Paper No: SC-19 **Smart Computing**

Student concentration level monitoring system based on deep convolutional neural network

U. B. P. Shamika*
Department of Statistic and Computer Science,
University of Kelaniya, Sri Lanka
shamikapawani137@gmail.com

P. K. P.G. Panduwawala
Department of Statistic and Computer Science,
University of Kelaniya, Sri Lanka
pkgkpanduwawala@gmail.com

W. A. C. Weerakoon
Department of Statistic and Computer Science,
University of Kelaniya, Sri Lanka
chinthanie@kln.ac.lk

K. A. P. Dilanka
Sri Lanka Telecom PLC
Research & Development Division, Sri Lanka
pasindu@slt.com.lk

Abstract **- As synchronous online classrooms have grown more common in recent years, evaluating a student's attention level has become increasingly important in verifying every student's progress in an online classroom setting. This paper describes a study that used machine learning models to monitor student attentiveness to distinct gradients of engagement level. Initially, the experiments were conducted using a deep convolutional neural network of student attention and emotions exploiting Keras library. The model showed a 90% accuracy in predicting attention level of the student. This deep convolutional neural network analysis aids in identifying crucial emotions that are important in determining various levels of involvement. This study discovered that emotions such as calm, happiness, surprise, and fear are important in determining a student's attention level. These findings aided in the earlier discovery of students with poor attention levels, allowing instructors to focus their assistance and advice on the students who require it, resulting in a better online learning environment.**

Keywords - Convolutional Neural Network, emotion, Keras, Machine Learning, online learning, student involvement

I. INTRODUCTION

Emotions have an important role in education and in many facets of human existence. Emotions are widely accepted to exist and to be judged. Student involvement is an essential notion in today's education system, and how much information the student receives is equally significant for learning.

The advancement of sophisticated teaching approaches, along with greater computational power, has investigated and resolved numerous research challenges linked to student involvement in the traditional classroom setting, with favorable outcomes. Despite these benefits, current global events have forced students to adjust to the online classroom model. A normal in-person classroom format helps students extend their concentration, develop their critical thinking, and reinforce their meaningful learning experience.

As a result, the research component has expanded to include the issues and obstacles encountered by students during synchronous online classes. Online learning has exploded in popularity in recent years, and it has become a necessary method of continuous learning in the midst of a crisis. Knowing the attention level of students in an online classroom setting is critical for creating an adaptive learning system. Emotions and facial expressions are important indicators that instructors use to determine a student's attention level, but this is not feasible when learning takes place in a digital environment.

Because of the COVID-19 epidemic, online learning and synchronous online classrooms have become a means of education in recent days. Recognizing students' attention levels with the system they are engaged in working with can change how any teacher interacts with their pupils. Identifying student attention levels will allow you to have a better picture of how they interact with the system and modify your teaching techniques. It also aids in recognizing and categorizing kids depending on their degree of attentiveness. The success of online classrooms is dependent on the outcome of students' knowledge and participation.

Other studies in this field focus on recognizing students' varied emotions (happy, sad, angry, puzzled, disgusted, astonished, calm, neutral) during lectures, laboratories, and class research. The majority of current research in this sector has largely focused on measuring a student's emotional state. Because there is no association model between a student's degree of involvement in class and their emotional state, such research is restricted in its value to teachers.

As a result, in order to make things easier for the teachers, research was conducted to determine if a student is attentive or not throughout class (binary classification on attentiveness). It is always useful to know if students are attentive or inattentive, but most of the time, students are not at either of these extremes. In practice, a student might be half attentive during lectures as well. As a result, a student's attention level may not always be restricted to 0 or 1.

A. Background

We broadened the study to see if there are several categories for classifying student involvement. Therefore, we used a multi-level categorization of student attention level (attentive, partly attentive, and inattentive) in an online classroom setting. The benefit of this approach is that it allows teachers to detect inattentive and partially attentive students early on and offer the necessary assistance, resulting in a better online learning environment.

