StoryDALL-E: Adapting Pretrained Text-to-Image Transformers for Story Continuation

Adyasha Maharana, Darryl Hannan, and Mohit Bansal

UNC Chapel Hill, NC 27514, USA
{adyasha,dhannan,mbansal}@cs.unc.edu

Abstract. Recent advances in text-to-image synthesis have led to large pretrained transformers with excellent capabilities to generate visualizations from a given text. However, these models are ill-suited for specialized tasks like story visualization, which requires an agent to produce a sequence of images given a corresponding sequence of captions, forming a narrative. Moreover, we find that the story visualization task fails to accommodate generalization to unseen plots and characters in new narratives. Hence, we first propose the task of story continuation, where the generated visual story is conditioned on a source image, allowing for better generalization to narratives with new characters. It is difficult to collect large-scale datasets to train large models for this task from scratch due to the need for continuity and an explicit narrative among the images in a story. Therefore, we propose to leverage the pretrained knowledge of text-to-image synthesis models to overcome the low-resource scenario and improve generation for story continuation. To that end, we enhance or 'retro-fit' the pretrained text-to-image synthesis models with task-specific modules for (a) sequential image generation and (b) copying relevant elements from an initial frame. Then, we explore full-model finetuning, as well as prompt-based tuning for parameter-efficient adaptation, of the pre-trained model. We evaluate our approach StoryDALL-E on two existing datasets, PororoSV and FlintstonesSV, and introduce a new dataset DiDeMoSV collected from a video-captioning dataset. We also develop a model StoryGANc based on Generative Adversarial Networks (GAN) for story continuation, and compare it with the StoryDALL-E model to demonstrate the advantages of our approach. We show that our retro-fitting approach outperforms GAN-based models for story continuation and facilitates copying of visual elements from the source image, thereby improving continuity in the generated visual story. Finally, our analysis suggests that pretrained transformers struggle to comprehend narratives containing several characters and translating them into appropriate imagery. Overall, our work demonstrates that pretrained text-to-image synthesis models can be adapted for complex and low-resource tasks like story continuation. Our results encourage future research into story continuation as well as exploration of the latest, larger models for the task. Code, data, demo and model card available at https://github.com/adymaharana/storydalle.

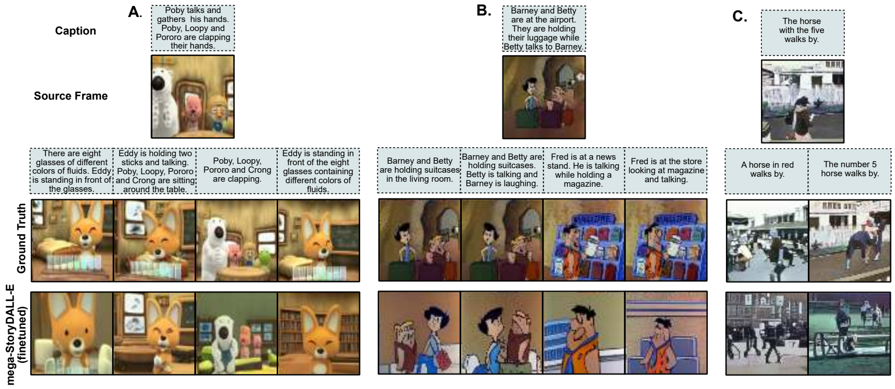

A.
Caption
Poby talks and gathers his hands. Poby, Loopy and Pororo are clapping their hands.
Source Frame
There are eight glasses of different colors of fluids. Eddy is standing in front of the glasses.
Eddy is holding two sticks and talking. Poby, Loopy, Pororo and Crong are sitting around the table.
Poby, Loopy, Pororo and Crong are clapping.
Eddy is standing in front of the eight glasses containing different colors of fluids.
Ground Truth
mega-StoryDALL-E (finetuned)
B.
Barney and Betty are at the airport. They are holding their luggage while Betty talks to Barney.
Barney and Betty are holding suitcases in the living room.
Barney and Betty are holding suitcases. Betty is talking and Barney is laughing.
Fred is at a news stand. He is talking while holding a magazine.
Fred is at the store looking at magazine and talking.
C.
The horse with the five walks by.
A horse in red walks by.
The number 5 horse walks by.

Fig. 1. Examples of predictions for (A) PororoSV (B) FlintstonesSV and (C) DiDeMoSV story continuation datasets from the mega-StoryDALL-E model. Source frame refers to the initial frame provided as additional input to the model.

1 Introduction

Pretrained text-to-image synthesis models like DALL-E [38] have shown unprecedented ability to convert an input caption into a coherent visualization. Several subsequent approaches have also leveraged powerful multimodal models [6, 37] for creating artistic renditions of input captions [7], demonstrating their potential for democratizing art. However, these models are designed to process only a single, short caption as input. In contrast, many use cases of text-to-image synthesis require models to process long narratives and metaphorical expressions, condition on existing visuals and generate more than one image to capture the meaning of the input text. In the past, multiple works have developed specialized Generative Adversarial Networks (GAN) models such as image-to-image translation [17], style transfer [20] etc. For instance, story visualization models [27] convert a sequence of captions into a sequence of images that illustrate the story. However, the recent advent of transformer-based large pretrained models opens up possibilities for leveraging latent knowledge from large-scale pretrained datasets for performing these specialized tasks more effectively, in a paradigm that is similar to finetuning of pretrained language models for performing downstream tasks based on language understanding. Hence, in this paper, we explore methods to adapt a pretrained text-to-image synthesis model for complex downstream tasks, with a focus on story visualization.

Story visualization is a challenging task that lies at the intersection of image generation and narrative understanding. Given a series of captions, which compose a story, an agent must generate a corresponding sequence of images that depicts the contents of these captions. While prior work in story visualization has discussed potential applications of the task [27, 31, 32, 44], the task itself presents some difficulties when being applied to real world settings. The model is limited to the fixed set of characters, settings, and events on which it is trained and has no way of knowing how to depict a new character that appears in a caption during test time; captions do not contain enough information to fully describe the character's appearance. Therefore, in order to generalize to new story elements, the model must have a mechanism for obtaining additional information about how these elements should be visually represented. First, we make story visualization more conducive to these use cases by presenting a new task called 'story continuation'. In this task, we provide an initial scene that can be obtained in real world use cases. By including this scene, the model can then copy and adapt elements from it as it generates subsequent images (see Fig. 1). This has the additional benefit of shifting the focus from text-to-image generation, which is already a task attracting plenty of research, and instead focuses on the narrative structure of a sequence of images, e.g., how an image should change over time to reflect new narrative information in the captions. We introduce a new dataset, DiDeMoSV [13], and also convert two existing visualization datasets PororoSV [27] and FlintstonesSV [10] to the story continuation setting.

Next, in order to adapt a text-to-image synthesis model to this story continuation task, we need to finetune the pretrained model (such as DALL-E [38]) on a sequential text-to-image generation task, with the additional flexibility to copy from a prior input. To do so, we first 'retro-fit' the model with additional layers to copy relevant output from the initial scene. Then, we introduce a self-attention block for generating story embeddings that provide global semantic context of the story during generation of each frame. The model is finetuned on the story continuation task, where these additional modules are trained from scratch. We name this approach StoryDALL-E and also compare with a GAN-based model StoryGANc for story continuation. We also explore the parameter-efficient framework of prompt-tuning and introduce a prompt consisting of task-specific embeddings to coax the pretrained model into generating visualizations for the target domain. During training of this prompt-tuning version of the model, the pretrained weights are frozen and the new parameters are learned from scratch, which is time as well as memory-efficient.

Results show that our retro-fitting approach in StoryDALL-E is useful for leveraging the latent pretrained knowledge of DALL-E for the story continuation task, and outperforms the GAN-based model on several metrics. Further, we find that the copying mechanism allows for improved generation in low-resource scenarios and of unseen characters during inference. In summary,

- We introduce the task of story continuation, that is more closely aligned with real-world downstream applications for story visualization, and provide the community with a new story continuation dataset.
- We introduce StoryDALL-E, an adaptation of pretrained transformers for story continuation, using retro-fitting. We also develop StoryGANc as a strong GAN baseline for comparison.
- We perform comparative experiments and ablations to show that finetuned StoryDALL-E outperforms StoryGANc on three story continuation datasets along several metrics.

- Our analysis shows that the copying mechanism improves correlation of the generated images with the source image, leading to better continuity in the visual story and generation of low-resource as well as unseen characters.

2 Related Work

Text-to-Image Synthesis. Most work in text-to-image synthesis has focused on the development of increasingly sophisticated generative adversarial networks (GANs) [8]. Recent works have leveraged multi-stage generation [56], attentional generative networks [49], dual learning [36], dynamic memory [28,57], semantic disentaglement [51], explicit object modelling [14] and contrastive loss [19,55] to further push performance on this task. DALL-E [38] is a large transformer language model that generates both text tokens and image tokens. VideoGPT [50] adapts the DALL-E architecture for conditional generation of videos from a first frame and trains it from scratch. In contrast, we adapt the pretrained DALL-E by *retro-fitting* the pretrained weights with task-specific modules for conditional generation of a sequence of images from a first frame.

Story Visualization. [27] introduce the CLEVR-SV and PororoSV datasets which are based on the CLEVR [18] visual question answering dataset and Pororo video question answering dataset [21] respectively. [31] adapt the Flintstones text-to-video synthesis dataset [10] into FlintstonesSV. While these datasets have served as challenging benchmarks, they contain recurring characters throughout the dataset. Complex datasets, requiring story visualization models to generalize to a more diverse set of test cases are needed to better guide research in this domain. We introduce the story continuation task and propose a new dataset for the task.

Most story visualization models follow the framework introduced in StoryGAN [27], which comprises a recurrent text encoder, an image generator, and image as well as story discriminators to train the GAN [46]. [54] add textual alignment models and a path-based image discriminator, while [25] add dilated convolution and weighted activation degree to the discriminators. [43] add figure-background segmentation to the model in the form of generators and discriminators. [32] and [31] use dual learning and structured inputs respectively to improve story visualization. We use their models as starting point and add modifications that leverage pretrained transformers for our proposed story continuation task.

Parameter-Efficient Training. Methods like adapter-tuning [12,15,30,45] and prompt-based tuning [23,26] add a small number of trainable parameters to the frozen weights of a pretrained model, which are then learned for the target task. Sparse updating of parameters [9,53] and low-rank decomposition matrices [16] also provide parameter-efficient methods for finetuning. [11,33] coaobine these approaches for a unified approach to finetuning pretrained models. [1] 'retro-fit' a pre-trained language model with cross-attention layers to retrieve relevant tokens at each timestep of word prediction in natural language generation. We use retro-fitting and prompt-tuning to adapt a pretrained image synthesis model to story continuation.

3 Methods

As discussed in Sec. 1, story visualization has limited applicability in real-world settings because the task formulation does not allow models to generalize to new story elements. Hence, we propose the story continuation task and present our StoryDALL-E and StoryGANc models for the task.

