

Plot and Rework: Modeling Storylines for Visual Storytelling

Chi-Yang Hsu^{1,3*}, Yun-Wei Chu^{2*}, Ting-Hao (Kenneth) Huang¹, Lun-Wei Ku³

Pennsylvania State University¹, Purdue University²,

Institute of Information Science, Academia Sinica³

{cxh5437, txh710}@psu.edu

{chu198}@purdue.edu

{lwku}@iis.sinica.edu.tw

Abstract

Writing a coherent and engaging story is not easy. Creative writers use their knowledge and worldview to put disjointed elements together to form a coherent storyline, and work and rework iteratively toward perfection. Automated visual storytelling (VIST) models, however, make poor use of external knowledge and iterative generation when attempting to create stories. This paper introduces **PR-VIST**, a framework that represents the input image sequence as a story graph in which it finds the best path to form a storyline. PR-VIST then takes this path and learns to generate the final story via a re-evaluating training process. This framework produces stories that are superior in terms of diversity, coherence, and humanness, per both automatic and human evaluations. An ablation study shows that both plotting and reworking contribute to the model’s superiority.

1 Introduction

Writing a story is a complicated task. Human writers use their knowledge to tie all the disjointed elements, such as people, items, actions, or locations, together to form a coherent storyline. Writers also re-evaluate their work constantly during the writing process, and sometimes even alter their writing goals in the middle of a draft. Flower and Hayes (1981) characterize a solo writer’s cognitive process as a series of components in which the writer’s own knowledge is described as the long-term memory, and the planning, translating, and reviewing steps can occur in a recursive, interconnected manner. These creative steps are essential to human writing. However, automated visual storytelling (VIST) models that compose stories given five images (Huang et al., 2016) do not make extensive use of human knowledge to tie the elements to-

gether, nor do they use human insight to evaluate the outputs and guide the generation process.

As for linking elements, most works generate visual stories in an end-to-end fashion (Huang et al., 2016; Kim et al., 2018), treating the task as a straightforward extension of image captioning. Recent works have begun to use relations between entities to improve visual storytelling, but often narrow in a particular subset of relations, such as relations between elements within the same image (Yang et al., 2019), relations between two adjacent images (Hsu et al., 2020), or relations between scenes (Wang et al., 2020). The full potential of rich real-world knowledge and intra-image relations have yet to be fully utilized. As for re-evaluation, recent work uses reward systems (Wang et al., 2018b; Hu et al., 2019) or estimated topic coherence (Wang et al., 2019) to automatically assess the output story and guide the generation process. However, these approaches are often optimized towards predefined aspects such as image relevancy or topic coherence, which do not necessarily lead to engaging stories from a human perspective. In the cognitive process of human writing, the writer’s judgment is critical, and visual storytelling models could benefit by considering human ratings.

This paper introduces **PR-VIST**, a novel visual storytelling framework that constructs a graph and captures the relations between all the elements in the input image sequence, finds the optimal path in the graph that forms the best storyline, and uses this path to generate the story. An overview of PR-VIST is shown in Figure 1.

- **Stage 1 (Story Plotting):** PR-VIST first constructs a story graph for the image sequence by extracting various elements (i.e., term nodes, object nodes) from all the images and linking these elements using external knowledge (i.e., VIST graph, VG graph). PR-VIST then finds