We suggested a system architecture that makes use of machine learning techniques and a computer vision service. Machine learning techniques are utilized to create a prediction model for each degree of student attention. The computer vision service is utilized to determine the pupils' emotional states. A model is constructed to link emotional states with the level of attentiveness of the pupil.

The first result is the output of one of the most common machine learning models, the Deep Convolutional Neural Network (CNN). Based on their facial expressions, this model was utilized to recognize student involvement. CNN scored the greatest average accuracy of 90.4 percent in the model, suggesting that it is absolutely possible to construct a prediction model for varying degrees of student involvement using information acquired from a recorded video.

The final result highlights the importance of emotion analysis and the prediction model of student attention levels in an online class environment using regression analysis.
The rest of the paper is organized as follows. Section II presents the related work. Section III introduces three algorithms and web scraping techniques are used. In section IV, the results and discussions are presented. Final Remarks and References are mentioned in Section V and VI, respectively.

II. Related Work

Monitoring the student learning process and delivering feedback to teachers in the classroom is a recent breakthrough in automated learning analytics. This notion of real-time feedback is made feasible by building the feature set with kinetic data collected from the Kinect One sensor device. In this study, seven different classifiers were evaluated to predict student attentiveness across time and average attention levels [1].

A methodology for detecting student emotions from student interactions with a cognitive tutor for mathematics was described. Cognitive tutors are programmed to respond to the student's actions inside the user interface. The software's log data was gathered, and observations were carried out in the school's computer lab. To evaluate the collected data, classification techniques such as decision tree, step regression, and naive bayes were utilized. The detectors evaluated on re-sampled data obtained 19% more accuracy than the set base rate [2].

A study was undertaken to increase student engagement in E-learning platforms by extracting mood patterns from their facial characteristics. The study aids in the assessment and identification of gaps in sustained attention by a student during an E- learning session. Analyzing moods based on a student's emotional states during an online lecture yielded data that could be easily used to improve the efficacy of the content delivery mechanism inside the E-learning platform. The study looks at whether facial expressions are the most important form of nonverbal communication and identifies the most prevalent facial characteristics that reflect a student's interest in a lecture. To train the models, such as the radial-based Neural Network (NN) model, the Hidden Markov Model, and the Support Vector Machine, a neural network method was employed (SVM). The outcome demonstrates a significant connection with feedback and a success rate of more than 70% in measuring the student's mood [3].

Early on, researchers established a link between visual attention and sadistic eye movement [4], employing the Viola-Jones algorithm to recognize face pictures [5]. To categorize the activities of eye movements, the Support Vector Machine (SVM) was used. These traditional principles served as the foundation for the development of different machine learning approaches.

III. Methodology

A. Dataset

The "DAISEE: Dataset for Affective States in E-Learning Environments" dataset contains 9068 video snippets captured "in the wild" from 112 users using an HD webcam setup to recognize user affective states, which are raw crowd annotated and associated with a standard annotation built by an expert team of psychologists. According to [6] research, each video was 10 seconds long since this length offered enough information for the labeling action. To mimic an E- learning environment, each participant was shown two separate 20-minute-long films. To capture a focused and comfortable atmosphere, one of the films was instructional and the other was recreational. It enables the capture of natural shifts in user attention levels. The students in this research ranged in age from 18 to 30 years old.

Because it was designed as an E-learning environment, the films were shot in a variety of settings, including dorm rooms, a busy lab area, and a library with varying lighting levels (light, dark and neutral). The video dataset was tagged with several emotional states such as boredom, confusion, engagement, and frustration. Each effect was further categorized into four labels: very low, low, high, and very high.

B. Data pre-processing

The first stage in our study project was to create a dataset from student photos collected in an E-learning environment. The video files were used to extract image frames. Every video is divided into 28 frames with a 20 minute gap. Fig. 1 shows the frames that we got from the videos. The picture dataset was difficult to set up since the films were shot in diverse places with varied lighting conditions. The difficulties included dark picture frames, students who were not within crucial proximity of the webcam, and students who were not within the image frame owing to other distractions. As a result, we concentrated on data collection by centralizing and cropping the face portions at identical pixel size for each frame.

