Generalizable Detection of Student Engagement in Online Learning Environments

No Author Given

No Institute Given

Abstract. Automated student engagement recognition in online learning is crucial as it enables the teachers to adapt the content delivery to improve learning. In this paper, we explore a method that finetunes a pretrained vision language model (VLM) to recognize student engagement markers. Our model learns to avoid incorrect answers during finetuning by using the emerging direct preference optimization techniques on self-generated preference pairs based on the correct and incorrect VLM answers. On publicly available student engagement datasets, our model shows superior performance over other approaches and substantially better generalizability over the traditional vision methods.

Keywords: Direct Preference Optimization · VLM · Engagement

1 Introduction

With the development of network and multimedia technology, online learning environments have become viable in education at all levels. Its adoption has skyrocketed following the COVID-19 pandemic. Online learning is now well-entrenched and increasingly preferred because of its numerous benefits both to the students (e.g., avoidance of physical transportation) and the providers (e.g., lower cost and ability to handle larger classes). However, online learning has some definite disadvantages due to the inability of the instructor to make face-to-face contact and assess comprehension or lack thereof. Research has demonstrated that student engagement provides a positive influence on academic achievement [1]. In the online learning environment, automated monitoring of the student engagement level becomes essential because scanning the camera feeds from a large number of students is distracting and impractical; furthermore, it requires that video stream from each student be transmitted to the instructor. Ideally, we would like to obtain the same kind of assessment that experienced teachers can easily do in real classrooms without the necessity to transmit student videos.

Traditionally, visual behavior recognition has been done via specially designed deep learning models that must be trained on well-crafted labeled datasets. While such methods can do extremely well in recognizing the targeted behaviors, they are limited by the difficulties in model crafting and the creation of good quality and varied training datasets. They also tend to struggle in generalizing well to out-of-distribution samples, as shown in our experiments.

Large Vision-Language Models (VLM) have recently been researched extensively to provide good descriptions (or "captions") of activities in images and short video segments. In this paper, we focus on determining how well the students are engaged in a virtual classroom environment. Video analytics determines this primarily based on the facial expressions and gestures, although the VLM based method explored here considers all visual features implicitly. We explore an open-source Vision Language Model, MiniGPT-4 [2] with Vicuna [3], a multi-modal chatbot based on LLaMA [4]. We further finetune it for a few student engagement marker tasks.

To enhance the generalizability of the model, we propose a novel variant of Direct Preference Optimization (DPO) [5] method that finetunes the VLM using self-generated preference pairs. In particular, we design a specialized DPO finetuning method that yields a VLM capable of accurately predicting student engagement markers from images (Sec. 2.2). Our design includes a self-generation pipeline to capture the broad distribution of the generated answers so that the preference pairs used to optimize the VLM can capture the actual distribution of incorrect answers. The proposed specialized DPO finetuning can be applied in any scenario with a labeled dataset.

We compare our finetuning results with various deep learning-based vision models and show that our method performs substantially better in predicting student engagement markers. We also evaluate the generalizability of our finetuned model to out-of-distribution samples by applying it to a different dataset. We focus on analyzing still images as opposed to videos since recent image-based VLMs are able to run on end-user devices in real-time, e.g., Llama 3.2 (Meta) and Molmo (Allen AI).

The rest of the paper is organized as follows. Section 2 describes the essential finetuning details of our method on the student engagement dataset. Section 3 explains the experimental setup, compares our method to other vision and VLM models, and analyzes the results. Finally, section 4 concludes the paper.

2 Methodology

2.1 Reinforcement Learning with Human Feedback

One challenge in training LLMs and VLMs is that it is difficult to measure the quality of output from these models automatically. Reinforcement Learning with Human Feedback (RLHF) provides a way for humans to be a source of the reward model without explicitly modeling the rewards. In LLMs and VLMs, RLHF is used to finetune the models, getting humans to choose between various outputs and learning a model to estimate the rewards.

Pretaining stage trains the language model to predict the next token given the prior text information using the large collection of datasets for natural language processing tasks. The loss used in the pretraining stage is normally the cross-entropy loss.

