Text-Conditioned Image Generation with Diffusion Models

Dhanya Srinivasan
Maheshwari M R

Project Guide: Dr. P. Mirunalini
Department of Computer Science and Engineering
SSN College of Engineering, Chennai

VIVA

May 2, 2024

Introduction

- Text-to-Image generation refers to the generation of visually realistic images, that match the context of the given text description.
- The integration of textual and visual information enhances various applications such as content creation and multimedia analysis.
- By combining textual descriptions with visual content, text-image understanding enables multimodal fusion, resulting in more comprehensive image generation.
- In this project, we intend to leverage the understanding of the relationship between image-text pairs in order to create a more effective image generative model.

Motivation

- Text-image understanding facilitates cross-modal understanding by integrating textual and visual information seamlessly. This enables multimodal fusion, where textual descriptions enrich visual content, and vice versa.
- Deep learning models use generative networks, are not able to correlate the produced images' semantics with the provided textual descriptions. Producing visually accurate visuals that are also semantically matched with the verbal descriptions is crucial.
- Most of the current systems often struggle with nuanced cross-modal understanding, leading to mismatches between textual descriptions and generated images.

Research Gaps Identified

Understanding the relationships between text-image pairs is crucial for generating images from text data, applicable in various fields. Some limitations of existing models include:

- Existing models like GAN and DALLE are exceptionally large. These not only take a large time to train but also require a very large dataset to be able to learn properly and generate a well defined image
- Generative models are often unable to generate images that are semantically aligned with the given captions. Hence, they often generate unintended images.
- Attention mechanisms are used to help the generator in focusing on different words, but this fails in producing global semantic consistency due to the difference in image and text modalities
- Visually realistic images are often not generated by existing T2I models

Problem Statement

To understand the relationship between text-image pairs through the utilization of an AI model, and use this understanding to generate images that are semantically aligned with the text description.

Dataset used

Training: MediaEval MUSTI Training Dataset

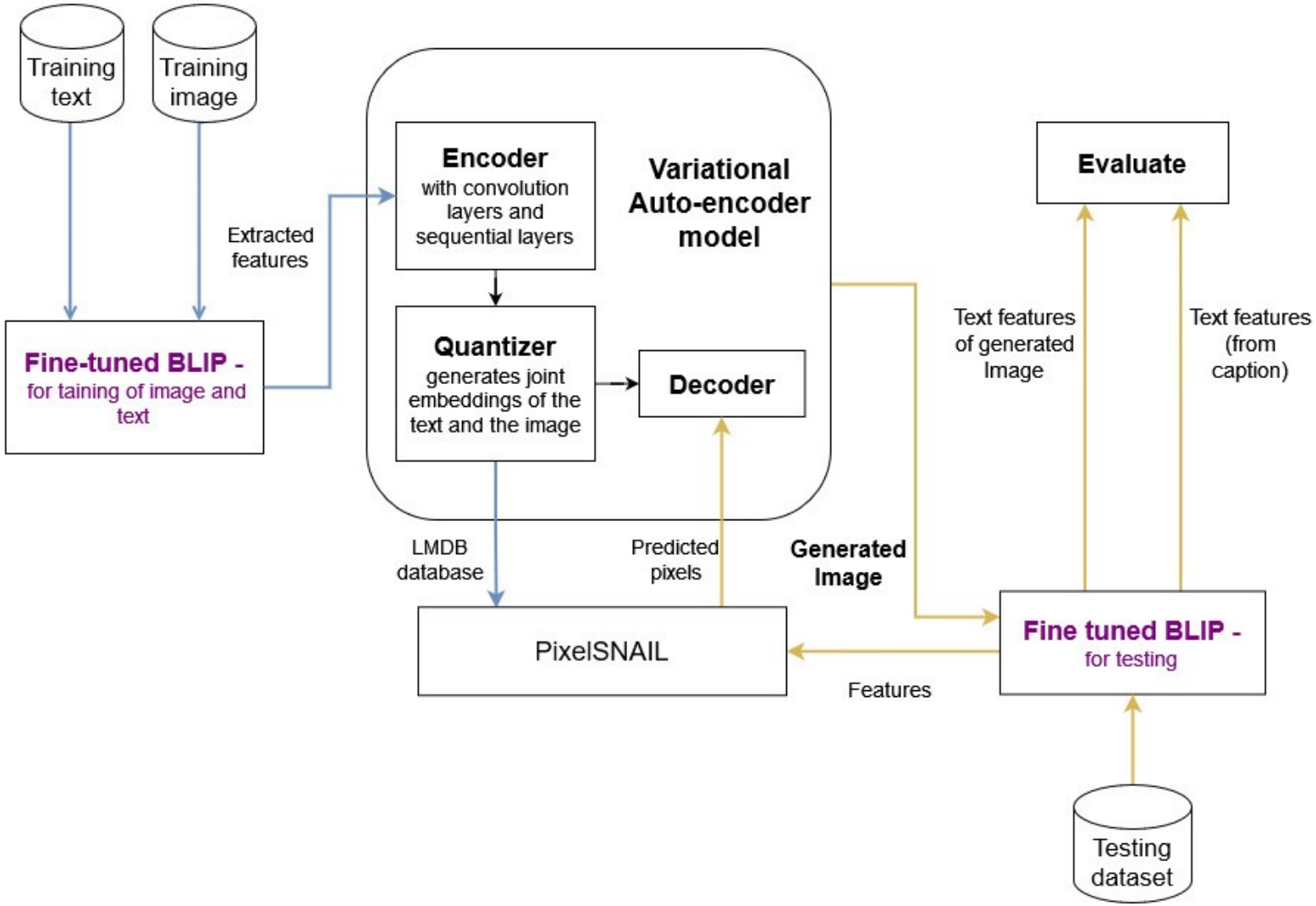
- Contains 795 historical art images along with captions and text in English

