CS F429 - Natural Language Processing
Group 2

Rudra Jewalikar - 2021A7PS0450P
Harsh Deshpande - 2021A7PS2225P

Codebase Structure

Apart from the previous submission of stage 1: The following additions are made

- The src folder contains well structured code as well as a final notebook ‘prism.ipynb’ for reproducing the code experiments with PRISM. The reference images used are also present as png files in the src folder itself. You can check prism.ipynb to have a look at the overall structure of PRISM.
- In the src folder itself you can view the overall iterative process by checking out the folders image_logs and interpretable_feedback_and_prompts.**These folders are crucial to the interpretability aspect which our pipeline touches.**
- Additionally image_generation_prism.ipynb and image_prompt_generation_prism.ipynb have been added. These show the modified code post adding PRISM.
- image_quality_evaluation.ipynb: This notebook contains the code to evaluate our final results using CLIP score.
- Image_prompts_prism and images_prism contain the image prompts and final images generated by the pipeline post incorporating PRISM.
- **Please do have a look at all the results in these folders too. Given the volume of experiments we have run, it did not seem feasible to add all the results in this documentation.**

Inspiration and Proposed Improvements

During our initial exploration, we noticed that a significant amount of time was consumed in the iterative process of improving prompts for tasks like summary generation and image generation. This involved producing an output, visually assessing its quality, updating the prompt, and repeating the process until satisfactory results were achieved. The manual nature of this workflow made it labor-intensive and time-intensive.

This observation inspired us to automate the process by leveraging a Large Language Model (LLM) to evaluate image outputs and provide constructive feedback. The paper ***Automated Black-Box Prompt Engineering for Personalized Text-to-Image Generation*** served as the

foundation for our approach. Its proposed PRISM algorithm demonstrates the use of LLMs to iteratively refine prompts while maintaining transferability and interpretability.

Interpretability was one of the main reasons we chose PRISM as the basis for our approach. Our previous architecture also emphasized interpretability, allowing us to clearly understand and track how prompts and outputs evolved. Similarly, PRISM ensures that the generated prompts and feedback suggested at each iteration are fully visible and human-readable.

This transparency makes the logical improvements comprehensible to users and enables the identification of potential biases or “cheating” introduced by the model. For example, if the model attempts to exploit overly simplistic patterns that achieve high scores without truly improving the output, such behavior can be spotted and corrected.(See example in output)

By incorporating this methodology, we aimed to achieve the following improvements:

1. **Reduction in Human Effort and Time**: Automating the prompt refinement process eliminates the need for manual intervention, significantly cutting down the time and labor involved.
2. **Introduction of Objectivity**: The reliance on human judgment for image evaluation is inherently subjective. Using an LLM-based judge ensures consistent and objective evaluations, leading to more reliable results.

This approach aligns with the goal of creating efficient, scalable, and model-agnostic solutions for generating high-quality prompts and outputs.

