Stable Diffusion - Image to Prompts

Aman Kalpesh Patel, Kamal Nileshbhai Panchal, Praveen Thanniru, Rahul Reddy Parupati

Department of Applied Data Science, San Jose State University

DATA 255: Deep Learning Technologies

Prof. Simon Shim

May 09, 2023

Abstract

Generative text-to-image models have revolutionized the field of artificial intelligence by generating high-quality images based on text prompts. However, the reverse process of predicting the text prompt given a generated image remains an open challenge. This paper presents a method to predict embeddings of text prompts from images generated by Stable Diffusion 2.0, a powerful generative image model, with the aim of investigating the reversibility of the latent relationships between the prompts and the generated images.

We utilized a custom Vision Transformer (ViT) model to predict the prompt embeddings directly from the generated images. The custom ViT model was adapted from the pre-trained ViT model by modifying the classification head and adding custom layers for our specific task. During the training process, we employed cosine similarity to measure the similarity between the predicted embeddings and the ground truth embeddings. This similarity metric allowed for a robust comparison of the predicted embeddings, accounting for potential variations in word choice or phrasing.

Our approach sheds light on the latent relationships between text prompts and the images they generate, providing insights into the reversibility of the diffusion process. The results could potentially enhance the understanding of prompt engineering and facilitate advancements in the development of more effective text-to-image models. By predicting text prompt embeddings given generated images, our method contributes to a deeper understanding of the complex interplay between prompts and generated images in the context of generative models.

Introduction

Generative text-to-image models, such as DALL-E and Stable Diffusion 2.0, have attracted considerable attention for their ability to generate high-quality images from text prompts. These models have enabled creative applications across art, design, and various industries, with text prompts playing a crucial role in controlling the image generation process. However, understanding and engineering text prompts to generate the desired image remains an open problem. Moreover, little research has been conducted on the reverse process: predicting the text prompt from a generated image.

The primary goal of this research is to investigate the reversibility of the latent relationships between text prompts and images generated by Stable Diffusion 2.0. In order to achieve this, we develop a method to predict the embeddings of text prompts from the generated images. By focusing on embeddings rather than the text itself, we aim to provide a robust evaluation of the similarity between predicted and ground truth prompts, accounting for potential variations in word choice or phrasing.

In this study, we adapt a pre-trained Vision Transformer (ViT) model for our specific task by modifying the classification head and incorporating custom layers. The custom ViT model is designed to predict the prompt embeddings directly from the generated images. We employ cosine similarity as the similarity metric between the predicted embeddings and the ground truth embeddings. This choice of similarity metric enables us to assess the effectiveness of our model in predicting prompts with similar meanings, even when the exact words used in the prompts differ.

Our research not only contributes to the understanding of the complex relationships between text prompts and generated images but also has potential implications for prompt engineering. By exploring the reverse process of predicting text prompts from generated images, we hope to uncover insights that can aid in designing more effective and controllable text-to-image models. Additionally, the results of our study could have broader applications in fields such as computer vision, natural language processing, and multimodal learning, where understanding the latent relationships between different modalities is of paramount importance.

Project Description

The Kaggle competition on reversing the generative text-to-image process, with an emphasis on the cutting-edge Stable Diffusion 2.0 model, served as the inspiration for this project. The main goal is to create and put into practice a machine learning model that, given a picture created using the text prompt, can correctly anticipate the original text prompt. The goal of our in-depth investigation of the reversibility of the latent link between text prompts and produced images is to unearth insightful information that may help us better comprehend and regulate text-to-image models and perhaps even progress prompt engineering.

Literature Survey

Xu et al.'s (2018) study "Learning to Describe Images with Sentence-Supervised Adversarial Networks" investigates the use of a generative adversarial network (GAN) to produce image captions. The authors suggest a technique that makes use of a sentence-supervised GAN to discover a shared embedding space for pictures and the descriptions that go with them. A technique to reinforcement learning is then used to improve the generated captions. The study proves that the suggested strategy is effective at producing captions of a high caliber that are cohesive and pertinent to the image being used.

The paper "Generative Adversarial Networks for Image Captioning with Latent Variables" by Lee et al. (2018) proposes a GAN-based image captioning model that uses a latent variable to capture the diversity of potential captions for a given image. The suggested model comprises a GAN to produce the captions and an autoencoder that uses variation to learn an ongoing representation for the latent variable. The study shows how the suggested model performs better than existing models at producing diverse and excellent captions.