Supervised Fine-tuning (SFT) stage finetunes the pre-trained language model on datasets for specific downstream tasks. The datasets are normally

high-quality datasets with appropriate instructions (prompts) and reasonable responses. The model obtained after the SFT finetuning is denoted as π_{ref}.

Preference sampling and Reward Learning: For a dataset D, for each input x, there is a pair of preferred/dispreferred answers (y_ω, y_ι), where y_ω denotes the answer that human labelers expressed preference for and y_ι denote the dispreferred answer. We can model this by a latent reward model r^* that generates the underlying preferences. A commonly used approach to model preferences is assuming the probability $p^*(y_\omega \succ y_\iota|x)$, which represents the probability of preferring answer y_ω over answer y_ι, as a sigmoid σ of the reward difference

$$p^*(y_\omega \succ y_\iota|x) = \sigma(r^*(x, y_\omega) - r^*(x, y_\iota)) \tag{1}$$

We need to use a reward model r_ϕ to estimate the true reward of human preference via maximum likelihood since the true reward function is not accessible. For this, we minimize the negative log-likelihood loss.

$$\mathcal{L}_R(r_\phi, D) = -\mathbb{E}_{x,y_\omega,y_\iota\sim D}[\log \sigma(r_\phi(x, y_\omega) - r_\phi(x, y_\iota))] \tag{2}$$

Reinforcement Learning Optimization: This phase uses the learned reward function to further optimize the language model. To prevent model collapse and maintain the proximity to the distribution of reference model π_{ref}, a KL divergence penalty is added.

$$\max_{\pi_\theta} \mathbb{E}_{x\sim D, y\sim\pi_\theta(y|x)}[r_\phi(x, y)] - \beta\mathbb{D}_{KL}[\pi_\theta(y|x) \parallel \pi_{ref}(y|x)]$$

where π_θ is the policy model that RLHF is optimizing, and β is the hyperparameter to restrict how far the model can deviate from the base reference model. Due to the non-differentiability, the objective is normally optimized with an RL algorithm such as Proximal Policy Optimization (PPO) [6]. PPO is a policy gradient algorithm that uses an objective function to find the best policy. The objective function minimizes the difference between the new and old policy. As a result, PPO avoids too large a policy update which may destabilize the learning.

2.2 Direct Preference Optimization

Direct Preference Optimization (DPO) [5] provides a simplification of traditional feedback-aligned LLMs. DPO avoids the Reinforcement Learning (RL) loop as RLHF and instead proposes a closed-form loss using a special choice of reward model parameterization in equation 2. Equation 3 shows the DPO loss, where y_ω is the preferred answer, y_ι the dispreferred answer, π_θ the policy model being optimized, and π_{ref} is the constraining reference model. The reference model provides stability in training and baseline for the policy model to improve against.

$$L_{DPO}(\pi_\theta, \pi_{ref}) = -\mathbb{E}\left[\log\sigma\left(\beta\log\frac{\pi_\theta(y_\omega|x)}{\pi_{ref}(y_\omega|x)} - \beta\log\frac{\pi_\theta(y_\iota|x)}{\pi_{ref}(y_\iota|x)}\right)\right] \tag{3}$$

One can rewrite the DPO loss to show that it maximizes the margin between preferred and dispreferred answer pairs:

$$r_\omega = \log(\pi_\theta(y_\omega|x)) - \log(\pi_{ref}(y_\omega|x)) \qquad r_\iota = \log(\pi_\theta(y_\iota|x)) - \log(\pi_{ref}(y_\iota|x))$$

$$L_{DPO}(\pi_\theta, \pi_{ref}) = -\mathbb{E}[\log\sigma(\beta(r_\omega - r_\iota))] \tag{4}$$

For the student engagement detection problem, we work with multi-class labeled datasets where the labels indicate the different engagement behaviors of the student. Therefore, we propose to transform the labeled datasets to preference dataset $D = \{x^i, y^i_\omega, y^i_\iota\}_{i=1}^N$, where N denotes the number of data samples. For each input image x, we assign an answer pair of preferred answer and dispreferred answer (y_ω, y_ι). A simple approach is to utilize the given ground truth to generate a preference dataset by using the given label of each image as the preferred answer y_ω and using the other labels in the dataset as the dispreferred answer y_ι. However, this approach will introduce bias to the vision-language model since the distribution of possible dispreferred answers has a much wider range.