Implementation and Incorporation

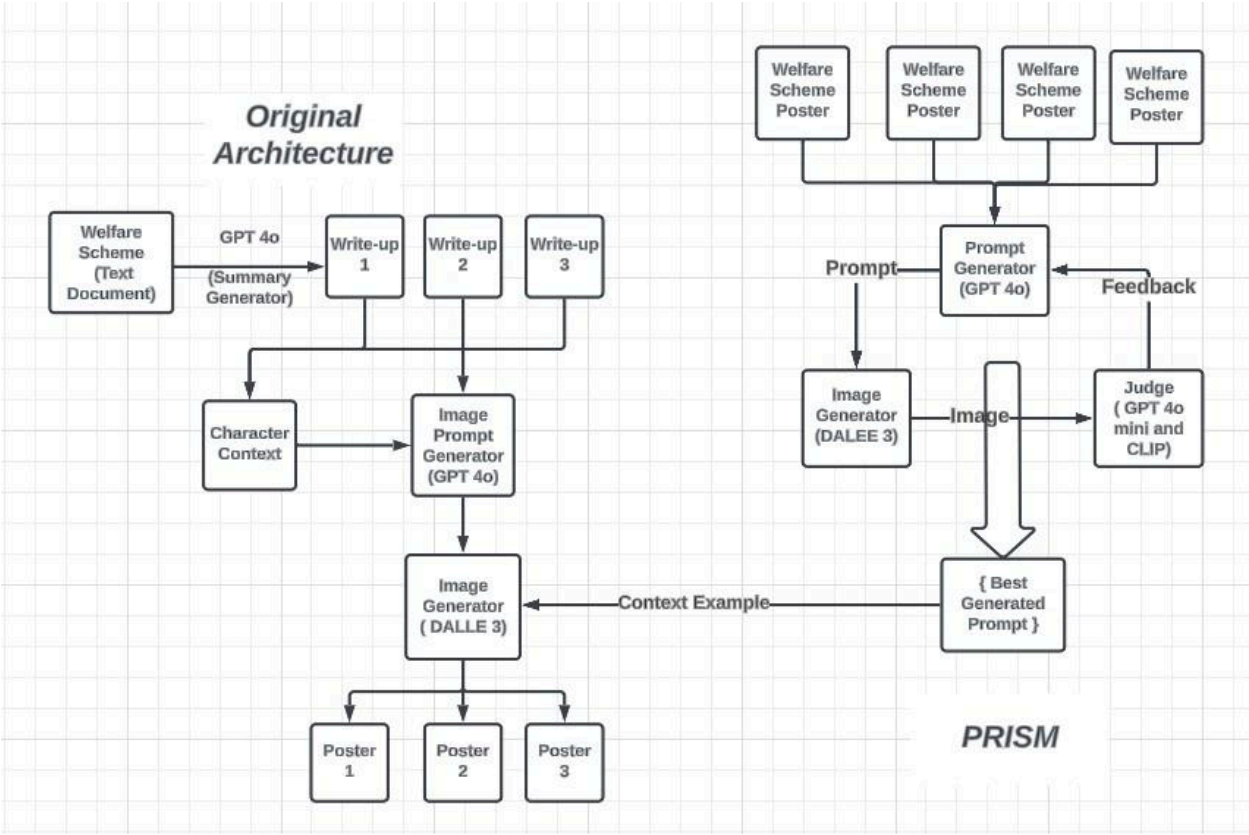
PRISM Architecture

PRISM (Prompt Refinement and Iterative Sampling Mechanism) operates through a structured, iterative process involving three key stages:

1. **Prompt Generator**: A Vision-Language Model (VLM) generates candidate prompts based on input data, such as reference images.
2. **Image Generator**: A Text-to-Image (T2I) model transforms these prompts into images, serving as visual representations of the input data.
3. **Judge**: Another VLM evaluates the similarity between the generated image and the reference images. This evaluation guides the refinement of prompts through iterative feedback, enhancing their quality and relevance over multiple iterations.

This iterative mechanism ensures that prompts evolve toward optimal results, capturing both structural and contextual nuances.

Combined Architecture



Implementation Details

- **Prompt Generator**:
  We utilized *GPT-4o* as the prompt generator. Leveraging its in-context learning capability, GPT-4o generated and refined prompts iteratively, adapting to the feedback received during each iteration.
  Generated prompt: The image features a banner for the Ayushman Bharat Pradhan Mantri Jan Arogya Yojana (PM-JAY). On the left, there's a circular logo with a green and orange design and text. Next to the logo, the program name is prominently displayed in bold pink letters. To the right, there are images of several health cards, showcasing their design and official emblems. On the far right, a collage of diverse individuals, including men, women, and children of varying ages and attire, is labeled "SPOTLIGHT." The background features a soft gradient with subtle patterns, adding a cohesive look to the banner.

What is important to understand is that by f**eeding minimal high quality posters, we are able to see improvements in performance for general welfare schemes**. This ensures that this pipeline is suitable for **low-resource tasks.**

