Icon Generation

Presented By:
Sachit Lele
Shivangi Singh
Shreyas Rai

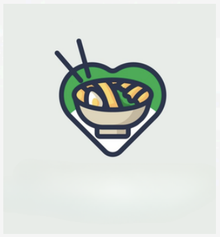
Introduction

In this project, we utilize GANs to generate a wide range of icon designs of specific types desired by the user. Our process involves training the model on existing icon datasets with its description, refining the results iteratively, and evaluating for quality. By showcasing this approach, we aim to inspire creative exploration in icon design.

The novelty here derives from the conditioning, which would make the resulting icons substantially more useful for our end users.

About the project

In the realm of digital design, icons are pivotal for enhancing user experience and interface aesthetics. Yet, crafting diverse and tailored designs can be time-consuming. Our project reimagines this process by utilizing Generative Adversarial Networks (GANs) to revolutionize icon design, driven by user prompts for personalized outcomes



Brand Alignment: Customizable designs ensure icons align with companies' brand identities and visual guidelines seamlessly.
Accessibility: Diverse iconography promotes inclusivity, enhancing user experiences for all individuals.

Dataset

mordern -logo-dataset (Hugging Face)

	"a logo of coffee shop, take-away coffee cardboard glass with white and brown stripes and dark brown lid, coffee circle with three cream drops, white background, brown foreground,…
	"a logo of coffee shop, White round background with black rim, cup, pretzel, horizontal stripe and cookery lettering, tan background, snow, darkslategray foreground, minimalism, modern"
	"a logo of coffee shop, image of a filled cup with steam in a square, white background, black, darkgray foreground, minimalism, modern"
	"a logo of cafe restaurant bar pizzeria with a slice of pizza at the top part of the circle, label with the year '2020' at the bottom and crossed rectangle in the middle, whitesmoke…

- 803 logo with their description.
- This dataset consists of 803 pairs (x,y)(x,y), where xx is the image and yy is the description of the image.
- The logos in the dataset are minimalist, meeting modern design requirements and reflecting the company's industry.

Literature on Topic

- **Scarcely** explored subject: **Less Literature Available**
- Generation on available Projects: **Poor Quality**

- Topic of Emphasis: **Fusion Approach (text +? image)**

- **Three** main types:
  - **Naive** Feature Concatenation (C-GAN 2014)
  - **Cross-Modal** Fusion (AttnGAN 2019)
  - **Conditional** Batch Normalization (SD-GAN 2018)

