Integrating Visuospatial, Linguistic and Commonsense Structure into Story Visualization

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

Abstract

While much research has been done in text-to-image synthesis, little work has been done to explore the usage of linguistic structure of the input text. Such information is even more important for story visualization since its inputs have an explicit narrative structure that needs to be translated into an image sequence (or visual story). Prior work in this domain has shown that there is ample room for improvement in the generated image sequence in terms of visual quality, consistency and relevance. In this paper, we first explore the use of constituency parse trees using a Transformer-based recurrent architecture for encoding structured input. Second, we augment the structured input with commonsense information and study the impact of this external knowledge on the generation of visual story. Third, we also incorporate visual structure via bounding boxes and dense captioning to provide feedback about the characters/objects in generated images within a dual learning setup. We show that off-the-shelf dense-captioning models trained on Visual Genome can improve the spatial structure of images from a different target domain without needing fine-tuning. We train the model end-to-end using intra-story contrastive loss (between words and image sub-regions) and show significant improvements in visual quality. Finally, we provide an analysis of the linguistic and visuo-spatial information.[1]

1 Introduction

Story Visualization is an emerging area of research with several potentially interesting applications such as visualization of educational materials, assisting artists with web-comic creation etc. Each story consists of a sequence of images along with a sequence of captions describing the content of the images. The goal of the task is to reproduce the images given the captions. It is more challenging than conventional text-to-image generation (Reed et al., 2016) because the generative model needs to identify the narrative structure expressed in the sequence of captions and translate it into a story of images. Some critical features of a good story include consistent character and background appearances, relevance to individual captions as well as overall story, and coherent narrative. While recent text-to-image models (Ramesh et al., 2021; Cho et al., 2020; Li et al., 2019a) are successfully generating high-quality images, they are not directly designed for narrative understanding over sequential text. Hence, story visualization necessitates independent research towards developing generative models for the task. In this paper, we explore the use of visuo-linguistic structured inputs and outputs for improving story visualization. Towards this end, we propose (V)isuo-spatial, (L)inguistic & (C)ommonsense i.e. VLC-STORYGAN which (1) uses constituency parse trees and commonsense knowledge as input using structure-aware encoders, (2) leverages a pretrained dense captioning model for additional position and semantic information,

Figure 1: Example of Generated Images from our model VLC-STORYGAN and Duco-StoryGAN (Maharana et al., 2021) for the PororoSV dataset.

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

Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pages 6772–6786
November 7–11, 2021.

and (3) is trained using intra-story contrastive loss for maximizing global semantic alignment between input captions and generated visual stories.

Grammatical structures like constituency parse trees are potentially rich sources of information for visualizing relations between objects (or characters), their actions, and their attribute (property) modifiers. Wang et al. (2019); Nguyen et al. (2020); Xiao et al. (2017); Cirik et al. (2018) demonstrate that inducing such tree-structures within the encoder guides words to compose the meaning of longer phrases hierarchically and improves various tasks like masked language modeling, translation, visual grounding of language etc., suggesting potential gains in other tasks. Most text-to-image synthesis as well as story visualization models (Li et al., 2019c; Maharana et al., 2021) perform flat processing over free-text captions using LSTM or Transformer-based encoders. Hence, in order to leverage the grammatical information packed in constituency parse trees, we propose a novel Memory-Augmented Recurrent Tree-Transformer (MARTT) to encode captions and promote forward flow of hierarchical information across the sequence of captions for each story. Further, we find that input captions in story visualization lack details about the visual elements in the image. Hence, we augment the captions with external knowledge. For instance, when one caption mentions *snow* while the other mentions *icy roads*, we provide the knowledge that both are related to cold weather, encouraging the model to learn similar representations for either of the phrases.

Dual learning has served as an effective method for promoting desirable characteristics in target output for both text-to-image generation (Qiao et al., 2019) and story visualization (Maharana et al., 2021). Song et al. (2020) use image segmentation to preserve character shapes while Maharana et al. (2021) use video captioning for global alignment between the input caption and the generated sequence of frames. Each of these auxiliary tasks generate uni-modal outputs, dealing either with image or text. In a bid to combine the benefits of learning signals from both visuo-spatial and language modalities, we propose the use of dense captioning as the dual task, which has proven useful as a source of complementary information for many vision-language tasks (Wu et al., 2019; Kim et al., 2020; Li et al., 2019b). Dense captioning models provide regional bounding boxes for objects in the input image and also describe the region. By using these outputs for dual learning feedback for story visualization, the generative model receives a signal rich in spatial as well as semantic information. The spatial signal is especially important for our task since the input captions do not contain any specifications about the shape, size or position of characters within the story. Further, we find that off-the-shelf dense captioning models, which are trained on realistic images from Visual Genome, transfer well to a markedly different domain like cartoon and can be used to provide visuo-spatial feedback without finetuning on target domain.

Finally, we want the model to recognize the subtle differences between frames in a story and generate relevant images that fit into a coherent narrative. Hence, we employ contrastive loss between image-regions and words in the captions at each timestep to improve semantic alignment between the caption and image. Adjacent frames in a story often contain subtle differences, as can be seen in an example in Fig. 1. We modify the region-word contrastive loss proposed in Zhang et al. (2021) for story visualization by sampling negative images from adjacent frames, forcing the model to recognize the difference between frames. Overall, our contributions are: (1) We propose VLC-StoryGAN to use linguistic information, augmented with commonsense knowledge, for conditional image synthesis. (2) use dense-captioning to provide complementary positional and semantic information during training and show that off-the-shelf models trained on Visual Genome can be effective without fine-tuning on the target domain. (3) propose intra-story contrastive loss between image regions and words to improve semantic alignment between captions and visual stories. (4) achieve strong improvements in visual quality compared to previous state-of-art, and show the usefulness of structured inputs and outputs to provide insights for future work.