- **Judge**:
  The evaluation component combined the reasoning capabilities of *GPT-4o* with the scoring precision of CLIP.
    - **CLIP Score**: CLIP evaluates the similarity between generated images and reference images by mapping both the iimages into a shared embedding space. This generates an objective score representing how well the generated image aligns with the reference image's concept.
    - **Integration with GPT**: The CLIP score was paired with GPT's contextual understanding and reasoning capabilities, enabling it to provide detailed feedback for refining prompts effectively. This dual-layer evaluation ensured both objective (numerical) and subjective (contextual) assessment of outputs. Feedback received: To provide feedback on the similarities and differences between these two images:
    -
    - 1. **Aspects Captured Well:**
    - - Both images focus on public information and awareness campaigns.
    - - Both use a circular central design element that creates focus.
    - - Government symbols and official seals are present, indicating authoritative sources.
    -
    - 2. **Important Elements Missed:**
    - - **Text Content:** The messages are different; the first image addresses worker schemes, while the second focuses on healthcare.
    - - **Imagery and Theme:** The first contains real photos of people, while the second uses illustrated figures, creating a stylistic mismatch.
    - - **Color Scheme:** The first image uses a pastel color palette, whereas the second is more orange and black.
    -
    - 3. **Improving the Prompt for Higher Similarity:**
    - - Include specific themes like “labor welfare” instead of general health.
    - - Specify “photographic elements” instead of “illustrations.”
    - - Use keywords for color schemes like “soft pastels” or name specific colors visible in the first image.
    - - Mention the use of real people and contextual objects or settings found in the first image.
    -

Such feedback also helps a human-in-the-loop to understand and make appropriate changes if necessary, this mitigates bias in important scenarios such as the creation of posters for welfare schemes.

- **Image Generation**:
  We employed *DALL-E* as the T2I generator, using the refined prompts from PRISM to create images that adhered to the desired style and structure.
- **Training Data**:
  A curated dataset comprising welfare scheme posters and the best examples of our previously generated posters was used to train the system. Importantly, these posters were not limited to the target welfare schemes for the final task. The intent was to enable PRISM to learn the universal structural characteristics of visually informative posters, independent of their specific content.
- **Training:** The entire PRISM loop is executed for 3 parallel times (Training 3 distinct prompt generators) on 10 posters (selected randomly) and the best prompt was selected on the basis of CLIP score.
- **Output :** After running PRISM for 3 parallel times (Training 3 distinct prompt generators) on 10 images (selected randomly), best prompt was selected on the basis of CLIP score.
  Best Generated Prompt :

```
Best Prompt: The image is a government promotional poster for the "Pradhan Mantri Matru Vandana Yojana" in Hindi. It features an illustration of a mother tenderly holding her baby with a warm, radiant background. The poster outlines financial incentives for pregnant women:

1. **Registration Incentive**: ₹1000
2. **First Check-up**: ₹2000
3. **Delivery in Government Hospital**: ₹1000
4. **Vaccination and Birth Registration of the Child**: ₹2000

The total amount offered is ₹6000. The poster is branded with the government emblem and logos related to the scheme, emphasizing maternal empowerment. Contact information and a website URL are provided for further details.
```

- While this prompt does have added context, it must be kept in mind that this was feeded as an example to the existing pipeline in an attempt to improve it's performance.

Integration into the Pipeline

The best-generated prompt from PRISM were incorporated into the existing image generation pipeline as **additional context example**. This approach offered the following benefits:

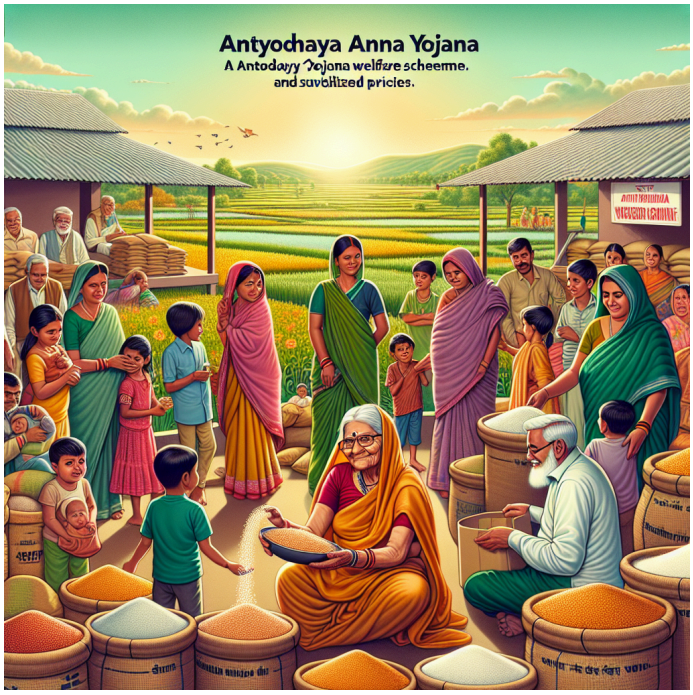
- **Preservation of Style and Structure**: The prompts ensured that the generated images retained the stylistic and structural features identified during the PRISM refinement process.
- **Avoiding Overriding Original Information**: By appending the prompts as context example, information of the structure and style were emphasized without overriding essential information about the original welfare schemes and character context.

