Improving Generation and Evaluation of Visual Stories via Semantic Consistency

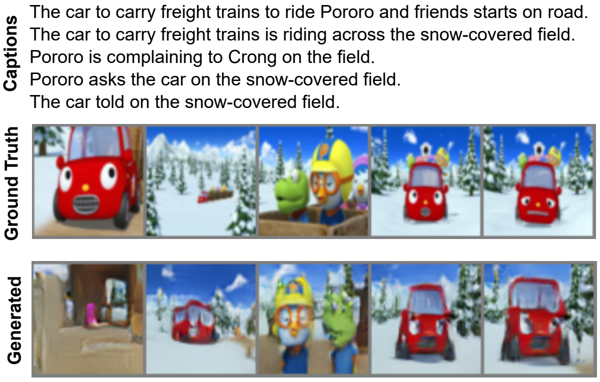
Adyasha Maharana **Darryl Hannan** **Mohit Bansal**
Department of Computer Science
University of North Carolina at Chapel Hill
{adyasha, dhannan, mbansal}@cs.unc.edu

Abstract

Story visualization is an underexplored task that falls at the intersection of many important research directions in both computer vision and natural language processing. In this task, given a series of natural language captions which compose a story, an agent must generate a sequence of images that correspond to the captions. Prior work has introduced recurrent generative models which outperform text-to-image synthesis models on this task. However, there is room for improvement of generated images in terms of visual quality, coherence and relevance. We present a number of improvements to prior modeling approaches, including (1) the addition of a dual learning framework that utilizes video captioning to reinforce the semantic alignment between the story and generated images, (2) a copy-transform mechanism for sequentially-consistent story visualization, and (3) MART-based transformers to model complex interactions between frames. We present ablation studies to demonstrate the effect of each of these techniques on the generative power of the model for both individual images as well as the entire narrative. Furthermore, due to the complexity and generative nature of the task, standard evaluation metrics do not accurately reflect performance. Therefore, we also provide an exploration of evaluation metrics for the model, focused on aspects of the generated frames such as the presence/quality of generated characters, the relevance to captions, and the diversity of the generated images. We also present correlation experiments of our proposed automated metrics with human evaluations.[1]

1 Introduction

While generative adversarial networks (GANs) have achieved impressive results on a variety of image generation tasks (Zhu et al., 2019; Qiao et al., 2019), the task of story visualization (Li et al., 2019b) is a variation of image generation that is more challenging and underexplored. In this setting, there is a story which consists of a sequence of images along with captions describing the content of the images, e.g., a web comic. The goal of the task is to reproduce the images given the captions (Figure 1). The benefits of investigating this task are far reaching. It combines two interesting and challenging sub-areas: text-to-image synthesis and narrative understanding, providing an excellent test bed for exploring and developing multimodal modeling techniques. From an application perspective, such a system could be used to enhance existing textual narratives with visual scenes. This tool would be especially useful to comic artists, who are infamously overworked, allowing them to automatically generate initial drawings speeding up their workflow. Additionally, such a system would have many applications in an educational setting, allowing educators to cater to a more diverse set of learning styles by automatically generating visualizations for a given topic, such as the water cycle in a science lesson. Furthermore, the data in this domain is cartoon-style, meaning the generated im-



The car to carry freight trains to ride Pororo and friends starts on road.
The car to carry freight trains is riding across the snow-covered field.
Pororo is complaining to Crong on the field.
Pororo asks the car on the snow-covered field.
The car told on the snow-covered field.

Figure 1: Illustration of the Pororo-SV dataset (Captions & Ground Truth) and the corresponding images generated from our model (Generated).

[1] Code and data: https://github.com/adymaharana/StoryViz.

Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 2427–2442
June 6–11, 2021. ©2021 Association for Computational Linguistics

ages avoid many of the ethical issues associated with real-world data. For a more detailed discussion, see Section 9.

The challenge of this task extends beyond tasks such as text-to-image or text-to-video synthesis. Namely, there is an explicit, narrative component to the data, which must first be accurately extracted from the text, and then consistently reproduced throughout the images. If the setting or a description of a character is provided in the first caption, this must be carried throughout the scene unless modified by a subsequent caption. Furthermore, the scenes in a single story can change drastically as the story progresses, requiring models to produce a greater variety of images than in a text-to-video task, which typically consists of short videos displaying a single action. To address these issues, we consider the task as proposed in Li et al. (2019b), which provides a baseline architecture, StoryGAN, along with datasets for the task. We introduce techniques that build on existing work and are focused on improving consistency across frames, resulting in images of higher visual quality.

First, we augment the model with Dual Learning via video redescription. The output images are fed through a video captioning model, which is trained to reproduce the ground truth story captions. This provides an additional learning signal to the model, forcing it to semantically align with the given narrative. Next, we add a Copy-Transform module that can take generated images from previous timesteps and copy the most relevant features of those images into the next generated frame, thus making the images more consistent in appearance. Finally, we propose the use of Memory-Augmented Recurrent Transformer (MART) (Lei et al., 2020) to model the correlation between word phrases in the input text and corresponding regions in the generated image. The recurrent nature of MART allows for the learning of sophisticated interactions between the image frames, yielding images that are more consistent in terms of character appearances and background imagery. We call the model architecture with the aforementioned additions DU(AL)-CO(PY)-STORYGAN or DUCO-STORYGAN.

Next, we focus on exploring alternative evaluation methods for story visualization models. While modeling improvements are crucial for progressing in this domain, evaluating these models is a challenge in itself. Like many other generative tasks, it is nontrivial to evaluate a story visualization model. Human evaluation is the most reliable option, but its monetary and time costs make this ill-suited to be the only evaluation method. Most prior work relies upon standard GAN evaluation metrics, which may provide some insight into how well the images were reproduced, yet miss out on other aspects of the story visualization task, such as the visual consistency of the setting across frames and global semantic alignment. Therefore, we make evaluation another focal point of the paper, exploring a variety of automatic evaluation metrics, which capture various aspects of the task, e.g., evaluating the quality of the images, the relevance to the story, the diversity of the generated frames, and the model's ability to accurately represent the characters. We present results from our model and baseline models on all metrics along with qualitative results, demonstrating the improvements from our proposed techniques. Using these metrics, we also provide ablation analyses of our model.

Our main contributions can be summarized as:

1. For the story visualization task, we improve the semantic alignment of the generated images with the input story by introducing dual learning via video redescription.

2. We enable *sequentially-consistent* story visualization with the introduction of a copy-transform mechanism in the GAN framework.

3. We enhance prior modeling techniques in story visualization with the addition of Memory Augmented Recurrent Transformer, allowing the model to learn more sophisticated interactions between image frames.

4. We present a diverse set of automatic evaluation metrics that capture important aspects of the task and will provide insights for future work in this domain. We also conduct correlation experiments for these metrics with human evaluation.

2 Related Work

Li et al. (2019b) introduced the task of story visualization and the StoryGAN architecture for sequential text-to-image generation. There have been a few other works that have attempted to improve upon the architectures presented in this paper. PororoGAN (Zeng et al., 2019) aims to improve the semantic relevance and overall quality of the images via a variety of textual alignment modules and a patch-based image discriminator. Li et al. (2020) also improve upon the StoryGAN architecture by upgrading the story encoder, GRU network, and discriminators and adding Weighted Activation Degree (Wen et al., 2019). Song et al. (2020) is a more recent work which makes improvements to the StoryGAN architecture; the primary contribution is adding a figure-ground generator and discriminator, which segments the figures and the background of the image. Our model improvements of MART, dual learning, and copy-transform build upon more recent techniques and we support them with a detailed series of ablations.

Text-to-Image and Text-to-Video Generation. While story visualization is an underexplored task, there has been plenty of prior work in text-to-image synthesis. Most papers in this area can be traced back to StackGAN (Zhang et al., 2017). Subsequent work then made various modifications to this architecture, adding attention mechanisms, memory networks, and more (Xu et al., 2018; Zhu et al., 2019; Li et al., 2019a; Yi et al., 2017; Gao et al., 2019). Huang et al. (2018) and Qiao et al. (2019) are direct precursors of our work. Both of these works subject the generated output as an image captioning task which attempts to reproduce the original text. Our proposed dual learning approach is an expansion of this module, where we use a state-of-the-art video captioning model based upon the MART (Lei et al., 2020) architecture to provide an additional learning signal to the model and increase the semantic consistency across images.

