20XD96 - Information Retrieval and Web Search Lab Package Report

Text to Image Retrieval

20PD09 - Jash Bhandari

20PD14 - Mahitej Reddy K

20PD22 - Rishi Ram S

24th October, 2024

Introduction

This project focuses on building a text-to-image retrieval system using Conditional Augmentation and StackGAN. The goal is to enable the generation of high-quality images from textual descriptions, providing a powerful tool for visualizing information based solely on text input.

Conditional Augmentation enriches the text embeddings by introducing controlled variability, which helps the model generate diverse and accurate images aligned with the input text. These embeddings are derived from a pre-trained Sentence Transformer and conditioned to ensure better mapping between text and image domains.

The project employs a **two-stage StackGAN architecture**. In the first stage, a low-resolution image is generated based on the conditioned text input, capturing the basic features. In the second stage, this image is refined, enhancing the details and producing a high-resolution output that more closely matches the input description.

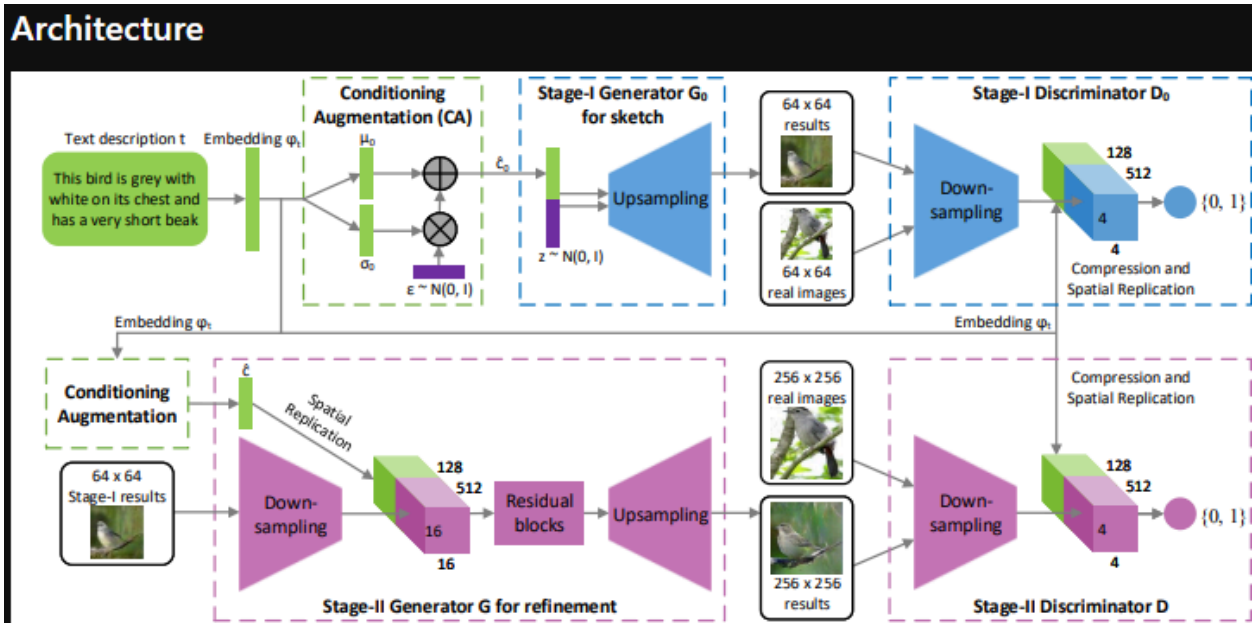
A key feature of the system is the **Streamlit interface**, allowing users to input textual descriptions and retrieve corresponding images in real-time. This interface simplifies the interaction with the model, providing a smooth experience for generating and viewing results.

This report covers the design, training, and implementation of the system, highlighting the use of Conditional Augmentation and StackGAN, along with the integration of a user-friendly interface for seamless text-to-image retrieval.

Tools and Techniques Used

- **TensorFlow and Keras**: Used for building and training deep learning models, including the generator and conditioning networks.
- **Sentence Transformers**: Employed for converting textual input into meaningful embeddings, which serve as the foundation for generating images.
- **Conditional Augmentation**: Introduced to add controlled randomness to text embeddings, helping the model generate varied and accurate images based on the text input.
- **StackGAN**: A two-stage generative model used to progressively generate images, starting from low-resolution to high-resolution, refining the details in each stage.
- **LeakyReLU and ReLU Activations**: Applied to introduce non-linearity in the model, aiding in better feature extraction and image generation.
- **Batch Normalization**: Used to stabilize training by normalizing the outputs of layers, reducing mode collapse during image generation.
- **UpSampling2D and Conv2D Layers**: Responsible for scaling up the image resolution and adding details through convolutions during the image generation process.
- **Streamlit**: Used to create a user-friendly interface for inputting text and displaying generated images in real time.

Flow



Conditional Augmentation

Conditional Augmentation is a technique used to add controlled randomness to text embeddings, enhancing the variety and accuracy of generated images. It introduces variability by generating random noise from a Gaussian distribution, which helps in producing diverse images from the same text input, avoiding repetitive outputs.