Testing: MediaEval MUSTI Testing Dataset

- Contains 200 historical art images along with captions and text in English


```
        {
            "id": 820,
            "image": "https://images.rkd.nl/rkd/thumb/
650x650/1f9a7843-23c5-7743-4c82-7070eea9ab09.jpg",
            "language": "en",
            "subtask1_label": "",
            "subtask2_labels": [],
            "text": "Whales are sometimes thrown upon the
coasts of Orkney , Shetland , and the Hebrides ; and ,
besides other fish which are caught for their oil , we may
mention the cearban or sun-fish , the fishery of which is
prosecuted with considerable success on the western
coasts .",
            "title": "Ship in a storm"
        },
```

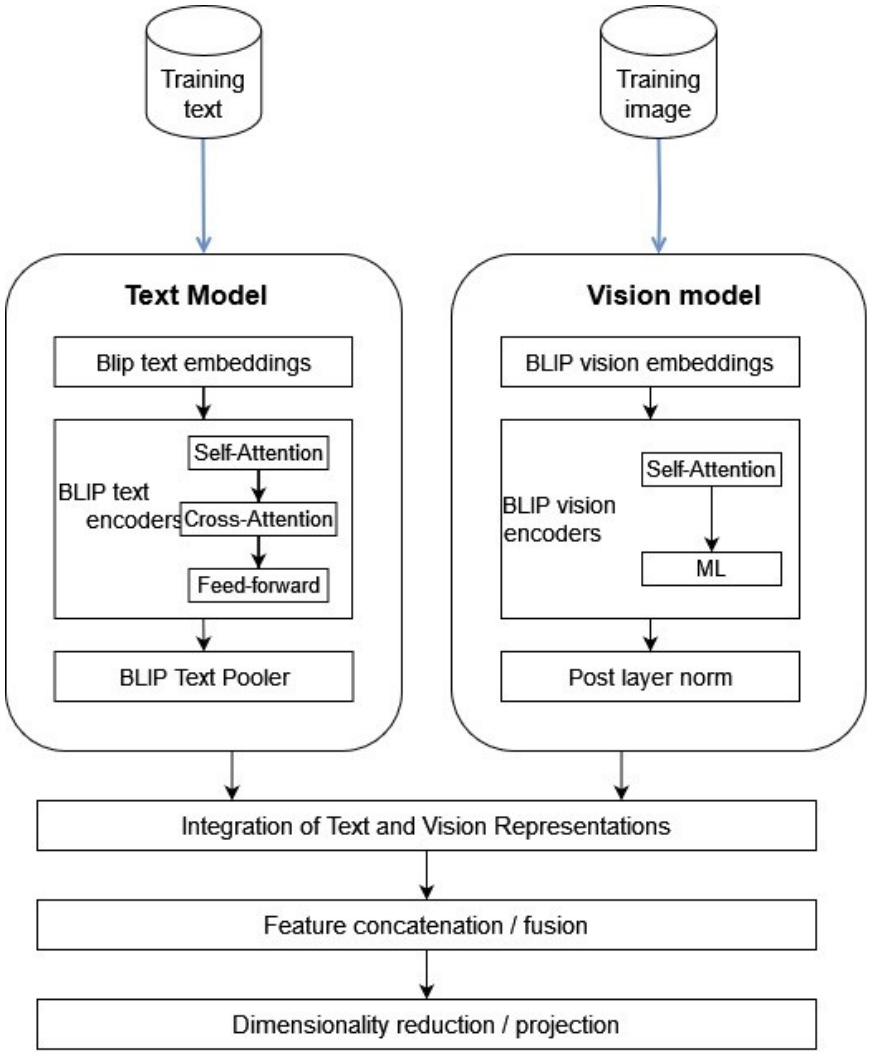

Proposed model - architecture



1. BLIP Fine-tuning

- The BLIP (Bootstrapping language-image Pre-training) model is a multimodal framework designed to learn joint representations of images and text.
- It consists of a text model based on BERT for processing textual data and a vision model based on CNNs and ViT for processing visual data.
- The text and vision models which are pre-trained individually, have been integrated into a joint text-vision model, which combines the representations learned by the text and vision models into a shared multimodal space which helps in cross-modal understanding.
- Fine-tuning using the MediaEval MUSTI 2023, BLIP is used to process the data and extract image and text embeddings which are further normalized and used to train the VQ-VAE model

1. BLIP Fine-tuning



The BLIP model

2. VQ-VAE

- VQ-VAE is a probabilistic model that finds latent, low-dimensional representations of data
- The encoders in the VQ-VAE encode images and text embeddings from the BLIP model into a lower dimensional latent space.
- The quantizer then takes this latent space as input and quantizes them into joint discrete embeddings according to the code-book. This is fed to the PixelSNAIL model.
- During the time of generation the decoder module reconstructs the quantized features back into the original features.

