CSCE 580- Introduction to AI

Final Exam- 12/11/2025

Shruti Jadhav

Q1a)

a) Write the name of the paper and student presenter:

Paper: Understanding Emotional Body Expressions via Large Language Models

Presenter: Yamuna Bobbala

b) Now, can you think and create a new example exemplifying the main conclusion of the paper

In one scenario, a 3D skeleton sequence captures a person engaging in fast, forceful "angry striding." The individual's torso is noticeably leaned forward, their steps are sharp and rapid, and their arms swing with exaggerated acceleration. Throughout the sequence, the person's hands remain tightly closed into fists, and their shoulders appear elevated and tense. These combined cues of forward throwing of the body, high-intensity temporal spikes in movement, and the contracted upper body, form a distinctive pattern of aggressive, high-arousal motion within the skeleton data.

LLM Output:

Emotion Recognition: "This is an angry person."

Generated Explanation: "The person moves with fast, forceful strides, leans their torso forward, and swings their arms sharply while keeping their fists tightly clenched. These tense and aggressive movements indicate anger."

c) Describe how the conclusion is supported in your example.

This example supports the paper's conclusion that 3D skeleton movement can be treated as a structured "language," enabling an LLM to both recognize emotional states and generate meaningful explanations. In this case, the model converts the movement sequence into semantic, spatial, and temporal tokens that capture details such as clenched fists, rapid strides, and forward-leaning posture. Drawing on these multi-

granularity tokens along with its background knowledge about how anger is expressed through body movements, the EAI-LLM correctly infers that the person is angry and is able to articulate why. By producing both an accurate emotion label and an explanation referencing tension, forceful motion, and aggressive posture, the model demonstrates its ability to integrate skeleton-token inputs with linguistic reasoning. This reflects the paper's conclusion that the proposed tokenizer and unified skeleton representation allow LLMs to achieve interpretable, human-like emotion understanding from raw 3D motion data.

Q2) Attendance Audit System

a) Describe your data preparation, if any, and why or why not.

I prepared the data by downloading the provided folder. I downloaded zip file with all the images, then unzipped and placed the folder in the location with the notebook file.

During my initial tries with using other Optical Character Recognition (OCR) tools, I compressed and cropped the images as I was having difficulty getting the models to provide accurate outcomes.

However, when these approaches failed, I left the original downloaded file sizes the same to do analysis with a different model.

b) Describe your steps to create a model – pre-trained, your own, manual

Initial failed models:

Initially I tried using various OCR models such as Tesseract, Easy OCR and even a transformer-based model TrOCR. However, the main issue I encountered during the use of these models were incoherent and incomplete results from text extraction that led to inaccurate analysis result.

For example, while using TrOCR a lot of the information was missing or misread. While this model had no problem with hallucination, it really struggled with reading the messy and incomprehensible handwriting. This led to the model dropping lines with student names and therefore an inaccurate count. Furthermore some handwriting was misread as numbers which not only led to interfering with the actual attendance count, but also created wrong minimum and maximum values. The way I checked this was by also checking the raw maximum values before they were bound to a logical attendance number. As a result, almost all the raw max values were either in hundreds, thousands (2025 made sense but other values did not) and one value even in 50,000s. Furthermore, the maximum

attendance values did not correlate with the evaluation dates. This led me to decide to move on with a different model.

Next, I tried using the tesseract model. However, the problem with this model was major hallucination and missing the provided text. Once I read the incoherent output by testing a cropped part of the image, I moved on to a different model. Using the EasyOCR produced the same problem of hallucination. It produced incoherent results even when provided with a cropped area to analyze the class number and date.

I then used the pretrained LLM- the Llava model (llava-hf/llava-1.5-7b-hf) using huggingface. I chose the 7MB model because it was the lighter version that I could run on my computer and colab. This model performed the best among the failed models. It was able to extract text much better as coherent names were produced when visually checked with one of the text files. However, this model still struggled with the messy handwriting as when the output was checked it was found that there would be a few entries that it would keep repeating in the response for attendance names/count, therefore leading to an inflated count.

Final model used:

Finally, for my last attempt in using a pretrained LLM I used the Text Extractor 5.1 model by ChatGPT found here: https://chatgpt.com/g/g-doLYgv5ks-text-extractor. I first tried to use this model by using the API method however I ran into the issue with billing credits. As a last resort I started prompting the images, one at a time using the prompt:

“Please extract text from this image. Put it in csv file with column headings: number, name, username, class, and date. Additionally, put class number and date as the file's naming convention. The date column data, put it in format mm/dd/yyyy. Additionally, file naming convention use ClassX_yyyy-mm-dd for example Class3_2025-08-26.”

Using this pretrained LLM, the model returned clean structured csv for each attendance sheet. This approach was chosen because training a model from scratch and fine-tuning OCR models as mentioned above were significantly more time-consuming and less accurate than a high-performance, pre-trained LLM optimized for vision & text extraction tasks. Furthermore, the model was also robust enough against discrepancies in data on the handwritten attendance sheet for example the serial numbering not being correlated to the number of students in class, however this model picked up on it and only counted the number of student entries.

c) Answer the questions from your analyses using the models (**ALSO PROVIDED IN CODE)**

a. What are the number of classes and their dates?

