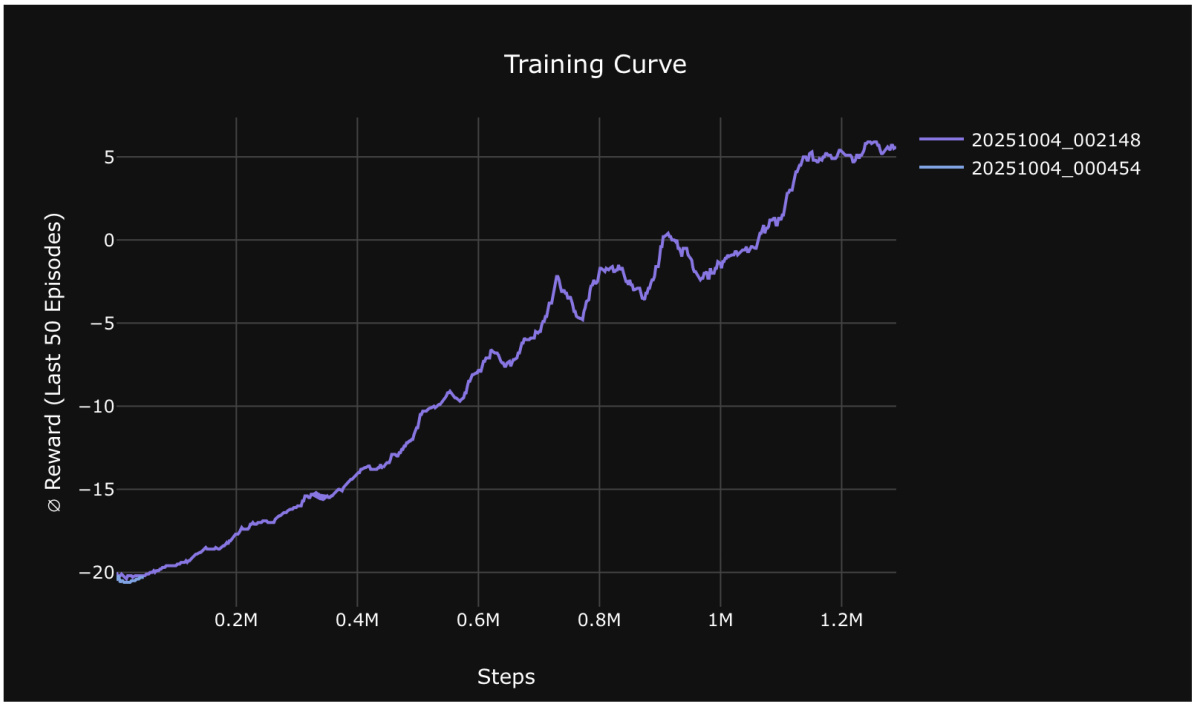
Ping Pong Dashboard

A real-time dashboard to visualize the training process for a Ping Pong agent, built with **FastAPI** (backend) and **React** (frontend).



Run	Model	Best Reward	Last ⌀ Reward	Time (HH:MM)	Plot
20251004_002148	PPO	9	5.6	14:57	☑
20251004_000454	-	-21	-20.3	00:16	☑

Hide Previous Runs (1)

Fig. 1: Ping Pong RL Training Dashboard – Real-time Performance Monitoring.

