Project Guidelines: Emoji Generation with VQ-VAE

Generative AI Course Project

Part A: Core (Required)

1. Dataset Preparation

- Collect a dataset of emojis (minimum ∼2500 images; e.g., Unicode emojis, custom smiley sets, or open datasets).
- Preprocess to uniform size (e.g., 32×32 or 64×64).
- https://huggingface.co/datasets/valhalla/emoji-dataset

2. Model Training

- Train a VQ-VAE for reconstruction.
- Report reconstruction quality (MSE, SSIM, FID).
- Visualize original vs. reconstructed emojis.

Part B: Generative Modeling (Required)

3. Latent Prior Modeling

- Train an autoregressive prior (PixelCNN, Transformer) or diffusion prior on discrete codes.
- Sample from the prior $\rightarrow$ decode $\rightarrow$ generate novel emojis.
- Compare generated samples with real dataset emojis.

Part C: Creative Extensions (Choose $\geq$1)

You are not limited to following, but some example extensions are:

1. Latent Interpolation – Morph between two emojis in latent space and visualize transition.
2. Conditional Generation – Condition prior on emotion labels (happy, sad, angry, etc.).
3. Style Transfer – Train on two emoji sets and perform cross-style emoji generation.
4. Super-Resolution – Train VQ-VAE to reconstruct high-res emojis from low-res inputs.
5. Inpainting – Mask part of an emoji and use latent codes to fill in missing regions.

Deliverables

1. **Code** – Well-structured Colab notebook (with training, generation, and visualizations).
2. **Report (4–6 pages, word document)**
   - Motivation & background.
   - Methodology (dataset, architecture, training setup).
   - Experiments (reconstruction, generation, creative extensions).
   - Results: visuals + metrics.
   - Discussion: strengths, limitations, insights.
3. **Presentation (8–10 minutes)** – Overview of approach, demo of generated emojis, key findings and creativity.

Evaluation Metrics

- **MSE (Mean Squared Error)** Measures pixel-wise error between reconstructed and original images. Lower = better.

$$\text{MSE} = \frac{1}{N}\sum_{i=1}^{N}(x_i - \hat{x}_i)^2$$

- **PSNR (Peak Signal-to-Noise Ratio)** Expresses reconstruction quality in dB scale. Higher = better.

$$\text{PSNR} = 10\log_{10}\left(\frac{MAX^2}{MSE}\right)$$

  where MAX is the maximum pixel value (e.g., 255).

- **SSIM (Structural Similarity Index)** Captures perceptual similarity (luminance, contrast, structure). Ranges $[0, 1]$, higher = better.

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)}$$

- **FID (Fréchet Inception Distance)** Compares feature distributions of generated vs. real images using Inception network embeddings. Lower = better.

$$FID = ||\mu_r - \mu_g||^2 + \text{Tr}\left(\Sigma_r + \Sigma_g - 2(\Sigma_r\Sigma_g)^{1/2}\right)$$

  where (μ_r, Σ_r) = mean & covariance of real embeddings, and (μ_g, Σ_g) = those of generated samples.

Marking Scheme (100 Marks)

1. **Implementation & Correctness (30 Marks)**
   - (10) Dataset preprocessing & training setup.
   - (10) Correct VQ-VAE implementation (encoder, decoder, codebook, loss).
   - (10) Stable training & demonstration of reconstructions.

2. **Generative Modeling (25 Marks)**
    - (10) Prior training (PixelCNN/Transformer/Diffusion).
    - (10) Novel emoji generation (visual quality + diversity).
    - (5) Quantitative evaluation (FID, reconstruction metrics).
3. **Creative Extensions (20 Marks)**
    - (10) Implementation of at least one extension.
    - (10) Quality of results + novelty.
4. **Analysis & Discussion (15 Marks)**
    - (5) Codebook usage analysis.
    - (5) Latent representation insights (visualization, clustering).
    - (5) Critical discussion of limitations & improvements.
5. **Report & Presentation (10 Marks)**
    - (5) Clarity, organization, and depth of written report.
    - (5) Presentation quality and effective communication of results.