CONDITIONAL VAE FOR TEXT TO IMAGE GENERATION USING SHORT-TEXT AND LONG-TEXT SYNTHESIS

Anonymous authors
Paper under double-blind review

ABSTRACT

We present models to perform text to image generation and image reconstruction using a conditional VAE (CVAE) trained on Fashion-MNIST for short-text and COCO for long-text with a CLIP feature extractor. In total, three main experiments were run. The first experiment involved short-text model architecture investigation between fully connected and a convolutional CVAE, where the convolutional CVAE was found to have a lower MSE and higher SSIM and FID. The second experiment tested long-text image generation and reconstruction on images of size 64x64, 128x128, and 256x256. The third experiment tested concatenating versus adding the conditional input in the CVAE from the COCO annotation using the features extracted by CLIP. Both conditioning methods resulted in similar results for sizes of 64x64 and 128x128, but the concatenation conditioning significantly outperformed the adding conditioning with 256x256 images.

1 INTRODUCTION

Generative modeling is a field within machine learning focused on learning the underlying distributions responsible for creating data. Understanding these distributions enables models to generalize across various datasets, facilitating knowledge transfer and effectively addressing issues of data sparsity. The applications of generative models include image denoising, inpainting, super-resolution, text-to-image synthesis, and more (Raut & Singh, 2024).

The history of generative modeling in the context of deep learning is limited to recent decades. Some examples of early generative models with deep learning architectures include Restricted Boltzmann Machines (Ackley et al., 1985) and Deep Belief Networks (Hinton et al., 2006). Both models were influential generative deep learning models, but had many limitations that prevented them from becoming more popular today. Historically, the gap between generative and discriminative models has been quite large, with significantly more success in the development of discriminative models. This gap has narrowed in recent years, with one of the most impactful advancements being the development of Variational Autoencoders (VAEs) just over a decade ago (Kingma & Welling, 2013).

The focus of this paper is on a specific type of VAE, called a Conditional Variational Autoencoder (CVAE). VAEs are unsupervised models that encode input data into a latent representation and reconstruct the input from this latent space, focusing on capturing data patterns. CVAEs extend VAEs by incorporating additional information such as class labels as conditional variables. This conditioning enables CVAEs to produce controlled generations. This project incorporates text as the conditional variable and can be broken down into two types of text input: short text and long text.

Short text conditional variables are represented as class labels to help the CVAE produce class-specific image generations. Long text conditional variables require the CVAE to have a more complex representation of the text, which is what the Contrastive Language-Image Pre-training (CLIP) neural network is used for in this paper. CLIP trains on pairs of images and captions, teaching the model to recognize which ones belong together (Radford et al., 2021). This allows CLIP to classify images into categories described in natural language, even without specific training on those categories.

2 Related Work

In their seminal paper, Kingma and Welling proposed a groundbreaking framework for VAEs (Kingma & Welling, 2013), which propelled further research into VAE models. They contributed two significant findings towards VAE research. First, the authors demonstrate that by reparameterizing the variational lower bound, it becomes possible to optimize the bound using standard stochastic gradient descent. Second, the authors demonstrate that by fitting an approximate inference model to the intractable posterior distribution, the paper significantly improves the efficiency of inference in generative models. These techniques are integral to our own work, where we leverage them to efficiently train Conditional VAEs (CVAEs).

After the introduction of VAEs, many variations have been developed and tested. One such variation is a hybrid model that integrates convolutional layers in the encoder and deconvolutional layers in the decoder with the vanilla VAE (Semeniuta et al., 2017). The addition of convolutional and deconvolutional layers resulted in improved computational efficiency, which will be helpful when dealing with the large scale of the COCO dataset for this project.

3 Methods

The aim of the project is to create an image generator using a CVAE for short text descriptions using Fashion-MNIST dataset and then for longer text descriptions using COCO dataset. Two different CVAE architectures were used to accomplish the short-text to image generation objective: a fully connected architecture and a convolutional-based architecture. CLIP along with several convolutional architectures with different input dimensions, latent space dimensions, and conditioning methods was used for the long text to image generation.