In the domain of text-to-video synthesis, Li et al. (2018), Pan et al. (2017), Gupta et al. (2018) and Balaji et al. (2019) generate videos from single sentences. In contrast to videos, story visualization does not have the requirement that the frames flow continuously together. Therefore, it allows for more interesting interactions and story-level dynamics to be captured that would only be present in longer videos.

Interactive Image Editing. Another task related to story visualization is interactive image editing. In this setting, rather than going from purely text to image, the model is given an input image along with textual instructions/directions, and must produce an output image that modifies the input image according to the text. This can take the form of high level semantic changes to the image, such as color and shape, as in Liu et al. (2020), Nam et al. (2018), and Chen et al. (2018), or this might take the form of Photoshop-style edits, as in Laput et al. (2013), Shi et al. (2020), and Manuvinakurike et al. (2018a). Alternatively, Cheng et al. (2020), Manuvinakurike et al. (2018b), and El-Nouby et al. (2019) are slightly closer to our task due to their sequential nature, where an image is modified repeatedly according to the textual feedback provided via a dialogue. However, unlike story visualization, these tasks do not have a narrative component. Furthermore, they involve repeatedly editing a single object at each timestep instead of generating diverse scenes with dynamic characters.

3 Methods

3.1 Background

Formally, the task consists of a sequence of sentences $S = [s_1, s_2, ..., s_T]$ and a sequence of images $X = [x_1, x_2, ..., x_T]$, where the sentence s_k describes the contents of the image x_k. The model receives S as input and produces a sequence of images $\hat{X} = [\hat{x}_1, \hat{x}_2, ..., \hat{x}_T]$, attempting to accurately reproduce X. As detailed in Li et al. (2019b), there are two aspects of this task. The first is local consistency, which is concerned with the quality of individual pairs in the sequence; an example is locally consistent if image $\hat{x}_k$ accurately represents the contents of sentence s_k. The second aspect is global consistency, which is concerned with the quality of the entire sequence. Namely, whether the sequence of images $\hat{X}$ accurately captures the content of the sequence of sentences S.

The general approach to this task as followed by StoryGAN (Li et al., 2019b) is as follows: The story encoder creates the initial representation h_0 of the story S. This is then passed to the context encoder, which is a recurrent model that takes a sentence s_k as input and forms a representation o_k. Each of these representations o_k are then fed to the image generator, which outputs an image $\hat{x}_k$. The generated images are passed to two discriminators, the image discriminator and story discriminator, which each evaluate the generated images $\hat{x}_k$ in different ways and produce a learning signal that can be used to adjust the parameters of the network.

3.2 DuCo-StoryGAN

The framework of our model is based on the StoryGAN architecture. We improve upon the context encoder and expand the network with dual learning and copy-transform mechanisms. The image and

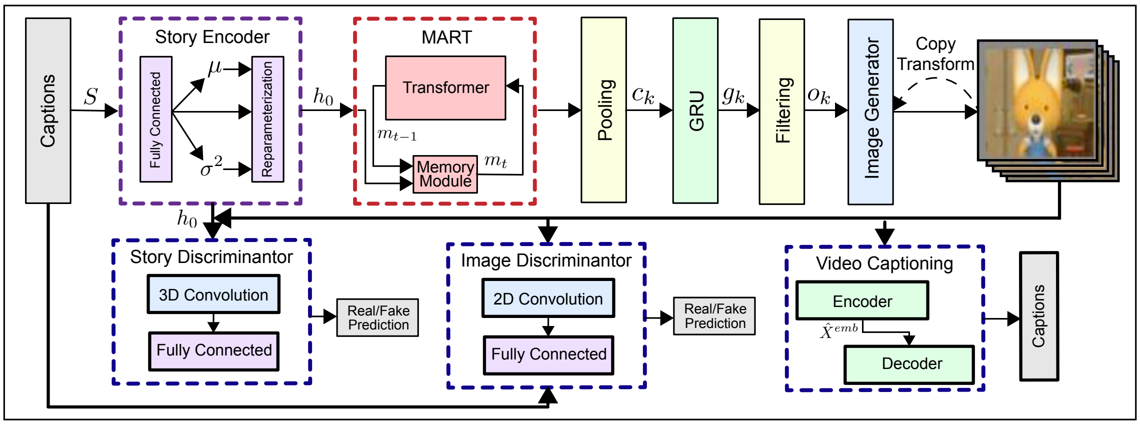


Figure 2: Illustration of DuCo-StoryGAN architecture. The story encoder is used to initialize memory module in MART context encoder, which encodes the captions for the image generator. The copy-transform mechanism copies features from images generated in previous timesteps to the image in current timestep. The generated images are passed to story and image discriminators, and dual learning video captioning model.

story discriminators, and the story encoder from the original model are retained in DuCo-StoryGAN; each contributes to a separate loss term i.e. $\mathcal{L}_{img}$, $\mathcal{L}_{story}$ and $\mathcal{L}_{KL}$ respectively. See Appendix for details on the loss terms. An overview of our model architecture can be seen in Figure 2.

MART Context Encoder. One of the primary challenges of story visualization is maintaining consistent background imagery and character appearances throughout the story. This is addressed with a recurrent context encoder which has access to the global narrative while encoding the caption in each time-step. We use the Memory Augmented Recurrent Transformer (MART) (Lei et al., 2020), where the memory is initialized with the conditioning vector h_0 from the story encoder. It takes word embeddings $W_k = [w_{k1}, w_{k2}, ...w_{kL}]$ where $w_{ij} \in \mathcal{R}^{1 \times d_w}$, corresponding to the frame caption at each timestep and produces contextualized embeddings which are then pooled to a single weighted representation c_k using attention. This allows the context encoder to capture sophisticated interactions among the words which the image generator can then capitalize on:

$$[m_{k1}, ...m_{kL}], h_k = \text{MART}([w_{k1}, ...w_{kL}], h_{k-1})$$

$$c_k = \sum_{i=1}^{L} \alpha_{ki} m_{ki};\ \alpha_{ki} = \frac{exp(m_{ki}^T u)}{\sum exp(m_{ki}^T u)}$$

where u is a query vector learned during training. The Transformer encoder is followed by a layer of GRU cells that take the contextualized embedding as input along with isometric Gaussian noise, ϵ_k, and produce an output vector g_k. The outputs c_k and g_k are concatenated and transformed into filters, and subjected to convolution with a projection of the sentence embedding s_k, resulting in output vector o_k. See Appendix for more details.

Image Generator. The image generator follows prior text-to-image generation approaches (Qiao et al., 2019; Xu et al., 2018; Zhang et al., 2017) and uses a two-stage approach. The first stage uses outputs o_k; the resulting image is fed through a second stage, which aligns the contextualized word encodings m_k from MART with image subregions generated in first-stage and reuses weighted encodings for image refinement.

Dual Learning via Video Redescription. Dual learning provides the model with an additional learning signal by taking advantage of the duality of certain tasks, i.e., if X can be used to produce Y, then Y can be used to produce X. Here, our primary task is story visualization, and we consider the secondary task of video captioning. We refer to this process as video redescription. To execute the idea of learning via video redescription, we employ a video captioning network which takes the sequence of generated images and produces a corresponding sequence of captions. The video captioning network is based on a recurrent encoder-decoder framework ($V_{enc}(.)$, $V_{dec}(.)$) and is trained using a cross-entropy loss on the predicted probability distribution (p) over its vocabulary. Specifically, $\mathcal{L}_{dual} = \sum_{k=1}^{T} \sum_{i=1}^{L} \text{log} p_{ki}(w_{ki})$. The hidden state in recurrent model helps the captioning network to identify narrative elements in the sequence of images and penalize the generative model for a lack of consistency in addition to semantic misalignment. We pretrain the video captioning network using ground truth data and freeze its parameters while training the generative model. We also include a multiplier, λ_{dual}, which allows us to scale the loss. The implementation of the encoder-decoder framework can vary. For our primary model, we adapt the MART video captioning network (Lei et al., 2020) to accept a 2D matrix of features at each time step where each column corresponds to an image sub-region (see Sec. 5).