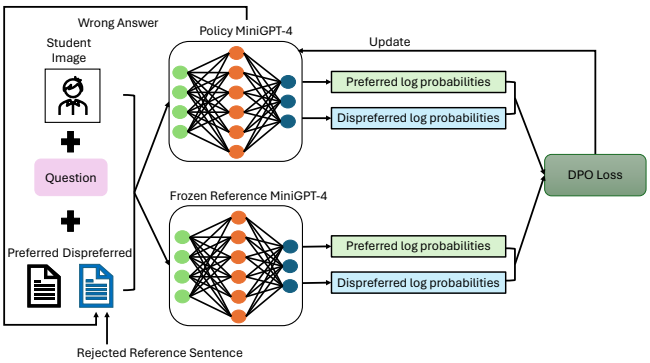

Fig. 1: DPO based Finetuning using a frozen reference & policy model.

In order to yield better answers, we propose to extend the reference data pairs D so that the dispreferred answers y^i_ι come from a more general distribution than ground truth. Since we do not have the distribution of generated answers of the vision-language model, **we use the wrong answers generated from the vision-language model itself to generate the dispreferred answers**. In practice, we form the dispreferred answers during training in the following manner. For a percentage of the time, if the generated answers from the Policy MiniGPT-4 are incorrect, we use these as dispreferred answers. Otherwise, if the generated answers are correct, we randomly choose from other labels besides the correct label as the dispreferred answers. For our experiments, this was done 30% of the time. The architecture of the DPO based finetuning is shown in Figure 1. We give example training pairs in the next section.

The probability $\pi_\theta(y|x)$ of any generated answer y is simply the product of the probabilities of its generated tokens. In contrast, when the answer y is provided, in order to compute $\pi_\theta(y|x)$, we only need to score how likely the LLM is to generate y if prompted with x. For this, we input x concatenated with y to the LLM and collect the log probabilities of all y tokens token by token.

2.3 Finetuning MiniGPT-4 on Engagement Datasets

We finetune MiniGPT-4 on the dataset using a set of prompts that are relevant to the student engagement problem. In order to achieve more general performance,

we optimize the MiniGPT-4 model with DPO using automatically generated preference pairs.

In our model, we introduce a finetuning that aims to generate specific image descriptions to describe behaviors or affective states indicating engagement levels of students. To be more specific, our approach uses student engagement categorization from the Student Engagement Dataset (SED) [7], the DAiSEE dataset [8], and the EngageNet dataset [9]. SED captures three student engagement markers. DAiSEE and EngageNet has four levels of engagement labels for each sample.

The following is an example of how we finetune with the SED dataset. SED uses the labels: 'looking at the paper', 'looking at the screen', and 'wandering'. We set the corresponding correct reference sentence as follows:

The person is looking down at the paper
The person is looking straight at the screen
The person is looking away

Since DAiSEE and EngageNet are composed of videos only, we subsampled them with a frequency 1 frame per second (fps), resulting in 10 frames per video clip. The 10 images are concatenated together and inserted into the finetuning and the VLM is asked to classify the student into one of "Highly-Engaged", "Engaged", "Barely-Engaged", and "Not-Engaged" classes.

Now we give two examples of preferred and dispreferred answers used for DPO finetuning. The VLM generated a wrong answer: 'Yes, the person is looking away. The blue headband is tied around their hair ...' We use it as a dispreferred answer, while the correct answer, 'The person is looking straight at the screen.' is used as the preferred answer. On the other hand, if the VLM generated a correct answer for a different image, 'The person is looking away.', we use it as a preferred answer and randomly select one of the remaining two answers as a dispreferred answer. 'The person is looking down at the paper.'