The loss function of CVAE is defined in Eq. 1 (William et al., 2022). The first term of the loss in this loss function is the reconstruction error. This is the expected negative log likelihood of the given data point. With a large error, the decoder does not output an image similar to the input image. The second term in the loss function is Kullback-Leibler divergence between the probability distribution $q_\phi(z|x,c)$ and $p(z,c)$. ϕ is the variational parameter.

$$-E_{z\sim q(Z|X)}[\log p(X|Z,C)] + KL(q_\phi(Z|X,C)||p(Z,C)) \tag{1}$$

3.1 Short-Text Preprocessing on Fashion-MNIST

Before passing information into a CVAE, preprocessing needed to be completed to pass in an image from Fashion-MNIST along with its class label. Fashion-MNIST is an MNIST-like dataset consisting of a training set of 60,000 examples and a test set of 10,000 examples. Each example is a 28x28 grayscale image, associated with a label from 10 classes (Xiao et al., 2017).

Preprocessing functions were created to extract the relevant key word containing the class name from the input short-text. Extra descriptors (synonyms) were used for most classes to account for similar fashion items included in each class (e.g. Coat & Jacket). The class name was then converted to its class number and used as the conditional input to the CVAE.

3.2 Short-Text Label Model with Fully Connected Layers

The first architecture created for short-text to image generation was a fully connected CVAE trained on the Fashion-MNIST dataset. The CVAE encoder consists of three linear layers. The first two layers contain a ReLU non-linearity. After the first two layers, the class number is embedded and concatenated along the feature dimension to the input. Thus, we reach the latent space dimension after applying the third layer, which is two-part for obtaining μ and σ. The reparameterization trick is then used with μ and σ to obtain the hidden variables z in the latent space. Once sampled, the output of the reparameterized latent space is once again concatenated with the embedded class number. This is passed to the decoder consisting of 3 linear layers. The first 2 layers have once again ReLU non-linearity while the final layer has a sigmoid non-linearity. The output of the decoder is a 784 dimensional flattened vector that can be reconstructed into a 28x28 image resembling a class from the Fashion-MNIST dataset.

In order to generate an image, the class label extracted from the short-test description through preprocessing is passed into the decoder of the model along with random samples from a Gaussian distribution to input the variables from the latent space. Before running the final model, hyperparameter tuning was performed for the weight decay values for the ADAM optimizer. The final model was trained for 25 epochs with a learning rate of $1e^{-3}$ and a weight decay of $1e^{-3}$.

3.3 Short-Text Label Model with Convolutional CVAEs

The second architecture created for short-text to image generation was a convolutional CVAE trained on the Fashion-MNIST dataset. The CVAE encoder consists of 3 convolutional layers each followed by a ReLU non-linearity. The output of the encoder is then flattened. The class number is then passed through an embedding layer and added to the encoder output. The reparameterization trick is then used with 2 linear layers to obtain a μ and σ in the latent space. Once sampled, the output of the reparameterized latent space is passed to the decoder now concatenated with the class number embedding layer output. The decoder consists of 3 transposed convolutional layers. The first two contain a ReLU non-linearity with the last layer containing a sigmoid non-linearity. The output of the decoder is a 28x28 generated image resembling a class from the Fashion-MNIST dataset. For generating images, the process is analogous to the fully connected CVAE. The model was also trained for 25 epochs with a learning rate of $1e-3$ to allow a comparison with respect to performance between the convolutional and the fully connected CVAE.

3.4 CLIP and Preprocessing COCO

For encoding image captions on the COCO dataset, we utilized OpenAI's CLIP embedding model in its ViT-B/32 configuration, which outputs embeddings of length 512. A limitation of the CLIP model is that it has a maximum token length of 77, with studies showing an even smaller effective length of 20 (Zhang et al., 2024). Thus, in instances where the input text contains multiple sentences, the text is split up by sentence and passed through the CLIP encoder. The resulting embeddings are averaged together to create the final output embedding. The CLIP embedding is passed into the decoder of the model along with random samples from a Gaussian distribution as the input to the latent space.