Fig. 1. Cropping the face portions

C. *Feature extraction and labeling*

The extracted face characteristics for each engagement level should be relevant and carefully evaluated for labeling in order to accurately classify the photos based on their attention. For categorizing the pictures, two situations must be considered. Then extract features of each frame and save them in CSV using clipID.

image name	m	z1	z2	z3	z4	z5	z6	z7	z8	z9	z10	z2112	z2113	ClipID
0.png	130.6886	162.4867	143.3633	159.1567	158.26	158.88	159.2	159.52	162.1	163	163	247	63.16667	1100011002
10.png	130.0993	162.4567	143.3067	158.9233	158.89	158.67	159.04	160.24	163.2	163.1667	163	247.3767	63.71333	1100011002
100.png	130.4105	163.4667	143.7667	159.0467	158.5	158.85	159.46	161.2	163	163.04	162.9	247.1567	63.99	1100011002
110.png	130.3882	163.1033	142.88	159.1067	158.1	158.38	158.97	159	159	159.48	159	246.7733	64.99	1100011002
120.png	131.8049	163.17	144.06	158.98	158.25	158.68	159.2	159.76	162.28	163.16	163	246.8267	65.95667	1100011002
130.png	132.9625	163.62	144.12	159.2467	158.61	158.49	159.25	159.8	161.62	162.66	163	91.32333	83.61667	1100011002
140.png	132.1711	163.1467	143.4333	158.9	158.1	158.17	158.73	159.5	159.9	162.52	161.8	63.43667	85.28333	1100011002
150.png	131.8355	163.45	143.9233	158.85	158.1233	158.3	158.55	159	161.8	162.4	160	78.66133	74.75	1100011002
160.png	131.4715	163.1067	144.62	158.6133	158.5	158.92	159.19	159.7	162.7	163	162.72	69.50333	71.98	1100011002
170.png	131.2912	162.4033	143.0433	158.8467	158.4733	159.16	159.16	159.65	162.7	162.58	160.2	63.57	70.70333	1100011002
180.png	131.2555	161.8767	143.8	158.56	158.2	158.2	158.3	159.52	162.1	160.28	159	57.33333	71.82667	1100011002
190.png	131.4562	162.5767	143.77	158.8267	158.3333	158.29	158.73	159.5	162.4	162.5067	160.7133	55.07333	74.07	1100011002
20.png	130.2002	163.46	144.7167	159.1	158.6333	159.1	158.9667	161.6133	162.97	163.84	163.1	247.02	64.91333	1100011002
200.png	131.3993	163.5633	144.2167	158.8867	159.3333	159.3333	159.1133	159.71	162.1	163	163	55.49	71.35	1100011002

Fig. 2. Feature extraction

As mentioned in Section dataset, for the first scenario, video files were identified based on their effects and assigned a score ranging from 0 to 3. (very low to very high). Images from video files with engagement effects at level-3 and other effects at level-0 are tagged as "Attentive." Similarly, pictures from video files with engagement effects at level-2 and other effects at level-0 are classified as "Partially attentive." Finally, pictures from video files with engagement effects at levels 0 and 1 are classified as "Inattentive". The pictures in this case are tagged using the carefully studied indications from face and behavioral features given by [7]. Based on the visual signals detected in the video files for the above- mentioned situation, the authors' criteria for involvement level categorization were changed by adding characteristics of facial expression, hand movements, and body postures. Still, the attention level definition remained the same.

For our classification issue, when each sample belongs to just one class, the labels are mutually exclusive. As a result, the neural network labeled the pictures using sparse category cross-entropy. While training the machine learning model, it lowered execution time and saved memory space. For example, instead of [1,0,0] or [0,1,0] or [0,0,1], the pictures will be labeled with [1] or [2] or [3] instead of [1,0,0] or [0,1,0] or [0,0,1] as in one-hot encoding in Fig. 3.

```
             ClipID  Boredom  Engagement  Confusion  Frustration
0    1100011002.avi        0           2          0            0
1    1100011003.avi        0           2          0            0
2    1100011004.avi        0           3          0            0
3    1100011005.avi        0           3          0            0
4    1100011006.avi        0           3          0            0
...             ...      ...         ...        ...          ...
5353 4599990246.avi        0           3          0            0
5354 4599990247.avi        0           3          0            0
5355 4599990248.avi        1           2          1            1
5356 4599990249.avi        0           3          0            0
5357  459999025.avi        1           3          0            0

[5358 rows x 5 columns]
```