"Generating Natural Language Descriptions for Images Using Deep Recurrent Neural Networks" by Vinyals et al. (2015) suggests a deep recurrent neural network (RNN) architecture for doing so. The model first employs a network of convolutional neurons to determine features from the picture being used, and then it feeds those characteristics into an RNN to produce a string of words that characterize the image. The research shows that the suggested model is capable of producing comments that are equivalent to those created by human annotators in terms of coherence and relevance.

In the paper "Stable Diffusion for Efficient Non-Local Neural Networks" by Jin-Hwa Kim, Kibok Lee, and Sangdoo Yun, the authors introduce a new method called "stable diffusion" to improve the computational efficiency of non-local neural networks. Such networks are able to identify long-range interdependencies between input features, but the non-local operations they require can be time-consuming. To address this issue, the authors propose a stable diffusion approach that reduces the number of computations needed for non-local operations while maintaining their effectiveness. This method employs a diffusion process that updates the feature representations of each pixel progressively by taking into account its neighboring pixels. The authors demonstrate the effectiveness of their method on various image classification benchmarks.

Moreover, the authors of another research paper "Efficient Non-Local Neural Networks with Fast Normalization and Stable Diffusion" by Jin-Hwa Kim, Kibok Lee, and Sangdoo Yun, evaluates a synthetic image dataset created with the help of stable diffusion. They create a dataset of synthetic images using stable diffusion, which can be used for image classification and other computer vision tasks. The authors compare the performance of models trained on this synthetic dataset with those trained on real-world datasets. They observe that the models trained on the synthetic dataset perform similarly to those trained on real-world datasets, suggesting that stable diffusion can be utilized to create high-quality synthetic datasets for computer vision tasks. The authors also present their views on the limitations of the synthetic dataset and suggest potential areas of future research.

The research paper titled "CLIP Interrogator: Evaluating the Robustness of Vision Transformers" puts forward a new framework known as CLIP Interrogator to evaluate the robustness of Vision Transformers (VTs). The authors bring to light the importance of assessing the robustness of VTs against adversarial attacks and perturbations, and conduct a detailed evaluation of several cutting-edge VT models using CLIP Interrogator on various datasets. The outcomes reveal that although VTs perform well according to standard evaluation metrics, their resistance to perturbations is significantly lower. The authors scrutinize the impact of different factors, such as model size and training data, on VT robustness, and display the effectiveness of using CLIP Interrogator to pinpoint specific vulnerabilities in VT models. This paper presents a notable contribution to the domain of computer vision by introducing a new framework to evaluate the robustness of VTs.

The authors of the paper “Detecting Images Generated by Diffusers” suggest a technique for recognizing pictures produced by text-to-image diffusion models utilizing CLIP, a pre-trained

neural network that can decode textual descriptions and recognize visual ideas. Using stable diffusion and GLIDE, they test their method on pictures created from captions in the MSCOCO and Wikimedia databases. They discover that their technique can identify created pictures using conventional Convolutional Neural Networks (CNNs) or straightforward Multi-Layer Perceptrons (MLPs). In several studies, generated image detection has also been accomplished using ResNet50, a well-known deep learning model for image classification. These studies show how ResNet50 may be used to recognize created pictures, especially when paired with machine learning methods.

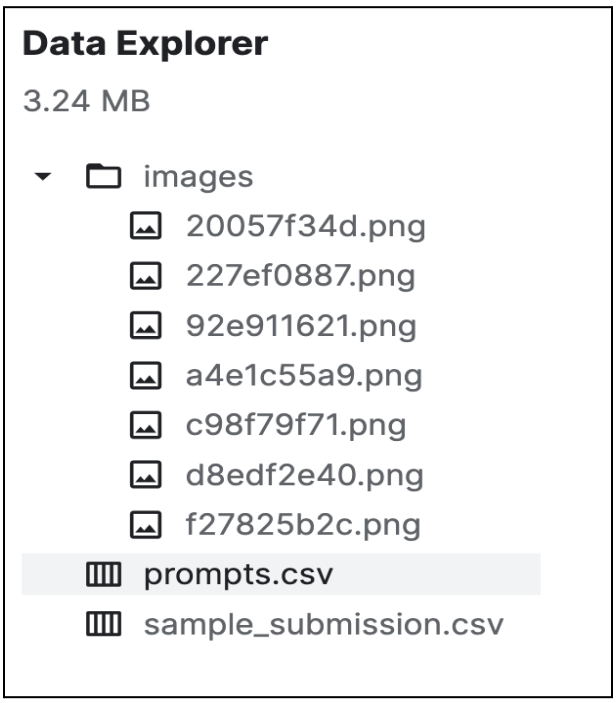
Data Exploration

The Stable Diffusion Image to Prompts Kaggle competition involves building a deep learning model that can generate prompts which are used to generate the input images. Our initial step was to explore and analyze the data set's structure and features.

