LLM for JOKE

Introduction and Purpose

- ➡ Welcome to our AI joke generation project.
- ➡ We used an LLM to create short and funny stories.
- ➡ Each step of the project will be explained today.

Model

- Used Hugging Face Transformers for model architecture.
- PyTorch was the base framework for training.
- Safetensors used for secure and fast model saving.
- Training started on Colab, continued locally.

Model

➡ Training started on Colab, completed locally.

➡ Training progress was monitored and evaluated.

➡ Output: short and humorous Turkish jokes.

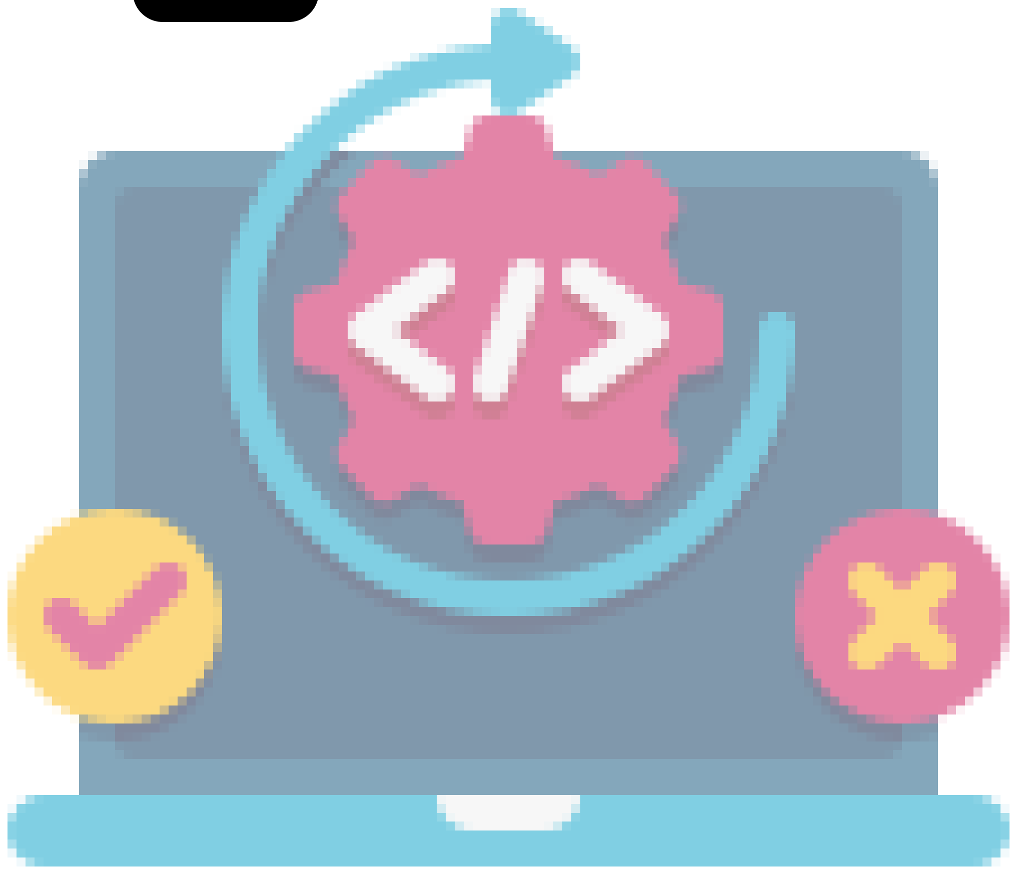
➡ Model quality can be improved with more tuning.

Frontend

✅ Built with Next.js and styled using Tailwind CSS.

✅ Used Shadcn/ui for clean and reusable components.

✅ Simple and responsive layout with one-click interaction



Backend

1️⃣ Developed with Python Flask for simple and fast API communication.

2️⃣ Single endpoint “/joke” handles joke generation requests.

3️⃣ Clean code structure for maintainability and future extensions.

Data and Dataset

➡ Jokes were collected in .txt format from open sources.

➡ Useless, incomplete, and repeated content was removed.

➡ Manual review ensured data consistency before training.

➡ Offensive or inappropriate language was carefully filtered out.

Summary & Conclusion

✅ We successfully built a joke-generating system using an LLM.

✅ The system can create jokes automatically with one click.

✅ Each part of the project was developed with simple and focused tools.

✅ The project shows that AI can also support creativity and humor.

🚀 **“This project proves that even AI can learn the art of telling a joke.”**

Thank you all for listening...

Emir ADIGÜZEL
210706011

Yasin Eren ŞAHİN
20070607

Harun Yahya ÜNAL
210706015

Fadıl Ahmet TÜFEKÇİ
210706031

Yağmur TANK
210706010

Melek Sude GÜNEN
220706052