We use the same prompt template as MiniGPT-4. The instruction of MiniGPT-4 is randomly sampled from a predefined instruction set which contains different instructions for the image caption task. It is well-known that prompting can influence the output of LLMs [10]. In our finetuning, we use the instruction set containing questions that are used specifically to determine the students' behavior. After extensive experimentation with Vicuna, the instruction set (when our goal is a natural language output) is chosen due to the consistently better performances. The following shows one example of our prompts:

Given the following image: <Img>ImageContent</Img>. You will be able to see the image once I provide it to you. Please answer my questions.

###Human: <Img><ImageFeature></Img> Is the person looking straight at the screen? Is the person looking down at the paper? Is the person looking away?
###Assistant:

During the finetuning, one of the questions is selected randomly from the instruction set. We found experimentally that using questions in this way produced a higher baseline.

Additionally, since the output in JSON format is easy to evaluate with keyword search, we use the following query to enforce Vicuna to respond in JSON format:

<Img><ImageHere></Img> Given label set:['looking at the screen','looking at the paper','wandering'] Question: What is the type of activity in the image and which category from the given label set would you use to describe this activity type? Answer me in the JSON format like {'label': 'activity_type'}

We evaluated two different versions of the MiniGPT-4 model in our experiments. One version is that we perform our finetuning using the MiniGPT-4 (Vicuna) checkpoint. Another version is that we perform our finetuning after using the MiniGPT-4 (Llama2) checkpoint. MiniGPT-4 (Vicuna) achieves an accuracy of 95.2% after our finetuning whereas MiniGPT-4 (Llama2) yields 88.6%. Therefore, we use MiniGPT-4 (Vicuna) as our base model.

3 Performance Evaluation

3.1 Datasets Used

Our evaluation utilizes three public datasets, Student Engagement Dataset (SED) [7], DAiSEE dataset [8] and EngageNet [9]. SED contains both an unbalanced and balanced component. The unbalanced one contains 18,721 frames sampled at one fps from 400 videos collected from 19 students. It has samples divided into 3 categories 'looking at the paper', 'looking at the screen', or 'wandering'. The first two are considered as "engaged" since the completion of their tasks requires one of those two activities. Note that "wandering" means that the student is not engaged (it does not mean "wandering-around"). The balanced dataset is a smaller 1973 frame version that removes similar samples for each of the three classes, resulting in a more balanced number of frames across the classes.

DAiSEE dataset is a large labeled student engagement level dataset that is collected by a web camera during the period of a student watching educational and recreational videos. The dataset contains 9068 video snippets collected from 32 female and 80 male subjects aged 18 to 30. The dataset is labeled with four different student engagement levels: Very Low, Low, High, and Very High. In addition to these engagement labels, we relabeled a subset of DAiSEE with SED labels for out-of-distribution evaluation with the model trained using SED dataset. The goal is to evaluate whether the model can apply the knowledge learned from the student engagement dataset to an out-of-distribution dataset under the same premise that both the DAiSEE and SED datasets have students in front of a web camera. With this premise, we annotated the DAiSEE dataset according to the evaluation framework of SED with labels 'looking at their paper', 'looking at their screen', and 'wandering'.

We have also identified 85 samples (hard samples) from SED that were selected based on misidentification by MiniGPT-4, which we call the hard SED dataset. These hard samples tend to contain images of students that face one direction whilst their gazes face one another, or students' face not being contained

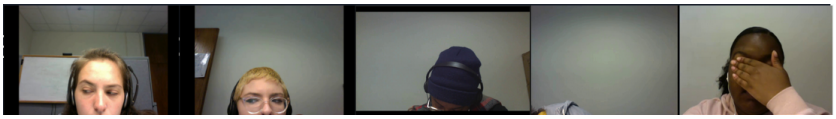

Fig. 2: Example hard samples we picked that tend to contain images of students face not being contained with in the image

with in the image. Examples of these hard Samples are shown in Figure 2. We further evaluated GPT-4V (OpenAI (2023)) as opposed to our finetuning methodology on this handpicked dataset.

