Engagement Tracking

Bharath Bangalore Somashekar and **Xiny Pan** and **Sabri Bektas**
st164288@stud.uni-stuttgart.de
st169949@stud.uni-stuttgart.de
st142566@stud.uni-stuttgart.de

Abstract

Detecting the engagement levels gives a huge help to measure the quality and attention in different situations. For learning process, student performance and class quality could be evaluated through the levels. In this task, we proposed a neural network within the attention pooling mechanism to achieve a more accurate predictions on the engagement scales.

1 Introduction

Engagement becomes part of the important criteria for judging the concentration, enthusiasm, and even satisfaction of users with a certain activity, which enables the technology of tracking engagement levels playing an important role in various situation.

For instance, traditional job interviews usually take place between the interviewers and interviewees. With the development of web techniques, more companies choose online conference as the main method in the hiring process. The video-based interview could benefit both sides in aspect of time saving. Interviewers could check the video recording again if it is permitted and make the decision.

The performance is usually scored in two aspect: speech and non-verbal behaviors. Besides the verbal demonstration addressed by the interviewers, non-verbal behaviors, such as eye contact, smiles, and head poses, could also reflect the adequate qualification and positive emotion the interviewee has. (Chen et al., 2016) presents a model which estimates the emotion of interviewees by extracting the facial features through Visage SDK's Face-Track. The detection engine could measure the emotional levels: positive, neural and negative according to the facial expressions. This tasks give a helpful reference to the final determination of the Human Resources.

The engagement performance could be tracked and evaluated through the vision analyzing and machine learning technology. Students learning performance in online classes, which is somehow the same as the situation in the interview, could be applied to the similar strategy to track the engagement level. This paper would proposed a model which based on the (Chang et al., 2018) to predict the engagement level by implementing the neural network.

2 Related Research

(Chang et al., 2018) focused on improving the precision detecting of engagement of learners. They demonstrated two models: one is consist of cluster-based models and the other one is an Neural Network (NN) model, which is the mainly baseline this paper would be investigated. These models aim to predict the engagement level when learning the MOOC videos. In NN model, they used the Bi-directional Long Short-Term Memory (BLSTM), one of the transformations of Recurrent Neural Network to improve the performance of dynamic features. The heuristic rules are also applied to the outcomes via the BLSTM to extract the changes of body postures.

In the experiments, the BLSTM performs better than the LSTM with the standard of Mean-Square error (MSE) because it has more feature information and in the meanwhile it calculates the dynamic features in bi-directions other than with only one information transferring. The attention mechanism plays an important role in decreasing the loss as well. However, the research shows a big difference in the result of test and validation. The size of dataset and distribution of labels might have an impact on the outputs, which should be noticed in the following study.

(Whitehill et al., 2014) mainly focused on the

finding the way of detecting the student engagement automatically by judging facial expressions in real-time. The dataset involved the recorded videos which are taken by under-graduated students participating in the cognitive skills training. These raw data are labelled with 4 different rate level and then calculated the differences from the scores that were made by various labelers. The length of the video for labelling are categorised into 2 kinds: 10 sec and 60 sec. They found that 10 sec videos give more useful information and gives relatively better context and reliability in the task.

The frame-by-frame recognition is applied to extract the features in the images, which could be evaluated by the labelers. Four binary classifiers are defined by utilizing 3 feature type algorithms: Boost (BF), SVM (Gabor) and MLR (CERT). The Boost (BF) and SVM (Gabor) outperform at the engagement $= 1$ as they are able to catch the eye gazes which mainly influence the labeling before.

However, CERT performs well in the engagement $= 4$ as the pose features are considered in this stage. To estimate the final engagement, they chose SVM (Gabor) as the binary classifiers applied to different strategies of regression. In the meantime, they used CERT feature weights to show the extent that how human labelers make a judgement concerning the engagement. The results shows that the student post-performance is highly accurate with the prediction which is observed by the pre-performance in machine learning. However, due to the specific experiment conducting, the result could not reflect the truth of long-term learning in the common life.

(Bartlett et al., 2006) developed an automatic system detecting facial expressions and coding the frame in the real-time environment. With the control of light and background, the system performs high accuracy on the CMU-MIT dataset. These facial action classifications could be achieved by SVM and AdaBoost.

The model are trained based on the Cohn and Kanade's DFAT504 and Ekman and Hager datasets. These data are achieved by different various people ranging from different ages and areas in the world. In the experiment, AU nose wrinkle and AU mouth stretch are easily to detected because they perform well even though with small training dataset. They found the data-driven classifers could know the expression intensity. Facial action intensity codes are highly related with the distance which divides the hyperplane and margin.