3. Code extraction module

- In this step, the trained VQ-VAE is used to extract latent codes from the images and store them in a Lightning Memory-Mapped Database (LMDB)
- The entire database is exposed in a memory map, and all data fetches return data directly from the mapped memory. This prevents from any memory copies or useless memory allocations during data fetch.
- Here, LMDB is used as it is highly efficient for data fetching from a large dataset and does not create any garbage data
- For each batch, images and text features are encoded using the VQ-VAE model to obtain latent codes. The latent codes are further converted into NumPy arrays which contain the latent code and filename. It is serialized and then stored in the LMDB database

4. PixelSNAIL model

- PixelSNAIL is an autoregressive generative model that is used in density estimation tasks with high dimensional data.
- The latent space of text and image embeddings that is stored in LMDB is used to train the model.
- This model helps in predicting the pixels while capturing long-range dependencies and incorporating conditional information.
- It works by leveraging causal convolutions, gated residual blocks, attention mechanisms, and conditional processing.
- The output of each PixelBlock is used as input to the next one in sequence. After processing through all PixelBlocks, the output is passed through a sequence of operations that include gated residual blocks, an ELU activation function, and a final convolutional layer to produce the final output.

5. Image generation module

- The PixelSNAIL predicts the pixels of the images. It is ensured that these pixel value is within the valid value (-1 to 1).
- After these samples are generated the VQ-VAE decoder is used to decode the sampled codes into actual images. The decoder of the VQ-VAE model performs the reverse process of the encoder, using convolutional layers and residual block to up-sample and capture details in the pixels generated by the PixelSNAIL to generate image.
- The generated images are finally saved into specified filenames.