When loading the COCO image dataset, images were normalized and downsampled to three sizes: 64x64, 128x128, and 256x256. Due to computing constraints and physical storage limitations, COCO was subsampled down to roughly 20 percent of its normal size when training our models, which still results in several thousand images for both the training and validation set. Due to computational limitations of storage, it was not possible to load more than 40 percent of COCO2017 on a machine at once to train without crashing.

3.5 Architecture and Different Conditioning Approaches on COCO

The architecture from the convolutional CVAE from Fashion-MNIST, with changes to the conditional input embedding model, was further scaled for use on the larger COCO dataset. The revised COCO architecture for the 128x128 and 256x256 images contains four convolutional layers in the encoder, each followed by a batch normalization layer and ReLU non-linearity. The 64x64 model only has 3 convolutional layers, also followed by a batch normalization layer and ReLU non-linearity. In the bottleneck, the convolutional input of the encoder is flattened, and the reparameterization trick is then used with 2 linear layers to obtain μ and σ for sampling of the latent space. Once sampled, the output of the reparameterized latent space is passed to the decoder, either concatenated or added with the output of the CLIP embedding layer depending on the conditioning method. The decoder contains either 3, 4, or 5 convolutional layers for the 64x64, 128x128, and 256x256 images respectively, with all but the last layer followed by a batch normalization layer and ReLU non-linearity. The final convolutional layer is only followed by a sigmoid non-linearity. The decoder output is a 3 channel color image of size 64x64, 128x128, or 256x256. The 64x64 model uses the COCO 2014 dataset, whereas the larger models use COCO 2017. These two datasets have the same images and number of classes, with the only difference being how the data is divided into train and test sets.

We tested two methods of conditoning in our COCO models. After being processed by CLIP and averaged if necessary, conditional inputs were integrated into the latent space either by adding them elementwise with the latent space, using a linear layer to resize them, or by concatenating them with the latent space before input to the decoder. The differences in these two methods can be observed in Figure 1. A major goal of our experimentation was to assess the effectiveness of these two conditioning methods.

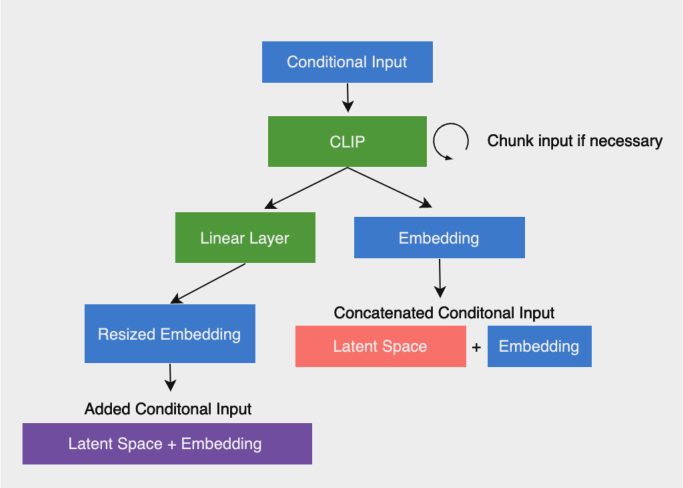


Figure 1: Pipeline for adding and concatenating conditional input

4 Experiments

In total, three large-scale experiments were run. The first experiment tested which model architecture would achieve better metrics on Fashion-MNIST: a convolutional or fully connected CVAE. The second experiment tested which image size would produce better metrics for generation and reconstruction for the long-text COCO model. This was tested for images of size 64x64, 128x128, and 256x256. The final experiment was to test the performance of adding vs. concatenating the conditional input in the CVAE. Every combination of conditioning method and image size was tested and compared using various quantitative metrics and qualitative observation.