Project Structure

```
pong-dashboard/
├── backend/
│   ├── main.py                # FastAPI backend
│   ├── requirements.txt       # FastAPI dependencies
│   └── .env                   # Env variables
├── frontend/
│   ├── public/
│   │   ├── index.html         # React root HTML file
│   │   ├── models/            # 3D models
```

```
│   │   └── images/             # Logos, icons, etc.
│   ├── src/
│   │   ├── App.js              # React main component
│   │   ├── App.css             # Styles
│   │   ├── index.js            # React entry point
│   │   ├── index.css           # Global styles
│   │   ├── reportWebVitals.js  # CRA performance metrics
│   │   └── TableTennisScene.js # 3D Table Tennis component
│   ├── package.json            # React dependencies and scripts
│   └── package-lock.json
└── README.md
```

Features

- **Live training chart**: Plot average rewards of the last 50 episodes over training steps.
- **KPI cards**: Display best reward, last average reward, and elapsed time.
- **Run summary table**: Shows previous runs, best reward, last average reward, elapsed time, and model type.
- **3D Table Tennis model**: Interactive view of a ping pong table using React Three Fiber
- **Rules panel**: Briefly explains Olympic table tennis rules alongside the model.

Backend Setup

1. Create a virtual environment:

```
uv venv --python 3.12
source .venv/bin/activate   # Mac/Linux
.venv\Scripts\activate      # Windows
```

2. Install dependencies:

```
uv pip install -r requirements.txt
```

3. Run the FastAPI server:

```
uvicorn main:app --reload --host 0.0.0.0 --port 8000
```

The backend exposes:

- http://localhost:8000/results — latest training run data
- http://localhost:8000/runs — summary of all runs

Frontend Setup

Navigate to the frontend directory:

```
cd frontend
```

2. Install dependencies:

```
npm install
```

3. Start the development server:

```
npm start
```

4. Open http://localhost:3000 to view the dashboard with the training metrics (Fig. 1).

5. For convenience, a summary of the official Olympic Table Tennis Rules can be found beneath the dashboard (Fig. 2).

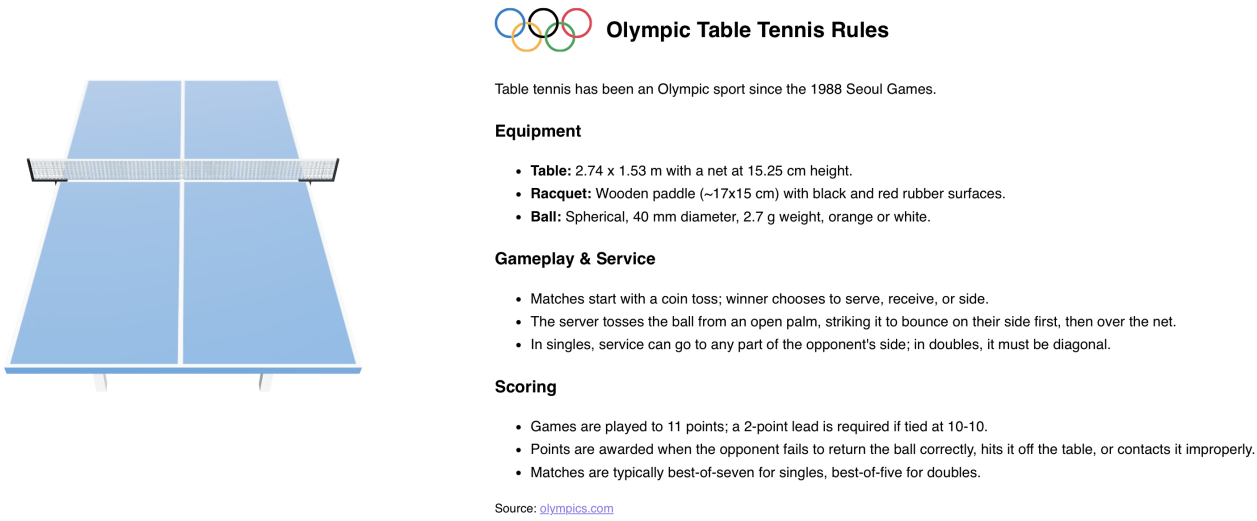


Fig. 2: Table Tennis 3D Model and Olympics Rules.

License

MIT

Credit

The 3D model of the Table Tennis titled "Low Poly Table Tennis" is the work by Xorshift licensed under CC-BY-4.0