EngageNet is a large-scale, multimodal dataset designed for user engagement prediction in real-world, in-the-wild settings. It comprises over 11.3K ten-second video clips (approximately 31 hours of data) from 127 participants recorded under diverse illumination conditions, and it captures both behavioral cues—such as facial expressions, head pose, and eye gaze—and cognitive responses from interactive questionnaires. Annotated into four engagement levels (Not Engaged, Barely Engaged, Engaged, Highly Engaged) using expert labels and self-reports, EngageNet provides a rich resource for developing and benchmarking deep learning models aimed at enhancing user experience in education, human-computer interaction, and related domains.

3.2 Data Preprocessing and Training

We used 80% of the data samples from the balanced SED dataset for finetuning and 20% of the samples for our evaluation with balanced data. We also evaluated our model with the raw SED dataset. We excluded the training balanced sample from the raw SED dataset and used the rest of the samples for evaluation. For both training and testing data samples, we combined the labels for the various categories and created an annotation JSON file that is a collection of data pairs where each image ID is associated with a reference sentence based on the category of that image accordingly. The detail of the reference sentence is discussed in section 2.3. We also compared the results of our finetuning prompts with the original prompts (Orig-P) used by MiniGPT-4[2]. Additionally, the evaluation results, using the original MiniGPT-4 checkpoint without finetuning (Orig-M) on the SED dataset, are shown as the baseline of the MiniGPT-4 model.

For the out-of-distribution evaluation on the DAiSEE dataset, we manually labeled 1046 frames with the three SED categories, assigned image IDs for each frame, and constructed an annotation JSON file. The annotation file for this dataset was constructed in a similar fashion to SED.

The model setup with reference and policy components is memory intensive. It was run on a system with two Nvidia RTX A6000 GPUs. Training ran with a global batch size of four. For the finetuning, we use AdamW optimizer. The learning rate is controlled with a cosine learning rate scheduler. The initial learning rate is $3e^{-5}$, minimum learning rate is $1e-5$ and warmup learning rate is $1e^{-6}$. The warmup steps is set to 200. The β used in DPO loss calculation is set to 0.1. The training time per epoch is about 25 minutes.

	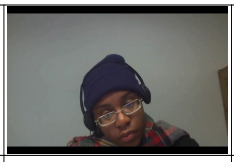	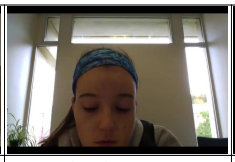	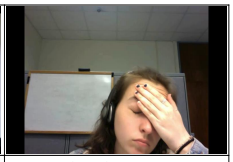
Reference sentence	The person is looking straight at the screen	The person is looking down at the paper	The person is looking away
m-GPT4 & DPO	The person is looking straight at the screen	The person is looking down at the paper	The person is looking away
m-GPT4 & DPO (JSON prompts)	{'label': 'looking at the screen'}	{'label': 'looking at the paper'}	{'label': 'wandering'}
m-GPT4 (JSON prompts)	{'label': 'looking at the screen'}	{'label': 'looking at the paper'}	{'label': 'looking at the screen'}
m-GPT4 (SED specific prompts)	The person is looking straight at the screen	The person is looking down at the paper	The person is looking straight at the screen
m-GPT4 (Original prompt)	The person is looking straight at the screen	The person is looking straight at the screen	The person is looking away
m-GPT4 (Original model)	The person is wearing a blue beanie and a red scarf and has a pair of headphones on. The person is looking down at the paper on the desk in front of them. The image is a close up of the person's face, with the headphones on their ears and the paper in front of them.	The person is looking straight at the screen. They are not looking down at the paper or away from the screen.	1. The person is looking straight at the screen. 2. The person is looking down at the paper. 3. The person is looking away from the camera.

Fig. 3: Comparison of the generated sentences using four different MiniGPT-4 based finetuning models for the three student engagement behavior markers. The image samples are selected from the SED dataset.

3.3 Assessing Correctness of Generated Answers

We use the output of the model to categorize the student engagements into one of the engagement labels. To evaluate MiniGPT4's output from the still-images, we use three different methods, keyword evaluation, sentence similarity (SS) [11] evaluation, and the Video-ChatGPT [12] evaluation benchmark of Correctness and Consistency.