4.1 Evaluation metrics

In order to assess the model generation, 3 quantitative metrics were used along with a visual qualitative test. Mean Squared Error (MSE) was calculated by summing the squares of the difference between the generated image and a ground truth image pixel-wise. Structural Similarity Index Measure (SSIM) is a metric that evaluates image quality by comparing two images based on structural information, luminance, and contrast (Nilsson & Akenine-Möller, 2020). SSIM can be used to compare images of any size while MSE is relative to pixel size. SSIM score ranges from -1 to 1, where 1 indicates identical images. Fréchet inception distance (FID) is a metric for quantifying the realism and diversity of images generated. As FID is a distance measure, lower scores are indicative of a better reconstruction of a set of images.

4.2 Quantitative Results from Short Text on Fashion MNIST

The quantitative metrics discussed are shown in Table 1 for models on Fashion-MNIST from the first experiment for the whole test set. The convolutional CVAE had a lower MSE and a higher SSIM than the fully connected CVAE, indicating better performance in image generation from the convolutional CVAE. FID is lower for the convolutional CVAE indicating more realistic images generated. Reconstructions and generated images are shown in Figures 2 and 3 for the convolutional CVAE as it had better metrics.

Table 1: CVAE Model performance comparison for FashionMNIST

Model	MSE	SSIM	FID
Fully Connected VAE	12.1598	0.6398	16.9910
Convolutional VAE	11.1087	0.7627	50.3034

4.3 Qualitative Results from Short Text on Fashion MNIST

Image reconstructions were provided from the validation dataset in Figure 2. These reconstructions as well as the 4 generations shown for 'dress' input in Figure 3 are all from the convolutional CVAE since it has better performance metrics. All reconstructed images can be qualitatively identified as belonging to one class. Generated images appear to be in the dress class and are of size 28x28, but appear blurry when viewed at a higher resolution.

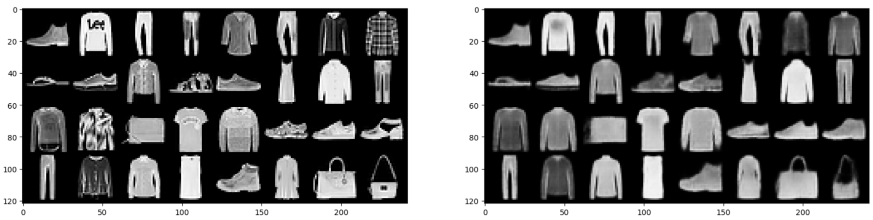


Figure 2: Image reconstruction with Convolutional CVAE for FashionMNIST

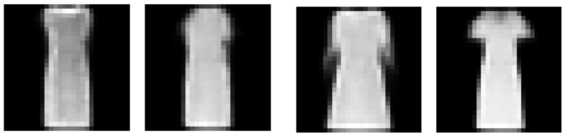


Figure 3: Image generation with Convolutional CVAE and input 'Make a dress' for FashionMNIST

4.4 Quantitative Results from Long Text on COCO

The next two experiments involved testing the architecture and image size. The second experiment involved differing image sizes of 64x64, 128x128, and 256x256. The third experiment involved and addition vs. concantenation method of conditioning. Results are shown in Table 2 for MSE, SSIM, and FID on all combinations of image size and conditioning.

Based on the data contained in Table 2, we found that the 64x64 addition conditioning model had the higher MSE, marginally SSIM, and generally underperformed the 64x64 concatenation conditioning model. In 128x128 and 256x256, the concatenation conditioning models had lower MSE and SSIM, but higher FIDs. We also note that SSIM remained relatively stable across different image sizes, but that FID slightly decreased as image size increased.