Fig. 3. Hot encoding

	image name	m	z1	z2	z3	z4	\
0	0.png	130.688572	162.486667	143.363333	159.156667	158.260000	
1	10.png	130.099319	162.456667	143.306667	158.923333	158.890000	
2	100.png	130.410485	163.466667	143.766667	159.046667	158.500000	
3	110.png	130.388157	163.103333	142.880000	159.106667	158.100000	
4	120.png	131.804935	163.170000	144.060000	158.980000	158.250000	
...	...	...	...	...	...	...	
29995	50.png	122.942326	176.426667	174.066667	172.246667	169.700000	
29996	60.png	122.902334	174.360000	172.900000	171.120000	170.706667	
29997	70.png	122.824133	175.030000	173.320000	171.306667	170.096667	
29998	80.png	122.539982	174.500000	172.370000	170.730000	169.770000	
29999	90.png	122.606157	174.700000	172.440000	172.150000	169.900000	

Fig. 4. Consider 300 data

Here we considered only 30000 in figure 4 data and merge feature CSV with label CSV (above two) using ClipId in Fig. 4.

```
       Boredom  Engagement  Confusion  Frustration
0            0           2          0            0
1            0           2          0            0
2            0           2          0            0
3            0           2          0            0
4            0           2          0            0
...        ...         ...        ...          ...
29995        2           3          0            3
29996        2           3          0            3
29997        2           3          0            3
29998        2           3          0            3
29999        2           3          0            3

[30000 rows x 2119 columns]
```

Fig. 5. Merge feature csv with label csv

Dropped boredom, confusion and frustration and considered only, the engagement levels in Fig. 5.

```
 2113      ClipID  Engagement
.6667  1100011002           2
.3333  1100011002           2
.0000  1100011002           2
.0000  1100011002           2
.6667  1100011002           2
  ...         ...         ...
.6667  1110031031           3
.0000  1110031031           3
.3333  1110031031           3
.3333  1110031031           3
.3333  1110031031           3
```

Fig. 6. Consider only engagement

D. *Splitting the dataset*

The dataset consists of 30000 preprocessed images with dimensions of 200 * 200 pixels. The dataset was randomized and divided into two phases: training and testing sets, as shown in figure 6. In this case, the training dataset is utilized to fit various models with weights defined by the prediction algorithm's accuracy and loss function.

We took a 30000 dataset and split it into 80% of the training dataset and 20% of the testing dataset. We had a training dataset of 24000 and a testing dataset of 6000.

```
print("Number of entries in each category:")
print("training: ", x_train.shape)
print("testing: ", x_test.shape)
```

```
Number of entries in each category:
training:  (24000, 2114)
testing:  (6000, 2114)
```

Fig. 7. Split dataset into training (80%) and testing (20%)

To avoid over fitting the network and fine-tune the model's hyperparameters, a validation set was employed. When the model encounters the training data values, it does not modify its weights. Instead, it aids in the specification of a stopping point for the back-propagation method. The trained model's efficiency was assessed using a test dataset. Each model's accuracy in the test data gives an unbiased assessment of the model's performance on unlabeled pictures and verifies the network's predictive capacity.

E. *Build a CNN to take the output.*

We created a CNN with an accuracy of 90.4% in order to increase student engagement in the E- learning platform. We used 10 iterations. Three densities were used to train the model: 100, 50, and 10.

F. Build a LSTM to take the output

We created a LSTM with an accuracy of 54.4% in order to increase student engagement in the E- learning platform. We used 10 epochs.