This integration streamlined the poster creation process, reducing manual intervention while ensuring high-quality, interpretable outputs.

Output Comparison

Original Posters Vs New Posters

Old Poster without PRISM (using only the pipeline proposed in Stage 1)



New Poster after applying PRISM

The newer posters clearly demonstrate a greater emphasis on the welfare scheme itself, over the character. Moreover, very minimal text (as required to ensure that people from various backgrounds can take benefit of the poster), is added just to give an introduction to the program. This text is also well formatted in most cases compared to without the PRISM algorithm implemented.

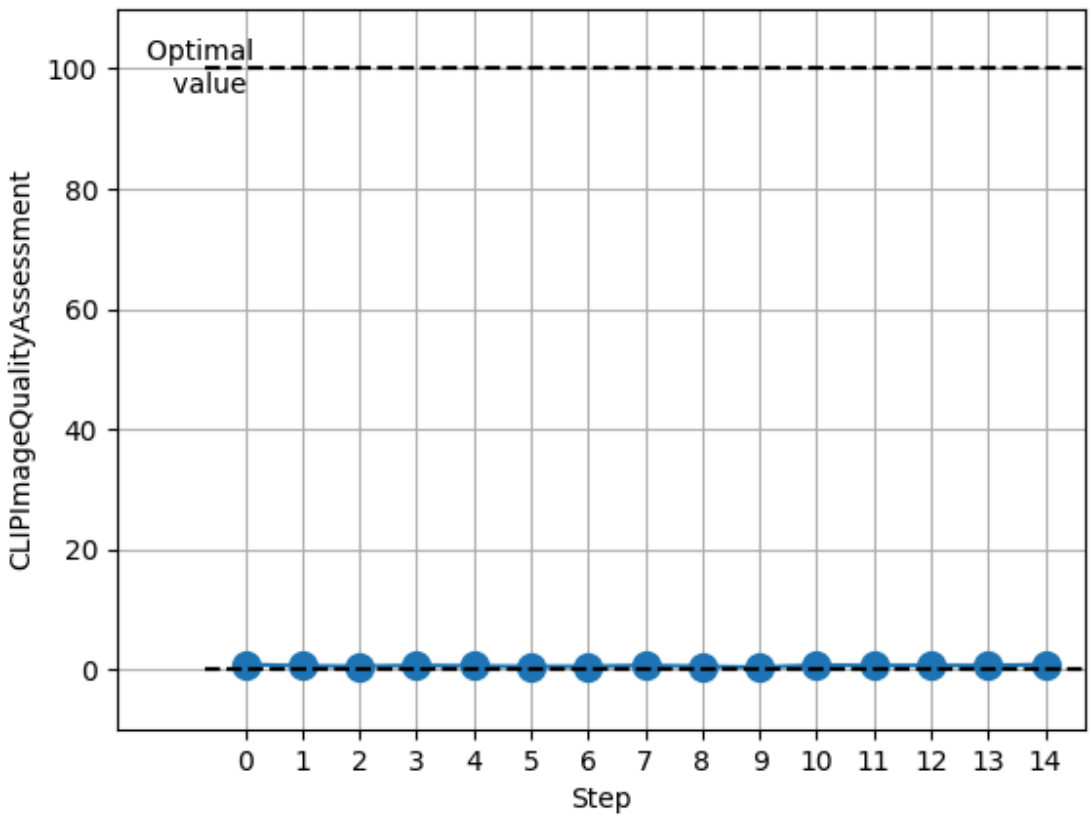
Other relevant examples to further depict the change are as follows:

(With only stage 1 pipeline)

(With PRISM algorithm implemented)

Observed Improvements

- **Improvements in CLIP Score:** We computed the **CLIP Score (Ranging from 0 to 1)** of previously generated and the newly generated posters with the write-ups, and there is a clear improvement in CLIP Scores of the newly generated posters. **Before the application of added formatted context with PRISM, we achieved an average CLIP score of 0.626. While, post application we achieved a score of 0.677.**



