CS 5787: Deep Learning
Controllable Text-Image Generation with Enhanced Text Encoder and Edge Loss

Xiaoyu Liang (xl778) Zeyu Wang (zw597) Cheng Wang (cw859) Zhenglun Chen (zc447)

Abstract

Text-to-image generation through generative adversarial networks (GANs) has become a popular research topic in recent years in the fields of natural language processing and high-quality image synthesizing. Many text-to-image GAN algorithms have been proposed to generate realistic images that align the text query to a large extent, and most of them are based on learning latent word features which produces realistic characteristics though, but usually doesn't capture the shape of the described item, generating images that contain out-of-shape items. In this paper, we remodeled an existing controllable text-to-image GAN [4] by 1) modifying the encoders and 2) adding more kinds of losses as an attempt to improve model performance. Experiments on benchmark datasets demonstrate that our method outperforms existing state of the art based on the results retrieved from the 15th epoch, and is able to effectively manipulate synthetic images using natural language descriptions.

1. Introduction

While text-to-image generators are widely applied in synthesizing images that match text description, big improvements were always achieved when the text/image encoding techniques were boosted. In the early days, most studies focused mainly on the adjustments to the image generation network, while generally regraded texts at sentence level instead of word level. As a result, those embedding outputs were relatively coarse since they lost many details of text information.

Later, in the conditional AlignDRAW model [5] and its subsequent work [12, 11], deeply mined text information was applied which extracted more fine-grained word-level text features. The performances of these works highly outperformed those from previous works.

Recently, the use of the scene graph [3] and semantic layout [2] enhanced the generating process even more. They not only took the fine-grained text characteristics into consideration, but also conducted in-depth mining of semantic information by modeling the entities and their relations in the text, so as to deal with more complex scenarios.

To preserve image features, Mao et al. [6] proposed an edge preserving mechanism and a multi-constraint framework to enhance the image encoding/decoding process. Its perceptual loss combines several statistical characteristics of the compressed images, including mean square error, feature loss, edge loss, and adversarial loss.

Inspired by their work, we utilized an edge-preserving perpetual loss calculation that takes edge information of the real and fake images to constrain the generation of shape. The loss of information in edge features will affect the learning process of the network.

The contribution of our work can be addressed in the following two aspects:

- Better exploit the text information by modifying the text encoder with the use of pre-trained language models
- Enforce model to pay attention to the edge by introducing additional constraint on perceptual loss

The above has accelerated the training process of ControlGAN [4] and achieved higher Inception Score [8] at the same epoch.

2. Related Work

2.1. Multi-stage generative adversarial networks

Multi-stage generative adversarial networks have become a commonly adopted architecture when building generative adversarial networks. Zhang et al. proposed a two-stage