* denotes equal contribution

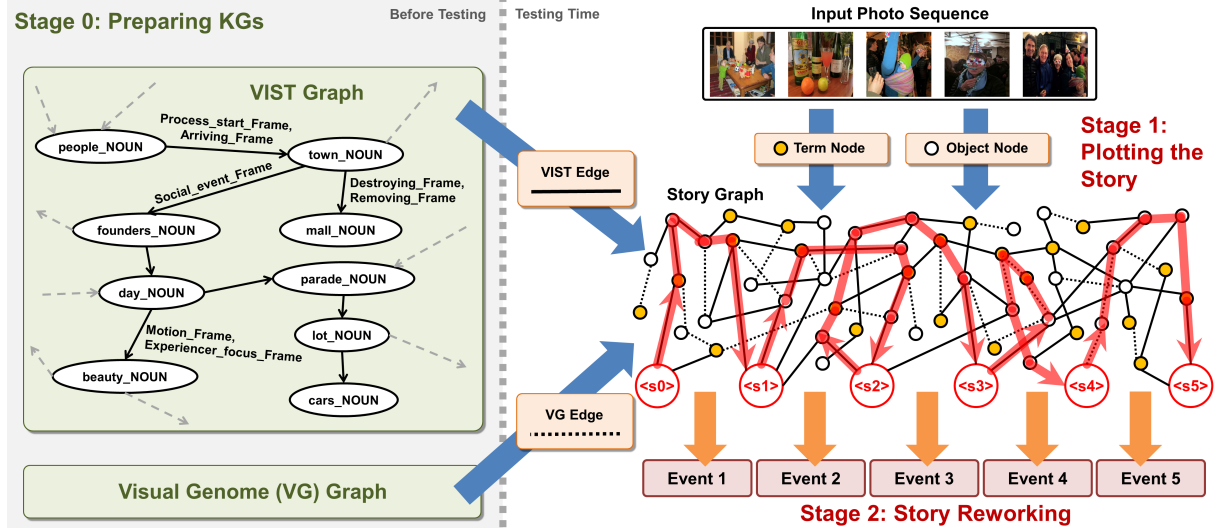


Figure 1: Overview of PR-VIST. In **Stage 1 (Story Plotting)**, PR-VIST first constructs a graph that captures the relations between all the elements in the input image sequence and finds the optimal path in the graph that forms the best storyline. In **Stage 2 (Story Reworking)**, PR-VIST uses the found path to generate the story. PR-VIST uses a story generator and a story evaluator to realize the “rework” process. In **Stage 0 (Preparation)**, a set of knowledge graphs that encode relations between elements should be prepared for the uses in Stage 1.

the best path in the story graph as the storyline and passes it to Stage 2.

- **Stage 2 (Story Reworking):** PR-VIST uses a story generator and a story evaluator to realize the reworking process: the generator takes the storyline produced in Stage 1 as the input to generate the story and backpropagates with an evaluator-augmented loss function. The evaluator, a discriminator model trained on human rating score data to classify good and bad stories, outputs a story quality score and modifies the loss. After a few optimization epochs, the generator eventually learns to generate stories that reflect human preferences.

In **Stage 0 (Preparation)**, a set of knowledge graphs that encode relations between elements are prepared for use in Stage 1. In this work, we prepare two knowledge graphs: a VIST graph and a visual genome (VG) graph. We construct the VIST graph based on the VIST dataset, representing in-domain knowledge; the VG graph is an existing resource (Krishna et al., 2017), representing generic knowledge. Note that as the PR-VIST framework is generic, it can use any knowledge graphs as needed.

Automatic and human evaluations show that PR-VIST produces visual stories that are more diverse, coherent, and human-like. We also conduct an ablation study to show that both story plotting (Stage 1) and reworking (Stage 2) contribute positively to

the model’s superiority. We believe this work also shows the potential of drawing inspiration from human cognitive processes and behavior to improve text generation technology.

2 Related Work

Visual Storytelling Researchers have been trying to advance the visual storytelling task since it was introduced by Huang et al. (2016). Some work modifies end-to-end recurrent models for better story generation (Hsu et al., 2018; Gonzalez-Rico and Fuentes-Pineda, 2018; Kim et al., 2018; Huang et al., 2019; Jung et al., 2020), and some use adversarial training to generate more diverse stories (Chen et al., 2017; Wang et al., 2018a,b; Hu et al., 2019). These methods produce legitimate stories and easier to implement because they relies only on one dataset. However, the generated stories can sometimes be monotonous and repetitive.