3.1 Story Continuation

Given a sequence of sentences $S = [s_1, s_2, ..., s_T]$ forming a narrative, story visualization is the task of generating a corresponding sequence of images $\hat{X} = [\hat{x}_1, \hat{x}_2, ..., \hat{x}_T]$, following [27]. S contains a story, where the captions are temporally ordered and describe the same narrative. This task has many different potential applications such as facilitating the creation of comics or creating visualizations in an educational setting. However, due to the way that the story visualization task is formulated, current models are far from being applied to these settings. The models rely on the images seen in the training data, to generate new visualizations for input stories during the inference phase. Thus, they can only recreate the characters as already found in the training set. Additionally, the captions in story visualization datasets are focused on the narrative, which limits the amount of information that is provided to the model, including descriptions of characters or settings, background etc. Much of this is inferred by the model, leading to generations that might be drastically different than expected, and it is unrealistic to expect the models to generate completely new visual attributes without sufficient instructions in the caption. Story continuation addresses these issues by providing initial information about the story setting and characters.

In the story continuation task, the first image of the sequence x_1 is provided as additional input to the model. By including an initial ground truth scene as input, the model has access to the appearances of characters, the setting in which the story takes place, and more. When making subsequent scenes, the model then no longer needs to create all the visual features from scratch, but can instead copy from the initial frame. This first image addresses both the generalization issue and the limited information issue in current story visualization models. We refer to this first frame as *source frame* and the remaining frames in the sequence $[x_2, ..., x_t]$ as *target frames.*

3.2 StoryDALL-E

The DALL-E generative model is trained using a simple language-modeling objective on the sequence of discrete image tokens for the task of text-to-image synthesis [38]. With massive amounts of data, such models learn the implicit

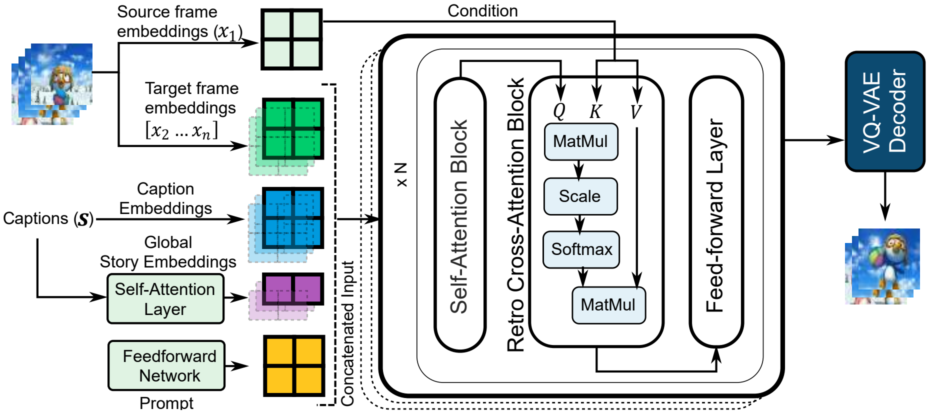


Fig. 2. Illustration of our StoryDALL-E architecture for the prompt-tuning setting. The frames are encoded using pretrained VQVAE and sent as inputs to the pretrained DALL-E. The inputs are prepended with input-agnostic prompt (in prompt-tuning setting only) and global story embeddings corresponding to each sample in the story continuation dataset. The output of StoryDALL-E is decoded using VQ-VAE to generate the predicted image.

alignment between text tokens and image tokens, which can be leveraged for downstream tasks like story continuation. The two main aspects that differentiate the story continuation task from text-to-image synthesis are: (1) sequence of captions vs. single caption, and (2) source frame vs. no source frame. Hence, in order to convert the text-to-image synthesis model into a story continuation model, we add two task-specific modules to the native DALL-E architecture. First, we use a global story encoder to pool information from all captions and produce a story embedding, which provides global context of the story at each timestep. Next, we 'retro-fit' the model with cross-attention layers in order to accept the source frame as additional input. We refer to our proposed model as StoryDALL-E (see Figure 2). All parameters of StoryDALL-E are updated during the finetuning of the model. In the parameter-efficient version of StoryDALL-E, we learn a sequence of embeddings for the story continuation task and provide it as a prompt to the model for task-specific instructions. During training, the pretrained model weights are frozen and these task-specific modules are trained from scratch.

Global Story Encoder. Most previous works in story visualization utilize recurrent encoders in the form of LSTM networks [27] or memory-augmented encoders [31, 32], to accept a sequence of captions as input. However, recurrent architectures are memory as well as time-intensive because of sequential processing. Hence, we propose to use a self-attention (f_{self}) based global story encoder, which takes the sentence embeddings for all captions as input and generates contextualized story embeddings for each time-step using parallel processing (see Figure 2). Additionally, we initialize sinusoid positional embeddings (S_{pos})

to provide information about the position of the target frame within the story, and add those to the story embeddings: $S_{global} = f_{self}(S + S_{pos})$. These embeddings are prepended to the word embeddings for the caption at that timestep and sent as input to the generative model.

Retro-fitted Cross-Attention Blocks. Next, we want to 'retro-fit' the DALL-E model with the ability to copy relevant elements from the source image, in order to promote generalizability to unseen visual attributes. This will allow the model to generate visual stories with completely new characters, as long as they are present in the source frame. Hence, we adapt the model to 'condition' the generation of target frame on the source frame by adding a cross-attention block to each self-attention block of the native DALL-E architecture. The image embeddings of the source frame are used in the cross-attention layer as *key* (K) and *value* (V), while the output from the preceding self-attention layer is used as *query* (Q). As shown in Figure 2, the DALL-E self-attention block consists of the self-attention (f^i_{self}), feed-forward (f^i_{dense}) and normalization (f_{norm}) layers. Given an input z_i to the ith self-attention block, the output z^{i+1} is: $z^{i+1} = f_{norm}(f^i_{dense}(f^i_{self}(z_i)))$. In StoryDALL-E, we insert a cross-attention layer such that the output z^{i+1} is:

$$z^{i+1} = f_{norm}(f^i_{dense}(f^i_{cross}(f^i_{self}(z^i), c_{img}))) \tag{1}$$

where f^i_{cross} is the cross-attention layer in the ith transformer block and c_{image} is the sequence of embedding representations for the conditioning image. The self-attention layers are constrained to perform causal masking for computing attention weights due to the nature of the image synthesis task. However, within the cross-attention layer, the input is free to attend over the entire source frame which eases the next token prediction task by augmenting the model with relevant information. The cross-attention layers are trained from scratch.

The StoryDALL-E architecture can be fully fine-tuned to learn the weights of the above-mentioned task-specific modules, while updating the weights of the pretrained model as necessary, on the target task as well as dataset. However, [1] show that freezing of pretrained weights during training of retro-fitted models can also lead to similar performance as models trained from scratch, with lesser training data. Further, it provides a parameter-efficient approach that can be trained/deployed with a smaller amount of computational resources. Hence, we additionally explore prompt-tuning [26] of the StoryDALL-E model.

Prompt. Prompt-tuning is an alternative [26] to full model fine-tuning where the pretrained model weights are frozen and instead, a small sequence of task-specific vectors is optimized for the downstream task. We initialize a parameterization network $MLP(.)$, which takes a matrix of trainable parameters $P^{'}_{\theta}$ of dimensions P_{idx} and $dim(h^i)$ as input and generates the prompt P_{θ}. These trainable matrices are randomly initialized and trained from scratch on the downstream task and dataset. P_{θ} is appended to the word embeddings of input caption, along with the global story embeddings. Together, these additional embedding vectors act as 'virtual tokens' of a task-specific prompt, and are attended to by each of the caption as image tokens. Formally, the input h^i to the ith self-attention layer in the auto-regressive transformer is organized as follows:

$$h^i = \begin{cases} P_\theta[j,:] & \text{if } j \in [0, P_{idx}) \\ S_{global} & \text{if } j == P_{idx} \\ f^i(z_j, h_{<j}) & \text{otherwise} \end{cases} \tag{2}$$

where $f^i(.)$ is the ith transformer block in StoryDALL-E.

With the aforementioned additions, we convert the pretrained DALL-E into StoryDALL-E model for the story continuation task. A pretrained VQVAE encoder [35] is used to transform RGB images into small 2D grids of image tokens, which are flattened and concatenated with the modified inputs in StoryDALL-E (see Appendix for details). Finally, StoryDALL-E is trained to model the joint distribution over the tokens of text s and image x: $p(x) = \prod_{j=1}^{d} p(x_j|x_{<i};s)$. New parameters as well as pretrained weights are optimized in full-model finetuning whereas only the parameters of the prompt, story encoder and cross-attention layers are optimized during prompt-tuning.

3.3 StoryGANc

Generative Adversarial Networks (GANs) have enjoyed steady progress at many image generation tasks such as style transfer [20], conditional image generation [49], image-to-image translation [17] over the last decade. Unlike transformers, they do not need to be pretrained on massive datasets, and can be trained for narrow domains with smaller datasets, which makes it an appealing method. Several recent works in story visualization have demonstrated the effectiveness of GANs for this task [27,32,44]. Hence, we also develop a GAN-based model, StoryGANc, for the story continuation task and compare its performance to that of StoryDALL-E on the proposed datasets (see Appendix for figure and details). StoryGANc follows the general framework of the StoryGAN model [27] i.e., it is composed of a recurrent text encoder, an image generation module, and two discriminators - image and story discriminator. We modify this framework to accept the source frame as input for the story continuation task, and use it for improving the generation of target frames. Our StoryGANc model is implemented as follows:

Pre-trained Language Model Encoder. We use a pretrained language model (such as RoBERTa [29] or CLIP text encoder [37]) as the caption encoder. These models are pretrained on large unimodal or multimodal datasets of language, which is of great utility for understanding the semantic concepts present in input captions. To ensure that the model has access to all captions, we append the captions together and use a special token to denote which caption is currently being generated.

Contextual Attention. The story representation from the encoder is combined with the image embeddings of the first frame of the image sequence using contextual attention [52] between the two inputs. The resulting representation is fed through a generator module which recurrently processes each caption, and produces a corresponding image.

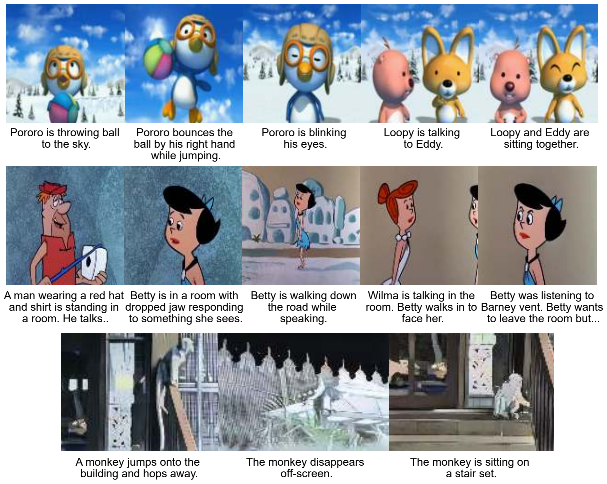
Discriminators. The story discriminator takes all of the generated images and uses 3D convolution to create a single representation and then makes a prediction as to whether the generated story is real or fake. The image discriminator performs the same function but only focuses on individual images. The KL-Divergence loss enforces gaussian distribution on the latent representations learnt by GAN. Finally, the model is trained end-to-end using the objective function: $\min_{\theta_G} \max_{\theta_I, \theta_S} \mathcal{L}_{KL} + \mathcal{L}_{img} + \mathcal{L}_{story}$, where θ_G, θ_I and θ_S denote the parameters of the text encoder + image generator, and image and story discriminators respectively. During inference, the trained weights θ_G are used to generate a visual story for a given input of captions.