Sequentially-Consistent Story Visualization. While certain components, such as character positions, will change from frame to frame, there are other components like background and appearances which usually carry over to adjacent frames. To take advantage of this continuity, we augment the model with a copy-transform mechanism. This mechanism can take into consideration the generated image from previous timesteps, and reuse aspects of those prior images during the current timestep. The copy-transform module $F^{copy}(.)$ performs attention-based semantic alignment (Xu et al., 2018) between word features $m_k \in \mathcal{R}^{D_w \times L}$ in the current timestep and image features $i_{k-1} \in \mathcal{R}^{D_i \times N}$ from previous step. Each column of i_{k-1} is a feature vector of a sub-region of the image. The word features are first projected into the same semantic space as image features i.e. $m'_k = Um_k$, where $U \in \mathcal{R}^{D' \times D}$. For the j^{th} image sub-region, the word-context vector is calculated as:

$$c_{jk} = \sum_{i=0}^{L} \beta_{ji} m'_{ik}; \;\; \beta_{jik} = \frac{\exp(h_j^T m'_{ik})}{\sum_{i=0}^{L} \exp(h_j^T m'_{ik})}$$

β_{jik} indicates the weight assigned by the model to the i^{th} word when generating the j^{th} sub-region of the image. The weighted word-context matrix is then concatenated with the generative image features from the current timestep and sent for upsampling to the image generator.

Objective. Bringing it all together, the final objective function of the generative model is:

$$\min_{\theta_G} \max_{\theta_I, \theta_S} \mathcal{L}_{KL} + \mathcal{L}_{img} + \mathcal{L}_{story} + \lambda_{dual}\mathcal{L}_{dual}$$

where θ_G, θ_I and θ_S denote the parameters of the entire generator, and image and story discriminator respectively. See Appendix for more details.

4 Experiments

Dataset. We utilize the Pororo-SV dataset from the original StoryGAN paper which has been adapted from a video QA dataset based on animated series (Li et al., 2019b)[2]. Each sample in Pororo-SV contains 5 consecutive pairs of frames and captions. The original splits of Pororo-SV from Li et al. (2019b) contain only training and test splits with nearly 80% overlap in individual frames. For a more challenging evaluation, we use the test split proposed in (Li et al., 2019b) as validation split (2,334 samples) and carve out an "unseen" test split from the training examples. The resulting dataset contains 10191, 2334 and 2208 samples in training, validation and test splits respectively. In this version, there is 58% frame overlap between the validation and train splits and 517 samples in the validation split contain at least one frame which is not present in the training set. Conversely, the test split has zero overlap with the training split.

Experimental Settings. Our model is developed using PyTorch, building off of the original StoryGAN codebase. All models are trained on the proposed training split and evaluated on validation and test sets. We select the best checkpoints and tune hyperparameters by using the character classification F-Score on validation set (see Appendix).

5 Evaluation of Visual Story Generation

As with any task, evaluation is a critical component of story visualization; however, due to the complexity of the task and its generative nature, evaluation is nontrivial. For instance, characters are the focal point of any narrative and similarly should be the focus of a model when producing images for the story. Hence, Li et al. (2019b) measure the character classification accuracy within frames of generated visual stories in order to compare models. However, it is also important that the characters and background are consistent in appearance, and together form a cohesive story rather than an independent set of frames. Inspired by insights such as this, we explore an additional set of evaluation metrics that capture diverse aspects of a model's performance on visual story generation.

Character Classification. We finetune the pretrained Inception-v3 (Szegedy et al., 2016) with a multi-label classification loss to identify characters in the generated image. Most earlier work in story visualization report the image-level exact-match (EM) character classification accuracy. However,

[2] We opt to not use the CLEVR-SV dataset as we believe that this dataset lacks a narrative structure and is not suitable for story visualization.

we contend that the exact match accuracy is not sufficient to gauge the performance of generative models, and the micro-averaged F-score of character classification should also be reported. For example, if Model A generates one of two characters in a frame with better quality than Model B (which generates none), it results in the same EM accuracy as Model B but an improvement in the recall/F-Score of the model, making the latter more reliable as a metric for quality. Our conclusion is based on the observation of consistent improvement in character classification scores with increasing training epochs and manual evaluation of image quality (see Fig 4).

Video Captioning Accuracy. In order to measure global semantic alignment between captions and generated visualizations, we propose to use video captioning models which have been pretrained on ground truth data to identify narrative elements in a sequence of frames. We use the Memory-Augmented Recurrent Model proposed in Lei et al. (2020) and add a CNN encoder (Sharma et al., 2018) on top of the Transformer encoder to extract image embeddings. The final convolutional layer (`Mixed_7c`) in finetuned Inception-v3 is used to extract a local feature matrix $f \in \mathcal{R}^{64\times2048}$ (reshaped from $2048 \times 8 \times 8$) for each image in the story. We then use this trained video captioning model to caption the generated frames. The generated captions are compared to the ground truth captions via BLEU evaluation[3], and this functions as our proposed metric for measuring global semantic-alignment between the captions and generated story. This pretrained model is also used as the video captioning dual network during training of DuCo-StoryGAN.

Discriminative Evaluation. Generative metrics such as BLEU are known to be noisy and unreliable. Hence, we also develop a discriminative evaluation setup. In order to compute similarity between generated image and ground truth, we compare the feature representations from either images in this discriminative setup. The training dataset for story visualization may contain one or more frames with the exact set of characters that are referenced in captions in the evaluation data. When we are checking for the presence of these characters in a generated image, we do not want to reward the model for copying the exact same frame from the training set instead of generating a frame suited to the input caption. In order to evaluate this consistency, we propose discriminative evaluation of the story visualization model. Using the character annotations for the final frame of each sample in the test splits, we extract a set of 4 negative frames which are taken from elsewhere in the video but contain those specific characters (see Fig. 7 in Appendix). The human evaluation accuracy on this dataset is 89% (κ=0.86) and is used as an upper bound when interpreting model accuracy performance. The cosine similarity between Inception-v3 features of final generated frame and candidate frames is computed and the frame with most similarity is selected as predicted frame. We report Top-1/2 accuracies.

R-Precision. Several prior works on text-to-image generation report the retrieval-based metric R-Precision (Xu et al., 2018) for quantifying the semantic alignment between the input text and generated image. If there are R relevant documents for a query, the top R ranked retrieval results of a system are examined; if r are relevant, the R-precision is r/R. In our task[4], $R = 1$. The encodings from a pretrained Deep Attention-based Multimodal Similarity Model (DAMSM) are used to compute cosine similarity and rank results. Since this model only evaluates a single text-image pair for similarity, it is not suitable for evaluating story visualization. Therefore, we train a new version of DAMSM to extract global representations for the story and sequence of images, referred to as Hierarchical DAMSM (H-DAMSM) (see Appendix).

The models used in the aforementioned evaluation metrics are trained independently of DuCo-StoryGAN on the proposed Pororo-SV splits and the pretrained weights are used for evaluation. See Appendix for other upper bounds.

6 Results

6.1 Main Quantitative Results

The results for Pororo-SV validation set can be seen in Table 1. The first row contains the results using the original StoryGAN model (Li et al., 2019b)[5]. The second row functions as another

[3] We use the `nlg-eval` package (Sharma et al., 2017) for BLEU evaluation.

[4] The R-precision score is obtained from 10 runs with 99 randomly picked mismatched story candidates in each run.

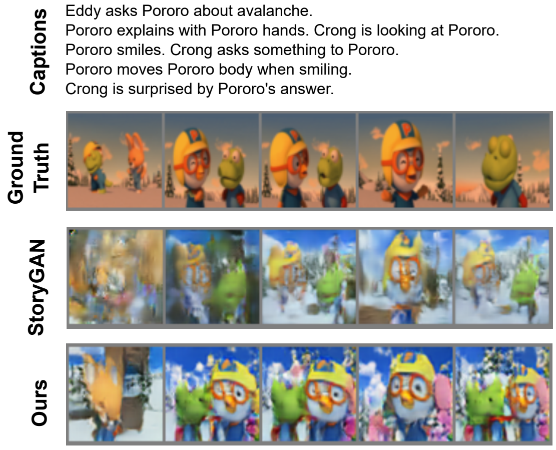
[5] We use a reduced training dataset as compared to the original StoryGAN paper (see Sec 4). However, we evaluate our StoryGAN code base on their exact splits and get 26.1% exact-match accuracy, which is approximately equivalent to the 27% reported in the original paper where they demonstrate that StoryGAN outperforms previous baselines such as ImageGAN, SVC, and SVFN.

Model	Char. F1	BLEU2/3	R-Precision	Frame Acc.	Top-1 Acc.	Top-2 Acc.
StoryGAN (Li et al., 2019b)	41.11	3.86 / 1.72	3.40 ± 0.01	21.90	22.42	45.40
StoryGAN + Transformer	42.45	3.92 / 1.73	4.03 ± 0.17	22.14	23.79	47.15
CP-CSV (Song et al., 2020)	43.79	3.96 / 1.73	3.97 ± 0.21	22.08	24.29	46.39
DuCo-StoryGAN	**48.27**	**4.51 / 1.92**	**6.10 ± 0.07**	**22.71**	**25.62**	**47.39**

Table 1: Results on validation split of Pororo-SV Dataset.

