Problem Statement 1: Neural Style Transfer

Imagine a world where you could transform any ordinary photograph into a masterpiece by applying the artistic style of renowned painters like Van Gogh, Picasso, or Monet. Your task is to develop a neural style transfer model that leverages state-of-the-art deep learning techniques to merge the content of one image with the style of another, creating stunning new visuals that blend the two seamlessly.

This model should be capable of extracting the stylistic features from a given artwork and applying them to a different image while preserving the original content's structure and details. The aim is to achieve a balance between the content and style, resulting in images that are visually appealing and artistically coherent.

You are encouraged to explore techniques such as convolutional neural networks (CNNs), optimization algorithms, and perceptual loss functions to accomplish this. However, the use of external APIs is not permitted.

You have the freedom to choose how to implement and showcase your model. Options include creating a web interface where users can upload images and select styles to apply or integrate the model into existing software tools for artistic creation.

Here are some references to help you get started:

https://arxiv.org/pdf/1508.06576.pdf https://arxiv.org/pdf/1603.08155.pdf

https://arxiv.org/pdf/1703.06868.pdf https://arxiv.org/pdf/1705.06830.pdf

https://arxiv.org/pdf/1804.03547.pdf https://arxiv.org/pdf/1912.07921.pdf

Your goal is to develop a model that not only produces visually striking

images but also demonstrates a deep understanding of both the content

and the artistic style, enabling users to create unique and personalized

artwork.

Problem Statement 2: Text to Image Generation

Imagine a world where you could describe an image in natural language, and an AI could generate that image for you with stunning accuracy and detail. Your task is to develop a text-to-image generative model that leverages state-of-the-art natural language processing techniques to create realistic images based on textual descriptions.

This model should be capable of understanding and translating textual descriptions into high-fidelity images, capturing intricate details and nuances while maintaining coherence with the input description. It should be adaptable to various domains, from everyday objects to complex scenes and landscapes.

You are encouraged to explore techniques such as attention mechanisms, transformer architectures, and adversarial training to achieve this. However, the use of external APIs is not permitted.

You have the freedom to choose how to implement and showcase your model. Options include creating a web interface where users can input text descriptions and receive corresponding images or integrating the model into existing software tools.

Here are some references to help you get started:

https://arxiv.org/pdf/1511.02793.pdf https://arxiv.org/pdf/1609.04802.pdf

https://arxiv.org/pdf/1802.06454.pdf https://arxiv.org/pdf/1812.04948.pdf

https://arxiv.org/pdf/2104.04843.pdf https://papers.nips.cc/paper/6125-

improved-techniques-for-training-gans.pdf

https://distill.pub/2018/differentiable-parameterizations/

https://www.coursera.org/specializations/generative-adversarial-

networks-gans

Your goal is to develop a model that not only generates visually appealing images but also demonstrates a deep understanding of the textual descriptions, opening new possibilities for creative expression and content creation.

Submission Form PS2: https://forms.gle/EdrbafxvuvvX8ZtN9

WhatsApp Group - https://chat.whatsapp.com/CHMT3BKV8SnHBni08Hxoq4

Problem Statement 3: Pdf Answering AI

Have you ever wondered if your pdf could talk to you and respond to your questions and queries? With this project you can create an AI that responds to your questions about the pdf that is being viewed at the moment.

Your task at hand is to create a model that leverages natural language processing techniques to provide quick and accurate responses from the context of the pdf that is being viewed. You are free to use any methods within the constrictions of the project like word2vec, glove embedding technique etc.., however, use of APIs is not allowed.

There are multiple ways to use your model and display it's working. You are free to use any method. Following is some of the methods recommended to use:

- Create a google extension (brownie points)

- Using a simple upload pdf and question interface (recommended)

Following are some references to help you get started:

- https://arxiv.org/pdf/1805.08092.pdf

- https://ieeexplore.ieee.org/abstract/document/9079274

- https://arxiv.org/pdf/1707.07328.pdf

- https://arxiv.org/pdf/1810.04805.pdf

- https://papers.nips.cc/paper/7181-attention-is-all-you-need.pdf

- https://nlp.seas.harvard.edu/2018/04/03/attention.html

- http://jalammar.github.io/illustrated-transformer/

- https://jalammar.github.io/visualizing-neural-machine-translation-mechanics-ofseq2seq-models-with-attention/

- https://www.coursera.org/learn/nlp-sequence-models

- https://www.youtube.com/playlist?list=PLam9sigHPGwOBuH4_4fr-XvDbe5uneaf6

Submission Form PS3: https://forms.gle/wfxbKzpFvcYLqc8r7

WhatsApp Group - https://chat.whatsapp.com/IgRkDwpDFPX6ZgyChzHr4Y

*Submission deadline for all three PS is **20th June 2024, 11:59 pm**.

Guidelines: -

To ensure the quality and originality of submissions, please adhere to the following guidelines when submitting your machine learning project for verification:

1. Originality and Integrity

- No Copied Projects: Submissions must be original. Any form of plagiarism will result in immediate disqualification.
- Source Verification: If your project is found to be copied or significantly derived from existing work without proper attribution, it will be discarded without review.

2. GitHub Repository

- Project Upload: Upload your entire project to a GitHub repository. Ensure that all necessary files are included.
- Repository Structure: Organize your repository clearly. Include folders for code, data, results, and documentation.
- README File: Include a comprehensive README file with the following:
    - Project Overview: A brief description of your project.
    - Installation Instructions: Steps to set up the environment and run the project.
    - Usage: Examples of how to use the code.
    - Dependencies: List of all required libraries and dependencies.

3. Detailed Report

- Length and Format: Submit a detailed report of 7-10 pages in PDF format.
- Content Requirements:
    - Introduction: Explain the problem statement and the objectives of your project.
    - Approach: Describe your methodology and the steps you followed to solve the problem.
    - Failed Approaches: Document any approaches that did not work, along with reasons for their failure.
    - Results: Present the results of your project, including any relevant metrics, graphs, or visualizations.
    - Discussion: Analyze the results, discussing their significance and any insights gained.
    - Conclusion: Summarize your findings and suggest possible future improvements.
    - References: Include citations for any external resources or papers you referred to.

4. Submission Process

- Deadline: Ensure your project is submitted by the specified deadline.
- Verification: All submissions will undergo a thorough review process to verify originality and quality and will be verified by ArIES upon completion.

Thank you for your participation!