3 Corpus

The dataset is selected as the same as (Chang et al., 2018) implemented from the challenge organizers. 197 videos *learn the Korean Language in 5 Minutes*, which are concerning the educational purpose, are split into training and validation dataset. Every video lasts approximately 5 minutes long and the environment of data collecting is wild nature, which means that variables influencing the circumstances are not controlled. Students who participate in the experiment could take this video in any places.

Label	0	1	2	3	$\sum$
training	5	35	81	28	149
validation	4	10	19	15	48
$\sum$	9	45	100	43	197

Figure 1:
Label distributions of the dataset
(Chang et al., 2018)

The Challenge includes 149 training videos and 48 validation videos (See Figure 1). There are 4 levels describing the extent of engagement intensity. Generally, 0 intensity represents that the participants does not show any interest towards the video. 1 stands for little engagement, which means that students are almost not engaged in the class. 2 means that the students are paying some attention to the content. 3 shows a highly positive towards the participation. In the evaluation stage, the Challenge organizers normalized level range to $[0 - 1]$ as 0.0, 0.33, 0.66 and 1.0.

4 Implementation

The implementation steps will be outlined in this section. Firstly we will preprocess the data starting with the video set's cutting and down sampling. Using OpenFace, an opensource face analysis implementation, we continue with the feature extraction. Following that, the extracted feature data will be prepared as input for the model to be trained. The model is based on a neural network, which monitors the engagement in the end.

4.1 Data Preprocessing

To adapt the data into a machine learning model the preprocessing of the data is necessary. A vector wise data representation is an ideal input for the

model for further processing. We divided the preprocessing into two sub processes. The first step will be to cut and downsample the videos and the second step will cover the feature extraction.

4.1.1 Cutting and Downsampling

All of the videos were downsampled to 10 frames per second to speed up the computation and save space. Furthermore, only the video portions from 00:30 to 4:30 were used for interaction prediction to remove any unintended effects from the beginning and end of the video. Here for the python library ffmpeg was used.

4.1.2 Feature Extraction

In the past years more and more face detection and analysis opensource software become popular. For the extraction of the Features we used OpenFace.(Tadas Baltrusaitis, 2021)
OpenFace is an open-source platform that incorporates facial behaviour analysis algorithms such as facial landmark identification, head pose tracking, eye gaze tracking, and estimation of facial Action Units, which are good intuitive cues about the subjects' attention (Tadas Baltrusaitis , Peter Robinson, Louis-Philippe Morency). For each frame, OpenFace was used to extract a 31-dimensional feature set:

- Head Pose: It is a 6-dimensional feature collection that describes the position of the head and its rotation in radians around the x, y, and z axes.
- Eye Gaze: It comprises two 3-dimensional eye gaze dircetion vectors for both eyes and one 2-dimensional averaged eye gaze direction in radians for both eyes. The eye gaze features have a complete dimension of 8.
- Facial Action Units (AUs): We use the detected strength (from 0 to 5) of these extracted AUs to form a 17-dimensional feature vector.

The extracted 31-dimensional frame-level features were then used to create segment-level features for each video. The creation is divided into 3 steps:

1. The original features were concatenated with the 1st and 2nd order delta coefficients, yielding a 93-dimensional feature vector for each frame.
2. We group 20 frames into one segment and apply 6 moment functions to each. The moment functions we use are min, max, standard deviation, kurtosis and skewness. For this purpose we assembled a sliding window. This sequentially goes through the frames, combines 20 frames into one segment, applies the 6 moment functions and recursively take the next 20 frames. As a result, each segment's cumulative function vector consists 31* 3* 6 = 558 attributes. Then we normalized all of the segment-level characteristics to zero mean and unit variance.
3. For computational efficiency, we used Principal Component Analysis (PCA) from the python sklearn library to reduce the normalized features to a lower dimension of d. The first principal variable is the function that induces the most variation. The second principal variable is the function that is responsible for the second largest variation, and so on.

4.2 Neural Network

Recurrent Neural Networks (RNN) models have recently become popular for automatically learning useful features from sequential results. A RNN is a type of neural network with loops that allow data to be stored within the network. RNNs use their logic from past interactions to predict future events. Recurrent models are useful because they can be applied to sequence vectors, allowing to handle more complex tasks.
Following this pattern, we investigated the use of RNN models for setting up our AI. RNNs that use gates to monitor the flow of information abstracted from a sequence are known as Long Short-Term Memory (LSTM). We used Bi-directional LSTM (BLSTM) in this study to help maintain the temporal dynamic characteristics. TensorFlow and Keras are used to implement these NN models and the attention function.

