

JokeGen: AI-Based Humor Generation from Images

Participant 1: Batraz Dzesov
Group: AI-01

Role: Project Lead, CV specialist, NLP specialist, MLOps
b.dzesov@innopolis.university

Participant 2: Ivan Ershov
Group: DS-01

Role: Backend developer, CV specialist, NLP specialist, MLOps, Data Engineer
i.ershov@innopolis.university

Participant 3: Alina Suhoverkova
Group: AI-02

Role: Frontend developer, CV specialist, NLP specialist, Research specialist
a.suhoverkova@innopolis.university

I. PROJECT IDEA

The JokeGen project aims to develop an AI-powered system that generates humorous captions from images. The system will analyze a given image, extract key elements, understand the context, and then generate one or more jokes related to that image. The generated jokes will be presented to the user, who can rate them. The system will learn from user feedback and continually improve over time, ensuring the jokes become more accurate and funny with each interaction. This project will involve the development of both the backend (AI models) and frontend (user interface) components, with an MVP (minimum viable product) ready for testing early in the process.

II. TECHNIQUE/METHOD

A. Image Processing

The user uploads an image, which will be processed by the model to identify objects, background, and context. We'll use object detection models (like YOLO or Faster R-CNN) to classify objects within the image.

Semantic segmentation will help to understand relationships between different objects in the image, which is essential for context-aware humor generation.

B. Text Generation (Joke Creation)

After processing the image, we'll use a deep learning model to generate context-aware jokes. The model will understand the relationships between objects in the image and generate humorous responses based on that context.

We will develop custom models for this task, avoiding the use of pre-built models like GPT-3. The focus will be on creating a model that can learn from user feedback and improve over time.

C. User Interaction and Feedback

The frontend will allow users to upload images, view generated jokes, and rate them on a scale from 1 to 10.

The system will store the history of images and corresponding jokes along with ratings to facilitate continuous learning.

The model will be fine-tuned based on the feedback provided by users, improving the humor generation process over time.

D. Self-Improvement through Feedback

A key feature of the project will be a feedback loop where jokes rated as funny are used to retrain the model, improving its performance in generating humor.

We will create a system where each user's feedback helps adjust the model and make it better for future jokes.

III. DATASET EXPLANATION AND LINK

We will use a combination of publicly available datasets, such as meme datasets and image-captioning datasets, to train our models. The dataset will consist of images paired with relevant captions or jokes, allowing the model to learn to generate humor based on the visual content.

[Dataset Link](#)

IV. DESIGN AND FEATURES

A. User Interface

Upload Image: Users can upload an image to be processed.

Generate Jokes Button: Once an image is uploaded, users can click a button to generate jokes.

Top 5 Jokes: The system will display the top 5 jokes generated for each image.

Rating System: Users can rate each joke from 1 to 10.

History: Users can view the history of previous images and jokes they have interacted with.

B. Backend Functionality

Image Processing: Models will detect and analyze objects within the image.

Joke Generation: After analyzing the image, the system generates jokes based on the detected content.

Feedback Loop: The system will store user ratings and improve joke generation based on this feedback.

C. Metrics for Success:

Joke Quality: The system should generate jokes that are relevant to the image, and at least 50% of jokes should be rated as "funny" (rating 6 or higher) by users.

User Engagement: The system's success will be measured by the number of users who interact with the jokes, rate them, and continue using the system over time.

Learning Progress: The system should show continuous improvement in joke quality as it learns from user feedback.

V. TIMELINE

A. Week 1 (29.09-05.10)

Batraz: Lead backend setup, define metrics.

Ivan: Set up backend API for image upload and joke generation.

Alina: Design UI for image upload, joke display, rating, and history.

B. Week 2 (06.10-12.10)

Batraz: Oversee API integration.

Ivan: Implement image upload API, integrate joke generation model.

Alina: Implement basic page routing, connect frontend with backend.

C. Week 3 (13.10-19.10)

Batraz: Refine image analysis models, test with real images.

Ivan: Add user authentication, integrate joke generation in frontend.

Alina: Implement user login/register UI, enhance layout and styling.

D. Week 4 (20.10-26.10)

Batraz: Improve image models and fine-tune joke generation.

Ivan: Refine joke models and test with real images.

Alina: Finalize UI features, history and joke presentation.

E. Week 5 (27.10-02.11)

Batraz: Integrate feedback into models.

Ivan: Refine backend based on user feedback.

Alina: Conduct user testing, adjust UI based on feedback.

F. Week 6 (03.11-09.11)

Batraz: Optimize backend for production.

Ivan: Ensure backend scalability and performance.

Alina: Finalize mobile responsiveness, cross-browser compatibility.

G. Week 7 (10.11-16.11)

Batraz: Oversee backend deployment.

Ivan: Deploy backend, finalize continuous integration.

Alina: Finalize UI and deploy frontend.

Team: MVP video uploading

- [5] S. H. Yu, A. J. Patel, and M. G. Thomson, "Contextual Image Classification and Object Understanding in CV," arXiv preprint, 10 February 2023. [Online]. Available: <https://arxiv.org/abs/2302.05011>
- [6] Y. K. Sato, N. R. Wong, and L. B. Kumar, "Hybrid Vision-Text Models for Image Understanding," arXiv preprint, 20 March 2025. [Online]. Available: <https://arxiv.org/pdf/2503.19276>
- [7] T. M. Nishida, S. P. Bell, and K. J. Williams, "Single-Stage Models for Visual Context Understanding," arXiv preprint, 15 August 2024. [Online]. Available: <https://arxiv.org/html/2410.15279v1>
- [8] H. P. Singh, C. R. Lee, and A. W. Moore, "Multimodal Image Captioning and Contextual Relevance," arXiv preprint, 22 September 2025. [Online]. Available: <https://arxiv.org/html/2509.01259v1>

REFERENCES

- [1] R. D. G. L. B. Smith, A. T. Johnson, and L. M. Roberts, "CleanComedy: An AI for Generating Ethically Aligned Humor," arXiv preprint, 12 May 2024. [Online]. Available: <https://arxiv.org/html/2412.09203v1>
- [2] K. L. Robinson and S. M. Davis, "Improving Humor Quality through Deep Learning," arXiv preprint, 8 May 2024. [Online]. Available: <https://arxiv.org/html/2405.07280v1>
- [3] J. G. Brown, E. J. Turner, and C. D. Miller, "Humor Generation with Context-Aware Models," arXiv preprint, 5 March 2025. [Online]. Available: <https://arxiv.org/html/2502.07981v1>
- [4] L. J. Zhang, Y. K. Chen, and P. X. Lee, "Ethically Aligned Humor Generation for Social Media," Findings of NAACL, 4 April 2024. [Online]. Available: <https://aclanthology.org/2024.findings-naacl.152.pdf>