4 Datasets

Since story continuation is a reframing of the story visualization tasks, existing story visualization datasets can be adapted for story continuation by assigning the first frame in the sequence as source frame and the rest as target frames. However, such existing story visualization datasets like PororoSV [27] and FlintstonesSV [10] are also homogeneous datasets with recurring characters i.e., the characters used during evaluation already appear in the training set. It is not possible to evaluate the generalization capacity of story continuation models using these datasets. Hence, we propose a new dataset in this paper.

DiDeMoSV. DiDeMo [13] is a video captioning dataset containing 10,000 short clips with more than 40,000 text descriptions temporally localized with the videos. Each of the clips was randomly sampled from the YFCC100M [47] dataset which is based upon Flickr. This results in videos that cover a large breadth of real-world scenarios, containing many different settings, actions, entities, and more. The dataset contains 11550/2707/3378 samples in training, validation and test respectively, with each sample containing three consecutive frames. This dataset challenges story continuation models to generate diverse inputs, covering many more story elements, in contrast to existing story visualization datasets. In order to do this, models must maximize their usage of the initial scene input and need to incorporate additional general visual knowledge, whether this is done through transfer learning or additional data.

We also use the existing PororoSV [27] and FlintstonesSV datasets [10], containing 10191/2334/2208 and 20132/2071/2309 samples respectively, to evaluate our story continuation models. Each sample contains 5 consecutive frames. There are 9 and 7 main characters in PororoSV and FlintstonesSV respectively, that appear throughout the dataset. For story continuation, we use the first frame as



Pororo is throwing ball to the sky.
Pororo bounces the ball by his right hand while jumping.
Pororo is blinking his eyes.
Loopy is talking to Eddy.
Loopy and Eddy are sitting together.
A man wearing a red hat and shirt is standing in a room. He talks..
Betty is in a room with dropped jaw responding to something she sees.
Betty is walking down the road while speaking.
Wilma is talking in the room. Betty walks in to face her.
Betty was listening to Barney vent. Betty wants to leave the room but...
A monkey jumps onto the building and hops away.
The monkey disappears off-screen.
The monkey is sitting on a stair set.

Fig. 3. Examples from the PororoSV (top), FlintstonesSV (middle) and DiDeMoSV (bottom) datasets. In the story continuation setting, the first frame is used as input to the generative model.

source frame and the rest of the four frames in the sequence as target frames. Evaluation is only performed on the generation of target frames. See Figure 3 for examples from the three story continuation datasets.

5 Experiments

We use the pretrained weights from popular open-source minDALL-E (1.3B parameters) which is trained on 14 million text-image pairs from the CC3M [42] and CC12M [3] datasets, to initialize our models.[1] minDALL-E uses the pretrained VQGAN-VAE [6] for discretizing image inputs. We experiment with pretrained CLIP [37] (38M parameters) and distilBERT [41] (110M parameters) text encoders for the StoryGANc models. The StoryDALL-E models are trained for 5 epochs with learning rates of 1e-04 (AdamW, Cosine Scheduler) and 5e-04 (AdamW, Linear Decay Scheduler) for full-model fine-tuning and prompt-tuning setups respectively. Checkpoints are saved at the end of every epoch. The StoryGANc models are trained for 120 epochs with learning rates 1e-04 and 1e-05 for the generator and discriminators respectively. Checkpoints are saved every 10 epochs. These models are trained on single A6000 GPUs.

We use the FID score for saving the best checkpoints in our experiments. The FID score calculates the difference between the ground truth and generated images by computing the distance between two feature vectors. Following [27] and [32], we also compute the character classification scores (F1 Score and Frame Acc.) for the PororoSV and FlintstonesSV datasets. See Appendix for details.

[1] https://github.com/kakaobrain/minDALL-E

Table 1. Results on the test sets of PororoSV, FlintstonesSV and DiDeMoSV (DSV) datasets from various models. Scores are based on FID (lower is better), character classification F1, and frame accuracy (F-Acc.; higher is better) evaluations.

Model	PororoSV			FlintstonesSV			DSV
	FID ↓	Char-F1↑	F-Acc↑	FID ↓	Char-F1↑	F-Acc↑	FID↓
StoryGANc (BERT)	72.98	**43.22**	17.09	91.37	70.45	55.78	91.43
StoryGANc (CLIP)	74.63	39.68	16.57	90.29	72.80	**58.39**	92.64
StoryDALL-E (prompt)	61.23	29.68	11.65	53.71	42.48	32.54	64.58
StoryDALL-E (finetuning)	**25.90**	36.97	**17.26**	**26.49**	**73.43**	55.19	**32.92**

6 Results

Main Quantitative Results. Table 1 contains the FID, character classification F1 score and frame accuracy results on the test sets of PororoSV and FlintstonesSV datasets using various models in our experiments. We train two variations of the StoryGANc model with the distilBERT and CLIP text encoders. Our model StoryDALL-E is trained under two settings, one where the pretrained weights are frozen during training and the other where the pretrained weights are also finetuned on the target dataset. In practice, we find it necessary to finetune the pretrained text and image embeddings within the transformers, which are pretrained on real-world images, in the prompt tuning setting in order to adapt them to different domains such as cartoons. This results in nearly 30% trainable parameters during prompt-tuning, as compared to full-model finetuning. With fully finetuned StoryDALL-E, we see drastic improvements in FID score for the PororoSV and FlinstonesSV datasets, over the StoryGANc model, demonstrating the superior visual quality of the generated visual stories. The character classification scores remain the same for FlintstonesSV and drop by 6% and 14% for PororoSV with use of finetuned and prompt-tuned StoryDALL-E respectively. GAN-based models like StoryGANc are able to recreate distinct and finer details of a character which leads to higher accuracy scores using a classification model, such as the Inception-v3 used in our experiments [32]. With prompt-tuning, we observe that StoryDALL-E models manage to capture the background elements of the scene but fail to properly recreate the characters in the frame. The frame accuracy score, which is based on exact match overlap of multiple characters in the predicted scene with those in ground truth, remains low for all models, suggesting that both methods struggle to compose multiple roles in a single image [5].

For the more challenging DiDeMoSV dataset, the fully finetuned StoryDALL-E model outperforms the GAN models by a wide margin in terms of FID score. It should be noted here that PororoSV and FlintstonesSV have a finite set of recurring animated characters throughout the dataset, whereas DiDeMoSV is derived from a multitude of real-world scenarios with no overlap in characters between training and evaluation sets. While the addition of a source frame makes it easier for the model to replicate it in the target frames, the generation is significantly more difficult due to the diversity in evaluation samples. However, since the

Table 2. Ablation results of finetuned StoryDALL-E on validation sets of PororoSV, FlintstonesSV and DiDeMoSV (DSV) datasets. Scores are based on FID (lower is better), character classification F1 and frame accuracy (F-Acc.; higher is better) evaluations.

Model	**PororoSV**			**FlintstonesSV**			**DSV**
	FID ↓	Char-F1↑	F-Acc↑	FID ↓	Char-F1↑	F-Acc↑	FID↓
StoryDALL-E	21.64	40.28	20.94	28.37	74.28	52.35	41.58
- Cross-Attention	30.45	39.32	34.65	35.04	73.94	53.28	55.89
- Story Embeddings	23.27	40.25	18.16	29.21	72.18	52.72	42.34
- Story Embeddings & Cross-Attention	31.68	35.29	16.73	36.28	72.44	51.32	58.14

Table 3. Results from human evaluation (Win% / Lose% / Tie%). Win% = % times stories from StoryDALL-E was preferred over StoryGANc, Lose% for vice-versa. Tie% represents remaining samples.

Dataset	**Visual Quality**	**Relevance**	**Consistency**
PororoSV	94/0/6	44/28/28	56/26/18
FlintstonesSV	90/2/8	32/38/30	42/32/26
DiDeMoSV	64/0/36	38/0/62	32/48/20

DiDeMoSV dataset contains images from the real-world domain, the pretrained knowledge of StoryDALL-E derived from Conceptual Captions is useful for generating relevant and coherent images for the dataset, while StoryGANc largely fails to do so.

Ablations. Table 2 contains results from ablation experiments on finetuned StoryDALL-E on the validation sets of the three story continuation datasets. The primary modifications we make to DALL-E in order to adapt it into StoryDALL-E, are the cross-attention layers and global story embeddings. We perform minus-one experiments on StoryDALL-E by removing each of these components and observing the effect on FID results on validation sets. First, we remove the cross-attention layers from StoryDALL-E, which reverts the model to the story visualization setting where the model no longer receives the first image as input, and is evaluated on the generation of the rest of the frames in the visual story. With this ablation, we see a large increase in FID scores across all datasets. Without a source image to guide the generated output, the quality of illustration drops rapidly, especially for the new DiDeMo dataset. The removal of global story embeddings results in a text-to-image synthesis setting with the first frame as additional input. In this scenario, we see smaller drops in FID, indicating that the global context is not as important as the ability to copy from an initial image. In the third row, we remove both, cross-attention layers and story embeddings, which relegates the setting to a text-to-image synthesis task, and observe a large increase in FID scores across all datasets.

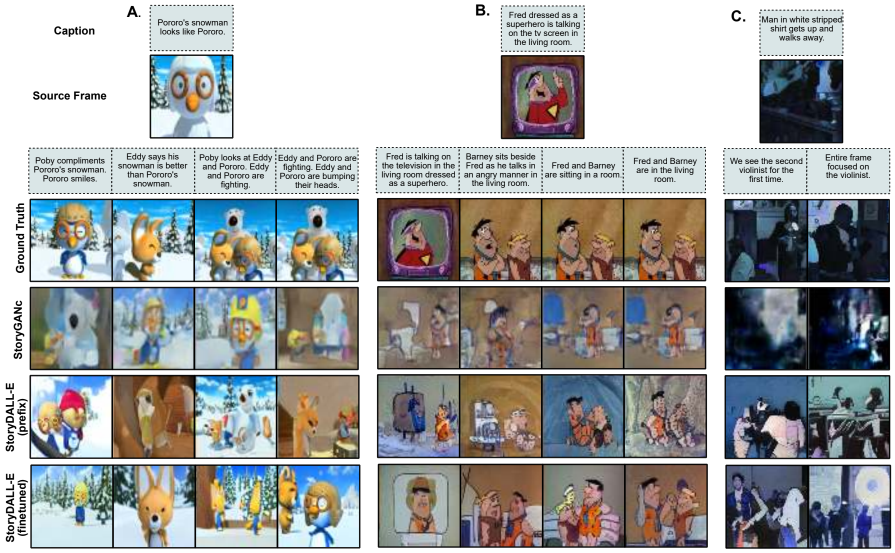

A.
Caption
Pororo's snowman looks like Pororo.
Source Frame
Poby compliments Pororo's snowman. Pororo smiles.
Eddy says his snowman is better than Pororo's snowman.
Poby looks at Eddy and Pororo. Eddy and Pororo are fighting.
Eddy and Pororo are fighting. Eddy and Pororo are bumping their heads.
B.
Fred dressed as a superhero is talking on the tv screen in the living room.
Fred is talking on the television in the living room dressed as a superhero.
Barney sits beside Fred as he talks in an angry manner in the living room.
Fred and Barney are sitting in a room.
Fred and Barney are in the living room.
C.
Man in white stripped shirt gets up and walks away.
We see the second violinist for the first time.
Entire frame focused on the violinist.
Ground Truth
StoryGANc
StoryDALL-E (prefix)
StoryDALL-E (finetuned)

Fig. 4. Examples of predictions for (A) PororoSV (B) FlintstonesSV and (C) DiDeMoSV story continuation datasets from finetuned StoryDALL-E and StoryGANc models. Source frame refers to the initial frame provided as additional input to the model.