2 Related Work

Story Visualization. The task of story visualization and the model StoryGAN was introduced by Li et al. (2019c). Zeng et al. (2019) and Li et al. (2020) used textual alignment modules and Weighted Activation Degrees respectively, to improve performance of StoryGAN. Song et al. (2020) add a figure-ground generator and discriminator to preserve the shape of characters. Maharana et al. (2021) demonstrate the effectiveness of video cap-

tioning as a dual task for story visualization and propose additional evaluation metrics. Notable recent models in the related field of text-to-image generation are large (Brock et al., 2018), trained on gigantic datasets (Ramesh et al., 2021) and are based on Transformer architectures Jiang et al. (2021). Mask-to-image generation modules (Koh et al., 2021) have proven effective for smaller datasets containing detailed captions and additional information for aligning image sub-regions to words within captions (Pont-Tuset et al., 2020). This is in sharp contrast to the datasets available for story visualization, which have been repurposed from video QA datasets and hence, contain short descriptions. Our work is based on exploring structured inputs and outputs for conditional image synthesis which has been largely unexplored in text-to-image synthesis and story visualization.

Story Understanding & Commonsense. Iyyer et al. (2016) introduced Relationship Modelling Networks to extract evolving relationship trajectories between two characters in a novel. Chaturvedi et al. (2017) use latent variables to weigh predefined semantic aspects like topical consistency to improve encoding for story completion. Guan et al. (2019); Chen et al. (2019) augment story encodings with structured commonsense knowledge to improve story ending generation. We focus on the use of structured commonsense as well as grammatical trees to improve story encoding for the end goal of visualization.

Tree Encoder. Tree structures have traditionally been encoded using Tree LSTMs (Tai et al., 2015; Miwa and Bansal, 2016; Yang et al., 2017b,a). In recent work, Wang et al. (2019) enforce a hierarchical prior in the self-attention layer of Transformer (Vaswani et al., 2017) and Harer et al. (2019) use a parent-sibling tree convolution block to perform structure-aware encoding. Nguyen et al. (2020) use sub-tree masking and hierarchical accumulation to improve machine translation. We propose a simpler Tree-Transformer architecture, augmented with memory units (Lei et al., 2020) for recurrence.

Contrastive Loss. Xu et al. (2018) first proposed the contrastive loss in text-to-image synthesis through the Deep Attentional Multimodal Similarity Model (DAMSM). ContraGAN (Kang and Park, 2020) performs minimization of contrastive loss between multiple image embeddings in the same batch, in addition to class embeddings (Miyato and Koyama, 2018). Zhang et al. (2021) combine inter-modality and intra-modality contrastive losses and observe complementary improvements. We adapt inter-modal loss for story visualization by sampling negatives from adjacent frames.

Dense Captioning. Dense captioning jointly localizes semantic regions and describes these regions with short phrases in natural language (Johnson et al., 2016). Wu et al. (2019) and (Kim et al., 2020) use dense captions for visual and video question answering respectively. We use a pretrained dense captioning model to first annotate our target dataset and then use it within a dual learning framework to improve image synthesis for story visualization.

3 Methods

3.1 Background

Given a sequence of sentences $S = [s_1, s_2, ..., s_T]$, story visualization is the task of generating a corresponding sequence of images $\hat{X} = [\hat{x}_1, \hat{x}_2, ..., \hat{x}_T]$. 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 story encoder $E(.)$ consists of a recurrent encoder which takes word embeddings $\{w_{ik}\}$ for sentence s_k at each timestep k and generates contextualized embeddings $\{c_{ik}\}$. $E(.)$ also learns a stochastic mapping from S to a representation h_0 which encodes the whole story and is used to initialize hidden states of the recurrent encoder (Li et al., 2019c; Maharana et al., 2021). The image generator $I(.)$ takes $\{c_{ik}\}$ and pools them into representations $\{o_k\}$ which are then transformed into images $\{\hat{x}_k\}$. We train the model within a GAN framework (Goodfellow et al., 2014). The generated images are passed to image and story discriminators, which evaluate the images in different ways and send back a learning signal.

In VLC-StoryGAN, we use constituency trees as input to a structure-aware encoder. Further, we impose losses based on visuo-linguistic structures and contrastive loss on the model during training. We outline each of these modules in detail.

3.2 Memory-Augmented Recurrent Tree Transformer (MARTT)

Given a sentence s of length n, let $G(s)$ be the constituency parse tree of s produced by a parser. $T(s)$ denotes the ordered sequence of n terminal