Performance analysis - BLIP Fine-tuning

- The BLIP model provides insights into the strength and nature of the relationship between an image and its corresponding caption. By testing the BLIP model, how accurately the BLIP model predicts that there is a strong relationship between the image and the text is gauged.
- Various metrics are used to gauge the performance of the fine- tuned BLIP model, These metrics include accuracy, precision, recall, F1 Score and similarity score.

BLIP evaluation

Accuracy	1.0
Precision	1.0
Recall	1.0
F1 Score	0.996
Mean similarity score	1.0
Median similarity score	1.0
Standard deviation of similarity scores	0.0

Table: Evaluation metrics of the BLIP model

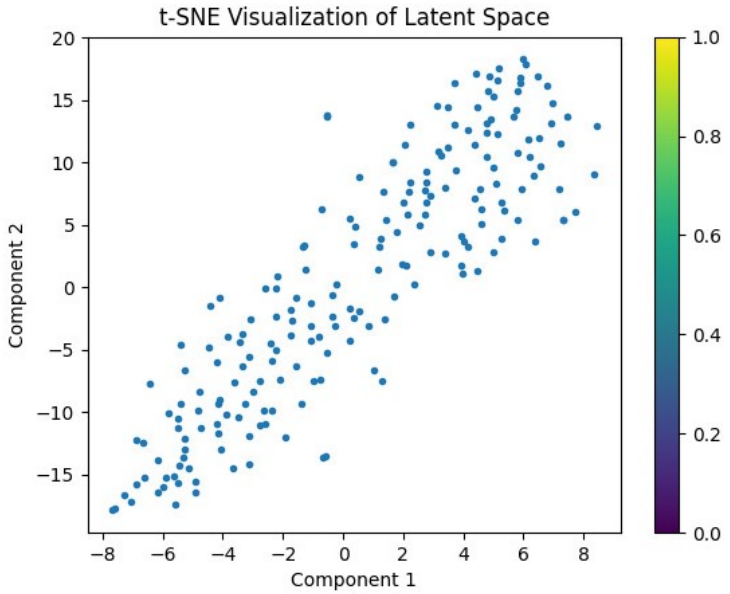
- Looking at the above metrics we can infer that the model's predictions are highly similar to the ground truth across all instances.
- Additionally, the standard deviation of similarity scores is 0.0, indicating very low variability in the model's performance.

Performance analysis - VQ-VAE model

To evaluate the encoder and quantizer, the codes of the testing dataset are extracted. 2 prominent algorithms are used to visualise the same, namely, the t-Distributed Stochastic Neighbor Embedding (t-SNE), Principal Component Analysis (PCA), and eventually, visualize the latent space generated.

Visualization of latent space - t-Distributed Stochastic Neighbor Embedding (t-SNE)

t-SNE is a powerful technique for dimensionality reduction and visualization of high-dimensional data.



This plot helps us visualize the structure of the latent space, with clusters of points indicating similarity among data points in the original space.

Visualization of latent space - Principal Component Analysis (PCA)

- PCA is a dimensionality reduction technique that identifies the directions (or principal components) that maximize the variance in the data and projects the data onto these components, reducing the dimensionality while preserving as much of the variance as possible.
- An MSE Loss of 0.240 is obtained, along with a PSNR Score of 6.5799. As seen, the MSE Loss is not as low as expected. Similarly, the PSNR Score is also only moderately high.
- The training and testing datasets are uniquely made of paintings and historical drawings, as opposed to the FFHQ face image datasets, which are made out of photographs, with clearly demarcated differences in colors and textures.

Evaluation of the images generated by the proposed system

- We extract the features of the test dataset using the BLIP model, and then feed it into the proposed system. The proposed system generates an image of resolution 64 x 64 px.
- As observed below, the images themselves are slightly pixelated and noisy, in comparison to the original images. This may be due to the smaller size of the PixelSNAIL model adapted to the hardware constraints.