Model	Char. F1	BLEU2/3	R-Precision	Frame Acc.	Top-1 Acc.	Top-2 Acc.
StoryGAN (Li et al., 2019b)	18.59	3.24 / 1.22	1.51 ± 0.15	9.34	23.14	42.27
StoryGAN + Transformer	19.29	3.29 / 1.23	1.49 ± 0.07	9.58	23.31	42.29
CP-CSV (Song et al., 2020)	21.78	3.25 / 1.22	1.76 ± 0.04	10.03	22.23	41.86
DuCo-StoryGAN	**38.01**	**3.68 / 1.34**	**3.56 ± 0.04**	**13.97**	**23.72**	**42.48**

Table 2: Results on test split of Pororo-SV Dataset.



Eddy asks Pororo about avalanche.
Pororo explains with Pororo hands. Crong is looking at Pororo.
Pororo smiles. Crong asks something to Pororo.
Pororo moves Pororo body when smiling.
Crong is surprised by Pororo's answer.

Figure 3: Sample results from StoryGAN and DuCo-StoryGAN on unseen test split.

baseline, where we replace the GRU-based context encoder in StoryGAN with a Bidirectional Transformer (Devlin et al., 2019). The conditioning augmentation vector is not used to initialize the context encoder in this model since a non-recurrent Transformer lacks a hidden state. We see 1-2% improvements in character classification and retrieval with this model over StoryGAN. The third row contains results from the more recent CP-CSV model (Song et al., 2020) which uses figure-ground segmentation as an auxiliary task for preserving character features. Consequently, it results in 2.68% improvement in character classification over StoryGAN and smaller improvements for other metrics. The final row contains results with DuCo-StoryGAN, which significantly outperforms previous models (including CP-CSV) across all metrics. The character classification F-Score improves by 7.16% suggesting that the characters generated in our images are of higher visual quality. Similarly, we see consistent improvements in BLEU as well as R-Precision with our model. As demonstrated in Sec 7.1, the improvement in BLEU can be attributed to the addition of dual learning, which directly optimizes the dual task of video captioning. The R-Precision indicates that our model learns better global semantic alignment between the captions and images. Lastly, the Top-1/2 accuracy scores show that our model is learning to generate diverse images, rather than copying scenes that feature the same characters from the training data.

DuCo-StoryGAN performs dramatically better than other models on the unseen test split (see Table 2). As can be seen in Fig 3, StoryGAN performs rather poorly on unseen samples compared to DuCo-StoryGAN. While the former produces images that are blurry and character shapes that are faint, the latter generates frames with sharp character features. This is reflected in the wide improvement margins on character classification scores in Table 2. Similar improvements are also observed for BLEU and R-Precision metrics, indicating that our model generates images which are more relevant to the input caption. When generating stories for the Pororo-SV test split, models tend to copy background elements from the samples seen in the training set, since the captions lack sufficient information about the setting. Hence, we observe little improvement over random chance in the discriminative accuracy scores for different models on test split. For instance, instead of generating the tinted background in ground truth in Fig. 3, the models produce a clear blue sky which is closer to samples seen in the training set. However, discriminative evaluation will be valuable for future work in this domain when inputs contain detailed information about the visual elements.

We also provide per character results for the Character F-Score. With DuCo-StoryGAN, we see up to 20% improvement for less frequent char-

	Win %		Mean Rating	
Attribute	Ours	StGAN	Ours	StGAN
Visual Quality	**82%**	3%	**2.06**	1.22
Consistency	**78%**	3%	**2.94**	1.78
Relevance	**26%**	2%	**1.28**	1.04

Table 3: Human evaluation on Likert Scale 1-5. Win% = % times stories from one model was preferred over the other (StGAN = StoryGAN). Tie% = % samples remaining after considering Win% of both models.

acters (see Table 6).

6.2 Human Evaluation

We conduct human evaluation on the generated images from DuCo-StoryGAN and StoryGAN, using the three evaluation criteria listed in Li et al. (2019b): visual quality, consistence, and relevance. Two annotators are presented with a caption and the generated sequence of images from both models, and asked to rate each sequence on a scale of 1-5. Results are presented in Table 3. With respect to pairwise evaluation, predictions from our model is nearly always preferred over those from StoryGAN (see Win% columns). Similarly, we see large improvements in mean rating of stories generated by DuCo-StoryGAN. However, we also see higher Tie% and low mean rating for the attribute Relevance, suggesting that much work remains to be done to improve understanding of captions.

Correlation Experiments: We also examine the correlation between our proposed metrics and human evaluation of generated images. We compute the Pearson's correlation coefficient between human ratings of 50 samples on three different attributes using the 1-5 Likert scale and their corresponding automated metric evaluation scores. Significant correlation ($\rho = 0.586$) was observed between our proposed Character F-Score metric and Visual Quality, lending strength to its use an automated metric for story visualization.

7 Discussion

7.1 Ablations

Table 4 contains plus-one ablations for DuCo-StoryGAN. The first row is the StoryGAN baseline and the second row is the StoryGAN + Transformer model, as discussed in Section 6. We then iteratively add each of our contributions and observe the change in metrics[6]. First, we upgrade the Transformer encoder to MART, which brings about the largest improvements across all metrics. The use of word embeddings with access to global conditioning vector and attention-based semantic alignment proves important to the task of story generation. Next, we use the MART context encoder with our proposed dual learning and copy-transform improvements. With the addition of video captioning as a learning signal, we see 0.20% (p=0.071) improvement in character F-score and 1.12% improvement in R-Precision (p=0.032) over MART. The highest improvements are observed for BLEU score, since the model is optimized on video captioning. Next, we evaluate the addition of the copy-transform mechanism where features from generated images in previous timesteps are copied to the image in current timestep. We observe 1.04% improvements for character classification and a slight drop in performance on video captioning. Similarly, there is 1.14% improvement in Top-1 accuracy for the discriminative dataset.

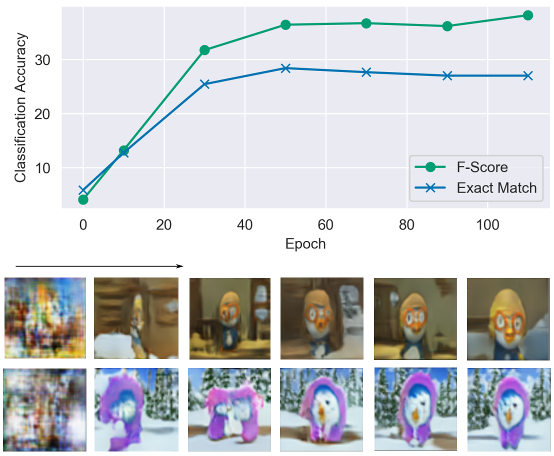


Figure 4: Progression of character classification scores (top) and generated images (bottom) with training.

As discussed in Section 3, we explore a variety of implementations for the dual learning component of our model. While MART-based video captioning works the best, we provide a discussion of other approaches in the Appendix.

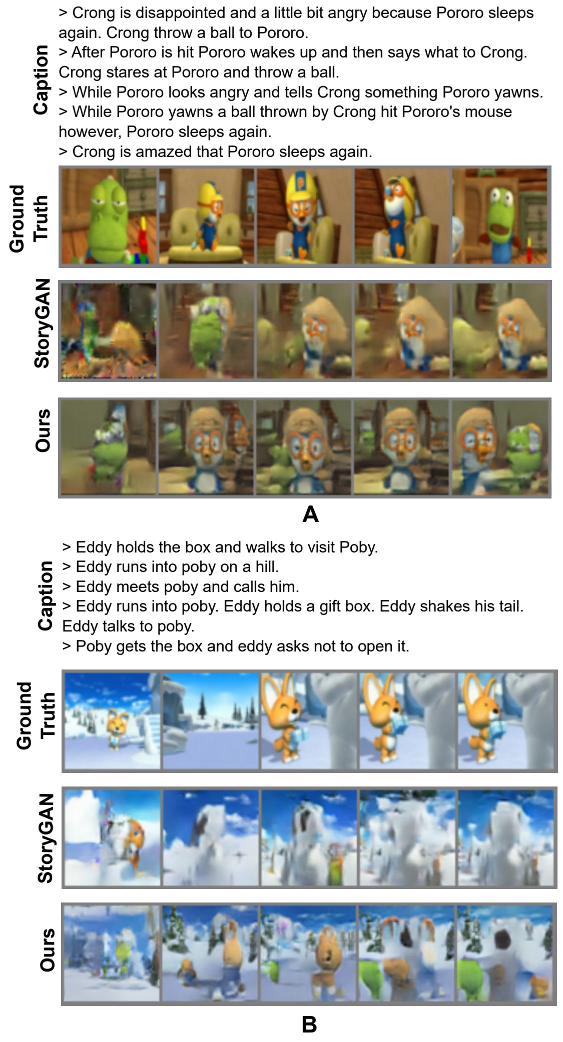
7.2 Qualitative Examples

Figure 5 contains two generated examples from the Pororo-SV dataset. The top row in each example contains the ground truth images, the middle row the images generated by StoryGAN, and the final row the images generated by our model. In

[6] Statistical significance is computed with 100K samples using bootstrap (Noreen, 1989; Tibshirani and Efron, 1993). All our improvements in DuCo-StoryGAN are statistically significant, except for discriminative evaluation and frame accuracy scores for the dual learning module.