6.1 Human Evaluation

We additionally conduct human evaluation on our models' outputs hoping to better capture the overall quality of the generated stories. We have a human annotator compare generated visual stories from our StoryDALL-E (finetuning) and StoryGANc (BERT) models. They are provided with predictions from each dataset and the corresponding ground truth captions and asked to pick the better prediction (or tie) in terms of visual quality, consistency, and relevance [27]. Results are presented in Table 3. The StoryDALL-E model outperforms StoryGANc model in terms of visual quality and relevance, achieving higher % of wins in each of the three datasets (except relevance in FlintstonesSV). These results follow from the fact that StoryDALL-E uses the VQGAN-VAE [6] which is designed for reconstructing higher resolution images. Moreover, it has access to large pretraining data, which improves alignment between semantic concepts in captions and regions in images. We see wins in terms of consistency for PororoSV and DiDeMoSV predictions from StoryDALL-E models. But, the absolute numbers for consistency and relevance show that there is still room for improvement.

7 Analysis

In this section, we perform experiments to analyze aspects of the StoryDALL-E model and the story continuation task. First, we perform qualitative analyses of the predictions from StoryDALL-E. Next, we quantify the effect of the retro-fitted cross-attention layers and visualize the attention heads. See Appendix for an analysis of the diverse semantic content in the DiDeMoSV dataset.

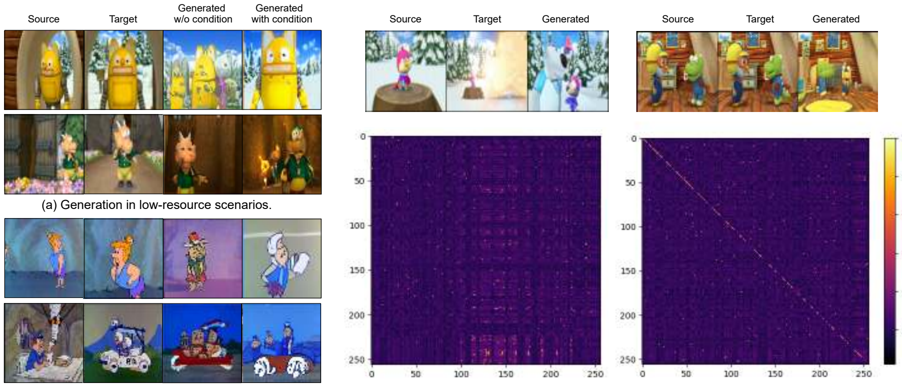
7.1 Qualitative Analysis

Figure 4 contains sampled outputs from both of our models for the three story continuation datasets. In each of these examples, StoryDALL-E generates higher quality images than StoryGANc. The difference is especially stark for PororoSV and FlintstonesSV datasets since StoryDALL-E is exposed to the characters during training and has additional guidance from source frame during inference. In the case of DiDeMoSV, the generations from StoryGANc are largely incomprehensible, which could be attributed to the unseen semantic concepts such as 'violinist' which did not appear in the training set. In contrast, StoryDALL-E is exposed to various real-world concepts during pretraining, which can be leveraged during generation. For instance, the pretrained knowledge, as well as the copying mechanism, help the StoryDALL-E model comprehend 'television' and generate an image for 'Fred is talking in the television' (see Figure 4(b)). However, the overall quality of the images from StoryDALL-E also does not approach human produced images. As discussed in Sec. 6, it is especially true for frames containing multiple characters. This suggests that while current models are able to attempt the task, there is still much work to be done before consistent and coherent images are commonly produced by the models.

We also examine the ability of StoryDALL-E to recreate scarce characters from the training set (see Figure 5(a)) and generate unseen characters (see Figure 5(b)), when guided by the copying mechanism via cross-attention layers. We find that the copying mechanism allows for better generation of shape and form for less-frequent characters in PororoSV. Similarly, we identified non-recurring characters in the FlintstonesSV dataset and observed the corresponding generated images, when StoryDALL-E has access to a previous frame where they appear. StoryDALL-E succeeds at partially copying visual aspects of the characters, such as the purple skirt (top) and blue uniform (bottom).

7.2 Retro-fitted Cross-Attention

We examine the attention scores computed in the retro cross-attention layer and present examples in Figure 5(c). The cross-attention layers in StoryDALL-E receive vector representations for the source image and compute the cross-attention output using the source frame as key/value and the target frame as query. In the first example (left), the target frame is copying visual attributes of the pink bird with the most emphasis, as be seen from the higher attention scores for the image tokens roughly in the center of the source frame. For the second example (right), the source frame and target frames are nearly similar; the attention scores are highest in the diagonal of the plot. The resulting images in both samples contain many visual attributes already found in the source image, demonstrating that the cross-attention layer is effective at enabling conditional



(a) Generation in low-resource scenarios.

(b) Generation of unseen characters.

(c) Visualization of attention heads in cross-attention layers of StoryDALLE.

Fig. 5. Examples of generation from StoryDALL-E in (a) low-resource scenarios and (b) of unseen characters. (c) Plots of attention scores computed in retro cross-attention layers for examples of source frames (x-axis) and target frames (y-axis).

image generation. See Appendix for correlation scores between source image and frames generated with or without condition using StoryDALL-E.

8 New Results & Demo with DALL-E Mega

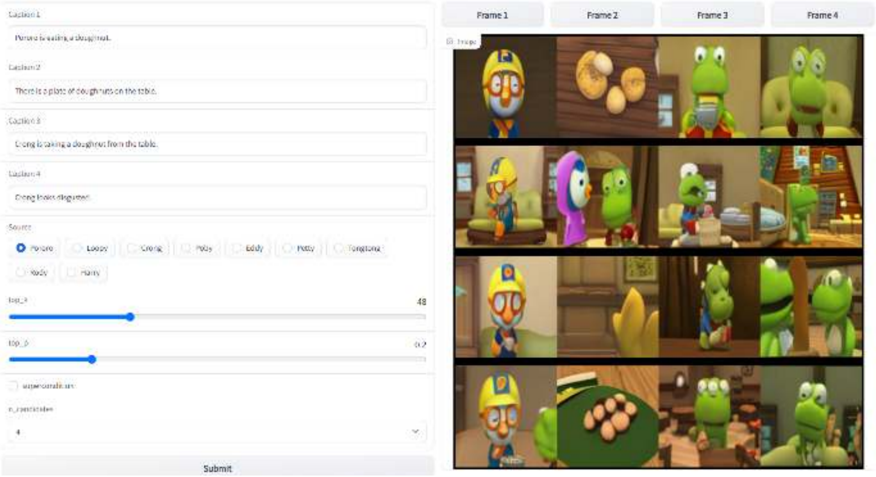
Following our approach of using pretrained text-to-image synthesis models for story continuation, we repeat our experiments with the recently released DALL-E Mega for the final version of the paper.[2] DALL-E Mega is pretrained on 15 million images from the Conceptual Captions dataset [42] and follows an encoder-decoder architecture, as opposed to the decoder-only architecture used in minDALL-E. It relies on the pretrained BART encoder [24] for encoding the input captions as well as an improved VQGAN-VAE for discretized encoding of images. In order to adapt DALL-E Mega for story continuation, we retro-fit the decoder in the pretrained model with cross-attention layers and a global story encoder as outlined in Sec. 3. These additional cross-attention layers facilitate copying from a source image, and the story encoder enables generation of a sequence of frames for the story continuation task. We refer to the StoryDALL-E model based on the pretrained DALL-E Mega as the mega-StoryDALL-E model in this paper. In the fully-finetuned version of the mega-StoryDALL-E model, we finetune the encoder as well as the decoder on story continuation datasets. Results are shown in Table 4. We observe up to 3% improvement in FID scores over StoryDALL-E across all datasets. Smaller improvements are observed for character classification scores with the use of DALL-E Mega. Examples are shown in Figure 1.

We make the mega-StoryDALL-E model trained on the Pororo dataset available for testing through an openly accessible and easy-to-use in-browser

[2] https://github.com/kuprel/min-dalle

Table 4. Results on the test sets of PororoSV, FlintstonesSV and DiDeMoSV (DSV) datasets from MEGA-STORYDALL-E. Scores are based on FID (lower is better), character classification F1 and frame accuracy (F-Acc.; higher is better) evaluations.

Model	PororoSV			FlintstonesSV			DSV
	FID ↓	Char-F1↑	F-Acc↑	FID ↓	Char-F1↑	F-Acc↑	FID↓
STORYDALL-E (finetuning)	25.90	38.48	17.26	26.49	73.43	**55.19**	32.92
MEGA-STORYDALL-E (finetuning)	**23.48**	**39.91**	**18.01**	**23.58**	**74.26**	54.68	**31.64**



Caption 1

Pororo is eating a doughnut.

Caption 2

There is a plate of doughnuts on the table.

Caption 3

Crong is taking a doughnut from the table.

Caption 4

Crong looks disgusted.

Source

● Pororo ○ Loopy ○ Crong ○ Poby ○ Eddy ○ Petty ○ Tongtong ○ Rody ○ Harry

top_k 48

top_p 0.2

□ supercondition

n_candidates

4

Fig. 6. A snapshot of the openly-available in-browser demo for MEGA-STORYDALL-E trained on the Pororo dataset. The right panel displays the images generated by the model for the captions entered by the user in the left panel.

demo system (see Figure 6).[3] From Figures 1, 6 and demo examples, we find that the model performs well at visualizing stories with up to three characters across all frames and struggles at generating coherent visuals for more than three characters, which is also in line with our findings in Sec. 7.1. The model copies visual elements from the source image and copies to each of the generated frames in the story, hence maintaining a continuous flow in narration by virtue of conditioning on an initial scene. MEGA-STORYDALL-E performs best at generating overtly visual actions such as 'making cookies', 'walking', 'reading a book'. Further, it is capable of generating semantic concepts that do not appear in the story continuation dataset, such as 'doughnut' and 'lion', by leveraging the pretrained knowledge of DALL-E Mega when possible. Most of the scenes in the Pororo dataset occur within the setting of a snowy village with wooden houses surrounded by trees and snow. Hence, the model usually generates scenes with similar visual elements.