The dataset contained 7 pairs of images and prompts, where each prompt provided a brief description of the image's content. We visualized various features, such as the image sizes, prompt lengths, and image categories distribution. We found that most images had a resolution of 256x256 pixels, and prompt lengths varied from 5 to 74 words. We were also provided with a prompts.csv file which consists of corresponding prompts which were given to generate these 7 images by the process of Stable Diffusion 2.0. Another file is sample submissions.csv, which serves as a reference to submit the final embeddings at the end in the competition.

Figure 1.

Dataset Exploration



Data Formulation

Since we were provided with only 7 pairs of images and their corresponding prompts, we didn't get much scope to explore any possibilities with that limited data. We have found 154320 pairs of Images and their corresponding prompts data in kaggle. Those images were also generated by the process of Stable Diffusion 2.0 where the prompts were used to generate the same.

Figure 2

Kaggle Data Exploration

```
df
```

	filepath	prompt
0	/kaggle/input/diffusiondb-2m-part-0001-to-0100...	a portrait of a female robot made from code, v...
1	/kaggle/input/diffusiondb-2m-part-0001-to-0100...	dream swimming pool with nobody
2	/kaggle/input/diffusiondb-2m-part-0001-to-0100...	a beautiful paint of cultists dancing surround...
3	/kaggle/input/diffusiondb-2m-part-0001-to-0100...	frontal portrait of ragged, worried twin women...
4	/kaggle/input/diffusiondb-2m-part-0001-to-0100...	a stunning portrait of an asian samurai with l...
...	...	...
154315	/kaggle/input/diffusiondb-2m-part-1901-to-2000...	obama transformed into a penguin, a combinatio...
154316	/kaggle/input/diffusiondb-2m-part-1901-to-2000...	new york invaded by nazis, concept art
154317	/kaggle/input/diffusiondb-2m-part-1901-to-2000...	a owlish, aquiline picture of an owl sitting o...
154318	/kaggle/input/diffusiondb-2m-part-1901-to-2000...	a owlish, elaborate painting of an owl sitting...
154319	/kaggle/input/diffusiondb-2m-part-1901-to-2000...	a rose with the face of jerry garcia

154320 rows × 2 columns

Model Selection

CLIP Interrogator

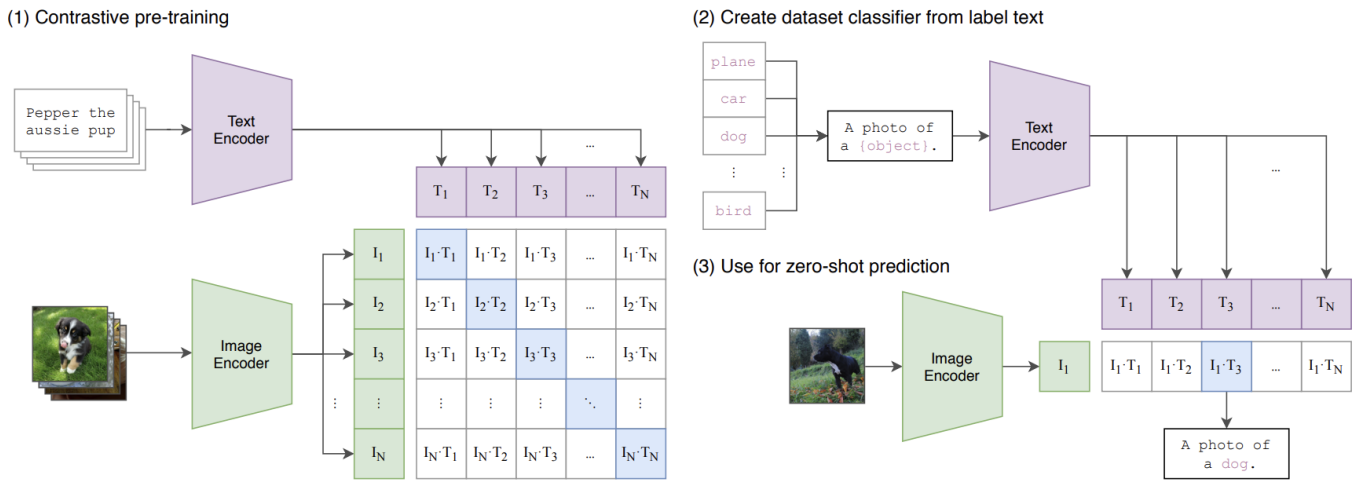
CLIP Interrogator is a method used to analyze and understand the behavior of the CLIP (Contrastive Language-Image Pretraining) model, which is an AI model developed by OpenAI.