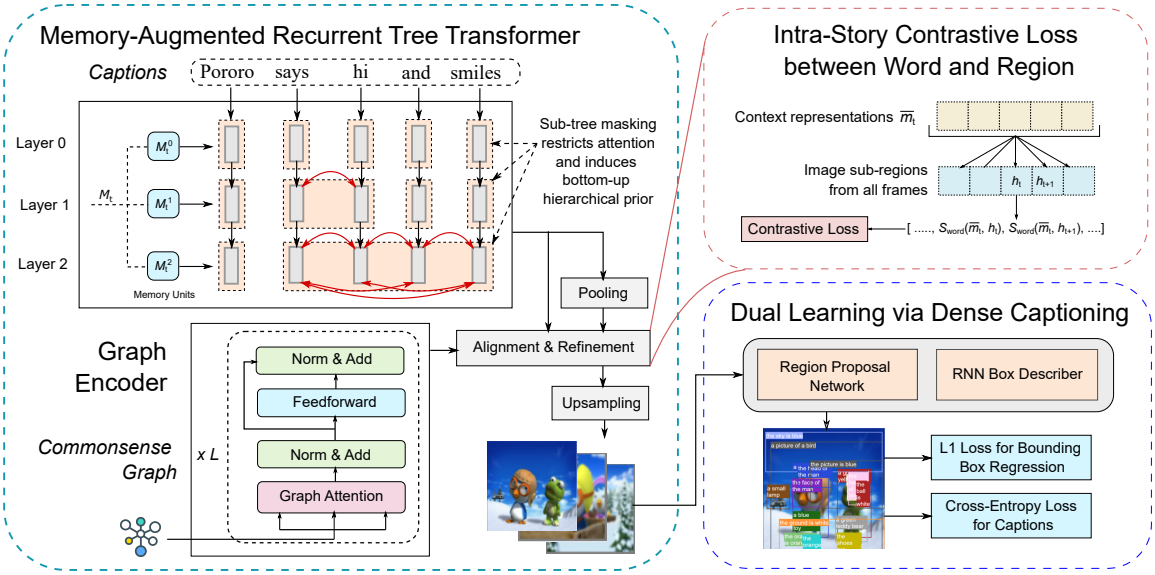


Figure 2: Illustration of VLC-STORYGAN architecture. The story encoder is composed of MARTT for encoding sequence of constituency parse trees, and Graph Transformer for encoding commonsense knowledge graphs. The intra-story contrastive loss optimizes semantic alignment while dense captioning loss provides visuo-spatial and semantic feedback about object/characters in generated images.

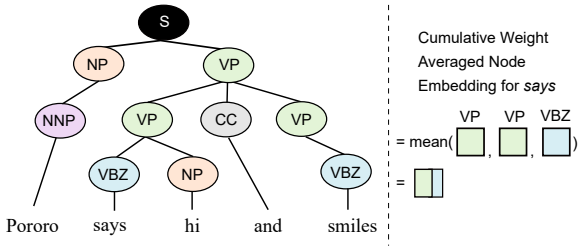


Figure 3: Upward cumulative avg. of node embeddings.

nodes (or leaves) of the tree and $N(s)$ denotes the set of nonterminal nodes (or simply nodes), each of which has a phrase label (e.g., NP, VP) and spans over a sequence of terminal nodes. Each leaf embedding is the concatenation of word embedding and a corresponding node embedding. To compute node embeddings, we perform upward cumulative average (Nguyen et al., 2020) over the nodes (phrase labels) that the respective non-terminal leaf token is a child of. For instance, as seen in Fig. 3, the node embedding for the word *Pororo* is the average of embeddings for NNP and NP. The node representations are learnt during training and provide information about the phrase label classes for each token. The encoder receives the sequence of leaf embeddings, in the same order as in the sentence, as input. Within the encoder, the hierarchical structure of a parse tree is promoted by introducing sub-tree masking for encoder self-attention (Wang et al., 2019). For each word query, self-attention has access only to other members of the sub-tree at that layer. In Fig. 2, each token only attends to itself in the first layer of Tree-Transformer. In the next layer, *says* and *hi* can attend to each other as they belong to the sub-tree rooted at VP. Consequently, all tokens within *says hi and smiles* can attend to each other in the third layer. This bottom-up approach, paired with node embeddings, induces the model to build a hierarchical understanding of the sentence through compositionality.

Tree Transformer was originally designed to encode a single tree input whereas in our task, we need to encode a sequence of trees for the sequence of images we plan to generate. Hence, we tie a series of Tree Transformers together by introducing memory cells and memory updater modules in each layer of self-attention. At time step t, the input query matrix within the self-attention layer attends over $[M^l_{t-1}; \bar{H}^l_t]$ where $M \in R^{T_m \times d}$ and $\bar{H} \in R^{T_c \times d}$ (T_m denotes memory state length and T_c denotes length of caption). The memory state M^l_{t-1} is updated to M^l_t following the steps outlined for memory updater in Lei et al. (2020).

3.3 Commonsense Knowledge

The input captions in most narrative datasets generally omit several relevant details about the plot or the background, which can be considered as commonsense. For example, in a scene where two characters are present outside on a sunny day, the

caption does not explicitly mention the presence of a sky in the background or the brightness of the sun. Hence, in order to introduce this external knowledge and enrich the input captions, we extract commonsense concepts relevant to each frame. To do so, we follow Bauer et al. (2018) and use a simple entity-based method to extract relevant paths from ConceptNet (see Sec 4 for details).

The commonsense knowledge paths are merged into a sub-graph which is then encoded using Graph Transformer. We use the Transformer-based graph encoder from Graph Writer (Koncel-Kedziorski et al., 2019) for structure-preserving encoding of graphs. First, the input graphs g_k are converted into unlabeled connected bipartite graphs $G_k = (v_k, E_k)$, where v_k is the list of entities and relations, and e_k is the adjacency matrix describing the directed edges (Beck et al., 2018). Next, v_k is projected to a dense, continuous embedding space V_k and is sent as input to the graph encoder. The encoder is composed of L stacked Transformer blocks; each Transformer block consists of a N-headed self-attention layer followed by normalization and a two-layer feed-forward network. The resulting encodings are referred to as graph contextualized vertex encodings. The entity encodings e_k are then appended to the output c_k from MARTT and used in the alignment module (see Fig. 2).

3.4 Image Generation

The image generator follows the two-stage approach in prior text-to-image generation works (Qiao et al., 2019; Xu et al., 2018; Zhang et al., 2017; Maharana et al., 2021). The alignment module performs attention-based semantic alignment (Xu et al., 2018) between image regions h_k and words $\bar{m}_k = [f_{entity}(e_k); f_{caption}(c_k)]$ in the current timestep. f_{entity} and $f_{caption}$ are dense layers for projecting commonsense and caption encodings respectively, into the same space as image embeddings. β_{jik} indicates the weight assigned by the model to the i^{th} word when generating the j^{th} sub-region of the image. For the j^{th} image sub-region, the word-context vector is calculated as:

$$a_{jk} = \sum_{i=0}^{L} \beta_{ji} \bar{m}_{ik}; \;\; \beta_{jik} = \frac{\exp(h_{jk}^T \bar{m}_{ik})}{\sum_{i=0}^{L} \exp(h_{jk}^T \bar{m}_{ik})}$$

The generated images are sent to image and story discriminators and the corresponding classification loss is used for training. We use the discriminator models proposed in Li et al. (2019c). Given the sentence s_k and the context information vector from the story encoder h_0, the image discriminator attempts to distinguish between the generated and ground truth image x_k, resulting in the loss $\mathcal{L}_{img}$. Similarly, the story discriminator classifies between the ground truth story and the generated sequence of images $\hat{X}$ to produce the loss $\mathcal{L}_{story}$. Additionally, the image discriminator is also used to classify the characters in the frame, when labels are available.