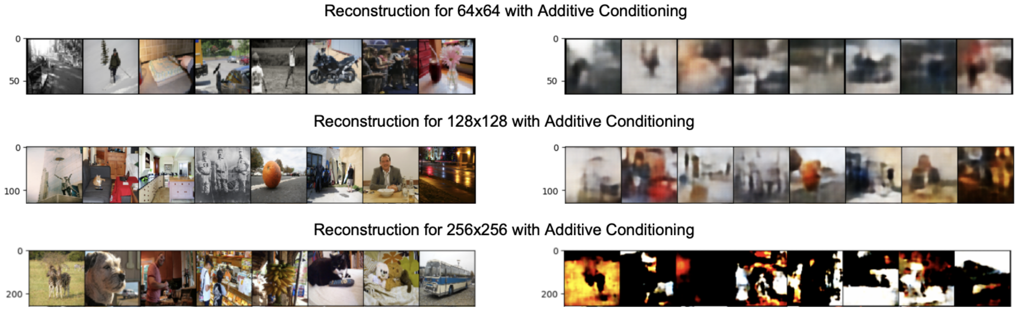
4.5 Qualitative Results from Long Text on COCO

Image reconstructions were provided from the validation dataset in the additive conditioning models in Figure 4. The 64x64 images appear the blurriest, while the 128x128 appears to be most easily recognizable. 256x256 images appear to learn the right features but wrong 3 channel RGB values despite normalization. Image reconstructions were provided from the validation dataset in the concatenation conditioning models in Figure 5. The 64x64 images appear the blurriest, while the 128x128 appears to be more easily recognizable. 256x256 images appear to learn the right features and have the most defined features.

Table 2: Experimental results with various image sizes and conditioning for COCO

Model	**MSE**	**SSIM**	**FID**
64x64 Addition Conditioning	195.2407	0.4135	274.2384
64x64 Concatenation Conditioning	184.8410	0.4148	272.1877
128x128 Addition Conditioning	628.7485	0.4725	231.6987
128x128 Concatenation Conditioning	615.0895	0.4615	299.9901
256x256 Addition Conditioning	20313.7465	0.1746	237.1675
256x256 Concatenation Conditioning	3656.8606	0.4421	243.7546

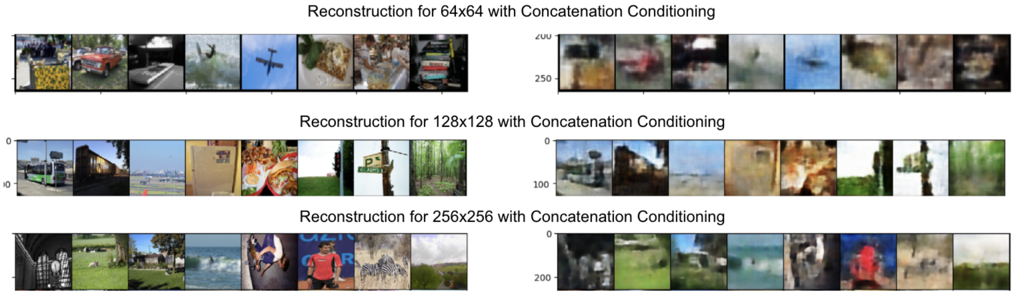
Image generations were provided using prompts and image results shown in Figure 6. The 64x64 images for both types of conditioning learn the background but appear blurry. The 128x128 images appear to learn some features and add detail to the image along with an object. The 256x256 images learn the most distinct features and the background.


Reconstruction for 64x64 with Additive Conditioning

Reconstruction for 128x128 with Additive Conditioning

Reconstruction for 256x256 with Additive Conditioning

Figure 4: Image reconstruction on COCO for additive conditioning models


Reconstruction for 64x64 with Concatenation Conditioning

Reconstruction for 128x128 with Concatenation Conditioning

Reconstruction for 256x256 with Concatenation Conditioning

Figure 5: Image reconstruction on COCO for concatenation conditioning models

5 Discussion

For comparing the Fashion-MNIST models, each quantitative metric in Table 1 told a slightly different story of the performance difference between using fully connected layers and convolutional layers. The convolutional CVAE performed better for the metrics MSE and SSIM. Note that the fully connected CVAE was better for FID. This might be due to the fact that the input for the fid function is a flattened vector for the fully connected CVAE, while we pass an reconstructed image 28x28 for the convolutional model. Thus, this metric is less appropriate for comparing both models here. Instead, we evaluate the performance by MSE and SSIM which are implemented analogously for both models and allow a fair comparison. In line with our expectation, the convolutional CVAE performs better w.r.t. MSE as well as SSIM. Considering that the convolutional CVAE performed better on those two metrics which are relevant here, along with the computational benefits of convo-