Methodology

Logo Generation

↓

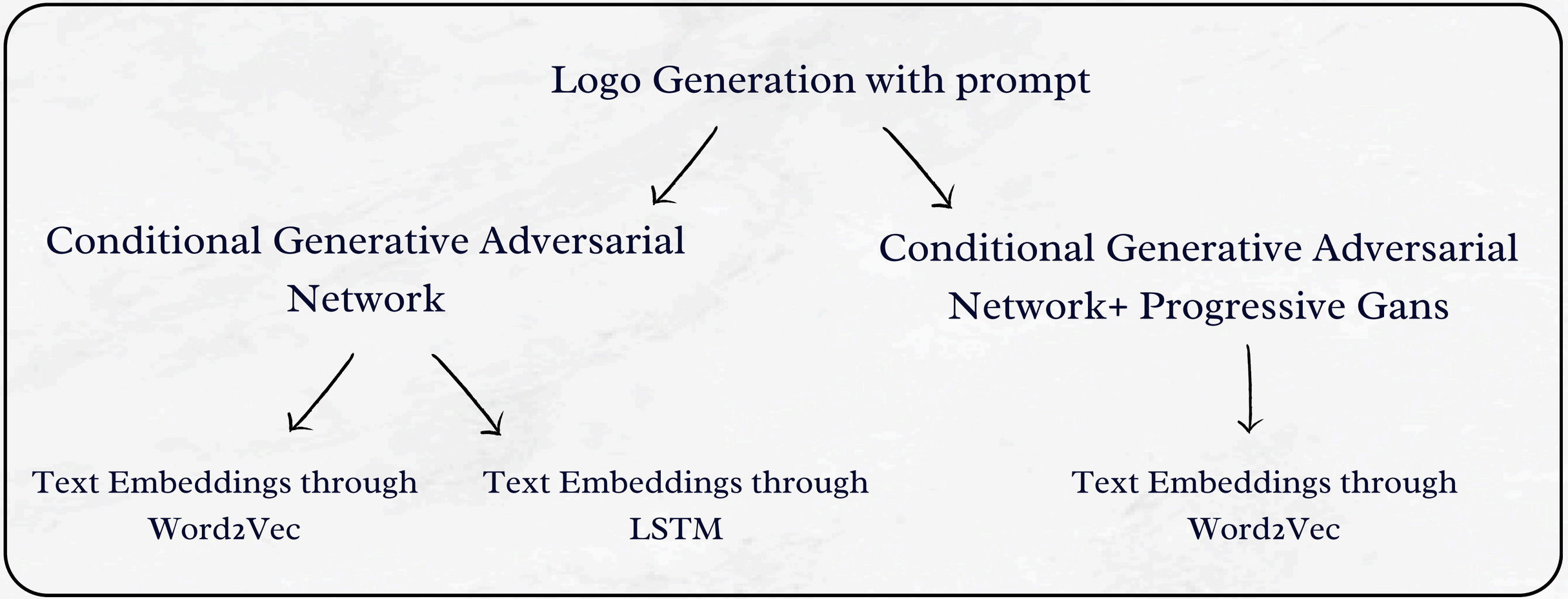
Generative Adversarial Network

Evaluation Metrics

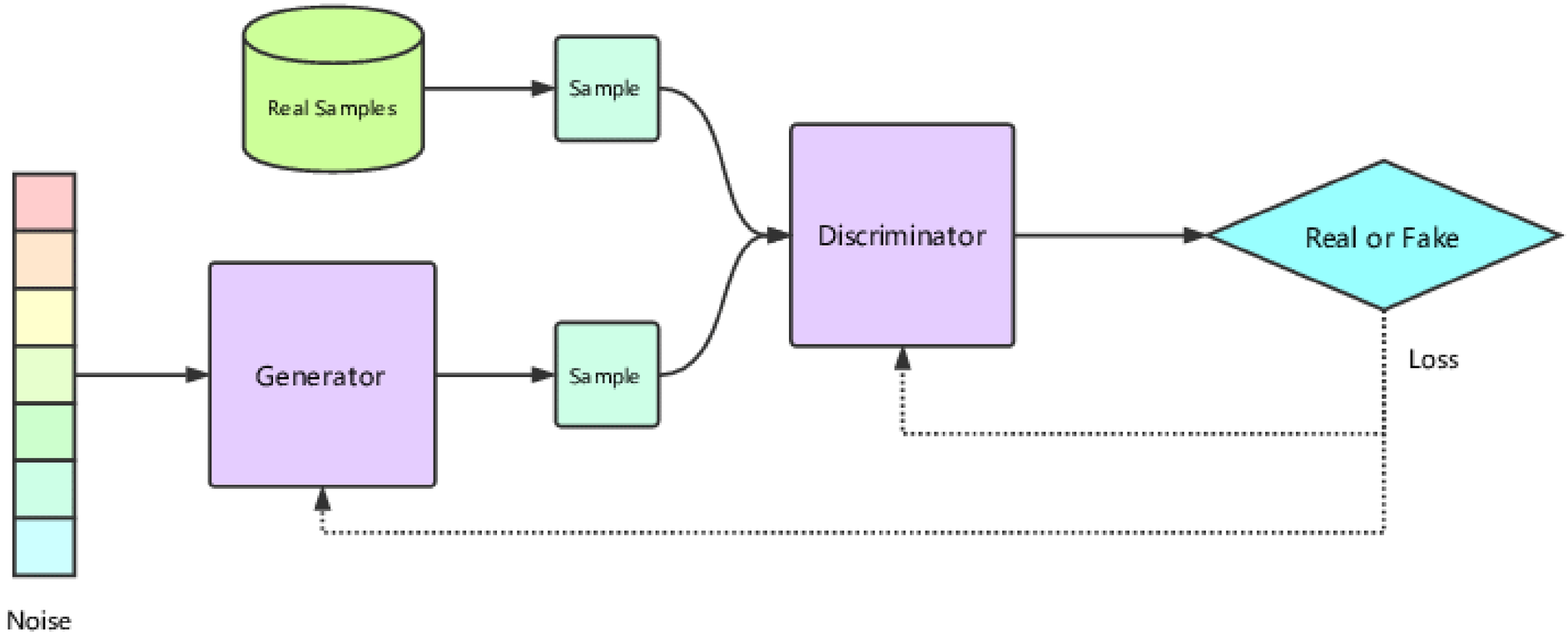
↓

Inception Score

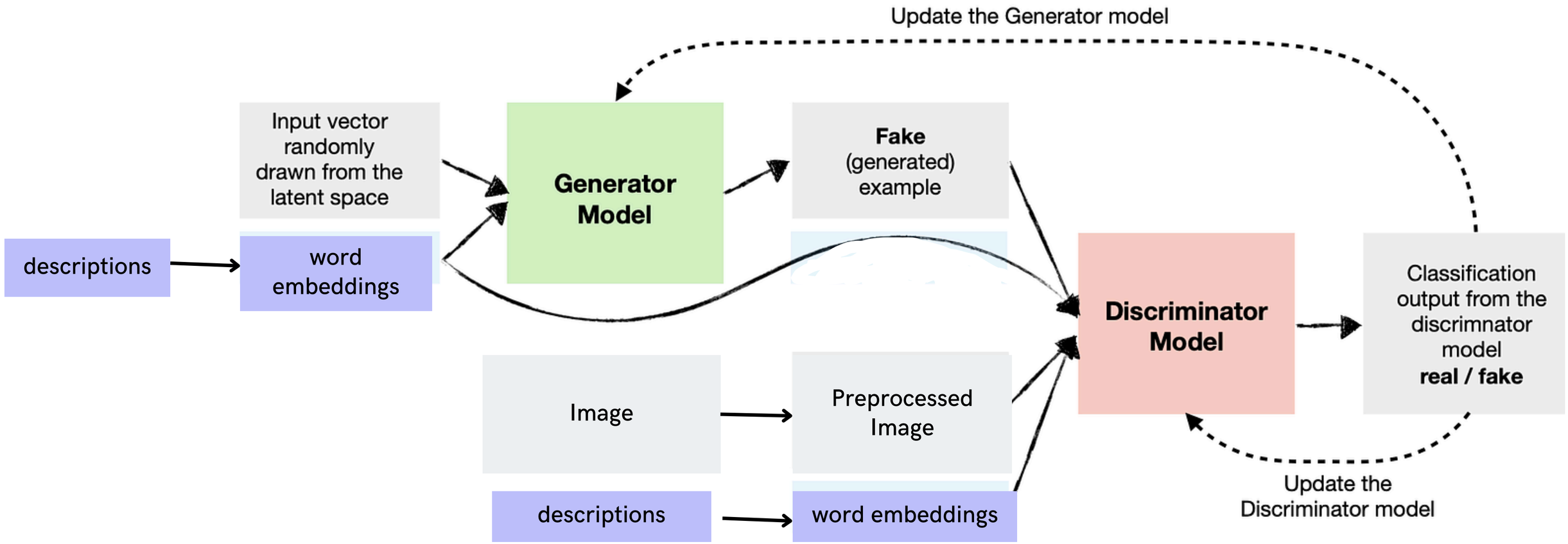
Methodology



Outline-Architecture of Generative Adversarial Networks (GANs) based on logo generation



Outline-Architecture of Conditional Generative Adversarial Networks (cGANs) based on logo generation through prompt



Preprocessing

Input Image Preprocessing:

- Convert the image stored in bytes format to an image object.
- Convert the image to RGB format if it's not already in that format.
- Resize the image to a standard size, such as 64x64 pixels.
- Convert the image to a numpy array for further processing.
- Normalize the pixel values of the image array to the range [0, 1] .

Text Embedding Generation using Word2Vec:

- Split the text into individual words or tokens.
- Word2Vec is a popular technique used to generate word embeddings, which are dense vector representations of words.
- Obtain the final text embedding representation, which can be used as input to the conditional GAN along with the preprocessed image for generating the logo.