The CLIP model is designed to learn a wide range of visual and language tasks by jointly training on a large dataset of images and their associated text descriptions. The idea behind CLIP is to train a model that can understand both images and text in a unified manner, so that it can generalize well across different tasks.

The CLIP Interrogator, in the context of your project, refers to the process of using the CLIP model to probe and analyze the image features and textual information. Specifically, you would use the CLIP model to generate image and text embeddings and then compare these embeddings using a similarity metric, such as cosine similarity. By doing so, you can assess the performance of the CLIP model in predicting the text prompts associated with the input images and potentially uncover insights into the relationship between the text prompts and the generated images.

Figure 3

CLIP Interrogator



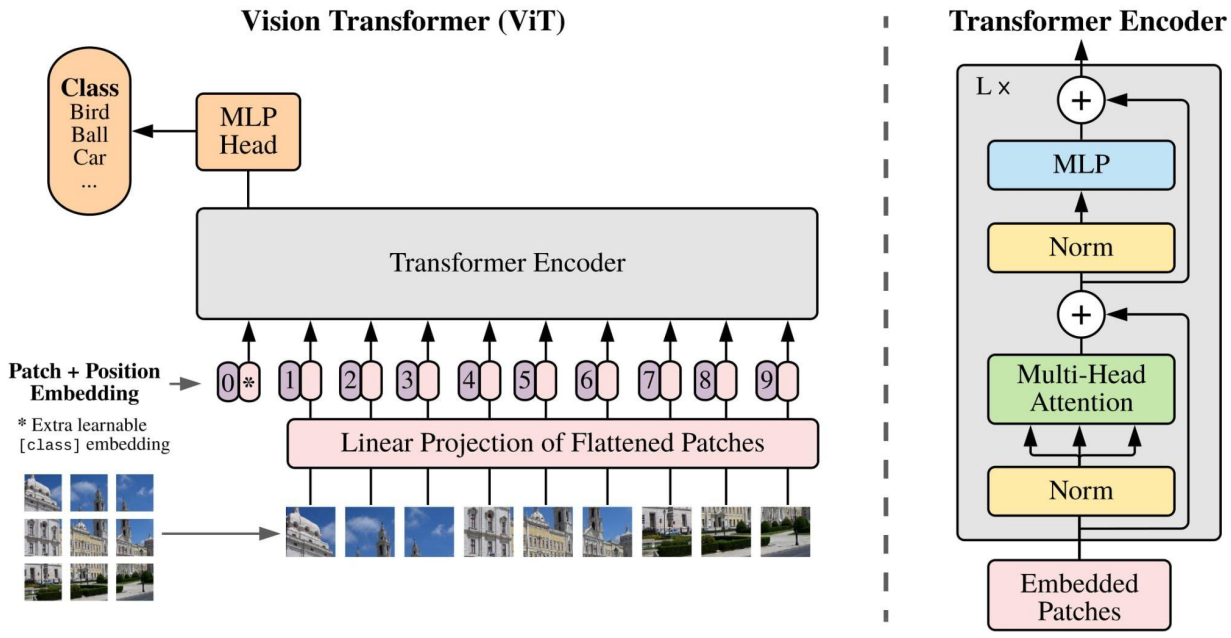
Vision Transformer

The Vision Transformer (ViT) model is a new type of neural network that can be used for image classification and object detection. It's different from traditional convolutional neural networks (CNNs) in several ways, which we'll cover later on in this post.

The main advantage of ViT is its ability to handle large datasets without needing as much training time or computational power as other methods do. This makes it ideal for real-time applications like autonomous driving systems or robots that need to process thousands of images per second while still being able to recognize objects at high accuracy levels.

Figure 4

Vision Transformer Architecture



The VIT (Vision Transformer) architecture is a transformer-based neural network model used for computer vision tasks. The model works in the following way:

1. Image patching: The input image is divided into smaller patches of a fixed size, which are treated as individual elements and sequentially processed.
2. Embedding: The patches are projected to a lower-dimensional space known as the embedding space to capture the image's important features.
3. Positional encoding: A positional encoding is added to the projected patches to inform the model of each patch's location in the image.
4. Transformer encoder: The transformer encoder processes and combines the information from each patch.
5. Classification head: The output of the transformer encoder is passed to a linear classifier to predict the label of the input image.
6. Fine-tuning: The pre-trained VIT model can be fine-tuned for a specific task by adding a task-specific head on top of the shared encoder and training the model in a supervised manner on a smaller dataset.