Model	Char. F1	BLEU2/3	R-Precision	Frame Acc.	Top-1 Acc.	Top-2 Acc.
StoryGAN (Li et al., 2019b)	41.11	3.86 / 1.72	3.40 ± 0.01	21.90	22.42	45.40
StoryGAN + Transformer	42.45	3.92 / 1.73	4.03 ± 0.17	22.14	23.79	46.15
StoryGAN + MART	47.03	4.15 / 1.81	5.11 ± 0.12	22.25	24.48	46.42
+ Story Captioning	47.23	4.78 / 1.87	6.32 ± 0.08	22.30	24.53	47.41
+ Copy Transform	48.27	4.51 / 1.92	6.10 ± 0.07	22.71	25.62	47.39

Table 4: Ablation results on validation split of Pororo-SV dataset.


> Crong is disappointed and a little bit angry because Pororo sleeps again. Crong throw a ball to Pororo.
> After Pororo is hit Pororo wakes up and then says what to Crong. Crong stares at Pororo and throw a ball.
> While Pororo looks angry and tells Crong something Pororo yawns.
> While Pororo yawns a ball thrown by Crong hit Pororo's mouse however, Pororo sleeps again.
> Crong is amazed that Pororo sleeps again.

> Eddy holds the box and walks to visit Poby.
> Eddy runs into poby on a hill.
> Eddy meets poby and calls him.
> Eddy runs into poby. Eddy holds a gift box. Eddy shakes his tail. Eddy talks to poby.
> Poby gets the box and eddy asks not to open it.

Figure 5: Comparative examples of generated images.

example A, we demonstrate the superior visual quality and consistency of the frames generated by DuCo-StoryGAN, as compared to StoryGAN. The MART encoder allows our model to comprehend long captions as well as attend to each word while generating images. The retention of native character features throughout the story during regeneration can be attributed to the copy-transform mechanism in our model. In contrast, we see that both models fail at generating defined characters in example B. This may be due to the fact that *Poby* is an infrequent character in the dataset and hence, both models fail to learn its features.

7.3 Linguistic Analysis

We perform visual analysis of the captions and predictions from DuCo-StoryGAN and observe two major recurring themes. First, the frequency of characters in the training data is a significant deciding factor for generated image quality. We looked at the samples that contained at least *Pororo* (most frequent character) and found that generated stories are better when there is only a single character in the frame's narrative as compared to multiple characters. This points to the inability of current story visualization models to align captions with multiple subjects/objects to the corresponding images. Second, generated images are poor for scenes containing infrequently occurring objects such as book/plate/boat/plane etc. in the caption. This behavior is expected since the model is unaware of real-world objects that do not already appear in the training set with sufficient frequency. Moreover, since the Pororo-SV dataset has been adapted from the annotations of a video QA dataset, the captions often contain information that can only span over multiple frames ("Pororo wakes up and then says what to Crong. Pororo stares at Pororo and throws a ball"), or cannot be visualized through images ("Poby gets the box and Eddy asks not to open it."). Hence, our results with metrics like BLEU and R-Precision which are supposed to capture the relevance between images and caption stay relatively low (see Tables 1 and 2).

8 Conclusion

In this paper, we investigate the underexplored task of story visualization. We improve upon prior modeling approaches and demonstrate the effectiveness of these new approaches by performing a robust set of ablation experiments. We also present a detailed set of novel evaluation methods, which we validate by demonstrating improvements across various baselines. Evaluation for story visualization is a challenging open research question in itself, and we hope that these methods will encourage more work in this domain.

9 Ethics/Broader Impacts

From an ethics standpoint, we provide a brief overview of the data that the model is trained on in Section 4 and a more detailed discussion in the Appendix. We provide some analyses of the data and refer the reader to the original StoryGAN paper, where the dataset was created, for further details. All of the language data consists of simple English sentences. Our experimental results are specific to the story visualization task. Pororo-SV is the most challenging story visualization task available; therefore, our results would likely generalize to other story visualization datasets. While story visualization is an exciting task with many potential future applications, the generated images still contain many obvious visual artifacts and therefore models trained on this task are still far from being deployed in any real world settings.

Story visualization minimizes many of the ethical issues associated with image and video generation. DeepFakes, which are algorithmically generated fake images, have become increasingly problematic (Nguyen et al., 2019). Oftentimes, these images are indistinguishable from real images, raising privacy concerns and providing a source of misinformation. The images that we generate here are not subject to this same issue, due to the fact that they are Cartoons, and are therefore unable to be confused with real images. The focus of the task is not on the realism of the images, but rather on the multimodal narrative. Therefore, cartoons are actually better suited for the task as real-world images only add additional visual complexity that is not relevant to the narrative.

Acknowledgments

We thank Peter Hase, Jaemin Cho, Hyounghun Kim, and the reviewers for their useful feedback. This work was supported by DARPA MCS Grant #N66001-19-2-4031, DARPA KAIROS Grant #FA8750-19-2-1004, ARO-YIP Award W911NF-18-1-0336, and a Google Focused Research Award. The views are those of the authors and not of the funding agency.

References

Yogesh Balaji, Martin Renqiang Min, Bing Bai, Rama Chellappa, and Hans Peter Graf. 2019. Conditional gan with discriminative filter generation for text-to-video synthesis. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*, pages 1995–2001. International Joint Conferences on Artificial Intelligence Organization.

Jianbo Chen, Yelong Shen, Jianfeng Gao, Jingjing Liu, and Xiaodong Liu. 2018. Language-based image editing with recurrent attentive models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8721–8729.

Yu Cheng, Zhe Gan, Yitong Li, Jingjing Liu, and Jianfeng Gao. 2020. Sequential attention gan for interactive image editing. In *ACM MM*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT (1)*.

Alaaeldin El-Nouby, Shikhar Sharma, Hannes Schulz, Devon Hjelm, Layla El Asri, Samira Ebrahimi Kahou, Yoshua Bengio, and Graham W Taylor. 2019. Tell, draw, and repeat: Generating and modifying images based on continual linguistic instruction. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 10304–10312.

Lianli Gao, Daiyuan Chen, Jingkuan Song, Xing Xu, Dongxiang Zhang, and Heng Tao Shen. 2019. Perceptual pyramid adversarial networks for text-to-image synthesis. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 8312–8319.

Tanmay Gupta, Dustin Schwenk, Ali Farhadi, Derek Hoiem, and Aniruddha Kembhavi. 2018. Imagine this! scripts to compositions to videos. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 598–613.

Qiuyuan Huang, Pengchuan Zhang, Dapeng Wu, and Lei Zhang. 2018. Turbo learning for captionbot and drawingbot. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, pages 6456–6466.

Kyung-Min Kim, Min-Oh Heo, Seong-Ho Choi, and Byoung-Tak Zhang. 2017. Deepstory: Video story qa by deep embedded memory networks. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17*, pages 2016–2022.

Gierad P. Laput, Mira Dontcheva, Gregg Wilensky, Walter Chang, Aseem Agarwala, Jason Linder, and Eytan Adar. 2013. Pixeltone: A multimodal interface for image editing. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '13, page 2185–2194, New York, NY, USA. Association for Computing Machinery.

Jie Lei, Liwei Wang, Yelong Shen, Dong Yu, Tamara L Berg, and Mohit Bansal. 2020. Mart: Memory-augmented recurrent transformer for coherent video paragraph captioning. *arXiv preprint arXiv:2005.05402*.