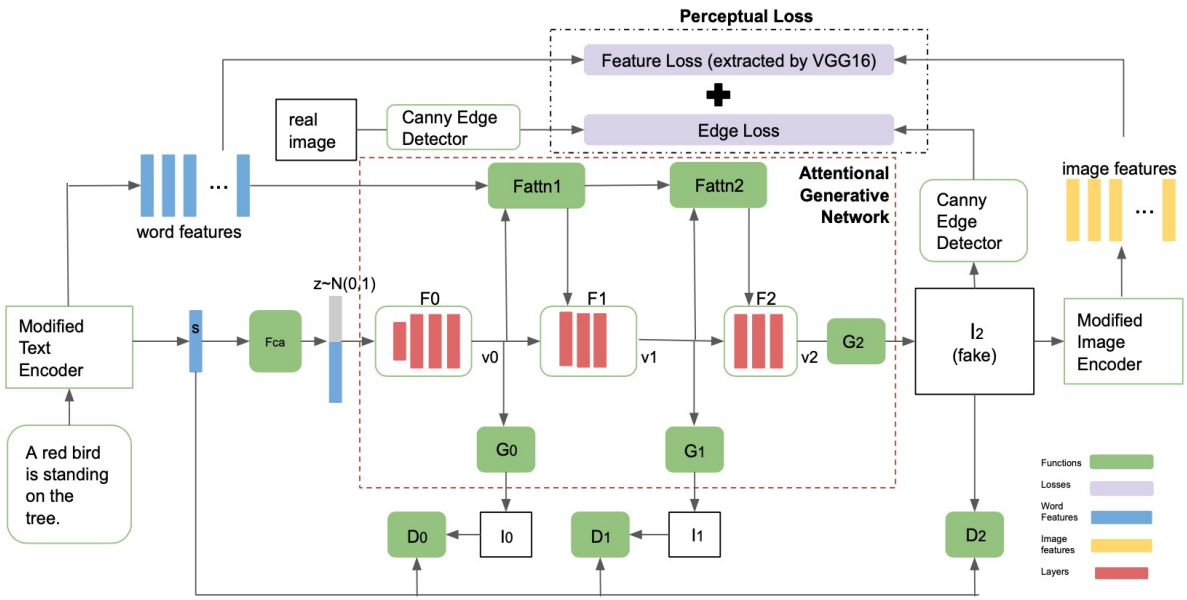

Figure 1. The architecture of our model. Given a sentence S, this text encoder encodes S to a sentence feature vector s as well as a word feature matrix w. We then feed the sentence feature vector and word feature matrix into the network. As shown above, the network has 3 generators which take the hidden visual features v as inputs and generate images I at each stage.

generative adversarial network architecture (StackGan++) [12] to generate high-resolution images: Stage-I GAN generates low-resolution images with primitive shapes and colors given textual description, and Stage-II GAN processes the results from the previous stage and text description to yield fine realistic images.

Huang et al. [11] proposed an Attentional Generative Adversarial Network (AttnGAN) to allow attention-driven, multi-stage refinement for fine-grained text-to-image generations which focus on drawing the sub-regional details driven by the most relevant words. Li et al. [4] also adopted this multi-stage architecture and built a controllable text-to-image generative adversarial network (ControlGAN) which synthesize realistic images through a word-level spatial, channel-wise attention-driven generator that distinguishes visual features. Based on the previous works, we can see that generating images in multi steps from rough to fine is a main trend and a widely used backbone for developing GANs.

2.2. Image-Text embeddings

Well-developed encoders are necessary during texts/images processing. In 2016, Wang et al. proposed a structure-preserving embedding method [9] by learning joint embeddings of images and texts using a two-branch neural network with multiple layers of linear projections. This model achieved high accuracy during image-to-text or text-to-image retrievals.

The Deep Attentional Multimodal Similarity Model (DAMSM) [11] also provides an image-text matching mechanism by measuring the similarity scores and reducing the mismatch losses of images and texts that are encoded by a CNN net and a LSTM net respectively. Li et al. [4] later on remodeled the image encoder by replacing the previous mix_6e layer of Inception-v3 net with a $relu2\_2$ layer of VGG-16 net. We noticed that a better encoding model can probably be achieved by integrating different embedding models and modifying the objective function to include more detailed losses.

3. Proposed Model

We adopted the proposed model of the channel-wise attention GAN as the main architecture, and explored 2 alternatives as an attempt for improving model performance: in the first approach, we modified the text encoder to make better use of text information. In the second approach, we captured other detailed losses (i.e., edge-loss), and added it to the objective function with a weight of 10*(-3).

In generally, for each given sentence S, our goal is to generate a realistic image I that semantically matches the given text. As shown in the Fig.1, our model has two main components: 1) an Attentional Generative Network

that captures latent visual attributes and emphasizes the most relevant subregions during generation. 2) an edge-preserving perceptual reconstruction that aligns item edges of the generated image with the original image.

The attentional generative network takes in the sentence feature vector s and word feature matrix w into the network which contains 3 generators and generates images I at each stage. Specifically,

$$v_0 = F_0(z, Fca(s))$$
$$v_k = F_k(v_{k-1}, Fattn_k(w, v_{k-1})), k = 1, 2$$
$$I_k = G_k(v_k), k = 0, 1, 2.$$

To address our ideas of modification, we conducted the following alternative approaches to modify the text encoder and the objective function:

3.1. Approach 1: Make better use of text information by modifying the text encoder

One of the main approaches to improve the text-to-image generation is to make better use of text information. In the work of controlGAN [4], the author adopted a bidirectional RNN [11] as text encoder to encode the sentence query. The encoder takes a sentence as input, and outputs two things: 1) a sentence feature s to represent the whole sentence, and 2) a word feature matrix that contains more detailed text information. To improve this existing encoder, we further trained the text encoder with pre-trained word embedding models, as well as adopted Part of Speech(POS) tagging to remove meaningless words in the text.