[3] See Model Card [34] & Demo at https://github.com/adymaharana/storydalle.

9 Conclusion

We introduce a new task called story continuation in order to make the story visualization task more conducive for real-world use cases. We present a new dataset DiDeMoSV, in addition to reformatting two existing story visualization datasets for story continuation. Our model StoryDALL-E, based on a retro-fitting approach for adapting pretrained transformer-based text-to-image synthesis models, outperforms GAN-based models on the story continuation datasets. We also added new, improved results and a demo system using the more recent, larger DALL-E Mega model. We hope that the dataset and models motivate future work in story continuation and that our work encourages the exploration of text-to-image synthesis models for more complex image synthesis tasks.

Acknowledgement. We thank the reviewers for their useful feedback. This work was supported by ARO Award W911NF2110220, DARPA KAIROS Grant FA8750-19-2-1004, NSF-AI Engage Institute DRL-211263. The views, opinions, and/or findings contained in this article are those of the authors, not the funding agency.

References

1. Borgeaud, S., Mensch, A., Hoffmann, J., Cai, T., Rutherford, E., Millican, K., Van Den Driessche, G.B., Lespiau, J.B., Damoc, B., Clark, A., et al.: Improving language models by retrieving from trillions of tokens. In: International Conference on Machine Learning. pp. 2206–2240. PMLR (2022) 4, 7
2. Cer, D., Yang, Y., Kong, S.y., Hua, N., Limtiaco, N., John, R.S., Constant, N., Guajardo-Cespedes, M., Yuan, S., Tar, C., et al.: Universal sentence encoder. arXiv preprint arXiv:1803.11175 (2018) 26
3. Changpinyo, S., Sharma, P., Ding, N., Soricut, R.: Conceptual 12m: Pushing web-scale image-text pre-training to recognize long-tail visual concepts. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3558–3568 (2021) 10, 26
4. Chen, Y., Lai, Y.K., Liu, Y.J.: Cartoongan: Generative adversarial networks for photo cartoonization. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 9465–9474 (2018) 25
5. Cho, J., Zala, A., Bansal, M.: Dall-eval: Probing the reasoning skills and social biases of text-to-image generative transformers. arXiv preprint arXiv:2202.04053 (2022) 11
6. Esser, P., Rombach, R., Ommer, B.: Taming transformers for high-resolution image synthesis. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12873–12883 (2021) 2, 10, 13, 26, 27
7. Frans, K., Soros, L., Witkowski, O.: Clipdraw: Exploring text-to-drawing synthesis through language-image encoders. arXiv preprint arXiv:2106.14843 (2021) 2
8. Goodfellow, I.J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A.C., Bengio, Y.: Generative adversarial nets. In: NeurIPS (2014) 4
9. Guo, D., Rush, A.M., Kim, Y.: Parameter-efficient transfer learning with diff pruning. In: Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers). pp. 4884–4896 (2021) 4

10. Gupta, T., Schwenk, D., Farhadi, A., Hoiem, D., Kembhavi, A.: Imagine this! scripts to compositions to videos. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 598–613 (2018) 3, 4, 9
11. He, J., Zhou, C., Ma, X., Berg-Kirkpatrick, T., Neubig, G.: Towards a unified view of parameter-efficient transfer learning. In: International Conference on Learning Representations (2021) 4
12. Henderson, J., Ruder, S., et al.: Compacter: Efficient low-rank hypercomplex adapter layers. In: Advances in Neural Information Processing Systems (2021) 4
13. Hendricks, L.A., Wang, O., Shechtman, E., Sivic, J., Darrell, T., Russell, B.: Localizing moments in video with natural language. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV) (2017) 3, 9, 24
14. Hinz, T., Heinrich, S., Wermter, S.: Semantic object accuracy for generative text-to-image synthesis. IEEE transactions on pattern analysis and machine intelligence (2020) 4
15. Houlsby, N., Giurgiu, A., Jastrzebski, S., Morrone, B., De Laroussilhe, Q., Gesmundo, A., Attariyan, M., Gelly, S.: Parameter-efficient transfer learning for nlp. In: International Conference on Machine Learning. pp. 2790–2799. PMLR (2019) 4
16. Hu, E.J., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W., et al.: Lora: Low-rank adaptation of large language models. In: International Conference on Learning Representations (2021) 4
17. Isola, P., Zhu, J.Y., Zhou, T., Efros, A.A.: Image-to-image translation with conditional adversarial networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1125–1134 (2017) 2, 8
18. Johnson, J., Hariharan, B., Van Der Maaten, L., Fei-Fei, L., Lawrence Zitnick, C., Girshick, R.: Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2901–2910 (2017) 4
19. Kang, M., Park, J.: Contragan: Contrastive learning for conditional image generation. In: NeurIPS (2020) 4
20. Karras, T., Laine, S., Aila, T.: A style-based generator architecture for generative adversarial networks. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 4401–4410 (2019) 2, 8
21. Kim, K.M., Heo, M.O., Choi, S.H., Zhang, B.T.: Deepstory: video story qa by deep embedded memory networks. In: Proceedings of the 26th International Joint Conference on Artificial Intelligence. pp. 2016–2022 (2017) 4
22. Lei, J., Wang, L., Shen, Y., Yu, D., Berg, T., Bansal, M.: Mart: Memory-augmented recurrent transformer for coherent video paragraph captioning. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. pp. 2603–2614 (2020) 23
23. Lester, B., Al-Rfou, R., Constant, N.: The power of scale for parameter-efficient prompt tuning. In: Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing. pp. 3045–3059 (2021) 4
24. Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., Stoyanov, V., Zettlemoyer, L.: Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. pp. 7871–7880 (2020) 15
25. Li, C., Kong, L., Zhou, Z.: Improved-storygan for sequential images visualization. Journal of Visual Communication and Image Representation **73**, 102956

(2020). https://doi.org/https://doi.org/10.1016/j.jvcir.2020.102956, http://www.sciencedirect.com/science/article/pii/S1047320320301826 4

26. Li, X.L., Liang, P.: Prefix-tuning: Optimizing continuous prompts for generation. In: Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers). pp. 4582–4597 (2021) 4, 7
27. Li, Y., Gan, Z., Shen, Y., Liu, J., Cheng, Y., Wu, Y., Carin, L., Carlson, D., Gao, J.: Storygan: A sequential conditional gan for story visualization. In: Proceedings of the IEEE Conference on CVPR. pp. 6329–6338 (2019) 2, 3, 4, 5, 6, 8, 9, 10, 13, 21, 22, 23, 24, 26
28. Liang, J., Pei, W., Lu, F.: Cpgan: full-spectrum content-parsing generative adversarial networks for text-to-image synthesis. arXiv preprint arXiv:1912.08562 (2019) 4
29. Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., Stoyanov, V.: Roberta: A robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692 (2019) 8, 23
30. Mahabadi, R.K., Ruder, S., Dehghani, M., Henderson, J.: Parameter-efficient multi-task fine-tuning for transformers via shared hypernetworks. In: Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers). pp. 565–576 (2021) 4
31. Maharana, A., Bansal, M.: Integrating visuospatial, linguistic, and commonsense structure into story visualization. In: Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing. pp. 6772–6786 (2021) 2, 4, 6, 23, 28
32. Maharana, A., Hannan, D., Bansal, M.: Improving generation and evaluation of visual stories via semantic consistency. In: Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. pp. 2427–2442 (2021) 2, 4, 6, 8, 10, 11, 23, 26
33. Mao, Y., Mathias, L., Hou, R., Almahairi, A., Ma, H., Han, J., Yih, S., Khabsa, M.: Unipelt: A unified framework for parameter-efficient language model tuning. In: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 6253–6264 (2022) 4
34. Mitchell, M., Wu, S., Zaldivar, A., Barnes, P., Vasserman, L., Hutchinson, B., Spitzer, E., Raji, I.D., Gebru, T.: Model cards for model reporting. In: Proceedings of the conference on fairness, accountability, and transparency. pp. 220–229 (2019) 16
35. Van den Oord, A., Kalchbrenner, N., Espeholt, L., Vinyals, O., Graves, A., et al.: Conditional image generation with pixelcnn decoders. Advances in neural information processing systems **29** (2016) 8, 21
36. Qiao, T., Zhang, J., Xu, D., Tao, D.: Mirrorgan: Learning text-to-image generation by redescription. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1505–1514 (2019) 4
37. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: International Conference on Machine Learning. pp. 8748–8763. PMLR (2021) 2, 8, 10, 23, 26
38. Ramesh, A., Pavlov, M., Goh, G., Gray, S., Voss, C., Radford, A., Chen, M., Sutskever, I.: Zero-shot text-to-image generation. In: International Conference on Machine Learning. pp. 8821–8831. PMLR (2021) 2, 3, 4, 5, 21, 22

39. Redmon, J., Farhadi, A.: Yolov3: An incremental improvement. arXiv (2018) 30
40. Rennie, S.J., Marcheret, E., Mroueh, Y., Ross, J., Goel, V.: Self-critical sequence training for image captioning. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 7008–7024 (2017) 25
41. Sanh, V., Debut, L., Chaumond, J., Wolf, T.: Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. In: 5th Workshop on Energy Efficient Machine Learning and Cognitive Computing (NeurIPS) (2019) 10, 26
42. Sharma, P., Ding, N., Goodman, S., Soricut, R.: 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). pp. 2556–2565 (2018) 10, 15, 26
43. Song, Y.Z., Rui Tam, Z., Chen, H.J., Lu, H.H., Shuai, H.H.: Character-preserving coherent story visualization. In: European Conference on Computer Vision. pp. 18–33. Springer (2020) 4
44. Song, Y.Z., Tam, Z.R., Chen, H.J., Lu, H.H., Shuai, H.H.: Character-preserving coherent story visualization. In: Proceedings of the European Conference on Computer Vision (ECCV) (2020) 2, 8
45. Sung, Y.L., Cho, J., Bansal, M.: Vl-adapter: Parameter-efficient transfer learning for vision-and-language tasks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5227–5237 (2022) 4
46. Szűcs, G., Al-Shouha, M.: Modular storygan with background and theme awareness for story visualization. In: International Conference on Pattern Recognition and Artificial Intelligence. pp. 275–286. Springer (2022) 4
47. Thomee, B., Shamma, D.A., Friedland, G., Elizalde, B., Ni, K., Poland, D., Borth, D., Li, L.J.: Yfcc100m: The new data in multimedia research. Communications of the ACM **59**(2), 64–73 (2016) 9
48. Van Den Oord, A., Vinyals, O., et al.: Neural discrete representation learning. Advances in neural information processing systems **30** (2017) 21
49. Xu, T., Zhang, P., Huang, Q., Zhang, H., Gan, Z., Huang, X., He, X.: Attngan: Fine-grained text to image generation with attentional generative adversarial networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1316–1324 (2018) 4, 8
50. Yan, W., Zhang, Y., Abbeel, P., Srinivas, A.: Videogpt: Video generation using vq-vae and transformers. arXiv preprint arXiv:2104.10157 (2021) 4
51. Yin, G., Liu, B., Sheng, L., Yu, N., Wang, X., Shao, J.: Semantics disentangling for text-to-image generation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 2327–2336 (2019) 4
52. Yu, J., Lin, Z., Yang, J., Shen, X., Lu, X., Huang, T.S.: Generative image inpainting with contextual attention. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 5505–5514 (2018) 9, 23
53. Zaken, E.B., Goldberg, Y., Ravfogel, S.: Bitfit: Simple parameter-efficient fine-tuning for transformer-based masked language-models. In: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers). pp. 1–9 (2022) 4
54. Zeng, G., Li, Z., Zhang, Y.: Pororogan: An improved story visualization model on pororo-sv dataset. In: Proceedings of the 2019 3rd International Conference on Computer Science and Artificial Intelligence. pp. 155–159 (2019) 4
55. Zhang, H., Koh, J.Y., Baldridge, J., Lee, H., Yang, Y.: Cross-modal contrastive learning for text-to-image generation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 833–842 (2021) 4

56. Zhang, H., Xu, T., Li, H., Zhang, S., Wang, X., Huang, X., Metaxas, D.: Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks. In: ICCV (2017) 4

57. Zhu, M., Pan, P., Chen, W., Yang, Y.: Dm-gan: Dynamic memory generative adversarial networks for text-to-image synthesis. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5802–5810 (2019) 4

A Background

In this section, we give a brief introduction to the original story visualization task and auto-regressive transformers for text-to-image synthesis.