IV. EXPERIMENT AND RESULTS

A. Model evaluation outcome for CNN classifier

By adjusting the validation split to 0.2, we were able to employ 10 epochs with the default batch size of 50. For every validation loss inside the model, the density was set to 100, 50, and 10 in the early stopping callback method. It monitors the loss quantity and, when it finds improvements, it is 2% when three densities are used at the same time. Before finishing the 10 epochs, the loss function for our model hit a saturation point of around 0.22, and the total accuracy achieved a high of 90.6 percent. The CNN model was evaluated using the accuracy and loss graphs. When the dataset is balanced across classes, evaluating the model efficiency solely on accuracy and loss value obtained from the validation set may cause difficulties. Figure 8 depicts the construction of the CNN model.

```
Epoch 1/10
2400/2400 [==============================] - 283s 118ms/step - loss: 0.6299 - accuracy: 0.7062 - val_loss: 0.4908 - val_accurac
y: 0.7728
Epoch 2/10
2400/2400 [==============================] - 266s 111ms/step - loss: 0.4673 - accuracy: 0.7830 - val_loss: 0.4062 - val_accurac
y: 0.8282
Epoch 3/10
2400/2400 [==============================] - 267s 111ms/step - loss: 0.3856 - accuracy: 0.8214 - val_loss: 0.3493 - val_accurac
y: 0.8392
Epoch 4/10
2400/2400 [==============================] - 270s 112ms/step - loss: 0.3333 - accuracy: 0.8464 - val_loss: 0.3213 - val_accurac
y: 0.8535
Epoch 5/10
2400/2400 [==============================] - 275s 115ms/step - loss: 0.3011 - accuracy: 0.8610 - val_loss: 0.2918 - val_accurac
y: 0.8753
Epoch 6/10
2400/2400 [==============================] - 279s 116ms/step - loss: 0.2755 - accuracy: 0.8739 - val_loss: 0.3001 - val_accurac
y: 0.8787
Epoch 7/10
2400/2400 [==============================] - 276s 115ms/step - loss: 0.2565 - accuracy: 0.8829 - val_loss: 0.2877 - val_accurac
y: 0.8797
Epoch 8/10
2400/2400 [==============================] - 283s 118ms/step - loss: 0.2455 - accuracy: 0.8903 - val_loss: 0.2533 - val_accurac
y: 0.8970
Epoch 9/10
2400/2400 [==============================] - 281s 117ms/step - loss: 0.2339 - accuracy: 0.8953 - val_loss: 0.2378 - val_accurac
y: 0.9082
Epoch 10/10
2400/2400 [==============================] - 283s 118ms/step - loss: 0.2166 - accuracy: 0.9019 - val_loss: 0.2283 - val_accurac
y: 0.9060
```

Fig. 8. Build a CNN model

By adjusting the validation split to 0.4, we were able to employ 10 epochs with the default batch size of 10. When it finds improvements, it is 2% when three densities are used at the same time. Before finishing the 10 epochs, the loss function for our model hit a saturation point of around 0.13, and the total accuracy achieved a lower of 54.4 percent. The LSTM model was evaluated using the accuracy and loss graphs. When the dataset is balanced across classes, evaluating the model efficiency solely on accuracy and loss value obtained from the validation set may cause difficulties. Figure 9 depicts the construction of the LSTM model

```
Training Progress
Epoch 1/10
2400/2400 [==============================] - 12012s 5s/step - loss: 0.1349 - acc: 0.5404 - val_loss: 0.1335 - val_acc: 0.5447
Epoch 2/10
2400/2400 [==============================] - 35860s 15s/step - loss: 0.1334 - acc: 0.5517 - val_loss: 0.1339 - val_acc: 0.5447
Epoch 3/10
2400/2400 [==============================] - 35857s 15s/step - loss: 0.1338 - acc: 0.5475 - val_loss: 0.1336 - val_acc: 0.5447
Epoch 4/10
2400/2400 [==============================] - 8081s 3s/step - loss: 0.1336 - acc: 0.5476 - val_loss: 0.1337 - val_acc: 0.5447
Epoch 5/10
2400/2400 [==============================] - 9093s 4s/step - loss: 0.1336 - acc: 0.5476 - val_loss: 0.1337 - val_acc: 0.5447
Epoch 6/10
2400/2400 [==============================] - 13220s 6s/step - loss: 0.1335 - acc: 0.5474 - val_loss: 0.1336 - val_acc: 0.5447
Epoch 7/10
2400/2400 [==============================] - 8897s 4s/step - loss: 0.1335 - acc: 0.5476 - val_loss: 0.1335 - val_acc: 0.5447
Epoch 8/10
2400/2400 [==============================] - 18161s 8s/step - loss: 0.1335 - acc: 0.5476 - val_loss: 0.1336 - val_acc: 0.5447
Epoch 9/10
2400/2400 [==============================] - 7937s 3s/step - loss: 0.1336 - acc: 0.5476 - val_loss: 0.1336 - val_acc: 0.5447
Epoch 10/10
2400/2400 [==============================] - 8003s 3s/step - loss: 0.1335 - acc: 0.5476 - val_loss: 0.1335 - val_acc: 0.5447
```