3.1.1 Train text encoder based on pre-trained word embedding model

Using pre-trained language embedding models is one of the most exciting directions in natural language processing (NLP), and is also an exploration of transfer learning. There are many state-of-the-models, such as BERT, RoBERTa, XLNet, and T5. These models, though differ in design, share the same idea of leveraging a large amount of unlabeled text to build a general model of language understanding before being fine-tuned on specific NLP tasks such as sentiment analysis and question answering.

In this project, we have experimented with pre-trained models including GloVe and BERT, to initialize the word embedding of the RNN text encoder model. GloVe(Global Vectors for Word Representation) [7] is a popular embedding technique, trained on aggregated global word-word co-occurrence from a corpus. BERT(Bi-directional Encoder Representations from Transformers) [1] is a even more successful model, which is able to single-handedly achieve state-of-the-art performances on several NLP tasks.

3.2. Approach 2: Add additional constraints by including external losses

External losses, such as box loss, pixel loss, image adversarial loss etc., are regraded as efficient methods for improving the performance of GANs.

3.2.1 Feature Loss extracted by VGG16

To mitigate the randomness of the generated images, perceptual loss calculation is an important approach to quantize the fidelity of the generated images. While pixel-wise difference is sensitive to the colors, a way of measuring distance between generated images and ground truth images is to apply metric learning. ControlGAN [4] backpropogated the mean square error (MSE) between visual features of real images and fakes images outputted from a relu layer of VGG16 as its perceptual losses. Measuring perceptual losses of real and fakes images mainly captures the visual feature disparities, and thus performs well when being included in the objective function.

3.2.2 Edge Loss captured by canny edge detector

We noticed that a big amount of generated images, through semantically align with the text query, are visually out of shape. As shown in the figure 2, this generated image obviously contains bird-related features like feathers and beaks, but lacks an acceptable shape of bird.

Figure 2. A generated bird with bird features but in a bad shape

In our attempt, we conducted an edge-preserving perpetual loss calculation that takes edge information of the real and fake images to generate images with meaningful shapes. For each image generated by G2 (the generator in the final stage), we extracted edge information by applying a canny edge detection algorithm to this fake image and its corresponding real image.

The canny edge detection process removes noise in the image with a 5x5 Gaussian filter, and then filters the image horizontally and vertically with a Sobel operator

to get the pixel gradients in both directions. After getting gradient magnitudes, edges are detected at which the pixel gradients change dramatically. For each image, this algorithm removes those unwanted pixels which don't contribute to edge formation, generating a new grayscale edge map that contains only the item edges.

3.2.3 Image-Text Embedding

To extract the edge image embedding, we adopted a deep structure-preserving image-text embedding approach [9]. This method is able to learn joint embeddings of images and text using a two-branch neural network. That's to say, the image and the text will share a latent space where vectors from the two modalities can be compared directly. Though in our task, we don't need to measure the semantic similarity between visual data and text data, but we need to measure the similarity of the edge from the generated image and the real image. We could utilize this method because in the learned latent space, images with similar meaning would be close to each other. Therefore, after training the two-branch model(Fig.3), we selected the image sub-network to encode the edge images to the target embeddings. Based on this, we could be able to calculate the edge loss that we proposed.

Model Architecture. The model architecture is in Fig.3. As shown in Fig.3, our model has two branches, each composes of fully connected layers with weight matrices W_l and V_1 . Two linear layers are separated by a ReLU activation function. After the last fully connected layer, we applied L2 batch normalization.

Training Objective. Our training objective is a stochastic margin-based loss that includes bidirectional cross-view ranking constraints. Given a training image x_i , let Y_i^+ and Y_i^- denote its sets of matching (positive) and non-matching (negative) texts, respectively. We want the distance between x_i and each positive text y_j to be smaller than the distance between x_i and each negative text y_k by some enforced margin m:

$$d(x_i, y_i) + m < d(x_i, y_k) \quad \forall y_j \in Y_i^+, \forall y_k \in Y_i^-.$$

Similarly, given a text $y_{i'}$, we have

$$d(x_{j'}, y_{i'}) + m < d(x_{k'}, y_{i'}) \quad \forall x_{j'} \in X_{i'}^+, \forall x_{k'} \in X_{i'}^-.$$

Loss Functions. We converted the constraints to our training objective in the standard way using hinge loss. The resulting loss function is given by

$$L(X, Y) = \sum_{i,j,k} max[0, m + d(x_i, y_i) - d(x_i, y_k)] + \sum_{i,j,k} max[0, m + d(x_{j'}, y_{i'}) - d(x_{k'}, y_{i'})]$$

Triplet Sampling. Our loss involves all triplets consisting of an anchor input, a positive match of the same class as the anchor, and a negative match of a different class from the anchor. To improve the optimizing efficiency, we sampled triplets within each mini-batch and optimized our loss function using SGD. As the Fig.4 shows, the triplet loss minimizes the distance between an anchor and a positive, both of which have the same identity, and maximizes the distance between the anchor and a negative of a different identity.