Text Embedding Generation using LSTM:

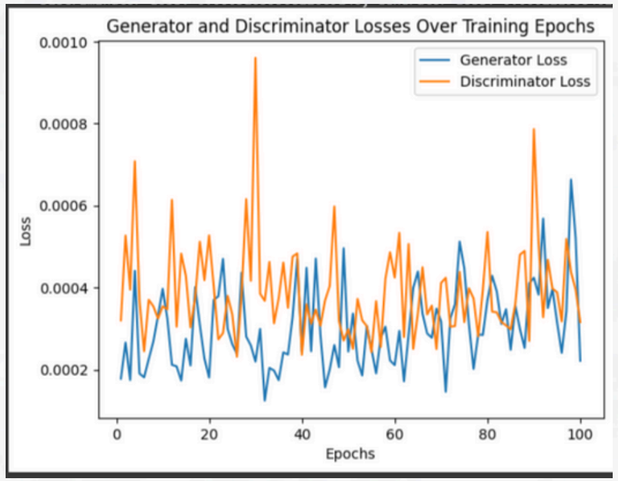
- the LSTM layer processes the text embeddings, capturing sequential information and dependencies within the text data.

Training


GAN TRAINING
ITER 0
imgflip.com
© Phil Bray/Netflix/Everett Collection/picture alliance