In this project, Conditional Augmentation is implemented as follows:

- The input text is converted into embeddings using a Sentence Transformer.
- These embeddings are split into two parts: the mean and the log-sigma (logarithmic standard deviation).
- A random noise vector is sampled from a Gaussian distribution, and this noise is added to the mean, scaled by the standard deviation (derived from the log-sigma), resulting in a conditioned vector.

- This conditioned vector, which represents the text in a more nuanced way, is then used in the image generation process.
- This augmentation allows the model to produce a wider range of images from the same text, improving both the diversity and realism of the generated outputs.

StackGAN Model

StackGAN is a generative model that generates high-resolution images from textual descriptions through a multi-stage process. It consists of two stages: the first generates a coarse, low-resolution image, and the second refines it into a detailed, high-resolution image.

In this project, **StackGAN** is implemented with the following stages:

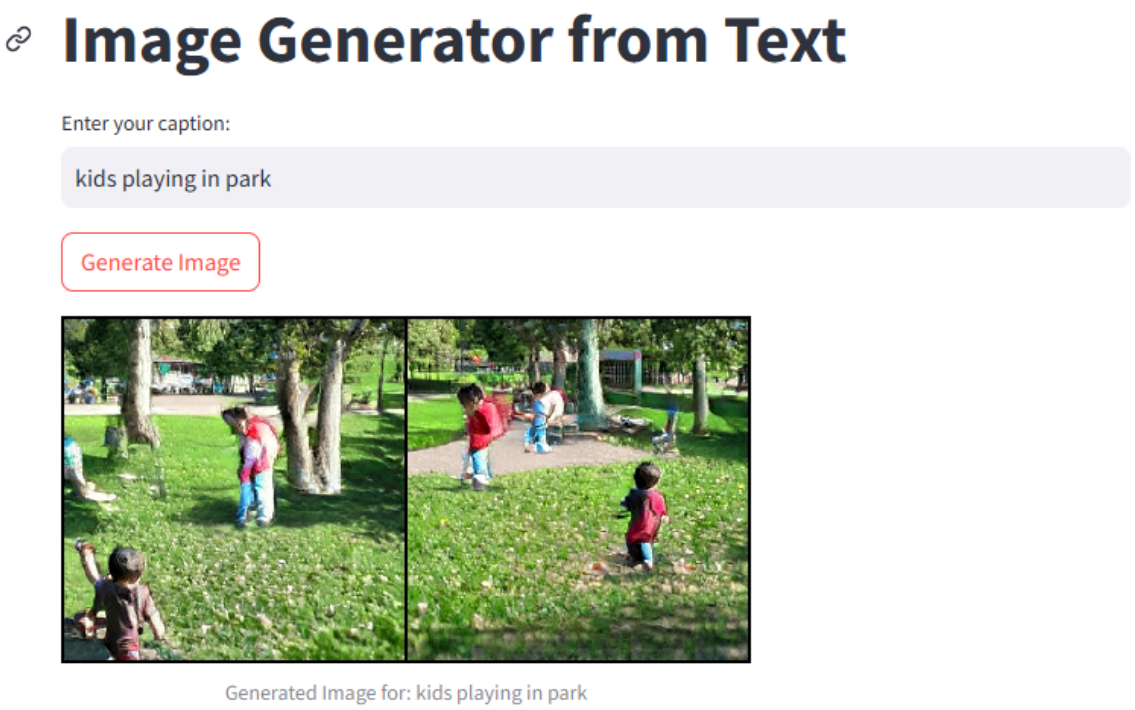
- **Stage 1**:
  - A low-resolution image (64x64) is generated based on the conditioned text embeddings.
  - The embeddings are passed through dense layers and upsampling blocks, progressively building the basic structure of the image.
  - This stage focuses on capturing the rough shape and color distributions of the object described in the text.
- **Stage 2**:
  - The image from Stage 1 is refined by adding more details and improving the resolution.
  - The refinement includes additional upsampling and convolution layers that help generate a more realistic and higher-resolution image.

In this project, **two levels** (stages) are used:

1. **Stage 1 Generator**: Produces the initial 64x64 image based on the conditioned embeddings.
2. **Stage 2** would typically take the output of Stage 1 and refine it to a higher resolution (usually 256x256).

The **StackGAN architecture** in this project ensures that the generated images are both contextually accurate and visually detailed, with each stage contributing to progressively improving the quality of the images.

Results



Conclusion

This project successfully demonstrates a text-to-image retrieval system using Conditional Augmentation and StackGAN, with a user-friendly Streamlit interface for seamless interaction. By leveraging Conditional Augmentation, the model effectively introduces controlled randomness into text embeddings, allowing for the generation of diverse and accurate images. The two-stage StackGAN architecture ensures that high-resolution, detailed images are progressively created from basic low-resolution outputs, refining visual quality at each stage.

The implementation shows that the combination of deep learning techniques, such as conditional augmentation and generative adversarial networks, can significantly enhance the task of converting textual descriptions into realistic images. The project's interface further simplifies user interaction, making it an intuitive tool for retrieving images from text inputs.

Overall, this system offers a powerful approach to visualizing textual information and demonstrates the potential of advanced generative models in text-to-image retrieval applications.