For the keyword evaluation, we consider the output as the correct answer if the generated sentence contains the desired keywords (i.e. paper, screen, away) representing the reference sentence. If no keywords match, the response is considered wrong. Our model, finetuned with DPO, did not return any results with multiple keywords matching. Though the other models we evaluated did. Therefore, we need a way to evaluate the meaning of the sentences. For example, an output generated by MiniGPT-4 (Orig-M) in Figure 3 says, "The person is looking straight at the screen. They are not looking down at the paper or away from the screen." Based on its meaning, it should be counted as "The person is looking straight at the screen" instead of one of the other categories. A keyword approach would incorrectly categorize the response. To address this issue, we opted for two other methods that have a deeper understanding of sentences.

For the sentence similarity (SS) evaluation, we use a pretrained sentence transformer, BGE-M3 [11], to determine whether the generated sentence conveys the same meaning as the reference sentences. We do this by comparing the embedding of the generated sentences and the candidate reference sentences representing the ground truth classes. If the reference sentence has the highest similarity score among all possible reference sentences, it is marked as correct,

otherwise, it is marked as incorrect. We use cosine similarity to capture the semantic similarity of sentence embeddings generated by BGE-M3 [11].

3.4 Results

First, we evaluate the performance of our model using both the accuracy and F1 score obtained by keyword and SS evaluation. We evaluate the performance of the finetuned m-GPT4 & DPO models (MiniGPT-4 model finetuned using DPO) against m-GPT4 models which finetuned differently depending on the dataset.

For SED dataset, We evaluate the performance of the finetuned m-GPT4 & DPO models with both the natural language prompts and JSP. These are compared against m-GPT4 models, which includes m-GPT4 model finetuned using the engagement specific prompt (EnP) we discussed in the previous section, m-GPT4 finetuned using the original prompts (Orig-P) of the MiniGPT-4 paper [2], and the original un-finetuned m-GPT4 model (Orig-M). We also compare our results against the deep learning vision model results. We finetuned MobileNet and Xception, both pretrained on ImageNet (Pre-IN), on the balanced SED dataset and obtained similar accuracy results on the balanced SED dataset to those presented in [7].

Method	Acc	F1	SS Acc	SS F1
m-GPT4 & DPO	96.7	96.7	96.7	96.7
m-GPT4 (EnP)	95.2	95.2	95.2	95.2
m-GPT4 (JSP/DPO)	96.0	95.9	×	×
m-GPT4 (JSP)	95.2	95.2	×	×
m-GPT4 (Orig-P)	87.6	87.4	87.6	87.4
m-GPT4 (Orig-M)	58.6	58.9	40.2	39.7
MobileNet (Pre-IN)	94	-	×	×
Xception (Pre-IN)	88	-	×	×
VGG16 (Pre-IN)	85	-	×	×

Table 1: Results on balanced SED.

Method	Acc	F1	SS Acc	SS F1
m-GPT4 & DPO	84.7	84.2	84.7	84.2
m-GPT4 (EnP)	81.2	80.8	81.2	80.8
m-GPT4 (JSP/DPO	85.9	85.6	×	×
m-GPT4 (JSP)	82.4	82.8	×	×
m-GPT4 (Orig-P)	70.6	72.1	70.6	72.1
m-GPT4 (Orig-M)	56.5	61.3	31.8	34.7
GPT-4V	74.2	72.6	×	×
MobileNet (Pre-IN)	82.3	83.5	×	×
Xception (Pre-IN)	84.7	85	×	×

Table 2: Results on hard SED samples

Method	Acc	F1	SS Acc	SS F1
m-GPT4 & DPO	94.6	95.8	94.6	95.8
m-GPT4 (EnP)	90.8	92.8	90.8	92.8
m-GPT4 (JSP/DPO)	94.3	95.6	×	×
m-GPT4 (JSP)	94.1	95.6	×	×
m-GPT4 (Orig-P)	89.5	90.0	89.2	89.8
m-GPT4 (Orig-M)	50.5	61.2	48.6	57.8
MobileNet (Pre-IN)	89.9	91.3	×	×
Xception (Pre-IN)	87	87.9	×	×

Table 3: Results on raw SED.