Features and Network Settings. To train the Image-Text Embedding network, we extracted edge information by applying a canny edge detection algorithm to CUB dataset. The canny edge detection process removes noise in the image with a Gaussian filter, and then filters the image horizontally and vertically, generating a new edge map that contains only the item edges. We removed color messages from the captions and applied BERT to encode text. The encoded text features and edge features are parsed into the network. The adversarial learning enforces edge features and text features to lean toward the same feature vector. Image embedding model takes 256*256 image and maps it to a 300-dimension feature vector, while text embedding model parses a sentence feature with dimension 300.

Application in perceptual loss. Besides the feature difference obtained from the pre trained VGG-16 network, we included the edge loss when computing perceptual loss. We connected the image embedding model that trained separately to the stage of perceptual loss, and parsed the edge into the image embedding network. A new perceptual loss is defined as:

$$L\_percep(I, I^{'}) = L\_feature(I, I^{'}) + 10^{-3} * L(edge)$$

4. Evaluation

4.1. Datasets

Our work is evaluated on CUB bird [10], consisting of 8,855 training images, 2,933 test images, and 10 corresponding captions per image.

4.2. Implementation

We maintained the training setup of the original work which used Adam optimiser with the learning rate 0.0002. To evaluate the enhanced text encoders with pre-trained models including BERT and GloVe, we changed the text-encoders and image-encoders (co-trained with the text-encoder), while keep other settings the same. To study the impact of edge loss, we adopted its own text-encoder and

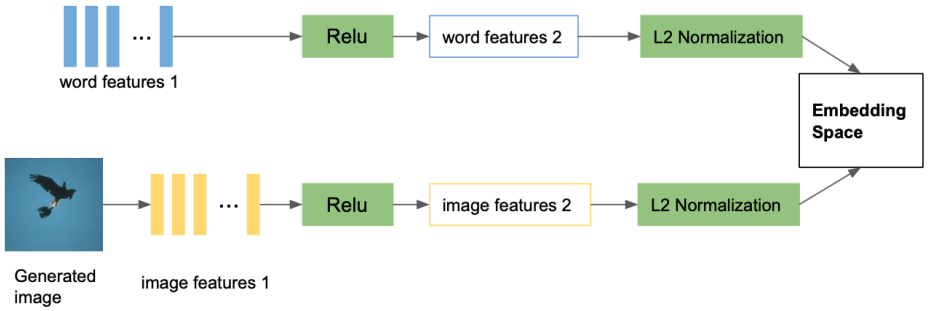

Figure 3. Model structure: there are two branches in the network, one for images (X) and the other for text (Y). Each branch consists of fully connected layers with ReLU nonlinearities between them, followed by L2 normalization at the end [9].

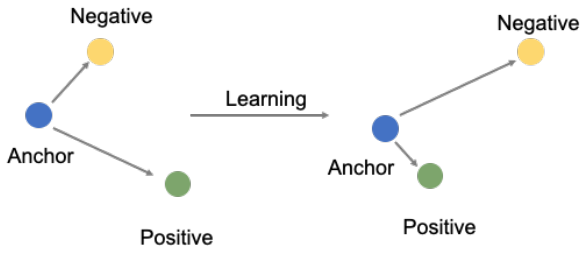

Figure 4. The Triplet Loss minimizes the distance between an anchor and a positive, both of which have the same identity, and maximizes the distance between the anchor and a negative of a different identity

image-encoder, and solely modified the layer of perceptual loss.

5. Ablation Studies

The original model was trained for 600 epochs. In order to assess the effectiveness of our work, we extracted the output from the 15th epoch of each model and used the Inception Score(IS) [8] of generated testing images. According to quantitative results shown in Table.1, our approaches achieved higher Inception Scores at the 15th epoch, indicating that the proposed changes on the model can effectively promote the training process at early stage. In addition, we presented a visual comparison(Fig.5) of the generated images from different model settings given the same caption.