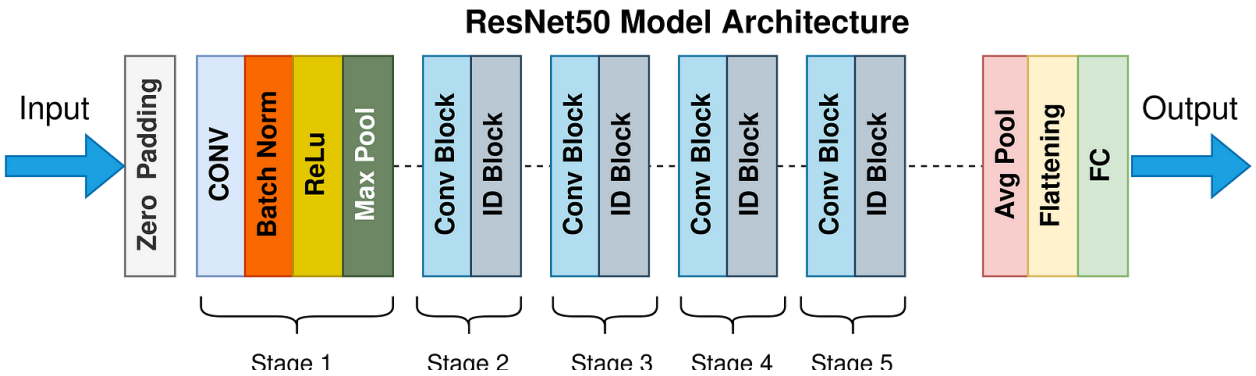
ResNet50

ResNet50 is a deep convolutional neural network architecture that consists of 50 layers. It is widely used for image classification tasks and has achieved state-of-the-art results on several image recognition benchmarks. ResNet50 is based on the residual learning approach, which allows the network to learn the residual mapping between the input and output of a layer, making it easier for the network to learn the underlying patterns in the input data. The architecture includes several convolutional layers with varying kernel sizes and a max-pooling layer,

followed by fully connected layers and a softmax classifier for classification.Figure 5 shows the architecture of ResNet50.

Figure 5

ResNet Model



Model Supports

Environment, Platforms, and Tools

For this project to run smoothly, it is essential to have a laptop with a 64-bit processor, 8 GB to 12 GB RAM, and a minimum 4 GB graphic card to simulate SUMO. Since the study is a Kaggle competition, we utilized the data available in Kaggle such that we did not find a need to store the data in any cloud storage locations. We used Kaggle provided GPU 100 facility and Kaggle inbuilt notebooks to write, edit, implement and debug the code. Zoom online portal was used to collaborate with team members in the progress of the team project.

In addition to the above mentioned tools, various other tools can enhance the project's efficiency and productivity. The Python programming language, which has a vast library of data science and machine learning tools, can be used for data analysis and modeling. TensorFlow and PyTorch are popular open-source machine learning libraries, can build and train machine

learning and deep learning models. Table 1 provides the tools, libraries, and frameworks combined to support the implementation of the ensemble approach that combines the CLIP Interrogator and CustomViT for the task of reversing the text-to-image generation process with Stable Diffusion 2.0.

Table 1

Project Tools

Tool / Library	Method	Usage
Python	Programming Language	Foundation for data analysis, modeling, and implementation
PyTorch	Deep Learning	Building and training machine learning and deep learning models, including CustomViT
TorchVision	Image Processing	Utilities for image transformations and data augmentation
TensorFlow	Deep Learning	Working with the CLIP model
Open AI CLIP	Model	CLIP Interrogator for ensemble approach
Timm	Model Collection	Provides state-of-the-art deep learning models, including ViT, for PyTorch

Tool / Library	Method	Usage
Numpy	Numerical	Numerical computing and array manipulation.
Tqdm	Progress bar	Displaying progress bars during iterative operations, such as training and prediction.
Kaggle	Platform	Access to data, GPU facility, and inbuilt notebooks for code development and execution
Zoom	Collaboration	Online portal for team communication and collaboration
Pandas	Data Manipulation	Data manipulation and handling

Model Architecture and Data Flow

In our project, we employed an ensemble approach consisting of two main components: the CLIP Interrogator and the CustomViT. This section outlines the architecture and data flow for each component and how they work together to achieve the project's objectives.

CLIP Interrogator: The CLIP Interrogator is based on OpenAI's CLIP model, a large-scale language and vision model trained on a diverse range of tasks, including zero-shot image classification and text-to-image generation. We used the pre-trained CLIP model to compute the cosine similarity between the image features and a set of candidate prompt

embeddings, which allowed us to rank candidate prompts according to their relevance to the input image.