Evaluation of the images generated by the proposed system

Bouquet of flowers in a vase on a marble table

Still-life of melons, mango's and a grasshopper

Whistling boy in pigpen

Figure: Image samples generated by the proposed model

Evaluation of the images generated by the proposed system

To evaluate the above model, we make use of several Natural Language Processing techniques. We pass the generated images through the BLIP model. This model generates captions for the images that we have generated.

The metrics are as follows:

- **The BLEU (Bilingual Evaluation Understudy)**
- **METEOR (Metric for Evaluation of Translation with Explicit Ordering)**
- **ROUGE-1 (Recall-Oriented Understudy for Gisting Evaluation)**
- **ROUGE-L (Recall-Oriented Understudy for Gisting Evaluation - Longest Common Subsequence)**

Evaluation of the images generated by the proposed system

Text Prompt	**Generated Image**	**Caption generated for evaluation**
Ship in a storm		Ship in ocean

Table: Depiction of the text prompt along with the generated image and a caption generated using this image which is used to evaluate contextual similarity

Evaluation of the images generated by the proposed system

Metric	**Value**
BLEU Score	0.47138
METEOR Score	0.13567
ROUGE-1 Score	0.4312
ROUGE-L Score	0.1563

Table: Evaluation metrics of the proposed system

As we can see in the above table, the BLEU score is fairly good, while the METEOR score is lower. This, we estimate, is due to the fact that the METEOR score gives high importance to word order and stemming. The ROUGE scores, similarly, are higher for the ROUGE-1 metric, due to calculation of unigram precision, and lower for the ROUGE-L metric due to the measurement of longest common subsequence.

Expected outcome

To evaluate the above model, the Learned Perceptual Image Patch Similarity (LPIPS) metric is used. LPIPS essentially computes the similarity between the activations of two image patches for the given pre-defined network. This measure has been shown to match human perception well. A low LPIPS score means that image patches are perceptual similar.

Model	**Value**
CycleGAN	0.134
DSVIB	0.129
Proposed system	0.124

Table: LPIPS Score Comparison

References

- Radford, Alec, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry et al. "Learning transferable visual models from natural language supervision." In International conference on machine learning, pp. 8748-8763. PMLR, 2021.
- J. Li, D. Li, C. Xiong, S. Hoi, Blip: Bootstrapping language-image pre-training for unified vision language understanding and generation, in: International Conference on Machine Learning, PMLR, 2022, pp. 12888–12900
- Ramzan, Sadia, Muhammad Munwar Iqbal, and Tehmina Kalsum. "Text-to-Image Generation Using Deep Learning." Engineering Proceedings 20, no. 1 (2022): 16.
- Reed, Scott, Zeynep Akata, Xinchen Yan, Lajanugen Logeswaran, Bernt Schiele, and Honglak Lee. "Generative adversarial text to image synthesis." In International conference on machine learning, pp. 1060-1069. PMLR, 2016.

References

- Qiao, Tingting, Jing Zhang, Duanqing Xu, and Dacheng Tao. "Mirrorgan: Learning text-to-image generation by redescription." In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 1505-1514. 2019.
- Xia, Weihao, Yujiu Yang, Jing-Hao Xue, and Baoyuan Wu. "Tedigan: Text-guided diverse face image generation and manipulation." In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 2256-2265. 2021.
- Kocasari, Umut, Alara Dirik, Mert Tiftikci, and Pinar Yanardag. "StyleMC: multi-channel based fast text-guided image generation and manipulation." In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp. 895-904. 2022.
- Nichol, Alex, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. "Glide: Towards photorealistic image generation and editing with text-guided diffusion models." arXiv preprint arXiv:2112.10741 (2021).

References

- Zhou, Yufan, Ruiyi Zhang, Changyou Chen, Chunyuan Li, Chris Tensmeyer, Tong Yu, Jiuxiang Gu, Jinhui Xu, and Tong Sun. "Towards language-free training for text-to-image generation." In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 17907-17917. 2022.
- Ramesh, Aditya, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. "Hierarchical text-conditional image generation with clip latents, 2022." URL https://arxiv. org/abs/2204.06125 7 (2022).
- Vector Quantized Diffusion Model for Text-to-Image Synthesis
- Generating Diverse High-Fidelity Images with VQ-VAE-2

Thank You