Effectiveness of advanced text-encoder. The models using enhanced text encoder with pre-trained model BERT and GloVe are able to generate yellow breast even with a smaller amount of training epochs. Since the model with edge loss utilizes the same text encoder as the ControlGAN, they both generated a blue bird. Adopting an advanced text-encoder can better capture semantic details and contribute to rendering correct colors to the image.

Model	IS Mean	IS Std
ControlGAN [4]	2.9145	0.0600
Ours_Bert	3.9473	0.4371
Ours_GloVe	4.5177	0.2305
Ours_EdgeLoss	3.8083	0.3646

Table 1. The test results extracted from the 15th epoch of each model. To evaluate the enhanced text encoders, we changed the text-encoders and image-encoders (co-trained with the text-encoder), while kept other settings the same. To study the impact of edge loss, we adopted its own text-encoder and image-encoder, and solely modified the layer of perceptual loss

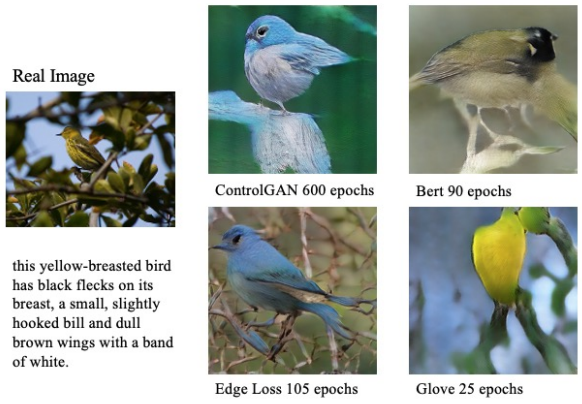

Figure 5. The generated samples

Effectiveness of additional edge loss. The model with edge loss seems to have a clearer shape of bird and a rich background than the others.

6. Conclusion

Based on ControlGAN[4], we proposed better text encoders to utilize the text information and enforced the model to pay attention to the edge by introducing additional constraint on perceptual loss. We effectively achieved higher Inception Score at the same epoch comparing against the original ControlGAN[4], which demonstrated the advantages of our method with respect both to high quality image-generation and efficiency of learning.

References

[1] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[2] S. Hong, D. Yang, J. Choi, and H. Lee. Inferring semantic layout for hierarchical text-to-image synthesis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7986–7994, 2018.

[3] J. Johnson, A. Gupta, and L. Fei-Fei. Image generation from scene graphs. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1219–1228, 2018.

[4] B. Li, X. Qi, T. Lukasiewicz, and P. Torr. Controllable text-to-image generation. In *Advances in Neural Information Processing Systems*, pages 2063–2073, 2019.

[5] E. Mansimov, E. Parisotto, J. L. Ba, and R. Salakhutdinov. Generating images from captions with attention. *arXiv preprint arXiv:1511.02793*, 2015.

[6] Q. Mao, S. Wang, S. Wang, X. Zhang, and S. Ma. Enhanced image decoding via edge-preserving generative adversarial networks. In *2018 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6. IEEE, 2018.

[7] J. Pennington, R. Socher, and C. D. Manning. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, 2014.

[8] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen. Improved techniques for training gans. In *Advances in neural information processing systems*, pages 2234–2242, 2016.

[9] L. Wang, Y. Li, and S. Lazebnik. Learning deep structure-preserving image-text embeddings. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5005–5013, 2016.

[10] P. Welinder, S. Branson, T. Mita, C. Wah, F. Schroff, S. Belongie, and P. Perona. Caltech-UCSD Birds 200. Technical Report CNS-TR-2010-001, California Institute of Technology, 2010.

[11] T. Xu, P. Zhang, Q. Huang, H. Zhang, Z. Gan, X. Huang, and X. He. Attngan: Fine-grained text to image generation with attentional generative adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1316–1324, 2018.

[12] H. Zhang, T. Xu, H. Li, S. Zhang, X. Wang, X. Huang, and D. N. Metaxas. Stackgan++: Realistic image synthesis with stacked generative adversarial networks. *IEEE transactions on pattern analysis and machine intelligence*, 41(8):1947–1962, 2018.