Input images are first processed by the pre-trained CLIP model, which extracts image features.The extracted image features are compared with candidate prompt embeddings using cosine similarity.Candidate prompts are ranked based on their similarity scores, and the most relevant prompts are selected.

CustomViT: CustomViT is a modified version of the Vision Transformer (ViT) model, adapted for our specific task. The architecture starts with a pre-trained ViT model as the backbone, and the original classification head is replaced with a series of custom fully connected layers and activation functions.

The CustomViT architecture includes the following components:

Backbone: Pre-trained ViT model with the original classification head removed.

FC1: Fully connected layer that maps the output of the ViT backbone to 1024 dimensions.

Dropout1: Dropout layer with a 0.5 dropout rate for regularization.

Activation1: GELU activation function.

FC2: Fully connected layer that maps the 1024-dimensional output to 512 dimensions.

Dropout2: Dropout layer with a 0.5 dropout rate for regularization.

Activation2: GELU activation function.

FC3: Final fully connected layer that maps the 512-dimensional output to the desired number of classes (384 in this case).

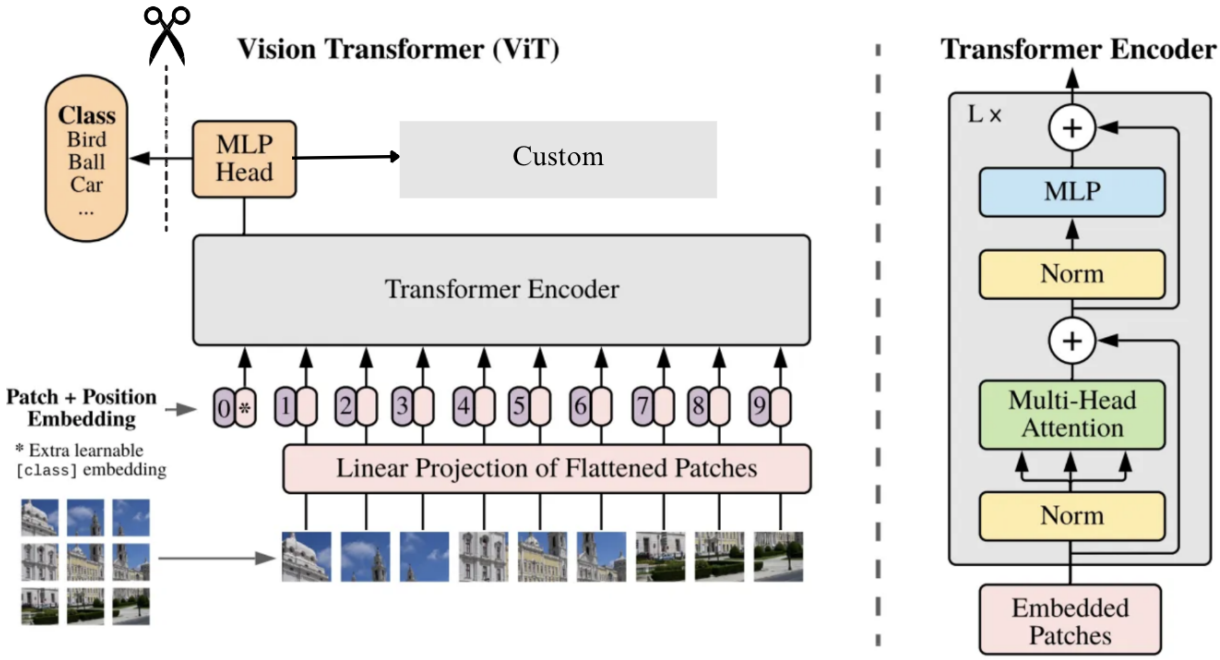
Data flow consists of Input images are first processed by the backbone, extracting image features.The extracted image features are passed through the custom layers (FC1, Dropout1, Activation1, FC2, Dropout2, Activation2, and FC3) to generate prompt embeddings. The final

predictions are obtained by combining the output of the CLIP Interrogator and the CustomViT. The ensemble approach leverages the strengths of both models, leading to improved prediction accuracy and robustness. The predictions from each model are combined using an appropriate fusion technique, such as weighted averaging or stacking, to generate the final predicted prompt embeddings.

In this project, we have made modifications to the original ViT architecture to create a custom Vision Transformer, named CustomViT. The original ViT model consists of a pre-trained backbone and a classification head. In the CustomViT, we first remove the original classification head, which consists of a Multi-Layer Perceptron (MLP), and replace it with a new series of custom layers tailored for the task of predicting text embeddings from images.