Fig. 9. Build a LSTM model

Figure 10 depicts the connection between the accuracy of the training set and the validation set for each epoch. The graph shows that the accuracy of both the training and validation sets has risen with each epoch. It is not always necessary to take into account the validation learning curve's last data point with the best accuracy of the model. The greatest accuracy of the model reached epoch in our investigation was epoch 10.

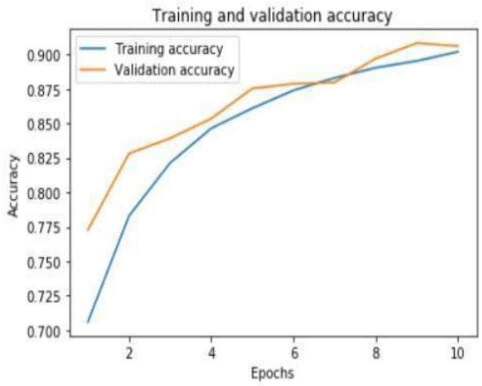

Fig. 10. Accuracy graph: Training vs. validation of CNN

Fig. 11 depicts the connection between the accuracy of the training set and the validation set for each epoch. The graph shows that the accuracy of the training set has risen up and after that I has gone down and in the same value, and validation sets have the same value with each epoch. It is not always necessary to take into account the validation learning curve's last data point with the best accuracy of the model. The greatest accuracy of the model was reached each epoch in our investigation.

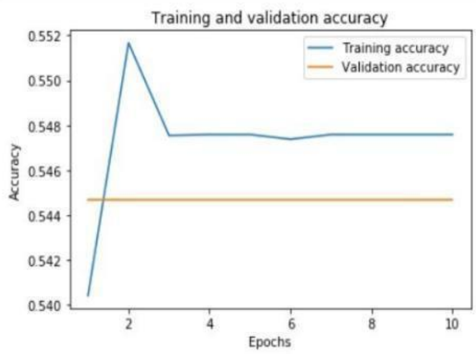

Fig. 11. Accuracy graph: Training vs. validation of LSTM

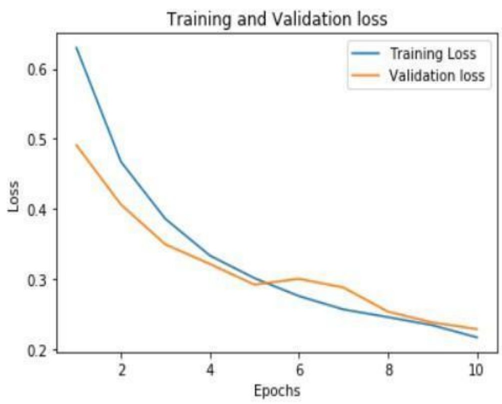

Fig. 12. Loss graph: Training vs. validation of CNN

Fig. 12 depicts the loss function connection between training and validation sets for each epoch. The graph terminates at epoch 10 with the patience parameter set to 6 due to the callbacks adjusted in the CNN model, since the validation loss function detected no progress.

