StoryGAN Report

Niyam Shyam Kotian
24B1100

June 2025

Contents

1 Introduction

The objective is to generate a sequence of images to describe a story from a multi-sentence paragraph. This presents two primary challenges:

- **Local and Global Consistency:** Each generated image must correspond meaningfully with its paired sentence and the entire sequence of images must coherently depict the whole story.
- **Sequential Scene Evolution:** The objects and backgrounds must transition smoothly across frames to ensure narrative consistency.

To address this, StoryGAN proposes a sequential GAN framework that uses RNNs and a new Text2Gist cell to dynamically update contextual information. The model uses a two-tier training scheme through an image-level discriminator and a story-level discriminator.

2 Methodology

2.1 Architecture

The key contributions include a Context Encoder with GRU and Text2Gist cells to capture evolving story context, a stochastic Story Encoder that maps the entire narrative to a latent vector to initialize the generation process and a joint adversarial training for both local and global consistency.

StoryGAN generates an image sequence $\hat{X} = [\hat{x}_1, \hat{x}_2, ..., \hat{x}_T]$ from a story $S = [s_1, s_2, ..., s_T]$, where each $\hat{x}_t$ corresponds to sentence s_t. The architecture comprises:

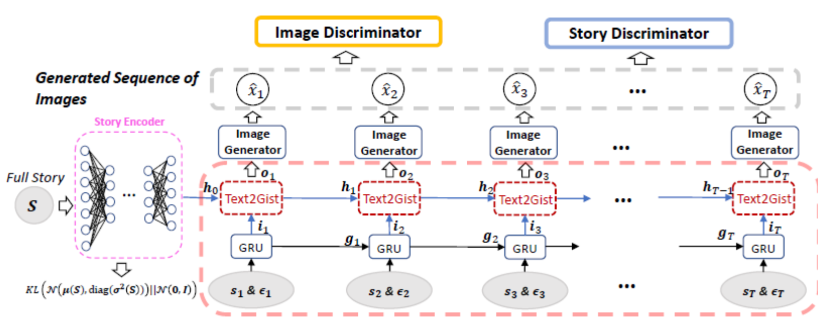

Figure 1: The framework of StoryGAN

Story Encoder:
It maps the entire story S to a low-dimensional latent vector $h_0 \sim \mathcal{N}(\mu(S), \sigma^2(S))$, where $\mu(\cdot)$ and $\sigma(\cdot)$ are MLPs. By using stochastic sampling, the Story Encoder

deals with the discontinuity problem in the original story space. It is regularized via KL divergence to match a standard Gaussian:

$$\mathcal{L}_{\text{KL}} = D_{\text{KL}}\left(\mathcal{N}(\mu(S), \sigma^2(S)) \parallel \mathcal{N}(0, I)\right)$$

h_0 initializes the Context Encoder's hidden state.
Context Encoder:
It is a two-layer RNN that dynamically updates the contextual information:

1. **GRU Layer:** It processes the current sentence s_t and noise ϵ_t, outputting intermediate vector i_t.

$$i_t, g_t = \text{GRU}(s_t, \epsilon_t, g_{t-1})$$

2. **Text2Gist Cell:** A modified GRU cell that integrates i_t with the story context h_{t-1} to produce an updated context h_t which reflects scene evolution via gating mechanisms and a gist vector o_t which aggregates the contextual and current-sentence features for image generation.

$$z_t = \sigma(W_z i_t + U_z h_{t-1} + b_z) \quad \text{(Update gate)}$$
$$r_t = \sigma(W_r i_t + U_r h_{t-1} + b_r) \quad \text{(Reset gate)}$$

$$h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \tanh(W_h i_t + U_h(r_t \odot h_{t-1}) + b_h)$$
$$o_t = \text{Filter}(i_t) \odot h_t$$

Image Generator:
Generates $\hat{x}_t$ from o_t using a deep convolutional network.
Discriminators:
Two adversarial components ensure consistency:

1. **Image Discriminator** (D_I)**:** It evaluates triplets $(s_t, h_0, \hat{x}_t)$ against real pairs (s_t, h_0, x_t). It also ensures local alignment between sentences and images.