3.5 Dual Learning with Dense Captioning

As we discussed in Sec. 1, dual learning can provide important visual or semantic signals for improving story visualization, depending on which auxiliary task is chosen for the feedback model. We propose the use of dense captioning for providing visuo-spatial as well as semantic learning signals during training and use the model in Yang et al. (2017b) as the feedback model.[2] The dense captioning model is not fine-tuned on images from the story visualization dataset since it lacks dense caption annotations and it is prohibitively time-consuming and expensive to gather such annotations for the task. Hence, we explore the use of Visual Genome-based predictions (Krishna et al., 2017) as "proxy" annotations for our dataset (see Fig. 2). Using these noisy predictions as ground-truth, we train the generative model to optimize for bounding box loss (L1 regression; $\mathcal{L}_{bbox}$) as well as captioning loss (cross-entropy; $\mathcal{L}_{caption}$).

Position Invariance via Bounding Box Loss: The input captions in our dataset do not specify positions for the characters. Unless there is explicit positional input, it is unreasonable to expect the model to get the ground truth positions correct in generated images. Hence, in order to enforce positional invariance, we augment our dataset with mirror versions of the stories.

3.6 Contrastive Loss

As discussed in Sec. 3.4, the alignment and refinement module computes a pairwise cosine similarity matrix between all pairs of image-regions and word tokens, followed by the soft attention $\beta_{i,j}$ for image region j to word i. The aligned word-context vector a_j for the j^{th} sub-region is the weighted sum of all word representations. Following Zhang et al. (2021), the score function

[2] We use the implementation at https://github.com/soloist97/densecap-pytorch

between all sub-regions h_k for image x_k and all words $\bar{m}_k$ corresponding to caption s_k is defined as $S_{word}(x_k, s_k) = log(\sum_{j=1}^{N} exp(cos(h_{jk}, a_{jk})))$, where N is the total number of sub-regions. Finally the contrastive loss between the words and regions in image x_k and its aligned sentence s_k with respect to the story is defined as:

$$\mathcal{L}_{word} = -log \frac{exp(S_{word}(x_k, s_k))}{\sum_{m=1}^{T} exp(S_{word}(x_m, s_k))}$$

where T is the total number of frames in a story.

Conditioning Mechanism. The story encoder $E(.)$ encodes the entire story S into a single representation, h_0, which functions as the initial memory state of the MARTT model, similar to Maharana et al. (2021). 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 KL-Divergence loss between the learned distribution and Gaussian distribution i.e., $\mathcal{L}_{KL} = KL(\mathcal{N}(\mu(S), \text{diag}(\sigma^2(S)))||\mathcal{N}(0, I))$.

Objective. The final objective function of the generative model is $\min_{\theta_G} \max_{\theta_I, \theta_S} [\mathcal{L}_{KL} + \mathcal{L}_{img} + \mathcal{L}_{story} + \lambda_{bbox}\mathcal{L}_{bbox} + \lambda_{caption}\mathcal{L}_{caption} + \mathcal{L}_{word}]$ where θ_G, θ_I and θ_S denote the parameters of the entire generator, and image and story discriminator respectively. λ values are weight factors for the respective losses.

4 Experimental Settings

Evaluation. We adopt the metrics proposed in Li et al. (2019c) and Maharana et al. (2021):

- **Character Classification**: Frame accuracy (exact match) and classification F1-score using finetuned Inception-v3 to measure visual quality of recurring characters in predicted images. (Szegedy et al., 2016).
- **Video Captioning Accuracy**: BLEU2/3 scores of captions generated for predicted images using pretrained MART (Lei et al., 2020).
- **R-Precision**: R-Precision for global semantic alignment between predicted images and groud truth captions using the Hierarchical-DAMSM (Maharana et al., 2021).
- **Frechet Inception Distance (FID)**: The distance between distributions of real images and generated images using pretrained Inception-v3.

Since story visualization datasets are adapted from a video captioning dataset, sometimes a single frame does not represent the caption perfectly. However, during training, we sample a frame from the video every time, thus providing coverage for the entire video and association between all characters in the story and their representation in the frame. With this process, the model is able to observe all characters from the caption in the target frames during training time. During inference, our target is a static story, and not a video. Hence, we evaluate the predictions under the assumption that all characters should appear in the frame.

Dataset. We use the PororoSV dataset proposed in Li et al. (2019c), and the splits proposed in Maharana et al. (2021) to evaluate our approach. Each sample in PororoSV has 5 frames and 5 corresponding captions that form a narrative. There are 9 recurring characters throughout the dataset. 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. There are 10191/2334/2208 samples in training, validation and test splits respectively. The constituency parses are extracted and pre-processed using spaCy (Kitaev and Klein, 2018) and NLTK (Bird et al., 2009).[3] For commonsense knowledge, we first extract nouns and verb words from all of the captions in a story, and find ConceptNet triples (Speer et al., 2017) containing at least one of those words in the subject and object phrases. Next, we use pretrained GloVe embeddings (Pennington et al., 2014) to find a broader pool of words which are related to the words and find additional relevant triples. These triples are combined into knowledge graph inputs for each frame. We use the top ten bounding box and caption predictions from a dense captioning model pretrained on Visual Genome (Krishna et al., 2017) for dual learning.

Experiments. Our model is developed using PyTorch. 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 the validation set.

