Este documento presenta algunos de los resultados obtenidos al momento de probar modelos de LLM existentes que se ofrecen como servicio para satisfacer la necesidad del proyecto, de estos servicios **GPT** y **LLava** son candidatos a los cuales sometimos a pruebas físicas.

Para realizar las pruebas se utilizaron imágenes del dataset Daisee, dataset sobre el cual otros artículos y papers de los se encuentran en las referencias utilizan.[ “**EmoAI Smart Classroom: The Development of a Student Emotional and Behavioral Engagement Recognition System**”]

**GPT**: GPT (Generative Pre-trained Transformer) es un modelo de lenguaje grande diseñado para entender y generar texto similar al humano. Aunque no es inherentemente visual, GPT se puede ajustar o usar con módulos adicionales para procesar descripciones y tareas relacionadas con imágenes.

**LLava** (Large Language Model Visual Assistant) es un modelo de inteligencia artificial multimodal diseñado para procesar y entender información visual y textual simultáneamente. Se utiliza comúnmente para tareas que requieren contexto visual, como la descripción de imágenes o la respuesta a preguntas visuales.

**Conjunto de Datos Daisee**: Daisee (Dataset for Affective States in E-Environments) contiene datos de video que capturan diferentes niveles de engagement (aburrido, feliz, sorprendido, etc.) en una variedad de escenarios. Se utiliza comúnmente para entrenar modelos que reconocen emociones humanas y niveles de engagement.

* Utilizar modelos pre entrenados como **LLava** y **GPT** nos permite evaluar rápidamente su efectividad en nuestro dominio sin la necesidad de un entrenamiento extenso desde cero. Estos modelos ya están optimizados para tareas específicas y pueden adaptarse con un esfuerzo mínimo.
* Esta es la alternativa a desarrollar y desplegar nuestra propia solución a través de medios convencionales que ofrecen más flexibilidad pero pueden tener un rendimiento menor junto con un esfuerzo considerablemente mayor.
* Al probar estos modelos, podemos establecer una referencia de rendimiento para entender los niveles de engagement, lo que ayuda a definir un estándar contra el cual nuestros modelos personalizados (como lo podría ser utilizar **CNN** y **KNN**) pueden compararse.

Gpt ofrece varias formas de probar sus servicios, por medio de la API y PromptEngineering se es posible rescatar con ayuda de String parsing la respuesta correcta del modelo a la pregunta.

| GPT |
| --- |
|  |
| El modelo en sí es capaz de clasificar correctamente la imagen, no obstante requiere de su considerable PromptEngineering para lograr el resultado esperado. Un punto débil de este modelo es el costo de GPT4.0, el análisis de imágenes estáticas tiene un costo considerable en el consumo de tokens de la plataforma y en caso de aplicarlo al caso de uso del proyecto ese sería aún mayor pues estaríamos haciendo video streaming, existen algunas maneras en teoría de compensar esto mediante ciertas técnicas pero son cosas que todavía no se han podido experimentar porque los recursos son muy costoso. |

| LLava |
| --- |
|  |
| El modelo es de uso OpenSource y Selfhost no obstante a pesar de no ser un servicio enorme como GPT este es capaz de reconocer patrones y clasificar correctamente en varios de los casos. Algunos puntos débiles de este modelo son que al ser self-host se requiere de un potente hardware físico por lo tanto al momento del testeo solo una persona cumple con los requerimientos físicos para implementar esto por lo tanto es bastante probable que esto presente una dificultad a la hora de implementarlo al desarrollo. |