In the figure 2 one can see the structure of RNN with the different layers. The initial input are the features of each segment we previously prepared. Those were fed into a dense layer with 16 neurons and no activation mechanism, followed by a second dense layer. Each segment will have a new encoding as a result of this.
The third hidden layer is the BLSTM. Build up of 16 hidden states the BLSTM gets the

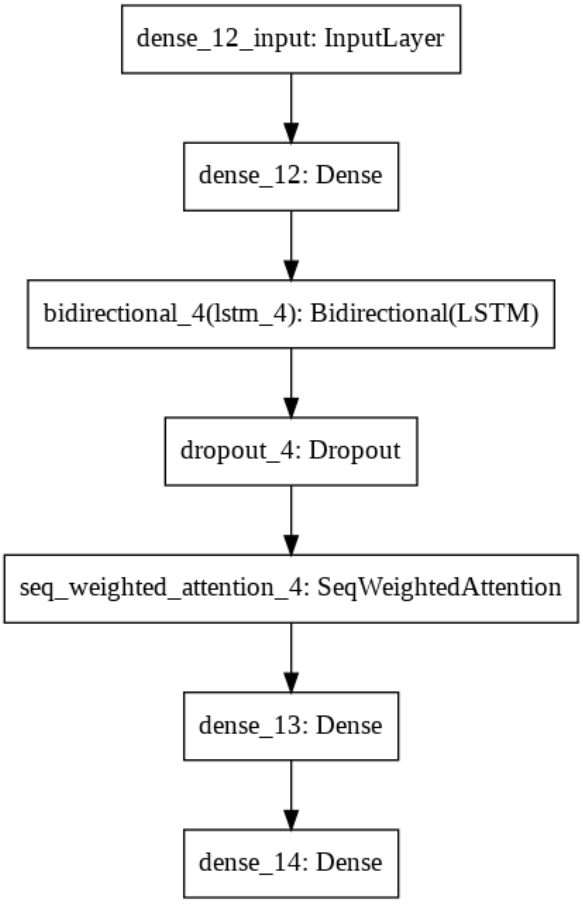

Figure 2: Neural Network for Engagement Tracking

output previous fully connected dense layer. This performs a new encoding in both the forward and backward temporal directions. The BLSTM layer flows into a dropout layer to average the hidden-state sequences and avoid overfitting, resulting in a fixed-dimension representation of the entire video.

The Deep Learning's Attention mechanism is built on the idea of focusing the attention, and it pays more attention to certain variables when processing data. It manages the interdependence of input and output. The idea behind using the attention mechanism is that a subject's level of engagement will change over time, and annotators can pay attention to various segments. For more accurate predictions, a NN model must pay more attention to certain key segments.(Colin Raffel, Daniel P. W. Ellis)

Finally, a two-layer dense layer was used to regress strength to a float number between 0 and 1. The first layer contains 16 neurons that are activated by reLU, and the second layer contains 1 neuron that is activated by sigmoid.

In the training step, we converted the labels into one-hot code vector. Intensity 0, 0.33, 0.66, 1 would be represented as $[1,0,0,0]$, $[0,1,0,0]$, $[0,0,1,0]$, $[0,0,0,1]$. The loss function would be taken by the Categorical Crossentropy. As the former part, we used the Binary Cross Entropy which somehow misunderstood the one-hot code. Despite the four different classes it has, it could be concluded into a regression problem even though it behaves more like a classification event. The MSE could also show some information concerning the training model. However, the Categorical Cross Entropy does shows a more stable accuracy for the model.

After training our model one can see the following results as presented in the figures 3 and 4. We implemented the 40 epochs and 32 batch size to train and validate the mode. The validation occupation would be 0.2. From the figure 3 we can conclude that our model overfits the data because the validation loss is higher than the training loss. From the figure 4 which indicates a growing accuracy around 0.57 we can conclude that our model performs a reasonable result on new data.