Chunye Li, Liya Kong, and Zhiping Zhou. 2020. Improved-storygan for sequential images visualization. *Journal of Visual Communication and Image Representation*, 73:102956.

Wenbo Li, Pengchuan Zhang, Lei Zhang, Qiuyuan Huang, Xiaodong He, Siwei Lyu, and Jianfeng Gao. 2019a. Object-driven text-to-image synthesis via adversarial training. In *CVPR*.

Yitong Li, Zhe Gan, Yelong Shen, Jingjing Liu, Yu Cheng, Yuexin Wu, Lawrence Carin, David Carlson, and Jianfeng Gao. 2019b. Storygan: A sequential conditional gan for story visualization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6329–6338.

Yitong Li, Martin Renqiang Min, Dinghan Shen, David E Carlson, and Lawrence Carin. 2018. Video generation from text. In *AAAI*.

Yahui Liu, Marco De Nadai, Deng Cai, Huayang Li, and Bruno Lepri Xavier Alameda-Pineda and, Nicu Sebe and. 2020. Describe What to Change: a text-guided unsupervised image-to-image translation approach. In *ACM MM*.

Ramesh Manuvinakurike, Jacqueline Brixey, Trung Bui, Walter Chang, Doo Soon Kim, Ron Artstein, and Kallirroi Georgila. 2018a. Edit me: A Corpus and a Framework for Understanding Natural Language Image Editing. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Ramesh Manuvinakurike, Trung Bui, Walter Chang, and Kallirroi Georgila. 2018b. Conversational image editing: Incremental intent identification in a new dialogue task. In *Proceedings of the 19th Annual SIGdial Meeting on Discourse and Dialogue*, pages 284–295.

Seonghyeon Nam, Yunji Kim, and Seon Joo Kim. 2018. Text-adaptive generative adversarial networks: Manipulating images with natural language. In *Advances in neural information processing systems*, pages 42–51.

Thanh Thi Nguyen, Cuong M Nguyen, Dung Tien Nguyen, Duc Thanh Nguyen, and Saeid Nahavandi. 2019. Deep learning for deepfakes creation and detection. *arXiv preprint arXiv:1909.11573*, 1.

Eric W Noreen. 1989. *Computer-intensive methods for testing hypotheses*. Wiley New York.

Yingwei Pan, Zhaofan Qiu, Ting Yao, Houqiang Li, and Tao Mei. 2017. To create what you tell: Generating videos from captions. In *ACM International Conference on Multimedia (ACM Multimedia)*.

Tingting Qiao, Jing Zhang, Duanqing Xu, and Dacheng Tao. 2019. Mirrorgan: Learning text-to-image generation by redescription. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.

Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. 2018. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2556–2565.

Shikhar Sharma, Layla El Asri, Hannes Schulz, and Jeremie Zumer. 2017. Relevance of unsupervised metrics in task-oriented dialogue for evaluating natural language generation. *CoRR*, abs/1706.09799.

Jing Shi, Ning Xu, Trung Bui, Franck Dernoncourt, Zheng Wen, and Chenliang Xu. 2020. A benchmark and baseline for language-driven image editing. In *ACCV*.

Yun-Zhu Song, Zhi-Rui Tam, Hung-Jen Chen, Huiao-Han Lu, and Hong-Han Shuai. 2020. Character-preserving coherent story visualization. In *Proceedings of the European Conference on Computer Vision (ECCV)*.

C Szegedy, V Vanhoucke, S Ioffe, J Shlens, and Z Wojna. 2016. Rethinking the inception architecture for computer vision. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2818–2826.

Robert J Tibshirani and Bradley Efron. 1993. An introduction to the bootstrap. *Monographs on statistics and applied probability*, 57:1–436.

Ziteng Wen, Linbo Xie, Hongwei Feng, and Yong Tan. 2019. Robust fusion algorithm based on rbf neural network with ts fuzzy model and its application to infrared flame detection problem. *Applied Soft Computing*, 76:251 – 264.

Tao Xu, Pengchuan Zhang, Qiuyuan Huang, Han Zhang, Zhe Gan, Xiaolei Huang, and Xiaodong He. 2018. Attngan: Fine-grained text to image generation with attentional generative adversarial networks. In *CVPR 2018*.

Zili Yi, Hao Zhang, Ping Tan, and Minglun Gong. 2017. Dualgan: Unsupervised dual learning for image-to-image translation. In *Proceedings of the IEEE international conference on computer vision*, pages 2849–2857.

Gangyan Zeng, Zhaohui Li, and Yuan Zhang. 2019. Pororogan: An improved story visualization model on pororo-sv dataset. In *Proceedings of the 2019 3rd International Conference on Computer Science and Artificial Intelligence*, pages 155–159.

Han Zhang, Tao Xu, Hongsheng Li, Shaoting Zhang, Xiaogang Wang, Xiaolei Huang, and Dimitris Metaxas. 2017. Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks. In *ICCV*.

Minfeng Zhu, Pingbo Pan, Wei Chen, and Yi Yang. 2019. DM-GAN: dynamic memory generative adversarial networks for text-to-image synthesis. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 5802–5810.

Appendices

A Methods

StoryGAN only uses pretrained sentence embeddings as a representation for the caption, while DuCo-StoryGAN uses a combination of sentence and word embeddings. The context encoder is responsible for encoding the captions and transforming them into image embeddings.

Story Encoder. The story encoder $E(.)$ encodes the entire story, S into a single representation, h_0, which functions as the initial memory state of the MART model. The input S is the concatenation of sentence embeddings $s_k \in \mathcal{R}^{1\times d_s}$ from all timesteps. The conditional augmentation technique (Zhang et al., 2017) is used to convert S into a conditioning vector by using it to construct and sample a conditional Gaussian distribution i.e., $h_0 = \mu(S) + \sigma^2(S)^{1/2} \odot \epsilon_S$, where $\epsilon_S \sim \mathcal{N}(0,1)$ and $\odot$ represents element-wise multiplication. This introduces a loss term which is the KL-Divergence between learned distribution and Gaussian distribution i.e.,

$$\mathcal{L}_{KL} = KL(\mathcal{N}(\mu(S), \text{diag}(\sigma^2(S)))||\mathcal{N}(0, I))$$

Discriminators. There are two discriminators in the model, each aimed at capturing a different aspect of the task. The image discriminator focuses on local consistency and is provided with the generated image $\hat{x}_k$, the sentence s_k, and the context information vector from the story encoder h_0, and must attempt to distinguish between this and a real triplet, containing the same information except for the real image x_k instead of the fake image ($\mathcal{L}_{img}$). Additionally, the image discriminator is also used to classify the characters in the frame, when labels are available. The story discriminator is instead concerned with the global consistency of the generated sequence. The generated image sequence $\hat{X}$ and story S are provided to the discriminator, which must distinguish it from an equivalently encoded real pair ($\mathcal{L}_{story}$)

MART Context Encoder The MART encoder, as described in the main paper, is followed by a layer of GRU cells that take the contextualized embedding as input along with isometric Gaussian noise, ϵ_k, and produce an output vector g_k. The outputs c_k and g_k are concatenated and transformed into filters, and subjected to convolution with a projection of the sentence embedding s_k i.e.

$$g_k,\ q_k = \text{GRU}(s_k, \epsilon_k, q_{k-1})$$
$$o_k = Filter([c_k; g_k]) \circ \tanh(W_I s_k)$$

where q_k is the hidden state of the GRU cells. $Filter(.)$ transforms the concatenated vector $[c_k; g_k]$ into a multi-channel filter of size $C_{out} \times 1 \times 1 \times len(W_I s_k)$, where C_{out} is the number of output channels. The convolution operation can be interpreted as the sifting of information from local context s_t with the use of filters that have access to the global context.

B GAN Training

The training procedure for our GAN architecture is similar to StoryGAN. The objective function for the generative model is:

$$\min_{\theta_G} \max_{\theta_I, \theta_S} \mathcal{L}_{KL} + \mathcal{L}_{img} + \mathcal{L}_{story} + \lambda_{dual}\mathcal{L}_{dual}$$

where θ_G is the parameters of the generator, θ_I is the parameters for the image discriminator, and θ_S is the parameters for the story discriminator. Note that the video captioning dual learning component is pretrained and then frozen while the rest of the model is trained.