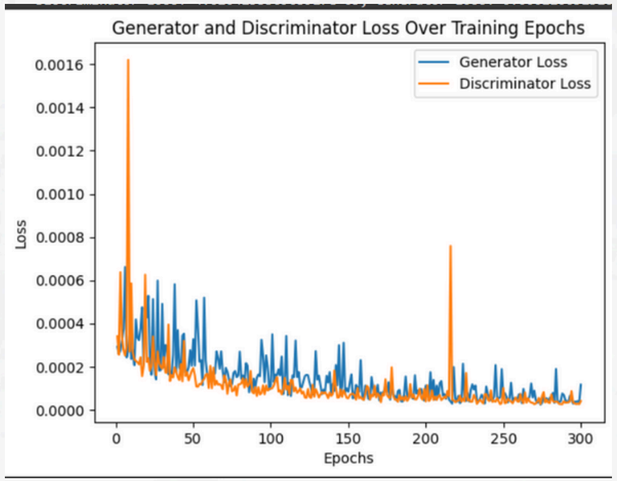
Output

For 100 epochs



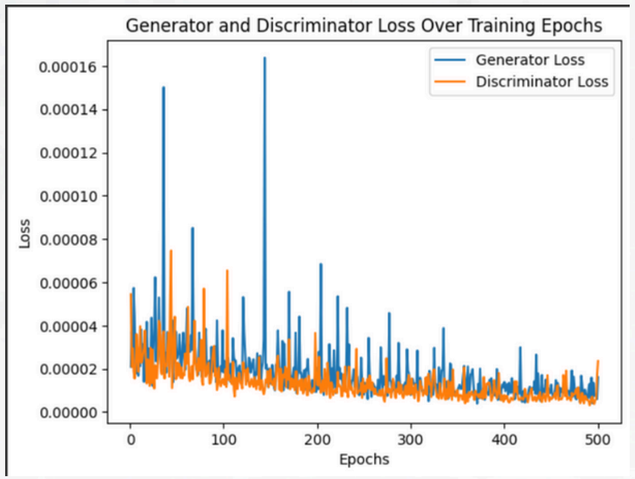
Inception Score: 3.9947908

For 300 epochs



Inception Score: 3.2692263

For 500 epochs

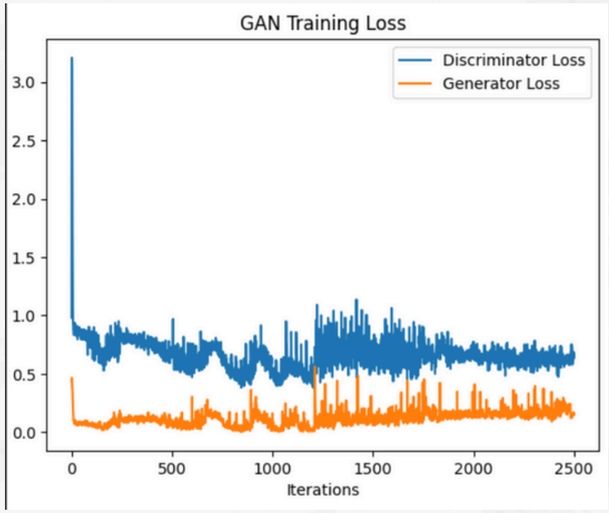


Inception Score: 3.1285596

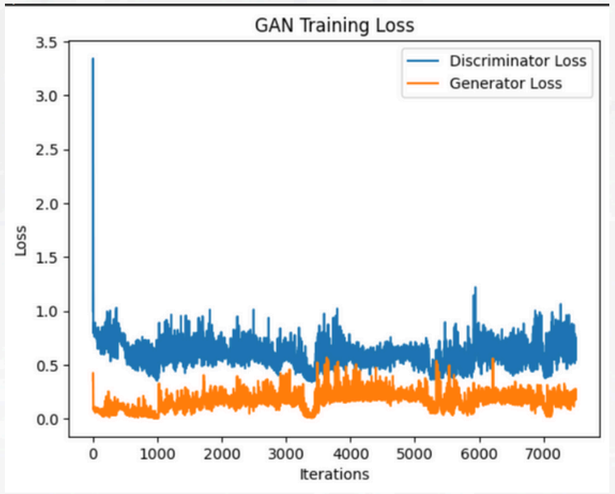
CGan through Word2Vec

Output

For 100 epochs



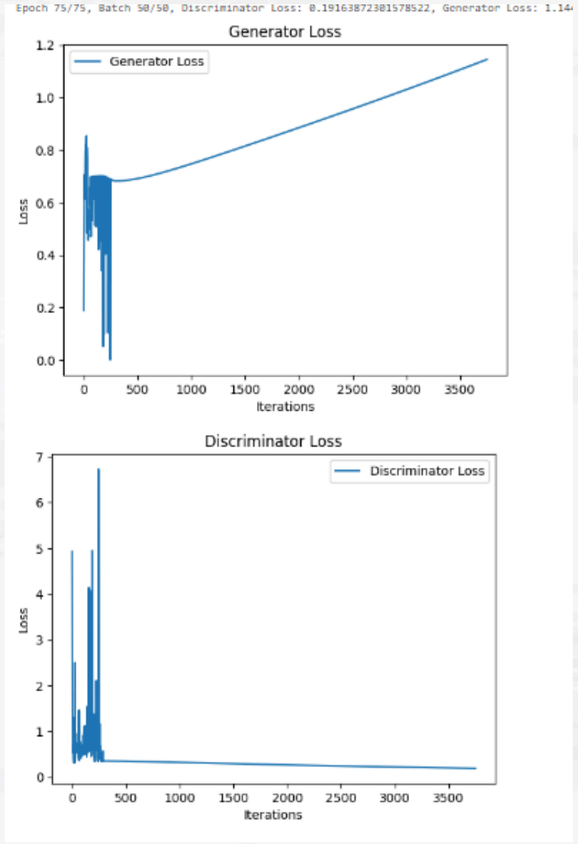
For 300 epochs



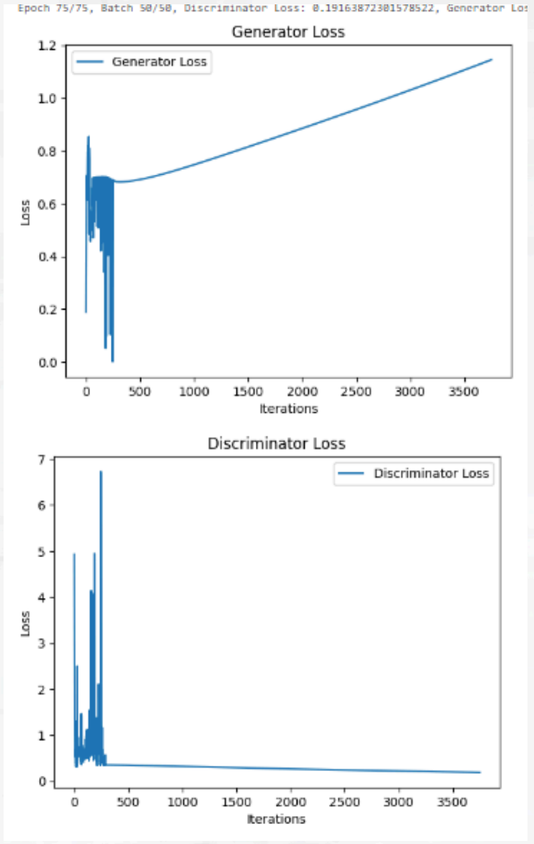
Simple Gan

Output

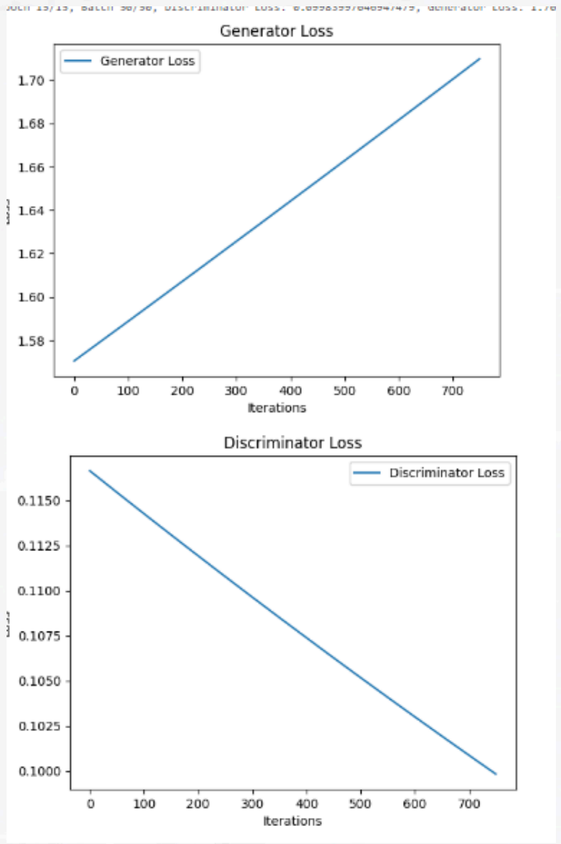
For 100 epochs



For 300 epochs