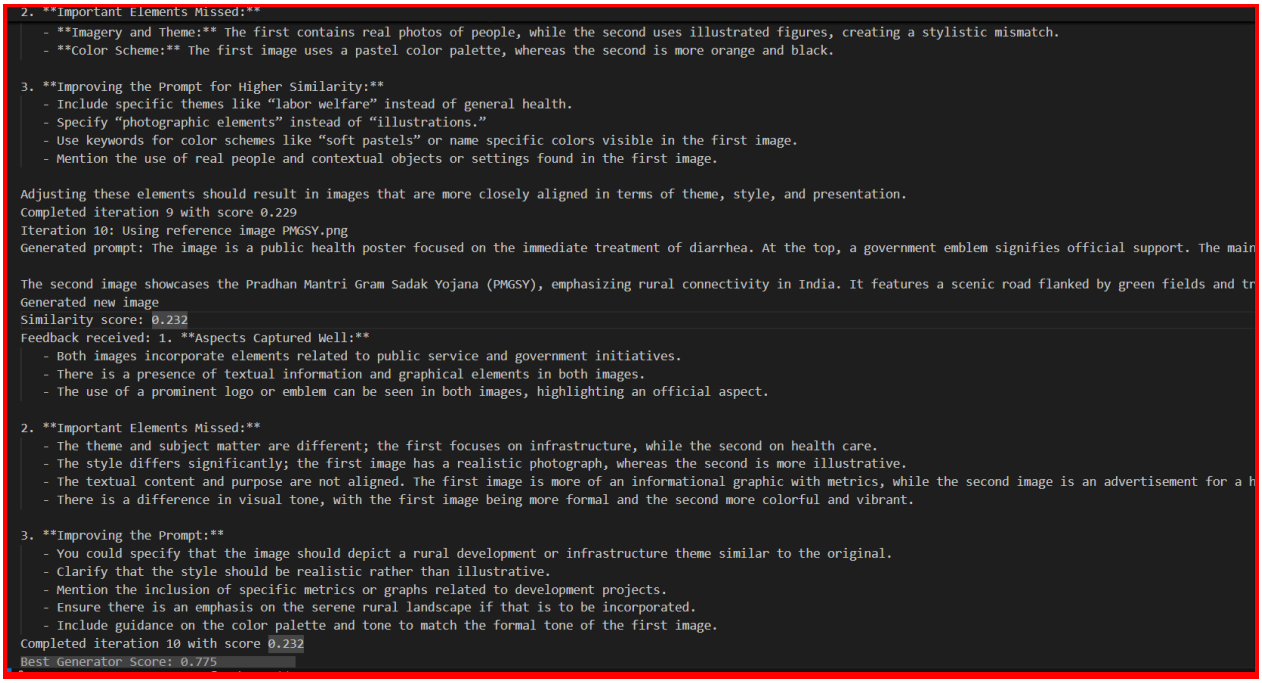
This is the final performance plotted graphically. (The scale given the library used always remains between 0 to 100, however the particular metric we used only takes values between 0 to 1).

- **Improvemet In Structure:** There is a consistent structure depicted across all the welfare schemes.
- **Improvement in conveying message:** The message is conveyed more clearly in the newly generated posters.
- **Reduction in Errors:** There are less errors related to the disfigurement of limbs, faces and objects in the newly generated posters. The texts which previously were just some random symbols have become legible. This is due to the highlighting the identified text and numbers in double quotes in the prompts. This was identified by the PRISM architecture.

Ineterpretability

While the entire architecture requires no human interaction, due to it's interpretability humans can still observe the intermediate prompts of the prompt generator and the feedback of the judge to identify bias. The model can sometimes also 'Cheat' by adding simplistic elements to boost the image to boost CLIP score. This can only be mitigated due to the interpretability of the model.

For example, during one of our training iterations we observed that all the intermediate generated images and corresponding prompts contains an emblem with 5 written on it in the middle.
Interpretability plays a vital role in mitigating the bias (if any) in AI-generated content by allowing humans to go through the entire process and make appropriate changes to the same.



By looking at the feedback generated as well as the refined image prompts at each step, the pipeline becomes interpretable. It not only allows humans to understand where toxic content is generated but also make appropriate changes if necessary.

References

1. chrome-extension://efaidnbmnnnibpcajpcglclefindmkaj/https://openreview.net/pdf?id=hIKsem01M5