[3] https://spacy.io/universe/project/self-attentive-parser

Model	Char. F1	Frame Acc.	FID↓	BLEU2/3	R-Precision
StoryGAN (Li et al., 2019c)	18.59	9.34	49.27	3.24 / 1.22	1.51 ± 0.15
CP-CSV (Song et al., 2020)	21.78	10.03	40.56	3.25 / 1.22	1.76 ± 0.04
DUCO-STORYGAN (Maharana et al., 2021)	38.01	13.97	34.53	3.68 / 1.34	$\mathbf{3.56 \pm 0.04}$
VLC-STORYGAN (Ours)	**43.02**	**17.36**	**18.09**	**3.80 / 1.44**	3.28 ± 0.00

Table 1: Results on test split of PororoSV Dataset. Lower FID is better; higher is better for rest of the metrics.

Attribute	Win%	Lose%	Tie%
Visual Quality	62%	28%	10%
Consistency	38%	30%	32%
Relevance	22%	18%	60%

Table 2: Results from human evaluation. Win% = % times stories from VLC-STORYGAN was preferred over DuCo-StoryGAN, Lose% for vice-versa. Tie% represents remaining samples.

5 Results

5.1 Main Quantitative Results

The results on the PororoSV test set can be seen in Table 1. We compare our model VLC-STORYGAN to three baselines: StoryGAN (Li et al., 2019c), CP-CSV (Song et al., 2020) and DUCO-STORYGAN (Maharana et al., 2021) for PororoSV. The final rows contain results with VLC-STORYGAN, which outperforms previous models across most metrics for PororoSV. We see drastic improvements in FID score and sizable improvements in charcater classification as well as frame accuracy scores. This demonstrates the superior visual quality of stories visualized via our proposed method. There is a small improvement in BLEU score and a slight drop in R-Precision.

The captions in PororoSV correspond more accurately to a video segment than a single image sampled from the segment (see example in Fig. 1). Hence, even though the metrics BLEU and R-Precision have been shown to be correlated with human judgement in text-to-image synthesis (Hong et al., 2018), the PororoSV dataset fails to be an appropriate testing bed for extending those metrics to story visualization. Since they are adapted from video datasets, there is poor correlation between a single frame and the caption that originally spanned an entire video clip. This leads to unstable results and smaller improvement margins for both metrics. Instead, the dataset presents a data-scarce scenario where the captions do not provide sufficient details for accurate generation of visual stories. This leaves ample scope for augmenting the input with external visual information such as scene graphs and dense captions, or structured knowledge such as commonsense graphs, as we have shown with our proposed model. The structured information in VLC-STORYGAN leads to better generation of multiple characters, as compared to Duco-StoryGAN (Fig. 1).

5.2 Human Evaluation

We conduct human evaluation on the generated images from VLC-STORYGAN and DuCo-StoryGAN, using the three evaluation criteria listed in Li et al. (2019c): visual quality, consistence, and relevance (see Appendix for details). Predictions from our model for PororoSV are preferred 62% of the times for better visual quality (see Win% columns). Our model also produces more consistent and relevant images, but the higher % of ties between the two models for these attributes indicate that much work remains to be done to improve global alignment between captions and images.

We also examine 50 random samples from the PororoSV dataset, and evaluate whether the bounding boxes predicted by the pretrained dense captioning model used in our approach are relevant to the task i.e. whether more than 50% of the predicted bounding boxes for each sample capture a meaningful part of the frame. We observe a high accuracy for PororoSV i.e. 68%.

5.3 Ablations

Table 3 contains minus-one ablations for VLC-STORYGAN on the PororoSV validation set. The fourth row shows results from the complete model VLC-STORYGAN. We then iteratively remove each of our contributions and observe the change in metrics. We obtain the largest drops in FID, character classification and frame accuracy by replacing MARTT with the structure-agnostic MART (fifth row). This suggests that the constituency tree, as well as the MARTT architecture, aids in comprehension of captions. We see similar but smaller

Model	Char. F1	Frame Acc.	FID↓	BLEU2/3	R-Precision
VLC-StoryGAN	50.07	25.33	18.08	4.57 / 2.14	6.06 ± 0.00
- MARTT	48.96	22.84	24.56	4.12 / 1.59	5.86 ± 0.01
- Commonsense Embeddings	50.02	25.17	18.41	4.57 / 2.10	6.08 ± 0.02
- Dense Captioning	49.87	24.68	20.02	4.32 / 1.84	6.07 ± 0.00
- Intra-Story Contrastive Loss	48.65	24.98	21.67	4.43 / 1.92	5.99 ± 0.01

Table 3: Ablation results for our model on validation split of PororoSV dataset. Lower FID indicates better performance, higher is better for rest of the metrics. Dense captioning includes both bounding box and captioning.

drops with the exclusion of dense captioning from VLC-StoryGAN, since it provides important positional and semantic information about visual elements (seventh row). The minor margins for commonsense knowledge (sixth row) suggest that while it is a promising source of additional data, more work is needed for its proper integration with input captions. Finally, the results in the last row show that the intra-story contrastive loss is effective for global semantic alignment.

We also ran an experiment for isolating the effect of memory augmentation in our model, by training a non-recurrent (no memory) Transformer with Tree representations for single image generation instead of story generation, and evaluated using the story visualization metrics. We observed significant drops across all metrics.

6 Analysis and Discussion

In this section, we take a closer look at the various data sources for VLC-StoryGAN.

6.1 Linguistic & Commonsense Knowledge

Results from Table 3 show that the grammatical structure of caption contributes to better understanding, which translates to improved visual stories. The improvement in frame accuracy further suggests that MARTT improves comprehension of multiple characters simultaneously present in the narrative. In order to further analyze this premise, we examine a story involving several characters and compare predictions from VLC-StoryGAN and DuCo-StoryGAN in Fig. 4. The constituency parse tree in Fig. 4 shows the hierarchical understanding of the caption that is inherent in the MARTT architecture. Sub-tree masking allows the model to attend over multiple characters independently in earlier layers and combine the encoding in later layers. This semantic understanding is reflected in the image generated by VLC-StoryGAN which generates both characters mentioned in the caption distinctly, whereas Duco-StoryGAN barely generates one of them, validating the idea that grammatical knowledge is beneficial for story visualization.