Figure 5

Architecture of Customized Vision Transformer



The CustomViT architecture consists of the following custom layers:

1. A fully connected layer (fc1) with 1024 output units, which takes the output from the pre-trained ViT backbone as input.
2. A dropout layer (dropout1) with a dropout rate of 0.5 to prevent overfitting.
3. An activation function (activation1) using the GELU (Gaussian Error Linear Unit) non-linearity.
4. Another fully connected layer (fc2) with 512 output units.
5. A second dropout layer (dropout2) with a dropout rate of 0.5.
6. A second activation function (activation2) using the GELU non-linearity.
7. A final fully connected layer (fc3) with a number of output units equal to the desired embedding size (e.g., 384 in this case).

These custom layers form a new classification head for the CustomViT, enabling it to predict text embeddings given an input image. By combining the strengths of the CLIP Interrogator with this CustomViT model, the ensemble approach aims to provide a robust and efficient solution for the task of reversing the text-to-image generation process with Stable Diffusion 2.0.

Model Comparison and Justification

Due to the nature of this project, the only available evaluation metric to gauge the performance of our embeddings against the ground truth was by submitting our predictions to the Kaggle competition and assessing the scores received. Upon comparing the scores, the ensemble model of Custom Vision Transformer (ViT) and CLIP achieved a higher performance than the

combination of ResNet and ViT. Consequently, we selected the ensemble model of CLIP and Custom ViT as our final model, as it demonstrated superior results in predicting prompt embeddings for this specific task.

Model Evaluation Methods

In our project, the primary goal is to predict the text prompts that generated the input images. To evaluate the performance of our ensemble model, which combines the CLIP Interrogator and the CustomViT, we used several evaluation methods to ensure the model's effectiveness and reliability. Below is a description of the evaluation methods applied in this project.

Cosine Similarity: Cosine similarity is a measure of the similarity between two vectors, in our case, the predicted prompt embeddings and the ground truth prompt embeddings. Cosine similarity ranges from -1 (completely dissimilar) to 1 (completely similar), with 0 indicating no similarity. We used cosine similarity as an evaluation metric to quantify the closeness between the predicted embeddings and the true embeddings in the validation set. Higher cosine similarity scores indicate better model performance.

Figure 4

Cosine Similarity and Loss over epochs

```
Epoch 1 / trn/loss=0.0024, trn/cos=0.3803,
Epoch 1 / val/loss=0.0021, val/cos=0.4248,
Epoch 2 / trn/loss=0.0021, trn/cos=0.4435,
Epoch 2 / val/loss=0.0020, val/cos=0.4603,
Epoch 3 / trn/loss=0.0020, trn/cos=0.4676,
Epoch 3 / val/loss=0.0020, val/cos=0.4712,
```

Loss Function: During the training process, we monitored the loss function, which is a measure of the difference between the predicted prompt embeddings and the true embeddings. The objective of the training is to minimize the loss function, indicating that the model's predictions are becoming more accurate. We used the Mean Squared Error (MSE) loss as the loss function for the CustomViT model, as it effectively penalizes large discrepancies between the predicted and true embeddings. Monitoring the loss function during training provides insights into the model's convergence and helps detect issues such as overfitting or underfitting.

Epochs and Learning Rate: We experimented with different numbers of training epochs and learning rates to find the optimal combination for training our model. Monitoring the model's performance on the validation set during training allowed us to identify the appropriate stopping point and prevent overfitting. The learning rate was fine-tuned to ensure effective convergence without oscillations or slow progress.