Each of the components in the model has a conditional loss, which is concerned with whether the input caption and generated image align. The adversarial loss function for the generator is then as follows:

$$\mathcal{L}_{G_i} = -\frac{1}{2}\text{E}_{\hat{x}_i \sim p_{\hat{x}_i}}[log(D_{img}(\hat{x}_i, s))] - \frac{1}{2}\text{E}_{\hat{X}_i \sim p_{\hat{X}_i}}[log(D_{story}(\hat{X}_i, S))]$$

where $\hat{x}_i$ is the generated image sampled from the distribution $p_{\hat{x}_i}$ during the i^{th} stage of generation. The first term is the conditional loss of the image discriminator, and the second term is the conditional loss for the story discriminator.

The adversarial losses for the discriminators are:

$$\mathcal{L}_{D_{img}} = -\frac{1}{2}\text{E}_{x_i \sim p_{x_i}}[log(D_{img}(x_i, s))] - \frac{1}{2}\text{E}_{\hat{x}_i \sim p_{\hat{x}_i}}[log(1 - D_{img}(\hat{x}_i, s))]$$

$$\mathcal{L}_{D_{story}} = -\frac{1}{2}\mathrm{E}_{X_i \sim p_{X_i}}[log(D_{story}(X_i, S))]$$
$$-\frac{1}{2}\mathrm{E}_{\hat{X}_i \sim p_{\hat{X}_i}}[log(1 - D_{story}(\hat{X}_i, S))]$$

where $\hat{x}_i$ is the generated image sampled from the distribution $p_{\hat{x}_i}$, and x_i is the real image sampled from the distribution p_{x_i}, during the i^{th} stage of generation.

For additional algorithmic details we refer readers to Li et al. (2019b).

C Experimental Settings

Our model is constructed using PyTorch, building off of the original StoryGAN codebase. All models are trained on the training set, tuned on the development set, and evaluated on the test set. We report results for each of the latter. We select the best checkpoints and manual tune hyperparameters for each model by using the validation character classification F-Score. We use the ADAM optimizer with betas of 0.5 and 0.999. We train the model on a single Nvidia 2080TI GPU. Each epoch takes 30 minutes, with the model being saved every 10 epochs. At 120 epochs of training, the total training time is nearly 60 hours for a batch size of 4. We did 1-5 runs for hyperparameter search using manual tuning. The number of trainable parameters in our proposed DuCo-StoryGANis 101,718,981.

D Hyperparameters

Many of our hyperparameters are shared with the StoryGAN model. The image size that we use is 64-by-64, and the length of the story is 5 images/captions. The learning rate of the generator is 2e-4, while the learning rate of the discriminator is slightly lower at 1e-4. We train the model for 120 epochs and set the learning rate to decay every 20 epochs. For each training update of the discriminators, we perform two updates for the generator network, with different mini-batch sizes for image and story discriminators (Li et al., 2019b). The image discriminator batch size is 20 and the story discriminator batch size is 4. We found in our experiments that all story visualization models are susceptible to mode collapse with small changes in the discriminator learning rate. Additionally, we attempted replacing the attention-based alignment module from Xu et al. (2018) with a cross-attention layer and observed mode collapse in later epochs for the first generated frame in the story. We also used an update ratio of 3:1 for generator vs. discriminator and did not find it useful.

The MART hyperparameters are as follows. The hidden size of the model is 192. The number of memory cells is 3. The number of hidden layers is 2. The dropout values across the model are 0.1. The layer normalization epsilon is 1e-12. The number of attention heads is 6. The word embedding size is 300, and the embedding is initialized using the 840B glove training checkpoint.

E PororoSV Dataset

We utilize the Pororo-SV dataset from the original StoryGAN paper (Li et al., 2019b). This dataset was originally a video QA dataset (Kim et al., 2017), consisting of one second video clips paired with multiple descriptions. A sequence of these video clips forms a story, which then has QA pairs associated with it. There are 9 characters frequently featured in the dataset; a distribution of them can be seen in the supplementary. Annotations are available for the distribution of characters in each frame. It can be seen that each character is featured in at least 10% of the frames, making it crucial for the model to be capable of generating each of them. To convert this to a story visualization task, Li et al. (2019b) sample the one second videos, obtaining a single, representative frame. Five sequential frame-description pairs are then considered to make up a single story. We use the training and test splits outlined in Li et al. (2019b) for comparable results. However, since this split is also used for tuning in both papers, we carve an equally-sized held-out split of unseen samples from the training set for fair evaluation of the models.

Character Frequency. The PororoSV dataset contains 9 characters that are frequently featured; a distribution of them can be seen in Figure 6.

F Evaluation

Video Captioning Accuracy. Video Captioning models use a sequence of image embeddings from the sequence of frames in a video segment as input and perform decoding on the processed features to produce a caption of single sentence or multiple sentences. However, they assume that there are multiple frames within a single video segment, unlike our story dataset where there is exactly one frame for each sentence in the story caption. There-

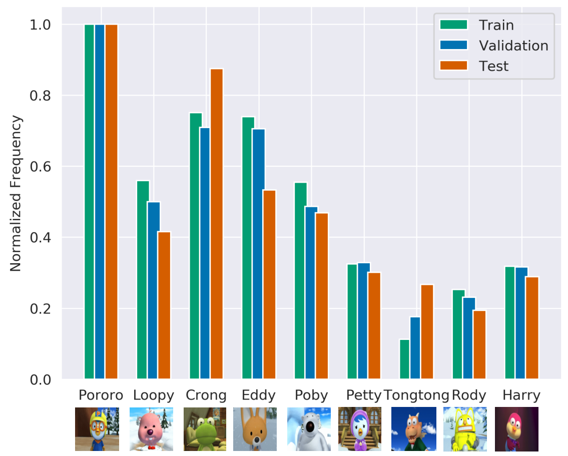


Figure 6: Distribution of PororoSV characters in various data splits.

fore, we adapt existing state-of-the-art video captioning models to perform decoding from a single frame for each sentence in the caption.

R-Precision. Several prior works on text-to-image generation report the retrieval-based metric R-Precision (Xu et al., 2018) for quantifying the semantic alignment between the input text and generated image. R-Precision is computed using the similarity between encodings extracted from a pretrained Deep Attention-based Multimodal Similarity Model (DAMSM). Since this model only evaluates a single text-image pair for similarity, it is not suitable for evaluating story visualization. Therefore, we train a new version of DAMSM to encode all text-image pairs in each story and compute the global similarity for consecutive frames from a story and their respective captions in addition to sentence and word similarity. We introduce an additional bidirectional LSTM network for encoding frame captions into a story representation and average pool the image features for individual frames to extract a global visual embedding for the story. The cosine similarity between these two vectors is used to rank the retrieval-based search between the query visualization and candidate story narratives. This improved model, referred to as Hierarchical DAMSM (H-DAMSM), is trained using two additional story-level losses $\mathcal{L}_{st0}$ and $\mathcal{L}_{st1}$ with a smoothing coefficient of γ=15, and the pretrained model is used for evaluation. We refer the reader to Xu et al. (2018) for details on the DAMSM model.

Example of Discriminative Dataset. Figure 7 shows an example from our discriminative dataset that is used in the discriminative evaluation.

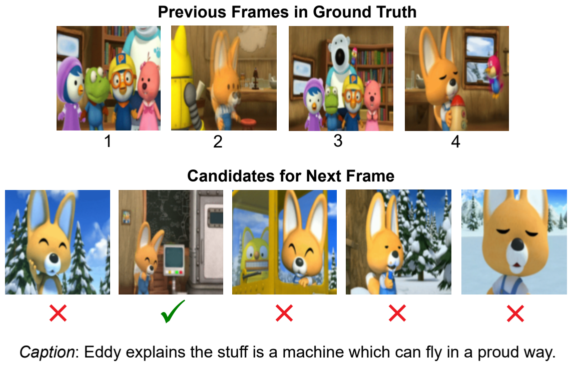

Previous Frames in Ground Truth
1
2
3
4
Candidates for Next Frame
✗
✓
✗
✗
✗
Caption: Eddy explains the stuff is a machine which can fly in a proud way.

Figure 7: Example of the Discriminative Dataset.

G Results

Dual Learning. The actual implementation of the encoder-decoder framework in our dual learning approach can vary. For our primary model, we adapt the MART architecture (Lei et al., 2020) to accept a 2D matrix of image features where each column corresponds to a sub-region, instead of a sequence of image features from adjacent frames in a video segment, for each time step (see details in the Evaluation section of the main paper). However, we compare this model with several variations of dual learning networks: (1) Transformer-based Image Captioning, (2) CNN-LSTM-based Video Captioning and (3) CNN-LSTM-based Image Captioning. The Transformer-based image captioning network is essentially a non-recurrent version of the MART-based video captioning. The CNN-LSTM based image captioning model is similar to Qiao et al. (2019). The generated image $\hat{x}_k$ is fed into a CNN, which produces a feature vector. The feature vector is then fed through an LSTM decoder, which produces the caption $\hat{s}_k$. The CNN-LSTM video-captioning model is an extension of this, using 3D convolutions to pool over all frames within a story. We pretrain these models on the Pororo-SV dataset and freeze the parameters before utlizing the weights to get the dual learning loss while training DuCo-StoryGAN.