Leveraging External Resources for VIST Another set of work leverages external resources and knowledge to enrich the generated visual stories. For example, Yang et al. (2019) apply Concept-Net (Liu and Singh, 2004) and self-attention for create commonsense-augmented image features; Wang et al. (2020) use graph convolution networks on scene graphs (Johnson et al., 2018) to associate objects across images; and KG-Story (Hsu et al., 2020) is a three-stage VIST framework that uses

Visual Genome (Krishna et al., 2017) to produce knowledge-enriched visual stories.

Editing or Optimizing Visual Stories A few prior work tries to post-edit visual stories or optimize the story content toward specific goals. VIST-Edit is an automatic post-editing model that learns from an pre- and post-edited parallel corpus to edit machine-generated visual stories (Hsu et al., 2019). While VIST-Edit is useful, it requires parallel training data, which is often unavailable. Hu et al. (2019) use a reward function to optimize the generated stories toward three aspects; Li et al. (2019) customize the emotions of visual stories. These methods use automatic metrics to optimize visual stories toward specific goals; our work, on the other hand, leverages the human evaluation data to guide the generation process.

Story Plotting in Story Generation Research in automatic story generation has demonstrated the effectiveness of story plotting (Yao et al., 2018; Fan et al., 2019), which typically involves organizing the “ingredients” into a well-organized sequence of events. Nevertheless, none of the studies applied story plotting for visual stories.

3 Stage 0: Preparation

To prepare for story plotting, we collect information from the images and knowledge from the knowledge graphs.

3.1 Story Element Extraction

To extract information from the images, two extraction methods are used to extract image-oriented and story-oriented story elements: *objects* and *terms*, respectively representing image and story intuition.

Objects These can be detected by current object detection models, for which we use a pre-trained object detection model—Faster-RCNN (Ren et al., 2015). To ensure the detected objects’ reliability, only those objects with the top five confidence scores are used in each image.

Terms These are story-like nouns such as events, time, and locations, which current object detection models are unable to extract. Therefore, we further use a Transformer-GRU (Hsu et al., 2020) to predict story-like terms. For each image and story pair, we use image objects as the input and the nouns in the corresponding human-written story as

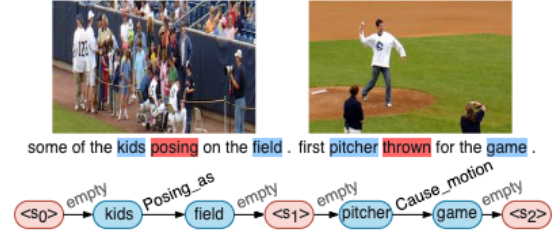


Figure 2: From the stories in VIST training data, a story is transformed into a golden storyline following the human reading direction, as part of the VIST graph.

the ground truth. The Transformer-GRU learns to convert objects to nouns commonly used in stories.

3.2 Knowledge Graph Preparation

To collect interactive relations between nouns, we prepare Visual Genome graph \mathcal{G}_{vg} and VIST graph \mathcal{G}_{vist} . These graphs contain interlinked real-world objects and terms, displaying visual and story-telling interaction. Table 1 summarizes the statistic of each graph.

Visual Genome Graph \mathcal{G}_{vg} describes pairwise relationships between objects in an image, describing visual interactions. No prepositional relations are included; only verb relations are preserved. All relations are converted into semantic verb frames using Open-SESAME (Swayamdipta et al., 2017), in which the semantic frames were pre-defined in FrameNet (Baker et al., 1998).