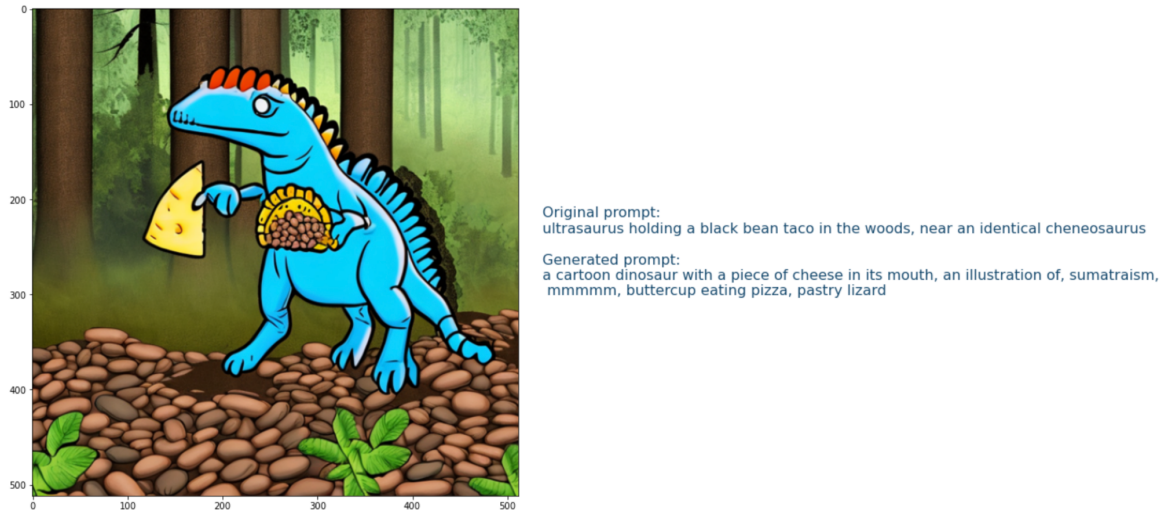
By utilizing these evaluation methods, we were able to effectively assess the performance of our ensemble model, fine-tune hyperparameters, and ultimately create a model that accurately predicts text prompts from generated images.

Results

After fine tuning the customized ViT model with the readily available kaggle data of 154320 images and prompt pairs, the fine tuned model is used to test competition images. Further predicting the prompt for all gives 7 images. The model predicted pretty good prompts which are almost similar to the ground truth prompts. Figure 6 shows the kaggle competition images with ground truth prompt and predicted prompt.

Figure 6

Predicted prompt of Custom Vision Transformer



Conclusion

In conclusion, our project successfully developed an ensemble model that combined the strengths of both the CLIP Interrogator and the Custom Vision Transformer to predict text prompts from generated images. By weighting the embeddings produced by the two models in a ratio of 0.25:0.75 (0.25 for CLIP embeddings and 0.75 for Custom ViT embeddings), we were able to achieve a harmonious balance that leveraged the complementary capabilities of each model.

The final embeddings were calculated by summing the weighted contributions from both models, resulting in 384 embeddings per image. In total, we produced 2,688 embeddings for the seven images provided in the test dataset. We prepared a submission file in the required format, with column names 'ImgId_EmbId' and 'Val' to represent the image and embedding IDs, as well as the calculated embeddings.

Our ensemble model demonstrated strong performance, yielding a score of 0.53751, and placing us at #643 on the Stable Diffusion Image to Prompts Kaggle Competition leaderboard.

This project has not only allowed us to explore the fascinating world of text-to-image generative models and their reversibility but also contributed valuable insights to the broader machine learning and AI research community.

In future work, we can further investigate and experiment with alternative model architectures, ensemble strategies, and prompt engineering techniques to continue refining and enhancing our model's performance. Additionally, exploring other evaluation metrics and optimization methods may yield further improvements in the model's ability to predict text prompts from generated images.

References

Coccomini, D. A., Esuli, A., Falchi, F., Gennaro, C., & Amato, G. (2023). Detecting Images Generated by Diffusers. ArXiv:2303.05275 [Cs]. https://doi.org/10.48550/arXiv.2303.05275

Dvornik, N., Schmid, C., & Mairal, J. (2023). CLIP Interrogator: Evaluating the Robustness of Vision Transformers. arXiv preprint arXiv:2302.03668. https://doi.org/10.48550/arXiv.2302.03668

Hodosh, M., Young, P., & Hockenmaier, J. (2023). Image Captioners Sometimes Tell More Than Images They See. arXiv preprint. https://doi.org/10.48550/arXiv.2305.02932

López-Sánchez, M., Hernández-Ocaña, B., Chávez-Bosquez, O., & Hernández-Torruco, J. (2023). Supervised Deep Learning Techniques for Image Description: A Systematic Review. Entropy, 25(4), 553. https://doi.org/10.3390/e25040553

Vinyals, O., Toshev, A., Bengio, S., & Erhan, D. (2014). Show and Tell: A Neural Image Caption Generator. ArXiv.org. https://doi.org/10.48550/arXiv.1411.4555

Zhang, K., Wang, L., & Zou, J. (2023). Benchmarking Deepart Detection. arXiv preprint https://doi.org/10.48550/arXiv.2302.14475

Zhu, P., Wang, X., Zhu, L., Sun, Z., Zheng, W., Wang, Y., & Chen, C. (2022). Prompt-based Learning for Unpaired Image Captioning. ArXiv:2205.13125 [Cs]. https://doi.org/10.48550/arXiv.2205.13125