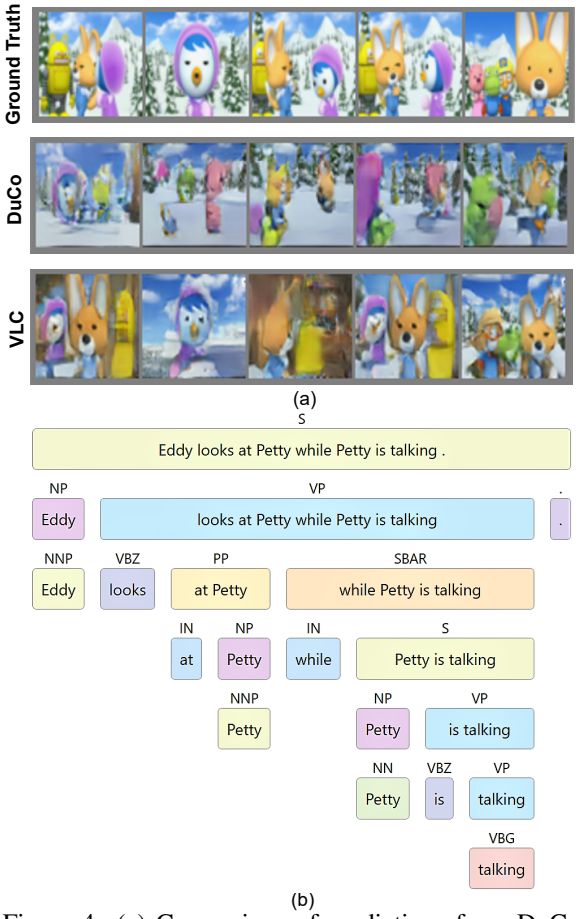
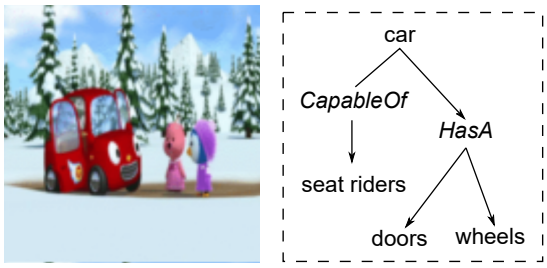


Figure 4: (a) Comparison of predictions from DuCo-StoryGAN and VLC-StoryGAN. (b) The constituency parse for caption of the fourth frame in (a).

In Fig. 5, we demonstrate an example of commonsense knowledge for a single frame in a story. We extract a sub-graph containing general information about car from ConceptNet (Speer et al., 2017) and use the graph contextualized embeddings from Graph Transformer for alignment with the generated image. The words *door* and *seat rider* correspond to specific sub-regions in the image and improve generalization.



Caption: The red car is offering Petty and Loopy a ride. The red car opens its doors for friends.

Figure 5: Example of commonsense knowledge for a given caption and its relevance to target image.

Figure 6: Dense captioning results on frames from the PororoSV dataset.

6.2 Analysis of Dense Caption Feedback

We use the dense captioning predictions on ground truth images in the PororoSV dataset in order to obtain the dual learning loss signal for VLC-StoryGAN during training. While we expected the predictions to be noisy, we found many of the predictions to be surprisingly relevant to the PororoSV dataset. For instance, most of the characters in PororoSV were identified as *teddy bear* or *stuffed toy or animal* and the dense captioning model provides roughly accurate bounding boxes for the entire character or prominent body parts (see Fig. 6). This explains the improvement in character classification scores with the addition of dual learning via dense captioning in our model. Many of the background elements in the stories, such as *blue sky*, *wooden table*, *snow*, and *green tree* look similar to their realistic counterparts in our cartoon setting. The captions are usually missing descriptions as well as positions of these minute details, whereas the dense captioning model provides precise locations and descriptions for the same.

Figure 7: Initial results on the Flintstones dataset.

6.3 Generalization to Flintstones Dataset

In order to measure the generalization of our approach to another dataset, we transformed the Flintstones dataset presented in the text-to-video synthesis work, CRAFT (Gupta et al., 2018), into story visualization. A single frame is sampled from each video clip and frames from adjacent clips are gathered into stories of length 5 (similar to PororoSV). The resulting dataset, FlintstonesSV, has 7 major recurring characters and has 20132/2071/2309 samples in the training, validation and test splits. Our model VLC-StoryGAN outperforms DuCo-StoryGAN on all metrics. We see 3.89% and 5.95% improvements in character F1-score and frame accuracy with our structured framework. Additionally, the FID drops by 9.23% suggesting large improvements in visual quality (see Fig. 7). Under human evaluation, predictions from VLC-StoryGAN are preferred as much or more than those from Duco-StoryGAN $\tilde{9}0\%$ of the times. The % of ties for all attributes is high, leaving scope for future research into this dataset.

7 Conclusion

In this paper, we investigate the use of structured knowledge for the task of story visualization. We propose a novel recurrent Tree-Transformer for encoding constituency trees and augment it with commonsense knowledge. We train the model using dense captioning loss and intra-story contrastive loss. Our results demonstrate the effectiveness of these approaches. We believe that these methods will encourage the use of structured knowledge for story visualization and text-to-image synthesis.

8 Ethics/Broader Impacts

The datasets and corresponding train/validation/test splits used in this paper were proposed by Li et al. (2019c), Kim et al. (2017), (Gupta et al., 2018) and Maharana et al. (2021). All the samples in the dataset consist of simple English sentences and cartoon images. Our experimental results are specific to the task of story visualization. The pretrained dense captioning model used in our paper is trained on English text and real-world images. All other models used and developed in our paper are trained on English text and cartoon images. By using cartoon images in our task, we avoid the egregious ethical issues associated with real-world usage of image generation such as DeepFakes. We focus not on generating realistic images, but on improved multi-modal understanding in the context of story visualization.