Method	Acc	F1	SS Acc	SS F1
m-GPT4 & DPO	88.4	87.9	88.5	88.0
m-GPT4 (EnP)	87.1	86.9	87.1	86.9
m-GPT4 (JSP/DPO	87.2	87.1	×	×
m-GPT4 (JSP)	86.8	87.0	×	×
m-GPT4 (Orig-P)	88.0	87.7	87.2	87.2
m-GPT4 (Orig-M)	62.2	70.7	56.7	67.3
MobileNet (Pre-IN)	26.7	33.2	×	×
Xception (Pre-IN)	55.9	65.5	×	×

Table 4: Results on relabeled DAiSEE samples.

Method	Acc	F1	SS Acc	SS F1
m-GPT4 & DPO	77.1	75.7	71.8	73.0
m-GPT4 (Inst tune)	55.5	40.3	52.5	46.3
ViT Facial Exprn. recog.	54.5	49.3	×	×
ViT	53.8	47.9	×	×
EmotionNet	51.1	×	×	×
DAiSEE	57.9	×	×	×

Table 5: DAiSEE engagement levels results

Method	Acc	F1	SS Acc	SS F1
m-GPT4 & DPO	65.1	64.2	62.6	64.4
m-GPT4 (Inst tune)	53.4	48.2	46.3	50.8
ViT Facial Exprn. Recog.	41.1	43.5	×	×
ViT	45.8	45.5	×	×
EngageNet	67.6	×	×	×

Table 6: EngageNet engagement levels.

SED Balanced	Correctness ↑	Consistency ↑
m-GPT4 & DPO	**4.84**	**3.67**
m-GPT4 (EnP)	4.77	3.36
m-GPT4 (Orig-P)	4.42	3.58
SED Raw		
m-GPT4 & DPO	**4.77**	**3.70**
m-GPT4 (EnP)	4.69	3.27
m-GPT4 (Orig-P)	4.54	3.48
DAiSEE		
m-GPT4 & DPO	**4.47**	3.64
m-GPT4 (EnP)	4.41	3.09
m-GPT4 (Orig-P)	4.43	**3.68**
SED Hardsamples		
m-GPT4 & DPO	**4.24**	**3.54**
m-GPT4 (EnP)	4.16	3.45
m-GPT4 (Orig-P)	4.2	3.46

Table 7: Evaluation with GPT-3.5 turbo

We show the evaluation results on the balanced SED dataset in Table 1. As we can see from this table, our method achieved 96.7% accuracy and F1 score, substantially better than the ones by the original deep learning vision models. Interestingly, the JSON prompts (JSP) achieved almost the same result, though

did perform a bit worse. The results of VGG16 are taken from [7]. Since they only report the accuracy, the other results for these methods are left blank. In all the tables, an ✕ denotes the sentence similarity is not relevant to the model.

On the evaluation of hard samples of SED (see Table 2), m-GPT4 & DPO model with JSON output performed the best, and this time it is better than m-GPT4 & DPO model with natural language output. Both m-GPT4 & DPO models significantly outperformed GPT-4V results. We also maintain superior performance compared to other m-GPT4 based models. At the same time, we find it interesting that the hard samples cause all the m-GPT4 models to perform worse. This performance hit does not manifest in the purely vision models.

For the out-of-distribution evaluation on raw SED dataset, our model outperformed the other models across all four measures (see Table 3). On the unbalanced, raw dataset, our model performs a bit worse compared to itself on the balanced dataset. Interestingly, the results of m-GPT4 (Orig-P) outperform itself on the unbalanced raw dataset compared to the balanced dataset. We suspect that even though m-GPT4 (Orig-P) is trained on the balanced dataset, it is skewing toward the predominant class found in the raw unbalanced dataset.

