MindSparksAI

Background:

In traditional classroom settings, teachers often face challenges in keeping track of each student's engagement and attentiveness. Despite their best efforts, it is difficult to accurately gauge when students are losing interest or becoming distracted, which can negatively impact their learning outcomes. Additionally, teachers lack real-time insights and actionable feedback to adapt their teaching methods dynamically to improve student engagement.

Problem:

The lack of real-time monitoring tools in classrooms prevents teachers from effectively identifying and addressing student disengagement. This gap results in:

1. **Inability to Track Individual Engagement**: Teachers struggle to monitor each student's level of engagement and attentiveness during lessons.
2. **Delayed Interventions**: Without real-time alerts, teachers cannot promptly address issues such as students falling asleep, yawning, or becoming distracted.
3. **Lack of Data-Driven Insights**: Teachers do not have access to detailed reports that highlight engagement trends and provide personalized suggestions for improvement.
4. **Challenges in Curriculum Adaptation**: Teachers need actionable insights to adjust their teaching strategies based on student engagement levels to foster a more interactive and effective learning environment.

Solution:

To address these challenges, we propose the development of a comprehensive AI-driven student engagement monitoring system. The system leverages machine learning models for emotion detection and pose detection to provide real-time insights into student attentiveness. Key features of the solution include:

1. **Real-Time Engagement Monitoring**: Using computer vision techniques, the system detects and analyzes students' emotions and poses to determine their level of attentiveness.
2. **Immediate Alerts**: The system provides real-time alerts to the teacher if a student shows signs of disengagement, such as falling asleep or yawning.

3. **Visual Dashboards**: A dashboard displays a list of students, allowing teachers to select an individual student and view their engagement levels and behavioral patterns over time.
4. **Comprehensive Reports**: At the end of each session, the system generates detailed reports using a Large Language Model (LLM) such as LLaMA 3.1. These reports include insights on each student's engagement, highlight trends, and offer personalized suggestions for improvement.
5. **Teaching Suggestions**: The system provides actionable suggestions to teachers on how to improve their teaching strategies based on the collected data.

Objective:

The primary objective of this project is to create a robust AI-driven platform that enhances classroom engagement by providing teachers with real-time monitoring, immediate alerts, and data-driven insights. By utilizing advanced machine learning techniques and integrating with visualization tools like Prometheus and Grafana, the system aims to foster a more interactive and effective learning environment.