A.1 Story Visualization

Given a sequence of sentences $S = [s_1, s_2, ..., s_T]$ forming a narrative, story visualization is the task of generating a corresponding sequence of images $\hat{X} = [\hat{x}_1, \hat{x}_2, ..., \hat{x}_T]$, following [27]. The sentences form a coherent story with recurring plot and characters. The generative model for this task has two main modules: story encoder and image generator. The sentence encoder $E_{caption}(.)$ takes word embeddings $\{w_{ik}\}$ for sentence s_k at each timestep k and generates contextualized embeddings $\{c_{ik}\}$. These embeddings are then used to generate the corresponding images. The terms *caption* and *sentence* are used interchangeably throughout the paper.

A.2 Pretrained Text-to-Image Synthesis Models (DALL-E)

The DALL-E model introduced in [38] is a text-to-image synthesis pipeline which comprises of a discrete variational autoencoder (dVAE) in the first stage and an autoregressive transformer in the second stage:

Stage 1. The Vector Quantized Variational Autoencoder (VQVAE) [35] consists of an encoder that learns to map high dimensional input data (x) to a discretized latent space, and a decoder that reconstructs x from the quantized encodings x^q. The model is trained using the reconstruction loss and commitment loss [48]. In DALL-E, the VQVAE is trained to transform RGB image into a small 2D grid of image tokens, where each token can assume a discrete value from a codebook of predefined length.

Stage 2. The VQVAE encoder from Stage 1 is used to infer the grid of discretized image tokens which is flattened and concatenated with the input text tokens, and an autoregressive transformer is used to model the joint distribution over the text and image tokens. For a given text input s and target image x, these models learn the distribution of image tokens $p(x)$ as,

$$p(x) = \prod_{i=1}^{d} p(x_i | x_{i<i}; s) | x < i) \tag{3}$$

The models are composed of stacked multi-head self-attention layers with causal masking and are optimized via maximum likelihood. Each self-attention block is followed by a MLP feedforward layer, as per the standard design of transformers. The prediction of image tokens at each time step is influenced by the text tokens and previously predicted image tokens via the self-attention layer.

Using this framework, DALL-E obtains impressive, state-of-the-art results on a variety of text-to-image tasks by leveraging large-scale pre-training on multimodal datasets.

B Additional Method Details

In this section, we provide additional details about the StoryDALL-E and StoryGANc models.

B.1 StoryDALL-E

Retro Cross-Attention Layer Density. We experiment with different densities of cross-attention layers in our implementation of StoryDALL-E. In the densest variation, we introduce the retro layer in every self-attention block of minDALL-E, effectively increasing the number of parameters in the model by nearly 60%. We vary the density of the retro layer for one in every 1-5 self-attention block(s), and run experiments for each of these variations. Our best model has a density of one retro layer in every third self-attention block.

Objective. Following the original DALL-E implementation [38], the StoryDALL-E model is trained on a combination of text loss and image loss. The losses are cross-entropy losses for the respective modalities, and the combined objective is,

$$\mathcal{L} = -\sum_{i=1}^{N_{text}} t_i log(p(t_i) - \sum_{i=1}^{N_{img}} m_i log(p(m_i)$$

where N_{text} and N_{img} are the caption lengths and image sequence lengths, set to 64 and 256 in our model respectively.

B.2 StoryGANc

StoryGANc follows the general framework of the StoryGAN model [27] i.e., it is composed of a recurrent text encoder, an image generation module, and two discriminators - image and story discriminator. We modify this framework to accept the source frame as input for the story continuation task, and use it for improving the generation of target frames. Our StoryGANc model is implemented as follows:

Pre-trained Language Model Encoder. In the current state-of-the-art story visualization models [32], recurrent transformer-based text encoders like MART [22] and MARTT [31] are learnt from scratch for encoding the captions. However, while the memory module contains information about prior captions, there is no way for the current caption to directly attend to words in prior or subsequent captions. This is crucial in a story where causality plays such a large role, e.g., which characters need to appear in the scene, even if they don't appear in the current caption, has there been any modifications to the background that need to appear in the current scene, etc. Furthermore, general world knowledge is crucial for successfully generating unseen stories in our datasets, which is possible with pretrained knowledge. Therefore, we propose using a pretrained language model (such as RoBERTa [29] or CLIP text encoder [37]) as the caption encoder. These models are pretrained on large unimodal or multimodal datasets of language; their latent knowledge of the world is of great utility for understanding the semantic concepts present in input captions. For the RoBERTa encoder [29], to ensure that the model has access to all captions, we append the captions together and feed all of them into each timestep. We use a special token to denote which caption is currently being generated. The representation from the first token h_0 is used as the caption representation. For the CLIP encoder [37], we add an additional self-attention block that takes the caption representation for each timestep and produces the contextualized representations that have been computed by attending to all other timesteps.

Contextual Attention. We then combine the story representation with the image embeddings of the first frame of the image sequence using contextual attention. First, we reshape the story representation as a 2D matrix and extract 3×3 patches $\{t_{x,y}\}$ as convolutional filters. Then, we match them against potential patches from the source frame $\{s_{x',y'}\}$ by measuring the normalized inner product as,

$$p_{x,y,x',y'} = \langle \frac{s_{x,y}}{||s_{x,y}||}, \frac{t_{x',y'}}{||t_{x',y'}||} \rangle \tag{4}$$

where $p_{x,y,x,y'}$ represents the similarity between the patch centered in target frame (x, y) and source frame (x', y'). We compute the similarity score for all dimensions along (x', y') for the patch in target frame (x, y) and find the best match from the softmax-scaled similarity scores. [52] implement this efficiently using convolution and channel-wise softmax; we use their implementation in our StoryGANc model. The extracted patches are used as deconvolutional filters and added to the target frame s. The resulting representation is fed through a generator module which processes each caption and produces an image. We use the generator module outlined in [27].

Discriminators. Finally, the loss is computed for the generated image sequence. There are 3 different components that provide the loss for the model. The first is a story discriminator, which takes all of the generated images and uses 3D convolution to create a single representation and then makes a prediction as to whether the generated story is real or fake. Additionally, there is an image

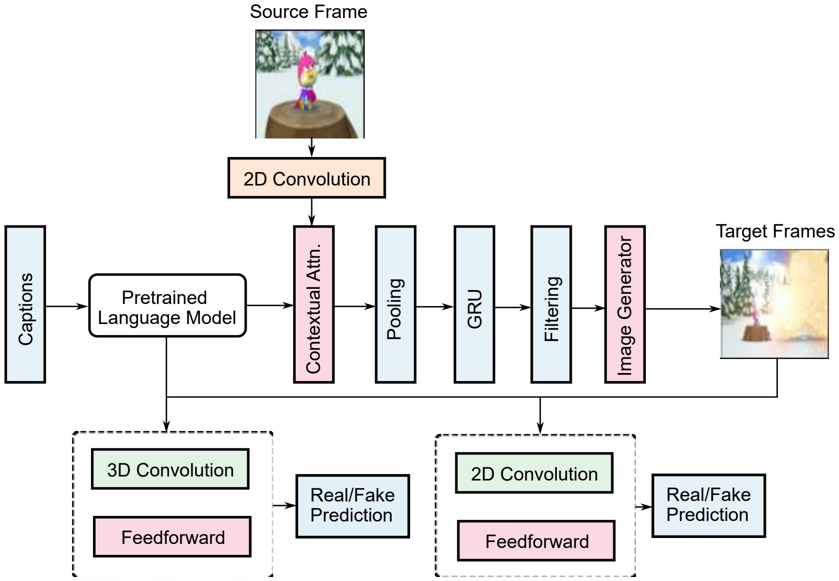


Fig. 7. Illustration of our StoryGANc architecture. The captions are first encoded using a pretrained language model to produce contextualized representations. These representations are sent to a contextual attention module along with the source frame, and the resulting representation is sent to the image generator. The generated frames are sent to a story and image discriminator, and the corresponding cross-entropy losses for detection real/fake images are used to train the StoryGANc model.

discriminator, which performs the same function but only focuses on individual images. Finally, the model is trained end-to-end using the objective function:

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

where θ_G, θ_I and θ_S denote the parameters of the text encoder + generator, and image and story discriminator respectively. $\mathcal{L}_{img}$ and $\mathcal{L}_{story}$ are cross-entropy losses for classifying ground truth and synthetic images into real and fake categories respectively. $\mathcal{L}_{KL}$ is the Kullback-Leibler (KL) divergence between the learned distribution h_0 and the standard Gaussian distribution, to enforce smoothness over the conditional manifold in latent semantic space [27]. During inference, the trained weights θ_G are used to generate a visual story for a given input of captions.

C Dataset Construction

We propose the new dataset DiDeMoSV, which is derived from the Didemo dataset [13]. Below, we present details about collection and cleaning of the dataset.

C.1 Dataset Construction

Prior work in story visualization has repurposed datasets from other tasks. We follow this trend and repurpose video captioning datasets in our work. Story visualization and video captioning share many components. In video captioning, an agent must produce a caption, or series of captions, that describe the content of a video. Story visualization can be thought of as video captioning in reverse, where frames are generated based on the captions. However, simply reversing the direction of the task is not sufficient in this case because the other difference between the two tasks is that story visualization has one frame per caption, whereas videos have many frames; a single caption is typically paired with a video time stamp, denoting which section of the video the caption aligns with. Therefore, to convert video captioning into story visualization, an appropriate method is needed to select which single frame should be used to represent the content of the caption.

