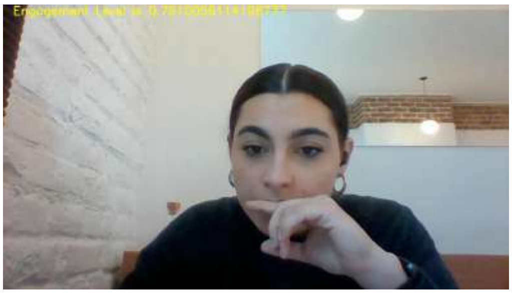

Engagement Level is 0.7556276321411133

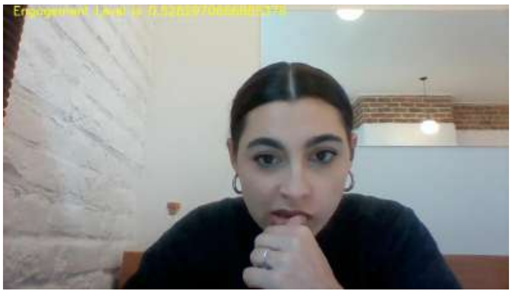

Engagement Level is 0.4363135099411011


Engagement Level is 0.014556407928466797


Engagement Level is: 0.9823780788860013


Engagement Level is: 0.9485800862317217


Engagement Level is: 0.7810056114188777


Engagement Level is: 0.5263970866880378