Table 4 shows the out-of-distribution evaluation results on the relabeled DAiSEE dataset. Again both versions of m-GPT4 & DPO exhibit the best performance across all evaluation measures, and JSON prompt performs slightly worse than the engagement specific prompt. This shows the generalization ability of the proposed approach since DAiSEE is an out-of-distribution dataset. In contrast, the results of MobileNet and Xception drop significantly, which clearly shows that they are not able to generalize well to out-of-distribution samples.

Tables 5 and 6 show our results on DAiSEE and EngageNet datasets using their original labels. Again, m-GPT4 & DPO model exhibits the best performance. Interestingly, it outperforms EmotionNet, which aims to recognize emotions in video data, even though we used only 10 frames from each video. m-GPT4 & DPO model does a bit worse than EngageNet. We believe that is because EngageNet was specifically designed to capture Eye Gaze, Head Pose, and Facial Action Units, which results in the better performance when determining the four levels of engagement, but that may not translate.

We also used GPT-3.5 turbo to evaluate the correctness of the natural language answers following the protocol described in [12]. The evaluation with GPT-3.5 turbo (see Table 7) outputs scores range from 0 to 5 for *Correctness* and *Consistency*, signifies the level of alignment between the model output and the ground truth. We find that these evaluation results are consistent with the keyword and Sentence Similarity evaluation results for both the SED and DAiSEE datasets. The proposed method (m-GPT4 & DPO) consistently scores higher across almost all evaluated datasets and performance measures.

4 Conclusions and Future Work

In this paper, we focused on the task of accurate recognition of engagement relevant visual behavior markers using VLMs. We exploited the direct preference

optimization (DPO) approach and proposed a modification to its finetuning that uses the model's responses to strengthen its performance. Unlike other preference alignment models, the proposed DPO finetuning generates preference data pairs using the wrong answers generated by the policy model during finetuning. This approach, which are finetuned to the student engagement domain, can leverage pretrained VLMs. This makes the proposed approach easily extensible for recognizing a large variety of visual markers relevant to engagement. We showed that our model's performance is superior to both pure vision models and other finetuning methods. We also demonstrated the generalizability to out-of-distribution samples, which is important for real-life applications.

References

1. Lei, H., Cui, Y. & Zhou, W. Relationships between student engagement and academic achievement: A meta-analysis. *Social Behavior and Personality: an international journal* **46,** 517–528 (Mar. 2018).
2. Zhu, D. *et al.* MiniGPT-4: Enhancing Vision-Language Understanding with Advanced Large Language Models. *arXiv:2304.10592* (2023).
3. Chiang, W.-L. *et al. Vicuna: An Open-Source Chatbot Impressing GPT-4 with 90%* ChatGPT Quality* Mar. 2023. https://lmsys.org/blog/2023-03-30-vicuna/.
4. Touvron, H. *et al.* Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971* (2023).
5. Rafailov, R. *et al. Direct Preference Optimization: Your Language Model is Secretly a Reward Model* 2023. arXiv: 2305.18290 [cs.LG].
6. Schulman, J. *et al.* Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347* (2017).
7. Delgado, K. *et al. Student Engagement Dataset* in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops* (Oct. 2021), 3628–3636.
8. Gupta, A. *et al. DAiSEE: Towards User Engagement Recognition in the Wild* 2022. arXiv: 1609.01885 [cs.CV].
9. Singh, M. *et al. Do i have your attention: A large scale engagement prediction dataset and baselines* in *Proceedings of the 25th International Conference on Multimodal Interaction* (2023), 174–182.
10. Schulhoff, S. *et al.* The Prompt Report: A Systematic Survey of Prompting Techniques. *arXiv:2406.06608* (2024).
11. Chen, J. *et al. BGE M3-Embedding: Multi-Lingual, Multi-Functionality, Multi-Granularity Text Embeddings Through Self-Knowledge Distillation* 2024. arXiv: 2402.03216 [cs.CL].
12. Maaz, M. *et al. Video-ChatGPT: Towards Detailed Video Understanding via Large Vision and Language Models* 2024. arXiv: 2306.05424 [cs.CV]. https://arxiv.org/abs/2306.05424.