Figure 13 depicts the loss function connection between training and validation sets for each epoch. The graph terminates at epoch 10 with the patience parameter set to 1 due to the callbacks adjusted in the LSTM model, since the validation loss function detected no progress.

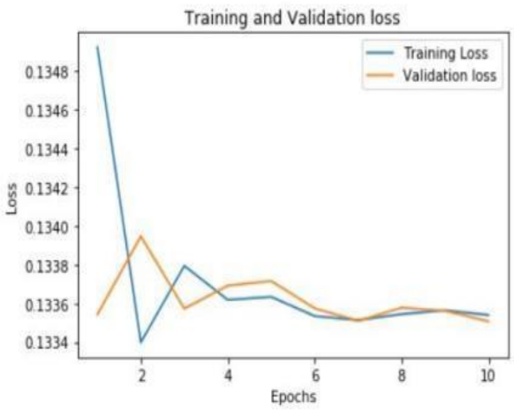

Fig. 13. Loss graph: Training vs. validation of LSTM

B. Result discussion

The efficiency and accuracy of forecasting student involvement levels was investigated in this study using a machine learning model. The machine learning model was evaluated using a balanced dataset. Based on the performance measures, it is possible to infer that the deep learning CNN model is more effective than LSTM. Despite its superior accuracy, the CNN model requires more time to train but the LSTM model wants more time than CNN. When compared to the LSTM models, the CNN model produced the largest proportion of erroneous classifications. In summary, the CNN model outperformed all other models in all measures, with the greatest accuracy of 90.6 percent.

V. Conclusion

The outcomes of this study enable teachers to properly detect inattentive and partially attentive pupils, which contributes to a better online learning environment. It enables teachers to help students in need, resulting in a better learning experience. Our research looked at three machine learning models for measuring student involvement based on their emotions. The CNN model was chosen as the appropriate machine learning model to measure a student's attentiveness based on their emotional state by the research methodology employed in this study, with a prediction accuracy of 90.6 percent. The influence of emotion state rage on the connection between emotion states and student engagement levels was also investigated in this study. Understanding the confounding influence of rage on other emotional states has enabled us to identify important emotions displayed by inattentive and partially attentive pupils statistically. Based on the findings of this study, we can infer that the deep CNN model provides a dependable and accurate platform for assessing different gradients of student involvement based on their facial expressions.

This study effort can be developed in a variety of ways. For future studies, the CNN model may be modified to use computational resource-intensive architectures such as VGG16, VGG19, and ResNet, which would increase the machine learning model's prediction accuracy.

This study could be expanded by incorporating a broader range of engagement levels to gain a more detailed understanding of students' attention levels and facial expressions. Furthermore, the research platform may be enhanced by integrating a web- based application that converts live video files into pictures, providing real-time data to the prediction model. A student survey may be included at the end of each online session to produce user-driven feedback data points to enhance and validate the machine learning models' prediction metrics.

Another goal of the research is to do picture auto- labeling rather than manual labeling. Once the relationship and relevance of emotions and engagement levels has been painstakingly determined, the cloud-based program may function as an AI expert in the labeling process. This approach is useful for dealing with big datasets.

References

[1] K. Janez Zaleteli, "Predicting students' attention in the classroom from Kinect facial and body features," 2017.

[2] S. j. R. S. P. B. A. C. D. Doborah Rudnick, "The Role of Landscape Connectivity in Planning and Implementing Conservation and Restoration Priorities. Issues in Ecology".*2012.*

[3] A. K. B.K. Poornima, "Predicting learner preferences from emotions using Deep Learning Techniques," 2016.

[4] S. Heiner Deubel Werner, "Saccade target selection and object recognition: Evidence for a common attentional mechanism," 1996.

[5] L. H. W. W. W. Thomas A. Dingush, "Development of models for on-board detection of driver impairment," 1987.

[6] Z. S.C. L. F. J. R. M. Jacob Whitehill, "The Faces of Engagement: Automatic Recognition of Student Engagement from Facial Expressions," 2014.

[7] S. E. H. Erin S. Lane, "A New Tool for Measuring Student Behavioral Engagement in Large University Classes," 2015.