We employ the self-critical image captioning model [40] for intelligently selecting the frame most aligned with the caption. Each of the clips that correspond to a caption is multiple seconds long. Not all of the frames will be equally aligned with the caption. Characters might be moving leaving blur effects, the scene might change a bit early or late in the clip, or there might be superfluous actions that occur. To initially shrink the number of frames that we must consider, we first sample frames at fixed intervals throughout the video. In the case of DiDeMoSV, we sample 10 frames. Each of the frames is then fed through the self critical image model and is ranked according to the sum of the log likelihood for each word in the caption being generated. We then use the top-ranked frame as the image for the given caption. The resulting image-caption sequence after this step is on average 4 frames long for DiDeMoSV. To maximize the amount of data that we have and make the task feasible, we split these image-caption sequences into a sequence of 3 frames. We use a sliding window approach to create these sequences, allowing for overlap between sequences. However, we also ensure that the train, val, and test splits contain separate videos. We then proceed with our image pre-processing steps.

The main pre-processing step that we explore is to convert the real-world images into cartoon images, to emphasize focus on the main characters of the image rather than the trivial details of the background. Rather than models focusing on making images realistic, we want them to focus on accurately representing the stories themselves in visual form. To cartoonize the images we use CartoonGAN [4]. Each of the extracted frames is fed through this network and the resulting output is used in the final dataset.

D Experimental Details

Pretrained Weights. While the VAE checkpoints for the original DALL-E model have been released, the transformer weights have not. We explored training the transformer component from scratch on our data, but found that it did not perform well. Therefore, we explored other publicly available efforts to reproduce

DALL-E and settled on a popular open-source version minDALL-E which is composed of 1.3 billion parameters and trained on 14 million text-image pairs from the CC3M [42] and CC12M [3] datasets.[4] minDALL-E uses the pretrained VQGAN-VAE [6] for discretizing image inputs. We adapt the pretrained model minDALL-E to StoryDALL-E and then prompt-tune/fine-tune the retro-fitted model on our target datasets.

We experiment with pretrained CLIP [37] (38M parameters) and distilBERT [41] (110M parameters) text encoders for the LM-StoryGAN models. The CLIP image encoder is used to extract image embeddings for the source frame in the story continuation task. The universal sentence transformer [2] is used to extract sentence embeddings for captions, that are sent as input to the global story encoder in StoryDALL-E.

Training Details. We conduct experiments in the story continuation setting, i.e., the models receive the first frame as input condition. The StoryDALL-E and mega-StoryDALL-E models are trained for 5 epochs with learning rates of 1e-04 (AdamW, Cosine Scheduler) and 5e-04 (AdamW, Linear Decay Scheduler) for fine-tuning and prompt-tuning setups respectively. We use a cosine schedule with warmup from 0 in the first 750 training steps. The minimum learning rate is 0.1 times the maximum learning rate. Checkpoints are saved at the end of every epoch. In full-model finetuning settings, the pretrained weights are finetuned with a smaller learning rate of 1e-05. The LMStoryGAN models are trained for 120 epochs with learning rates 1e-04 and 1e-05 for the generator and discriminators respectively. Checkpoints are saved every 10 epochs. These models are trained on 1-2 A6000 GPUs.

For the publicly available demo, we have continued training the StoryDALL-E and mega-StoryDALL-E models for up to 50 epochs, which takes up to 10 days on 2 A6000 GPUs and exhibits improved performance over the checkpoints reported in the paper. See the codebase for links to the demo and the checkpoints used therein.[5]

Evaluation Metrics. We consider 3 automatic evaluation techniques. The first is FID score, which calculates the difference between the ground truth and generated images by computing the distance between two feature vectors. We follow prior work and use Inception-v3 as our image encoding model.

Following [27] and [32], we also compute the character classification scores for the Pororo and Flintstones datasets, which are adapted from video QA datasets with recurring characters. We use the Inception-v3 models trained for character classification on these respective datasets for computing the F1 Score and frame accuracy (exact match). Since the DiDeMoSV dataset does not have recurring characters, we do not evaluate performance of our models on these datasets using character classification.

[4] https://github.com/kakaobrain/minDALL-E

[5] https://github.com/adymaharana/storydalle

Table 5. Results on the validation sets of PororoSV, FlintstonesSV and DiDeMoSV (DSV) datasets from various models. Scores are based on FID (lower is better), character classification F1 and frame accuracy (F-Acc.; higher is better) evaluations.

Model	**PororoSV**			**FlintstonesSV**			**DSV**
	FID ↓	Char-F1↑	F-Acc↑	FID ↓	Char-F1↑	F-Acc↑	FID↓
StoryGANc (BERT)	63.94	54.02	24.53	87.65	71.98	55.68	93.21
StoryGANc (CLIP)	65.13	**54.83**	**25.29**	87.02	72.30	**59.35**	93.26
StoryDALL-E (prompt)	45.68	31.91	22.14	67.05	54.17	26.23	72.61
StoryDALL-E (finetuning)	**21.64**	40.28	20.94	**28.37**	**74.28**	52.35	**41.58**

E Additional Results

In this section, we present the results on validation sets of the three story continuation datasets discussed in Table 1 in main text.

Validation Set Results. We present results on the validation set of the three story continuation datasets discussed in main text i.e. PororoSV, FlintstonesSV and DiDeMoSV, in Table 5. The fully-finetuned StoryDALL-E model performs the best across all datasets in terms of FID score. The gains are seen in FID, due the high visual quality of the images generated by StoryDALL-E. However, the character classification and frame accuracy scores for the StoryDALL-E are close to those of StoryGANc for the FlintstonesSV dataset and relatively lower for the PororoSV dataset, in spite of being of better visual quality (as per manual analysis). This might be attributed to the fact that GAN-based models tend to generate some finer details of a character while sacrificing shape and form, which is recognized by character classification models as a faithful reconstruction. On the other hand, StoryDALL-E models focus on shape and form and tend to blur other defining characteristics, which are appealing to human eyes but fail to be recognized by the classification model.

Due to the higher resolution images generated by VQGAN-VAE [6], the visual quality of images produced by StoryDALL-E is highly preferred over predictions from the StoryGANc models. Similarly, the latent pretrained knowledge of DALL-E promotes generation of images that align well with the input captions, and results in higher wins for the StoryDALL-E model. The %wins and %loss are nearly uniform for the attribute *consistency* in this larger experiment, for the PororoSV and DiDeMoSV datasets. Predictions from the StoryDALL-E model are found to be more consistent than those of StoryGANc for the FlintstonesSV dataset. See predictions from StoryDALL-E for the PororoSV, FlintstonesSV and DiDeMoSV datasets in Figures 10, 11 and 12 respectively.

F Additional Analysis

In this section, we examine various aspects of the story continuation task, models and datasets. First, we demonstrate the advantages of the story continuation task over the story visualization task. Next, we calculate correlations between the source images and generated images from StoryDALL-E, with and without condition, to demonstrate the utility of cross-attention layers. Finally, we discuss the semantic content of our proposed DiDeMoSV dataset.

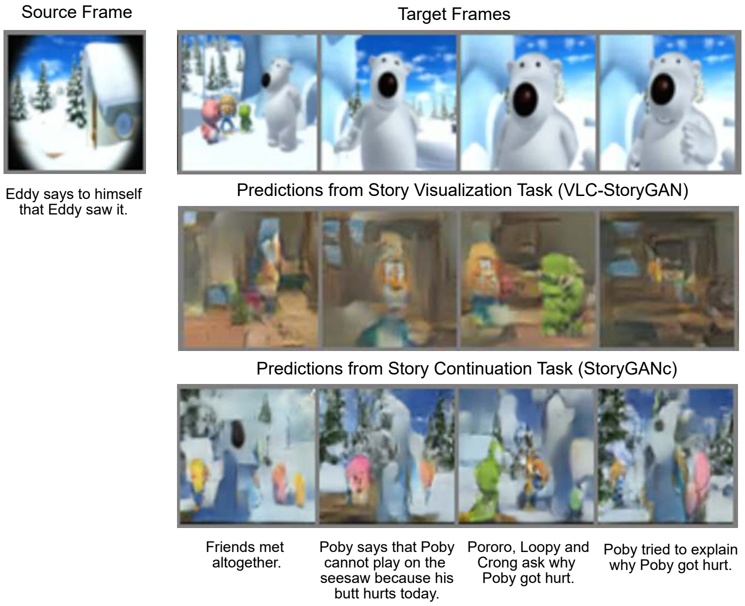

Source Frame
Target Frames
Eddy says to himself that Eddy saw it.
Predictions from Story Visualization Task (VLC-StoryGAN)
Predictions from Story Continuation Task (StoryGANc)
Friends met altogether.
Poby says that Poby cannot play on the seesaw because his butt hurts today.
Pororo, Loopy and Crong ask why Poby got hurt.
Poby tried to explain why Poby got hurt.

Fig. 8. Comparison of predictions from state-of-the-art story visualization model VLC-StoryGAN (middle) and our story continuation model StoryGANc (bottom) for a sample from the PororoSV dataset (top).

F.1 Story Visualization vs. Story Continuation

In Figure 8, we present a comparison of predictions from the state-of-the-art story visualization model VLCStoryGAN [31] and our story continuation model StoryGANc for a sample from the test set of the PororoSV dataset. Story Visualization relies only on the input captions to generate the images from scratch. However, as discussed in Section 3.1 in the main text, the captions in story visualization datasets are short and do not contain information about the setting and background elements. As a result, the predictions from story visualization models rely on data seen in the training set to infer arbitrary visual elements. In Figure 8, the story takes place in a snowy field with trees (top), but the prediction from VLCStoryGAN (middle) depicts the story as taking place indoors. When the first frame is given as additional input to our model StoryGANc in the story continuation task, the models borrows the snowy fields from the source frame and creates the story within that setting (bottom). Hence, story continuation is a more realistic and practical version of story visualization that can enable significant progress in research and faster transfer of technology from research to real-world use cases. Our experiments and datasets demonstrate the utility of this task.

F.2 Correlation between Source and Generated Images

We also measure the cosine similarity between the source frames and the generated frames from StoryDALL-E, with and without the retro-fitted cross-attention layer for conditioning on a source image, as a representation of the correlation between the two sets of images. We encode the images using the CLIP image encoder ViT-B/16 and report the mean and standard deviation of cosine similarity values for each dataset (see Table 6). We see up to 0.3 points increase in correlation between the source image and generated image for all three datasets with the use of the conditioning mechanism.

Table 6. Mean and standard deviation of correlation between source image and generated images from StoryDALL-E without and with conditioning on the source image.

Dataset	**without condition**	**with condition**
PororoSV	0.23 ± 0.04	0.26 ± 0.04
FlintstonesSV	0.38 ± 0.05	0.41 ± 0.03
DiDeMoSV	0.16 ± 0.04	0.19 ± 0.01

F.3 Semantic Analysis of the DiDeMoSV dataset.

Figure 9 contains counts for (A) noun chunks, (B) verbs and (C) object classes in DiDeMoSV. As discussed in Section C, DiDeMoSV is collected from Flickr and the most common nouns indeed reflect this. Most of the captions are descriptive in that they describe the contents of the scene, the location of the objects/people in the scene, and the actions that are taking place in the scene. In DiDeMoSV, the focus is on the breadth of information that must be considered in the form of actions, objects, and settings.