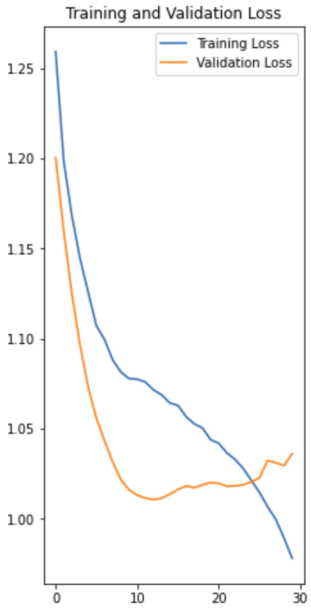

Figure 3: Loss Plot

5 Evaluation

In the evaluation step, we converted the one-hot code labels into the normal label representations. The maximum value of the element located in the

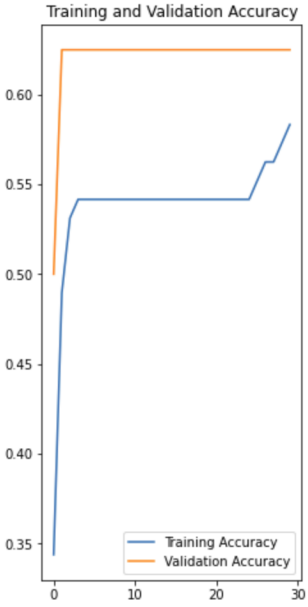

Figure 4: Accuracy Plot

vector would be assigned to the specific label for testing stage.

As a comparison dataset, 45 videos were used to validate the model. The mean square error is the evaluation metric used. Table 1 displays the outcome of the base model as well as the evaluation of our model. The MSE indicates how well the model predicts data. The lower the MSE, the better the model performs in general. Our model has nearly doubled the MSE of the Base model, as in Table 1 can be seen.

Table 1: Evaluation for the base and our mode.

Model	MSE
Base Model	0.049
Our Model	0.096

6 Conclusion and Future Work

In this project, we attempted to incorporate a interaction monitoring model based on the paper from (Cheng Chang , Cheng Zhang, Lei Chen, Yang Liu).

This paper (Abhay Gupta, Arjun D'Cunha, Kamal Awasthi and Vineeth Balasubramanian) presents DAiSEE, a dataset designed to aid research and development in the field of user interaction identification. The DAiSEE dataset, which contains video recordings of subjects in an e-learning environment annotated with crowdsourced labels for engagement, anger, uncertainty, and boredom. The dataset captures "in the wild" settings that are common in the real world, and it will be made public, along with the crowd's individual annotations, to encourage open study.(Abhay Gupta, Arjun D'Cunha, Kamal Awasthi and Vineeth Balasubramanian)

The engagement levels from the DAiSEE dataset can be easily applied to our model. It has in total 4 levels like we use to train our model. The results of this massive dataset may be used to improve accuracy.

References

Abhay Gupta, Arjun D'Cunha, Kamal Awasthi and Vineeth Balasubramanian. Daisee: Towards user engagement recognition in the wild. *LATEX CLASS FILES*, 14(8).

Marian Stewart Bartlett, Gwen Littlewort, Mark G Frank, Claudia Lainscsek, Ian R Fasel, Javier R Movellan, et al. 2006. Automatic recognition of facial actions in spontaneous expressions. *Journal of multimedia*, 1(6):22–35.

Cheng Chang, Cheng Zhang, Lei Chen, and Yang Liu. 2018. An ensemble model using face and body tracking for engagement detection. In *Proceedings of the 20th ACM International Conference on Multimodal Interaction*, ICMI '18, page 616–622, New York, NY, USA. Association for Computing Machinery.

Lei Chen, Gary Feng, Chee Wee Leong, Blair Lehman, Michelle Martin-Raugh, Harrison Kell, Chong Min Lee, and Su-Youn Yoon. 2016. Automated scoring of interview videos using doc2vec multimodal feature extraction paradigm. In *Proceedings of the 18th ACM International Conference on Multimodal Interaction*, ICMI '16, page 161–168, New York, NY, USA. Association for Computing Machinery.

Cheng Chang , Cheng Zhang, Lei Chen, Yang Liu. An ensemble model using face and body tracking for engagement detection. *ICMI*.

Colin Raffel, Daniel P. W. Ellis. Feed-forward networks with attention can solve some long-term memory problems. *ICLR 2016*.

Tadas Baltrusaitis. 2021. Openface.

Tadas Baltrusaitis , Peter Robinson, Louis-Philippe Morency. Openface: an open source facial behavior analysis toolkit.

Jacob Whitehill, Zewelanji Serpell, Yi-Ching Lin, Aysha Foster, and Javier R Movellan. 2014. The faces of engagement: Automatic recognition of student engagementfrom facial expressions. *IEEE Transactions on Affective Computing*, 5(1):86–98.