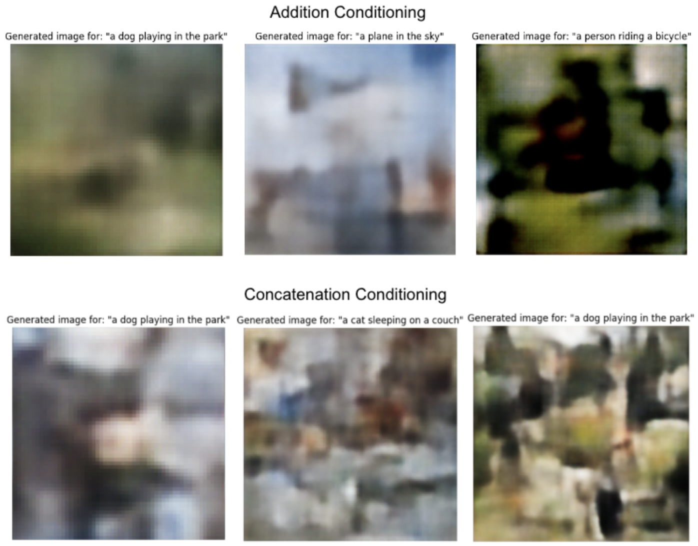

Addition Conditioning
Generated image for: "a dog playing in the park"
Generated image for: "a plane in the sky"
Generated image for: "a person riding a bicycle"
Concatenation Conditioning
Generated image for: "a dog playing in the park"
Generated image for: "a cat sleeping on a couch"
Generated image for: "a dog playing in the park"

Figure 6: Image generation on COCO, from left to right 64x64, 128x128, 256x256

lutional layers (Semeniuta et al., 2017), we chose to continue with a convolutional approach when scaling up for long text on the COCO dataset.

For COCO, convolution is even more important in order to implement invariances in the network. This was probably less relevant for fashion MNIST, as all objects have the same rotation and are placed in the middle of the picture. In contrast, images from COCO include objects with different positions. Therefore, convolution might improve shift or translational invariance to some degree. The models likely performed well as the resolution was low at 28x28 resulting in 784 total pixels. This information for the 1 channel greyscale image could easily be contained with a latent space of dimension 128. The short text approach with one-hot encoding and embeddings is rudimentary compared to the CLIP feature extractor. However, since the CVAE is a simplistic architecture compared to newer state-of-the-art models, the simplicity from the short-text encoding along with the small image size allows for strong results.

From the second experiment, a smaller image size leads to higher performance metrics. This logically follows as smaller images have less total pixels and will be easier for the CVAE to learn and reconstruct features. Since there are less details to reconstruct, it also logically follows that smaller images are easier to generate. It is also faster and less computationally intensive to train CVAEs with smaller images since, making it a more feasible approach. Architectures for the different image sizes can be extremely similar with varying levels of convolutions to filter the image along with different transform cropping sizes from COCO. It testing these models, mini experimentation was done on the latent space size for each model. As expected, increasing the size of the latent space allows the model to capture more information and better reconstruct the image. However, a larger latent space results in a model with longer training time and more parameters. This leads to a tradeoff between latent space size performance and computational efficiency.