The graph for the frequency of verbs across the captions in the DiDeMoSV dataset (see (B) in Figure 9) illustrates the complexity of the actions that are being undertaken by agents in the story. It can be seen that most of the actions are simplistic and related to movement, such as “walks", “comes", “starts", “turns", “goes", etc. A lot of the verbs are also centered around vision, such as “see", “seen", and “looks". While these words corroborate our prior insights reflecting the relative simplicity of the stories in DiDeMoSV, they also are crucial for understanding simplistic event chains. An understanding of these simple verbs and the way that they affect the story goes a long way towards facilitating story continuation, especially in the many settings of DiDeMoSV.

Part (C) in Figure 9 contains a breakdown of the objects that appear in the DiDeMoSV images. To generate these graphs, we use Yolov3 [39] to process each of the images in the respective datasets. The 'person' class is the dominant class in both datasets. This intuitively makes sense due to the initial data sources from which the respective video captioning datasets were constructed. Additionally, it matches the pattern that is observed in the caption noun analysis, where the nouns in both datasets are most frequently referring to people. However, we can also see that there are limitations of the Yolov3 model. There are frequently occurring nouns, such as 'camera' in DiDeMoSV that are not able to appear in our image analysis because these do not have corresponding classes in the model. We use the default confidence threshold of 0.25 in the Yolo model, which generates predictions for only 76% of DiDeMoSV images.

Our analysis demonstrates the diversity of the DiDeMoSV dataset, and showcases it as a challenging benchmark for the story continuation task, in addition to PororoSV and FlintstonesSV.

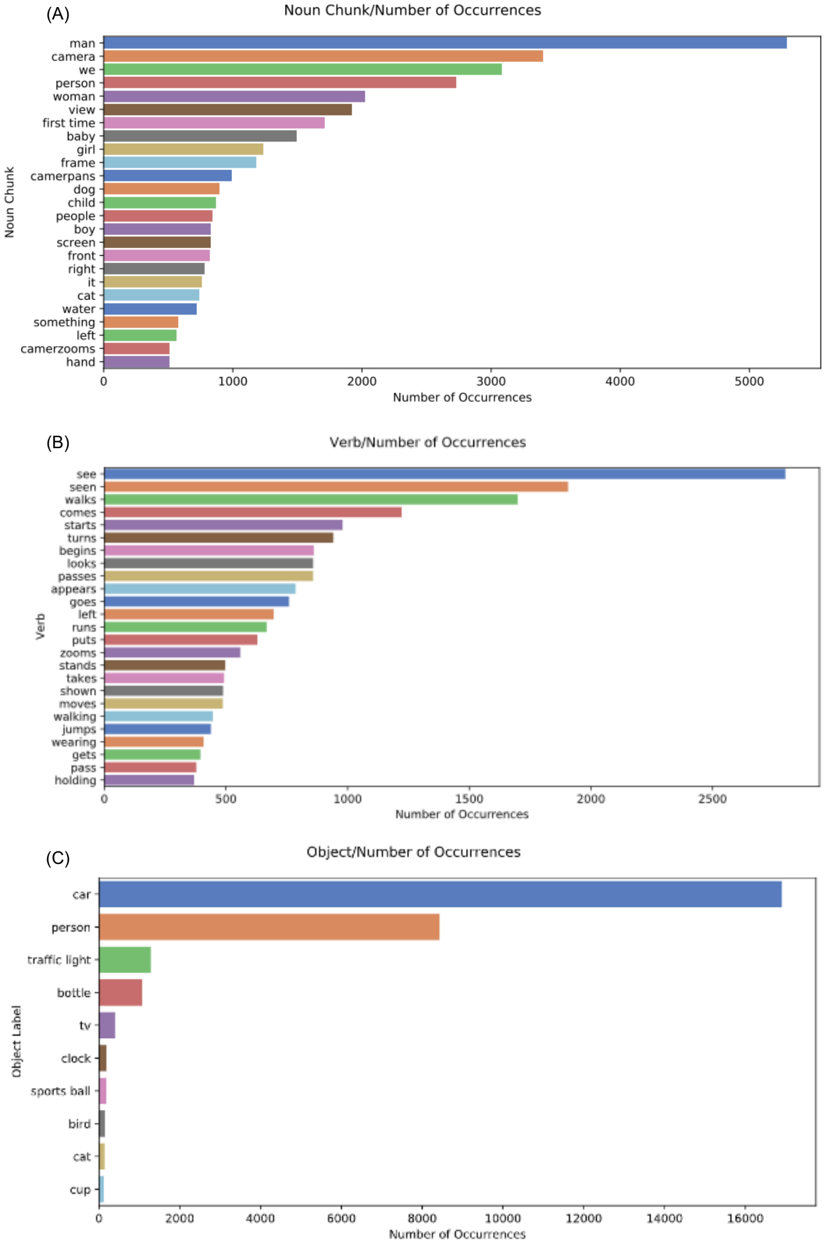


Fig. 9. Plots for frequency of (A) noun chunks and (B) verbs in the captions and (C) objects in the frames of the DiDeMoSV dataset.

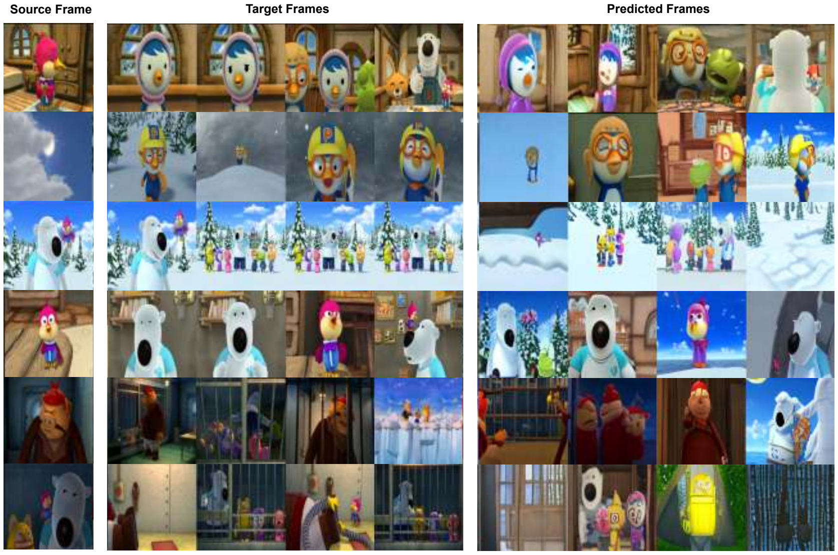


Fig. 10. Generated samples from StoryDALL-E for the PororoSV dataset.

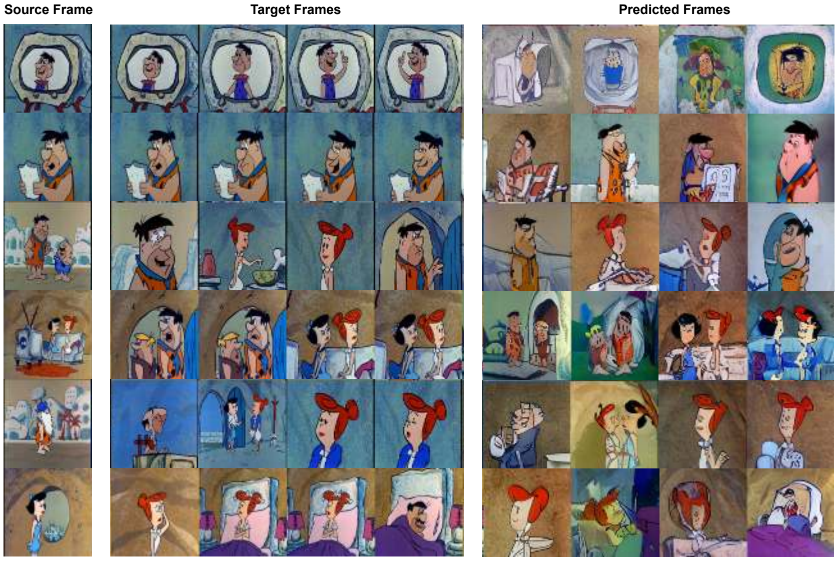


Fig. 11. Generated samples from StoryDALL-E for the FlintstonesSV dataset.

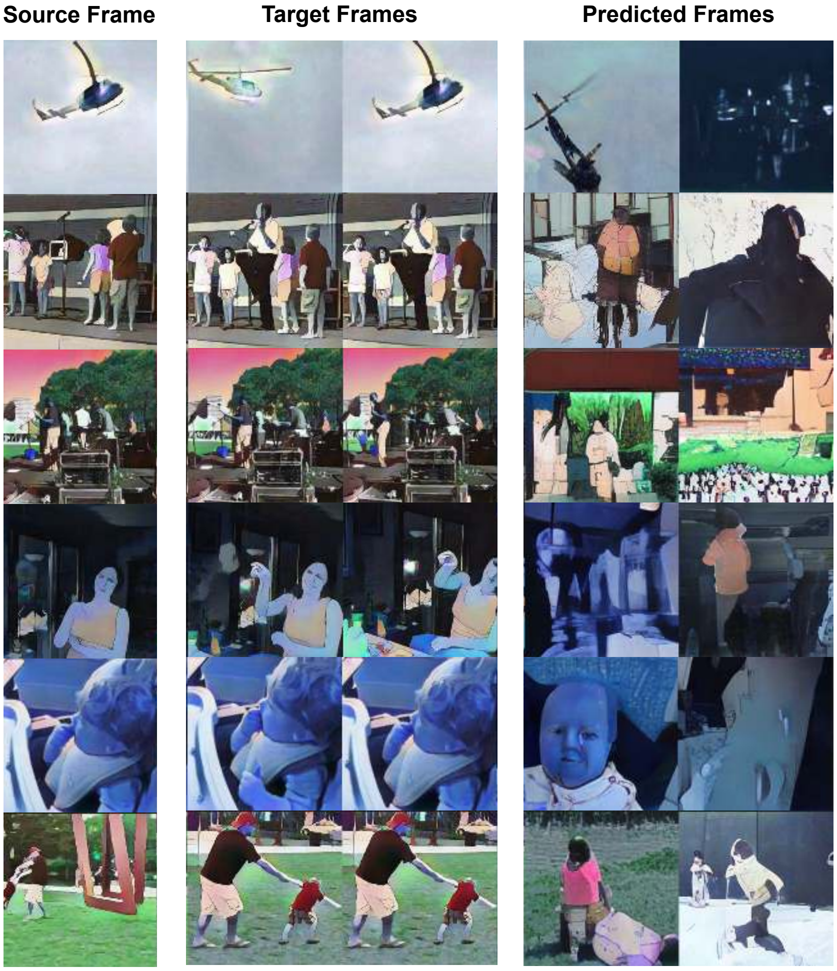


Fig. 12. Generated samples from StoryDALL-E for the DiDeMoSV dataset.