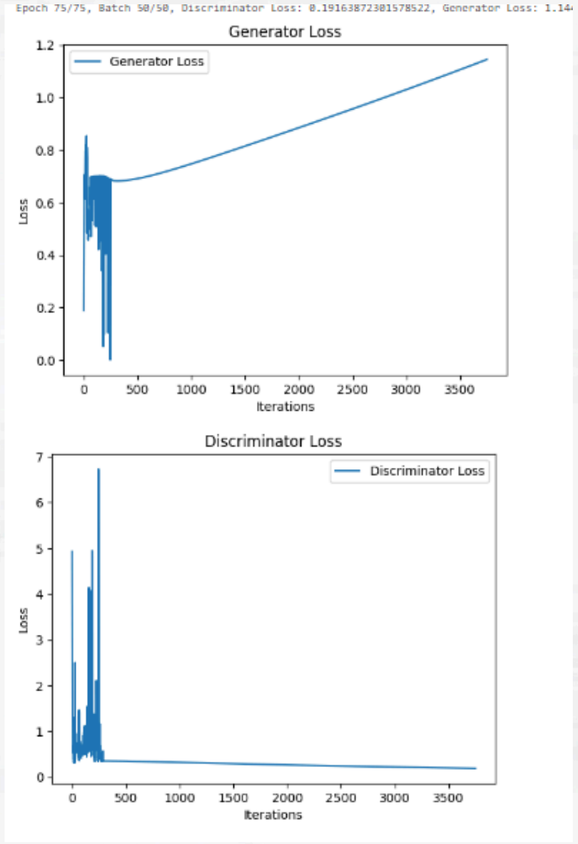
For 500 epochs



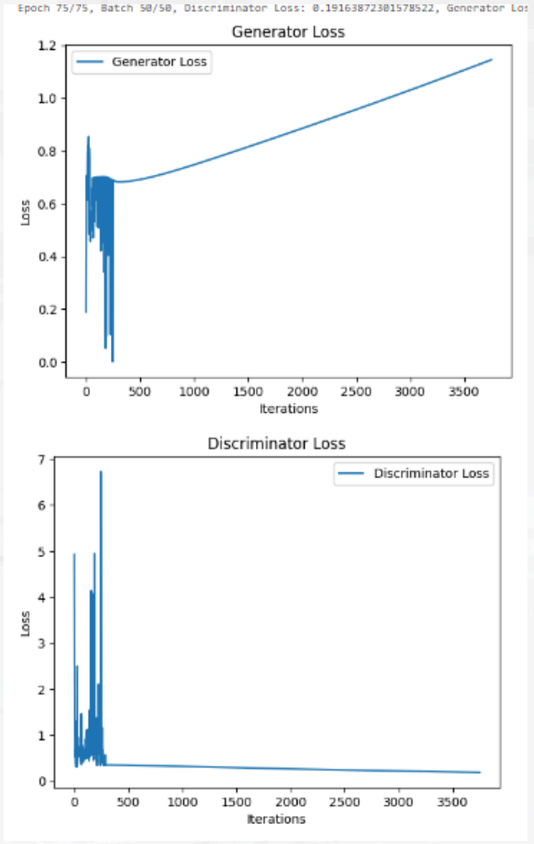
Cgan LSTM

Output

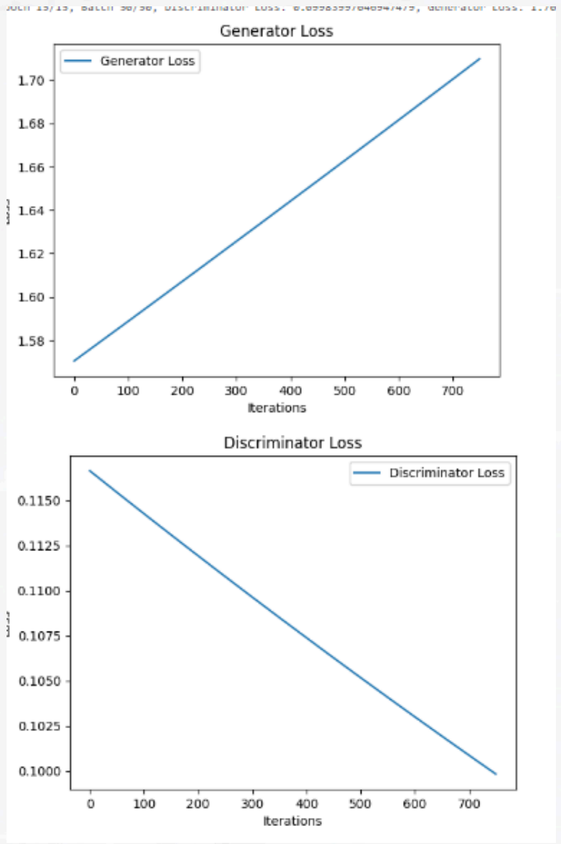
For 100 epochs



For 300 epochs



For 500 epochs



Cgan LSTM

Output

Conclusion

1. Progressive GANs have demonstrated superior image generation quality relative to computational resources expended.
2. The generative loss trend for LSTM GANs exhibited unexpected behavior, as it consistently increased linearly per epoch instead of the anticipated decrease.
3. The Simple GAN was initially employed to validate the feasibility of generating images and to identify areas for enhancement to achieve superior outputs.

Tools used

1. Colab T4 GPU
2. PyCharm
3.Kaggle
4. Github
5. Canva
6. Google Scholar
7. Papers with Code

Reference

1. https://machinelearningmastery.com/how-to-develop-a-conditional-generative-adversarial-network-from-scratch
2. Generative Adversarial Text-to-Image Synthesis https://arxiv.org/abs/1605.05396
3. Conditional GAN(2014): 1411.1784 (arxiv.org)
4. Attention Gan(2018): 1711.10485 (arxiv.org)
5. Semantic Disentanglement Gan(2019): 1904.01480 (arxiv.org)

thankyou ⟶