Acknowledgments

We thank Darryl Hannan, Hanna Tischer, Hyounghun Kim, Jaemin Cho, 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 W911NF18-1-0336, and a Google Focused Research Award. The views are those of the authors and not of the funding agency.

References

Lisa Bauer, Yicheng Wang, and Mohit Bansal. 2018. Commonsense for generative multi-hop question answering tasks. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4220–4230.

Daniel Beck, Gholamreza Haffari, and Trevor Cohn. 2018. Graph-to-sequence learning using gated graph neural networks. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 273–283.

Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural language processing with Python: analyzing text with the natural language toolkit*. " O'Reilly Media, Inc.".

Andrew Brock, Jeff Donahue, and Karen Simonyan. 2018. Large scale gan training for high fidelity natural image synthesis. In *International Conference on Learning Representations*.

Snigdha Chaturvedi, Haoruo Peng, and Dan Roth. 2017. Story comprehension for predicting what happens next. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1603–1614.

Jiaao Chen, Jianshu Chen, and Zhou Yu. 2019. Incorporating structured commonsense knowledge in story completion. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 6244–6251.

Jaemin Cho, Jiasen Lu, Dustin Schwenk, Hannaneh Hajishirzi, and Aniruddha Kembhavi. 2020. X-lxmert: Paint, caption and answer questions with multi-modal transformers. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8785–8805.

Volkan Cirik, Taylor Berg-Kirkpatrick, and Louis-Philippe Morency. 2018. Using syntax to ground referring expressions in natural images. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

Ian J Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron C Courville, and Yoshua Bengio. 2014. Generative adversarial nets. In *NIPS*.

Jian Guan, Yansen Wang, and Minlie Huang. 2019. Story ending generation with incremental encoding and commonsense knowledge. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 6473–6480.

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.

Jacob Harer, Chris Reale, and Peter Chin. 2019. Tree-transformer: A transformer-based method for correction of tree-structured data. *arXiv preprint arXiv:1908.00449*.

Seunghoon Hong, Dingdong Yang, Jongwook Choi, and Honglak Lee. 2018. Inferring semantic layout for hierarchical text-to-image synthesis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7986–7994.

Mohit Iyyer, Anupam Guha, Snigdha Chaturvedi, Jordan Boyd-Graber, and Hal Daumé III. 2016. Feuding families and former friends: Unsupervised learning for dynamic fictional relationships. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1534–1544.

Yifan Jiang, Shiyu Chang, and Zhangyang Wang. 2021. Transgan: Two transformers can make one strong gan. *arXiv preprint arXiv:2102.07074*.

Justin Johnson, Andrej Karpathy, and Li Fei-Fei. 2016. Densecap: Fully convolutional localization networks for dense captioning. In *Proceedings of*

the IEEE conference on computer vision and pattern recognition, pages 4565–4574.

Minguk Kang and Jaesik Park. 2020. Contragan: Contrastive learning for conditional image generation. In *NeurIPS 2020*. Neural Information Processing Systems.

Hyounghun Kim, Zineng Tang, and Mohit Bansal. 2020. Dense-caption matching and frame-selection gating for temporal localization in videoqa. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4812–4822.

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.

Nikita Kitaev and Dan Klein. 2018. Constituency parsing with a self-attentive encoder. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2676–2686.

Jing Yu Koh, Jason Baldridge, Honglak Lee, and Yinfei Yang. 2021. Text-to-image generation grounded by fine-grained user attention. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 237–246.

Rik Koncel-Kedziorski, Dhanush Bekal, Yi Luan, Mirella Lapata, and Hannaneh Hajishirzi. 2019. Text generation from knowledge graphs with graph transformers. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2284–2293.

Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. 2017. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International journal of computer vision*, 123(1):32–73.

Jie Lei, Liwei Wang, Yelong Shen, Dong Yu, Tamara Berg, and Mohit Bansal. 2020. Mart: Memory-augmented recurrent transformer for coherent video paragraph captioning. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2603–2614.

B. Li, Xiaojuan Qi, Thomas Lukasiewicz, and P. Torr. 2019a. Controllable text-to-image generation. In *NeurIPS*.

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

Hui Li, Peng Wang, Chunhua Shen, and Anton van den Hengel. 2019b. Visual question answering as reading comprehension. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6319–6328.

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

Adyasha Maharana, Darryl Hannan, and Mohit Bansal. 2021. Improving generation and evaluation of visual stories via semantic consistency. In *Proceedings of NAACL*.

Makoto Miwa and Mohit Bansal. 2016. End-to-end relation extraction using lstms on sequences and tree structures. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1105–1116.

Takeru Miyato and Masanori Koyama. 2018. cgans with projection discriminator. In *International Conference on Learning Representations*.

Xuan-Phi Nguyen, Shafiq Joty, Steven CH Hoi, and Richard Socher. 2020. Tree-structured attention with hierarchical accumulation. In *ICML*.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Jordi Pont-Tuset, Jasper Uijlings, Soravit Changpinyo, Radu Soricut, and Vittorio Ferrari. 2020. Connecting vision and language with localized narratives. In *European Conference on Computer Vision*, pages 647–664. Springer.

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*.

Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. 2021. Zero-shot text-to-image generation. *arXiv preprint arXiv:2102.12092*.

Scott Reed, Zeynep Akata, Xinchen Yan, Lajanugen Logeswaran, Bernt Schiele, and Honglak Lee. 2016. Generative adversarial text to image synthesis. In *International Conference on Machine Learning*, pages 1060–1069. PMLR.

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)*.

Robyn Speer, Joshua Chin, and Catherine Havasi. 2017. Conceptnet 5.5: An open multilingual graph of general knowledge. In *Thirty-First AAAI Conference on Artificial Intelligence*.

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.

Kai Sheng Tai, Richard Socher, and Christopher D Manning. 2015. Improved semantic representations from tree-structured long short-term memory networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1556–1566.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pages 6000–6010.

Yaushian Wang, Hung-Yi Lee, and Yun-Nung Chen. 2019. Tree transformer: Integrating tree structures into self-attention. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1060–1070.