The results of the third experiment are nuanced by the second experiment as image size affected results of concatenation vs additive model architecture. For 64x64 and 128x128, the addition vs. concatenation models results are very similar, indicating that there is little effect on performance. However, for the 256x256, the concatenation model largely outperformed the additive model. This is hypothesized to be because in concatenating, the model can learn to lower the weights for inputs that are not useful, and this will not affect image information as annotations are separated from in the image. For adding, annotations and images are not separated, so adding distorts image information if they are not useful. This likely manifests only as resolution increases as image information becomes more important as the total number of pixels increases. It is hypothesized this will occur in images larger than 256x256 as well, but it is extremely computationally intensive.

Image reconstruction within all CVAEs worked to some extent, showing that the CVAE models were all learning the intended task. The largest issue came from image generation across any CVAE trained with COCO. This was expected as COCO annotations are primarily literal and descriptive. Aesthetic details, artistic elements, or style information are missing from the annotations that would be needed to reasonably reconstruct the image after being passed through the CLIP feature extractor. COCO also does not contain information about location of the object in the image and is a relatively small dataset (with even further subsampling on top of that). For these reasons, quality image generation was not a feasible task, but all models seemed to at least learn to generate a background and some resemblance of an object near the center of the image.

6 Conclusion

With a fully connected and convolutional CVAE, images from Fashion-MNIST could successfully be reconstructed and generated. However, the convolutional CVAE was a more robust and marginally more performant with respect to the quantitative metrics. Utilizing CLIP and convolutional CVAEs on the COCO dataset, experimentation concluded that images 128x128 could be reconstructed with a slight blur. As resolution increased to 256x256, it became far more difficult to reconstruct the image. The model on the smallest image size of 64x64 could recreate images with a strong blur with objects that are sometimes hard to identify, although some of the blurriness is a result of aggressively downsampling the image. The other main experiment ran with long text on COCO concluded that at image sizes up to 128x128, concatenating and adding the conditional input in the encoder and decoder are both feasible. However, it is theorized that at 256x256 or in cases where it is difficult for the model to learn, concatenation is better than addition in model architecture. Generated images based on COCO was difficult due to the limitations of the CVAE and the simple annotations provided on the COCO dataset.

For future work, it is recommended to utilize larger-scale and multiple GPUs along with more memory storage to be able to load 80,000 images within COCO with no computational limitations. With multiple powerful GPUs and the complete COCO dataset, it is likely the models would achieve a lower loss and better quantitative metrics for reconstruction. This would also allow for increasing the latent space size in the CVAEs without any computational issues, which we theorize would also provide better quantitivative metrics. For image generation from long text, future work includes using newer state of the art architectures like conditional diffusion. It would also be recommended to train on a dataset with more information contained in the annotations such as LAION.

REFERENCES

David H Ackley, Geoffrey E Hinton, and Terrence J Sejnowksi. A learning algorithm for boltzmann machines. *Cognitive Science*, 1985.

Geoffrey E Hinton, Simon Osindero, and Yee-Whye Teh. A fast learning algorithm for deep belief nets. *Neural computation*, 2006.

Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

Jim Nilsson and Tomas Akenine-Möller. Understanding ssim. *arXiv:2102.12037*, 2020.

Alec Radford, Jong Wook Kim, and Chris Hallacy. Learning transferable visual models from natural language supervision. *arXiv:2103.00020*, 2021.

Gaurav Raut and Apoorv Singh. Generative ai in vision: A survey on models, metrics and applications. *arXiv:2402.16369*, 2024.

Stanislau Semeniuta, Aliaksei Severyn, and Erhardt Barth. A hybrid convolutional variational autoencoder for text generation. *arXiv:1702.02390*, 2017.

Harvey William, Saeid Naderiparizi, and Frank Wood. Conditional image generation by conditioning variational auto-encoders. *arXiv:2102.12037*, 2022.

Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv:1708.07747*, 2017.

Beichan Zhang, Pan Zhang, and Xiaoyi Dong. Long-clip: Unlocking the long-text capability of clip. *arXiv:2403.15378*, 2024.

We disclose the use of ChatGPT for this group project.