	ClassNumber	ClassDate
0	1	2025-08-19
1	2	2025-08-21
2	3	2025-08-26
3	4	2025-08-28
4	5	2025-09-02
5	6	2025-09-04
6	7	2025-09-09
7	8	2025-09-11
8	9	2025-09-16
9	10	2025-09-18
10	11	2025-09-28
11	12	2025-09-25
12	13	2025-09-30
13	14	2025-10-02
14	15	2025-10-07
15	16	2025-10-14
16	17	2025-10-16
17	18	2025-10-21
18	19	2025-10-23
19	20	2025-10-28
20	21	2025-10-30
21	22	2025-11-04
22	24	2025-11-11
23	25	2025-11-13
24	26	2025-11-18
25	27	2025-11-20

b. What is the median class attendance per class?

Median Attendance is: 33 students `33.0`

c. What are the dates with lowest and highest attendance?

```
Lowest attendance of 14 students on 2025-11-20
Highest attendance of 49 students on 2025-08-21
```

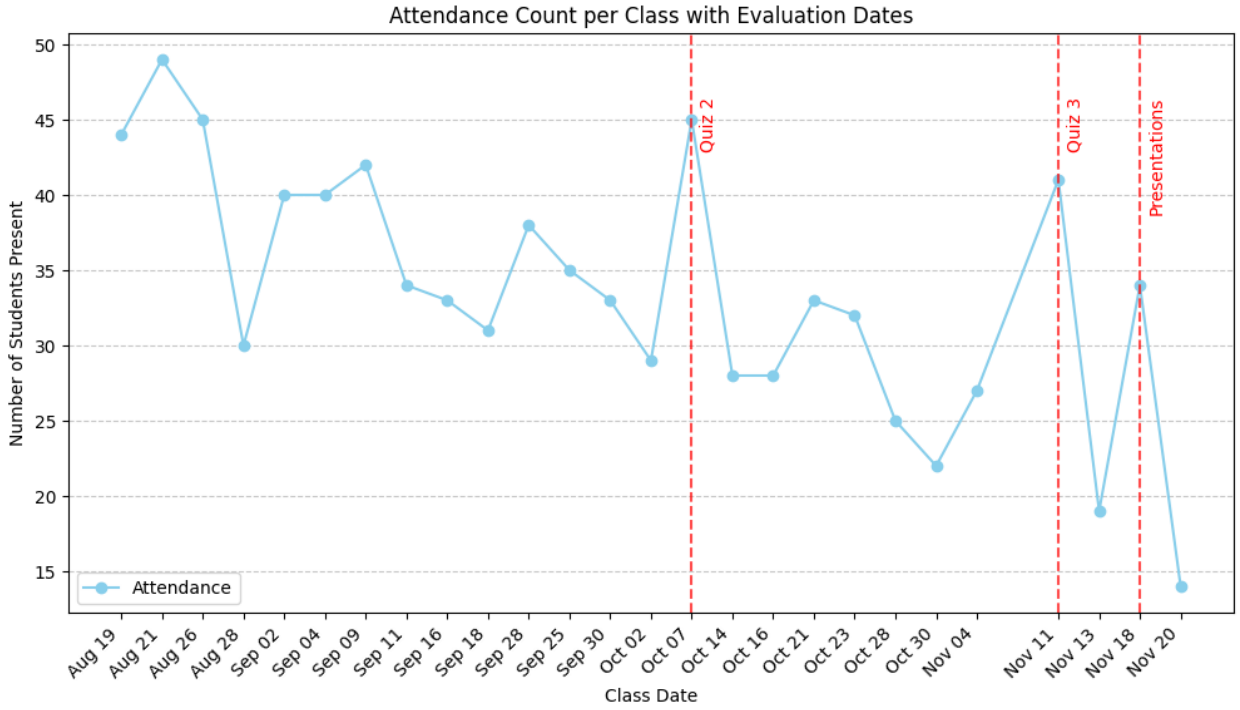
d. Is there a correlation of high attendance with course evaluations dates? When is the attendance highest?

As seen above, the highest attendance is 2025-08-21.

However, there is a correlation among high attendance with course evaluation dates. Furthermore, the highest attendance among the course evaluation dates occurs on Oct 7 as below:

ClassNumber	15
ClassDate	2025-10-07
AttendanceCount	45
IsEvaluation	True
EvaluationType	Quiz 2

Here is the graph that shows the attendance across the course, as well as the course evaluation dates.



d) If you had more time (say a week), what more could you have done to improve performance?

If I had more time to improve the system, I would focus on enhancing the OCR quality and reducing the amount of manual work required in the pipeline. The dataset of 27 attendance-sheet images was too small to meaningfully fine-tune an OCR model, but with more time I could construct a much larger labeled dataset by automatically or manually cropping each sheet into smaller segments for example 2–3 names per crop. These labeled text snippets could then be used to fine-tune a model such as Llava, TrOCR, or another vision–language model on the handwriting and formatting style specific to these attendance sheets. This would reduce name-recognition errors and improve consistency across images. I would also improve the overall preprocessing pipeline. Currently, I process images one at a time to maintain accuracy. With more time and a better OCR model, I could automate this step using methods such as detecting the table region automatically

or applying light image cleanup to make text easier for any OCR model to read. These do not require expertise in image processing but would still offer measurable improvements.

Finally, if I had API access without rate or quota limits, I could build a fully automated end-to-end script that does everything in one place such as takes in all images at once, preprocesses them, extracts text, validates the CSV outputs, and produces the final analytic summaries. This would significantly reduce human involvement, reduce spending time switching between two platforms for analysis, and increase scalability.