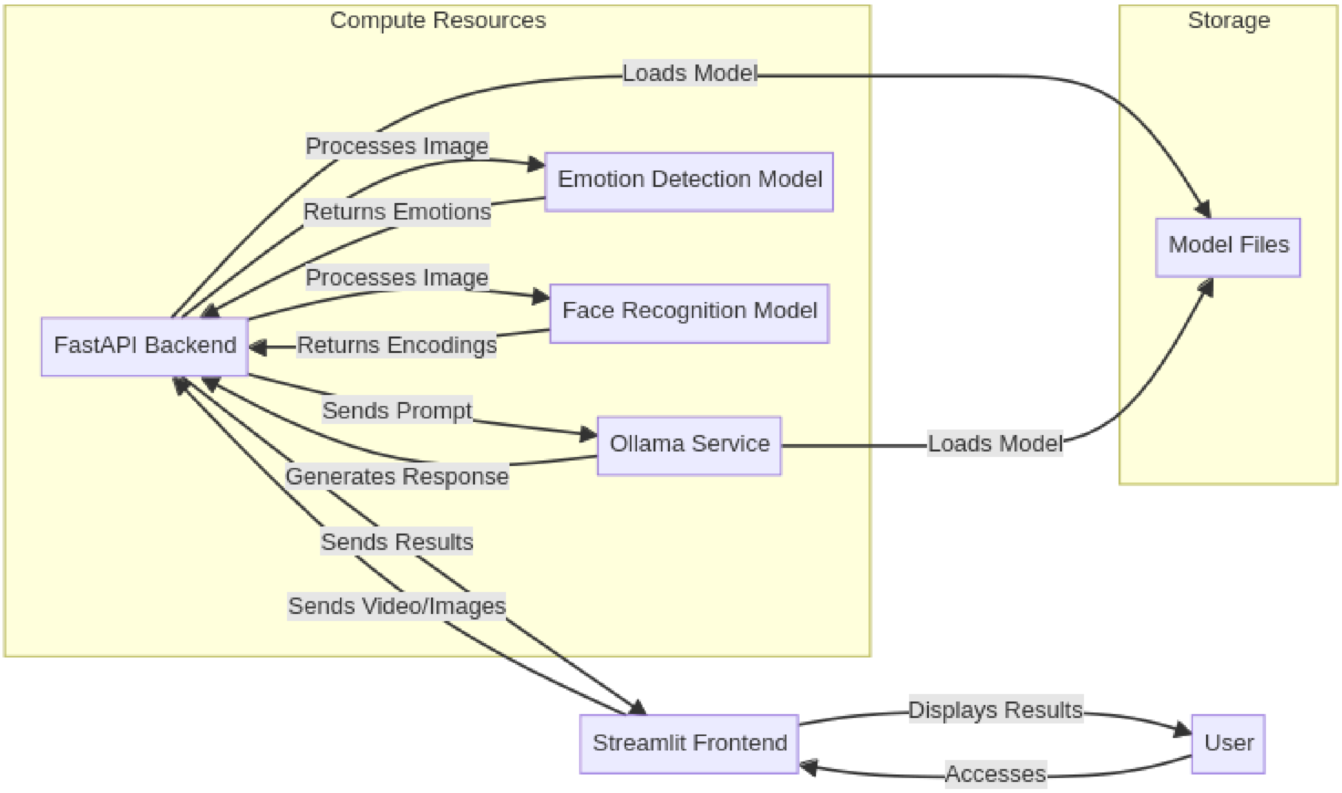
Expected Outcomes:

- Improved student engagement and attentiveness in classrooms.
- Empowered teachers with real-time data and actionable insights to adapt their teaching methods.
- Enhanced learning outcomes through timely interventions and personalized feedback.
- Comprehensive engagement reports that help in identifying trends and making informed decisions.

Technologies and Tools:

- **Machine Learning Models**: For emotion and pose detection.
- **Large Language Model (LLM)**: For generating personalized reports and teaching suggestions.
- **FastAPI**: Backend framework for serving the ML models and handling API requests.
- **Streamlit**: UI for visualizing the results and recommendation
- **Docker**: For containerizing the application components.
- **Docker Compose**: For orchestrating multi-container Docker applications.