2. **Story Discriminator** (D_S)**:** It encodes the full image sequence $E_{img}(X)$ and story $E_{txt}(S)$ into feature vectors. It also computes global coherence through a similarity score.

$$D_S = \sigma\left(w^\top (E_{\text{img}}(X) \odot E_{\text{txt}}(S)) + b\right)$$

2.2 Loss Functions

Let $\theta, \psi_I, and\ \psi_S$ denote the parameters of the whole generator $G(\cdot;\theta)$, the image discriminator, and the story discriminator, respectively. The objective function for StoryGAN is

$$\min_{\theta} \max_{\psi_I, \psi_S} [\alpha \mathcal{L}_{Image} + \beta \mathcal{L}_{Story} + \mathcal{L}_{KL}]$$

where α and β balance the three loss terms. $\mathcal{L}_{KL}$ is the regularization term of the Story Encoder previously defined. $\mathcal{L}_{Image}$ and $\mathcal{L}_{Story}$ are defined as

$$\begin{aligned}\mathcal{L}_{Image} = \sum_{t=1}^{T} \big[& E_{(\mathbf{x}_t, \mathbf{s}_t)}[\log D_I(\mathbf{x}_t, \mathbf{s}_t, \mathbf{h}_0; \psi_I)] \\ & + E_{(\epsilon_t, \mathbf{s}_t)}[\log(1 - D_I(G(\epsilon_t, \mathbf{s}_t; \theta), \mathbf{s}_t, \mathbf{h}_0; \psi_I))]\big]\end{aligned}$$

$$\begin{aligned}\mathcal{L}_{Story} = & E_{(\mathbf{X}, \mathbf{S})}[\log D_S(\mathbf{X}, \mathbf{S}; \psi_S)] \\ & + E_{(\epsilon, \mathbf{S})}[\log(1 - D_S([G(\epsilon_t, \mathbf{s}_t; \theta)]_{t=1}^{T}, \mathbf{S}; \psi_S))]\end{aligned}$$

$D_I(\cdot, \psi_I)$ and $D_S(\cdot; \psi_S)$ are the image and story discriminator, parameterized by ψ_I and ψ_S, respectively.

2.3 Training Method

The parameters of the story and image discriminators, ψ_I and ψ_S, are updated in two separate **for** loops, respectively, while the parameters of the image generator θ are updated in both loops. The initial hidden state of the Text2Gist layer is the encoded story feature vector h_0 produced by the Story Encoder.

Algorithm 1 StoryGAN Training Procedure

Initialize parameters θ, ψ_I, ψ_S
for $iter = 1$ to max_iter **do**
    for $iter_I = 1$ to k_I **do**
        Sample mini-batch $\{(s_t, S, x_t)\}$ from training set
        Compute $\mathbf{h}_0$ via Story Encoder (Eq. 1)
        Generate single image $\hat{\mathbf{x}} = G(\epsilon_t, s_t; \theta)$
        Update ψ_I and θ using $\nabla_{\psi_I, \theta} \mathcal{L}_{Image}$
    end for
    for $iter_S = 1$ to k_S **do**
        Sample mini-batch $\{(S, X)\}$ from training set
        Compute $\mathbf{h}_0$ and update $\mathbf{h}_t$ $\forall t$ via Text2Gist
        Generate sequence $\hat{\mathbf{X}} = [G(\epsilon_t, s_t; \theta)]_{t=1}^{T}$
        Update ψ_S and θ using $\nabla_{\psi_S, \theta} \mathcal{L}_{Story}$
    end for
end for

We will be using the Adam optimizer for parameter updates. We also find that using different mini-batch sizes for image and story discriminators may accelerate training convergence, and that it is beneficial to update generator and discriminator in different time steps in one epoch.

3 Conclusion

StoryGAN effectively generates coherent sequences of images from a set of sentences using the Text2Gist cell and the dual-discriminator framework. The model outperforms previous methods due to its dynamic context propagation and adversarial training.

References

[1] Yitong Li, Zhe Gan, Yelong Shen, Jingjing Liu, Yu Cheng, Yuexin Wu, Lawrence Carin, David Carlson, and Jianfeng Gao.
StoryGAN: A Sequential Conditional GAN for Story Visualization.
Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2019.