As seen in Table 5, the image captioning approach using CNN-LSTM has a limited impact on performance. Next, we explore Transformer for implementing the captioning model and see larger improvements for character classification and BLEU scores. However, there is limited improvement in performance on R-Precision using image captioning as dual learning. We hypothesize that this is due to the image captioning model's inability to capture information across frames; essentially,

Dual Model	**Char. F1**	**BLEU2/3**	**R-Precision**	**Frame Acc.**	**Top-1 Acc.**	**Top-2 Acc.**
Image Captioning (CNN-LSTM)	47.08	4.29 / 1.83	5.23 ± 0.06	22.29	24.47	46.48
Video Captioning (CNN-LSTM)	46.19	3.98 / 1.73	4.04 ± 0.29	22.12	23.93	46.22
Image Captioning (Transformer)	47.21	4.58 / 1.81	5.37 ± 0.11	22.47	24.47	46.51
Video Captioning (Transformer)	47.23	4.78 /1.87	6.32 ± 0.08	22.30	24.53	47.41

Table 5: Results from variations of Dual Learning on Pororo-SV dataset.

Character	**Support**	**StGAN**	**TF**	**DuCoGAN**
Pororo	4400	**0.59**	0.59	0.58
Loopy	2279	0.07	0.08	**0.21**
Crong	3327	0.50	**0.51**	0.49
Eddy	3154	0.48	0.50	**0.58**
Poby	2346	0.25	0.26	**0.44**
Petty	1564	0.16	0.17	**0.49**
Tongtong	717	**0.15**	0.16	0.14
Rody	1073	0.21	0.20	**0.41**
Harry	1503	0.40	0.41	**0.42**

Table 6: Character Classification F-Scores on Pororo-SV validation set (StGAN=StoryGAN, TF=StGAN+Transformer).

Model	**Metric**	Score
Inceptionv3	Frame Acc.	41.93
	Precision	74.66
	Recall	64.12
	F-Score	68.99
	Accuracy	80.68
MART	METEOR	15.06
	ROUGE_L	18.13
	CIDEr	102.34
H-DAMSM	R-Precision	88.05 ± 0.00

Table 7: Upper Bounds of models used for Metrics on Pororo-SV validation set.

this method of dual learning is only capable of considering local consistency and not global consistency. Therefore, we use a video captioning model, where all frames are considered simultaneously, allowing it to capture both local consistency and global consistency. The performance of CNN-LSTM based video captioning model on the captioning validation set was low. Hence, using this model for dual learning loss negatively affected performance of our story visualization model. The Transformer-based image-captioning model outperforms video-captioning with CNN-LSTM, suggesting that a sophisticated dual model is as important as global context for story visualization. Consequently, the MART-based video captioning model leverages additional global context and outperforms Transformer-based image captioning across all metrics.

Individual Character Accuracy. As detailed in the Experiments section in the main paper, there are 9 characters which are featured throughout the Pororo-SV dataset. The distribution of characters varies across scenes, with some occurring more frequently than others. Using StoryGAN, Pororo, the most frequently occurring character in the dataset, has the highest F-Score, while the decrease in F-Score for other characters roughly correlates with their frequency in the data. With DuCo-StoryGAN, we saw marginal improvements for Pororo and up to 30% absolute improvement in F-Score for less frequent characters like Loopy. See Figure 6 for a detailed breakdown of each character. While this confirms the data intensive nature of story visualization, it also shows that advanced modelling approaches can alleviate the issue of data scarcity to some extent. However, models in this domain will ultimately need to be extended to more diverse datasets with more characters and settings before they can be useful in practical applications (see Introduction).

Evaluation Metric Upper Bounds. Many of the evaluation metrics that we use take advantage of other external model architectures (see Evaluation section in main paper), similar to prior work in this domain (Li et al., 2019b, 2020). Therefore, the quality of the evaluation metrics is contingent upon the accuracy of these models. Table 7 contains the upper bound results for these models on the Pororo-SV dataset. The finetuned Inceptionv3 model achieves high overall accuracy i.e. more than 85% on validation and test sets. Video captioning model MART achieves high scores on the Pororo-SV validation set for several NLG metrics. The H-DAMSM model achieves 88.05% R-precision on the validation set.

More Generated Examples. Figure 8 contains additional examples that our DuCo-StoryGAN model generated.

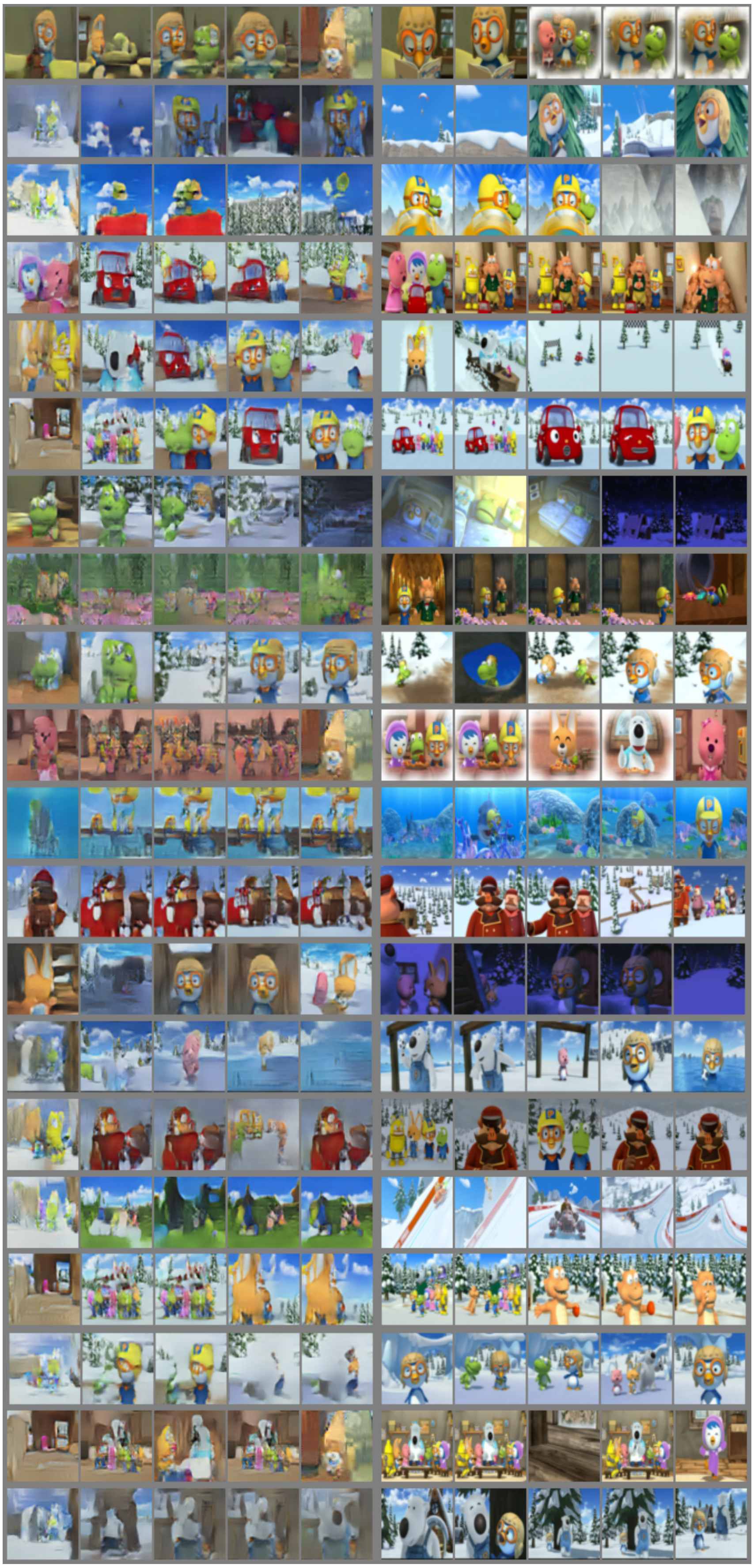


Figure 8: Additional generated examples using our model. On the left is the generated examples and on the right is ground truth.