Architecture



Getting Started

1. Download the codebase.
2. Change directory into MindsparksAI:

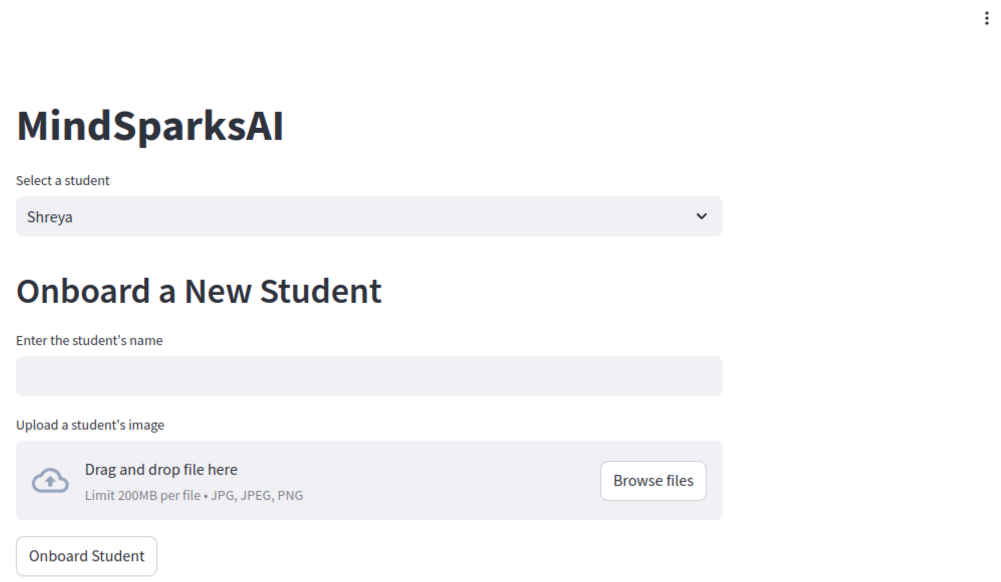
```
cd MindsparksAI
```

3. Run the docker command:

```
docker compose up --build
```

4. On your browser of choice, go to `localhost:8501`.

5. The following Streamlit interface will be displayed.



MindSparksAI

Select a student: Shreya

Onboard a New Student

Enter the student's name

Upload a student's image

6. Enter name of student, upload their image, and click on onboard student. Their face encodings will be saved.
7. On the dropdown menu, select the student's name. Click on Run, and the camera will start streaming and render interactive graphs in real-time.
8. After the class ends, uncheck the `Run` option.
9. The video stream will close and you can hit the button `Generate Recommendations`.
10. The following recommendation gets printed out:
    Recommendations: Based on the provided data, I've analyzed the student's engagement levels across different time points and identified patterns, trends, and insights that can inform recommendations for both the student and the teacher.

    Student Recommendations:

    1. **Self-reflection:** The student should take some time to reflect on their engagement patterns during lessons. They can ask themselves questions like:
        - What triggers my attention or lack thereof?
        - Are there specific topics or activities that I find more interesting than others?
        - Am I comfortable asking questions or participating in discussions? If not, why?
    2. **Identify interests:** The student should explore their interests and hobbies outside of the classroom. They can:
        - Join a club or organization related to their favorite subject
        - Participate in online forums or communities centered around their interests
        - Share their passions with friends and classmates
    3. **Improve note-taking skills:** The student should work on improving their note-taking techniques to stay engaged during lessons:
        - Experiment with different methods (e.g., Cornell Notes, Mind Maps)

   - Organize notes by topic or concept
   - Summarize key points in their own words
4. **Seek help when needed:** The student should not hesitate to ask for help if they're struggling or need clarification:
   - Reach out to teachers during office hours or after class
   - Form study groups with classmates who can provide support and guidance
   - Utilize online resources (e.g., Khan Academy, YouTube tutorials)
5. **Develop problem-solving skills:** The student should focus on developing their critical thinking and problem-solving abilities:
   - Practice solving problems or puzzles during free time
   - Engage in activities that require creative solutions (e.g., writing, art)
   - Participate in debates or discussions to improve argumentation skills
6. **Stay organized:** The student should maintain a clean and organized workspace to minimize distractions:
   - Prioritize tasks and set realistic goals
   - Use calendars, planners, or apps to stay on track
   - Establish routines for regular breaks and self-care

(Note: This is just a sample recommendation. Based on the student's emotion, yours will be different.)

Limitations of the Prototype

- Currently, it detects emotions of only one student at a time.

Challenges Faced

- Collection of database for the labels `bored`, `attentive`, and `neutral` was a huge task. Pictures were scraped from stock images repository.
- Tested and tried Hugging Face's repository of LLaMa3.1, which wasted a lot of time because of the low computational resources.
- Mapping student recognition with their emotions took some time to accomplish.

Future Plans

- Scale this project and introduce this to classrooms.

- Enhance the UI using React.js or Vue.js.
- Optimize the ML models so that they are able to capture every student's emotions in a classroom and by the end of the session, generate personalized recommendations by querying each student ID.
- Gamify the app and provide credit points to students.
- Introduce this solution in other domains as well.