Jialin Wu, Zeyuan Hu, and Raymond Mooney. 2019. Generating question relevant captions to aid visual question answering. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3585–3594.

Fanyi Xiao, Leonid Sigal, and Yong Jae Lee. 2017. Weakly-supervised visual grounding of phrases with linguistic structures. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5945–5954.

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*.

Baosong Yang, Derek F Wong, Tong Xiao, Lidia S Chao, and Jingbo Zhu. 2017a. Towards bidirectional hierarchical representations for attention-based neural machine translation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1432–1441.

Linjie Yang, Kevin Tang, Jianchao Yang, and Li-Jia Li. 2017b. Dense captioning with joint inference and visual context. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2193–2202.

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, Jing Yu Koh, Jason Baldridge, Honglak Lee, and Yinfei Yang. 2021. Cross-modal contrastive learning for text-to-image generation. *arXiv preprint arXiv:2101.04702*.

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*.

A Methods

A.1 Dense Captioning

For position invariance, we augment the PororoSV dataset with the mirror versions of the images and the corresponding mirror versions of the bounding box region predictions. When computing bounding box loss in dual learning, we compute the loss with the original bounding box prediction as well as its mirror version as target and retain the one which is lower. This way, we avoid penalizing the model for inverted positions of the characters since we do not provide explicit positional input to the model.

A.2 Story & Image Discriminators

We use the story and image discriminators as outlined in StoryGAN (Li et al., 2019c). The image discriminator is given the generated image $\hat{x}_k$, the sentence s_k, and the context information vector from the story encoder h_0, and distinguishes between a corresponding 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 also classifies the characters in the frame. The story discriminator evaluates the entire story S and the generated sequence of images $\hat{X}$.

A.3 Image Generation

The image generator follows the two-stage approach in prior text-to-image generation works (Qiao et al., 2019; Xu et al., 2018; Zhang et al., 2017; Maharana et al., 2021). The first stage uses outputs from the encoder; the resulting image is fed through a second stage, which weighs the outputs from the structure-aware context encoder as well as commonsense encoder, according to the image sub-regions and reuses for generation. The alignment module performs attention-based semantic

alignment (Xu et al., 2018) between image regions h_k and words $\bar{m}_k = [f_{entity}(e_k); f_{caption}(c_k)]$ in the current timestep. f_{entity} and $f_{caption}$ are dense layers for projecting commonsense and caption encodings respectively, into the same space as image embeddings. β_{jik} indicates the weight assigned by the model to the i^{th} word when generating the j^{th} sub-region of the image. For the j^{th} image sub-region, the word-context vector is calculated as:

$$a_{jk} = \sum_{i=0}^{L} \beta_{ji} \bar{m}_{ik}; \;\; \beta_{jik} = \frac{\exp(h_{jk}^T \bar{m}_{ik})}{\sum_{i=0}^{L} \exp(h_{jk}^T \bar{m}_{ik})}$$

B Experimental Settings

B.1 Evaluation

Li et al. (2019c) propose the character classification accuracy (exact match) within frames of generated visual stories as a measure of visual quality. Maharana et al. (2021) propose an additional set of automated evaluation metrics that capture diverse aspects of a model's performance on visual story generation. We adopt those metrics for evaluating our models:

- **Character Classification**: We use the finetuned Inception-v3 (Szegedy et al., 2016) and report frame accuracy and character F1-score.
- **Video Captioning Accuracy**: We use the pretrained MART video captioning model (Lei et al., 2020) and report BLEU2/3 scores for the generated captions.
- **R-Precision**: We use the Hierarchical-DAMSM (Maharana et al., 2021) to report R-Precision scores on the pairs of ground truth captions and generated stories.
- **Frechet Inception Distance (FID)**: We report the FID score, which is a metric used for evaluating the distance between real images and generated images for text-to-image synthesis datasets.

B.2 Hyperparameters

The image size that we use is 64-by-64, and the length of the story is 5 images/captions, same as DuCo-StoryGAN. The learning rates of the generator and discriminator are 2e-4. The model is trained for 150 epochs and the learning rate is decayed every 30 epochs. For each training update of the discriminators, two corresponding updates are performed for the generator network, with different mini-batch sizes for image and story discriminators (Li et al., 2019c). The image discriminator batch size is 60 and the story discriminator batch size is 12. We found in our experiments that story visualization models are prone to mode collapse at lower batch sizes, which is not resolved with perceptual loss in contrast to conventional knowledge. The above-mentioned hyperparameters are optimized using 12 iterations of manual tuning.

The MARTT 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 4. 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 which is initialized using the 840B glove training checkpoint. The node embedding size is 50.

The total number of trainable parameters in the VLC-StoryGAN is approximately 100M. We use the ADAM optimizer with betas of 0.5 and 0.999. We train the model on a single RTX A6000. Each epoch takes 50 minutes, with the model being saved every 10 epochs. At 150 epochs of training, the total training time is nearly 4 days.

C Results

See examples of predictions for PororoSV and FlintstonesSV from VLC-StoryGAN in Figures 8 and 9 respectively.

C.1 Human Evaluation

We conduct human evaluation on the generated images from VLC-StoryGAN and DuCo-StoryGAN, using the three evaluation criteria listed in Maharana et al. (2021): visual quality, consistence, and relevance. Two annotators are presented with a caption and the generated sequence of images from both models, and are asked to state their preferred sequence for each attribute. They also have the option to pick none if both images fare the same. In terms of visual quality, predictions from our model are preferred 62% of the times, as compared to 28% for DuCo-StoryGAN (see Win% columns) for PororoSV. Our model is also preferred more times for the attributes consistency and relevance, but the higher % of ties between the two models for these attributes indicate that much work remains to be done to improve global alignment between captions and images.

Figure 8: Example of generated images (left) from VLC-StoryGAN and corresponding ground truths (right).

Figure 9: Example of generated images (left) from VLC-StoryGAN and corresponding ground truths (right).