VIST Graph we propose \mathcal{G}_{vist} to collect the storytelling interactions. We develop this novel story knowledge graph by converting references in the VIST training and validation datasets (Huang et al., 2016) to graphical data. Following the reading direction, in each reference, we extract nouns and semantic verb frames using SpaCy¹ and Open-SESAME to obtain noun-verb-noun (NVN) tuples. Using nouns and semantic verb frames as nodes and edges, these are collectively assembled into a golden storyline. For example, for “first pitcher thrown for the game” in Figure 2, we extract *pitcher*, *game*, and *Cause_motion*, which is a semantic verb frame for *thrown*, as a NVN tuple. Additionally, we include a noun token $\langle s_i \rangle$ as the transition point to the next sentence or termination point of a story, and a verb frame token *empty_frame* to interlink two nouns when a semantic frame is absent. To conclude, all of the golden storylines are assembled into \mathcal{G}_{vist} .

¹SpaCy: <https://spacy.io/>

| | Nodes | Relations | Links |
|-------------------------|-------|-----------|-------|
| \mathcal{G}_{vg} | 3,323 | 564 | 22.31 |
| \mathcal{G}_{vist} | 2,048 | 531 | 11.75 |
| $\mathcal{G}_{vg+vist}$ | 4,158 | 880 | 22.78 |

Table 1: The statistics of knowledge graphs. The table shows the number of distinct nodes and relations in each graph. It also shows the average link per node. Note that the nodes and relations from \mathcal{G}_{vg} and \mathcal{G}_{vist} have overlaps.

4 Stage 1: Story Plotting

4.1 Storyline Predictor

In Stage 1, PR-VIST uses a storyline predictor to find what it deems the best path in the story graph as the storyline and then pass this to Stage 2. For the storyline predictor, we use UHop (Chen et al., 2019), a non-exhaustive relation extraction framework. A single hop is defined as searching from one entity to another entity by a single relation. UHop performs multiple single-hop classifications consecutively in the graph to find the path representing the storyline, that is, a path that consists of a sequence of nouns and verb frames.

Single-hop classification can be described as Equation 1 and Figure 3. In step i , at the current head entity h_i , the model is given a list of candidate relations $r_i \in R_i$ and tail entities $t_i \in T_i$. Each r_i is in [verb. t_i] or [verb.noun] format, containing information for both the verb frame and the tail noun entity. The scoring model F is given *objects* and predicted relations r_1, \dots, r_{i-1} as input. The model predicts a score for each r_i and selects the highest verb-noun pair r_i from Q :

$$r_i = \arg \max_{q \in Q} F(\text{objects}, r_1, \dots, r_{i-1}). \quad (1)$$

Training UHop learns to find a path for the storyline from the golden storyline. The training procedure starts with an initial noun token entity $\langle s_0 \rangle$ in the golden storyline for single-hop classification, where $h_1 = \langle s_0 \rangle$. It learns to select the correct relation r_i from a list of candidate relations R_i in \mathcal{G}_{vg} and \mathcal{G}_{vist} . Then, it calculates the error to the noun and verb frame in the golden storyline for backpropagation. In the next hop, the framework proceeds to the next noun in the golden storyline and repeats the single-hop classification.

Testing In PR-VIST’s testing step, for each story, five images are transformed into a story graph

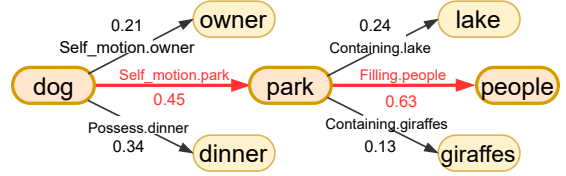


Figure 3: Storyline pathfinding process. All entities are from *object* or *term* list, and all relations are in [verb.noun] format, which the verbs are verb frames from knowledge graphs and the nouns are the tail entities. The single-hop classification begins with *dog*. The storyline predictor is given three candidate relations. The framework selects the highest score relation and move on to the next entity *park*. Then, the single-hop classification repeats.

\mathcal{G}_{story} . As demonstrated in Figure 1, we first extract the *object* and *term* story elements for each story, and then link these together using the verb frames in \mathcal{G}_{vist} and \mathcal{G}_{vg} as edges. This yields a well-defined graph presenting a comprehensive view of five images for each story— \mathcal{G}_{story} . Next, a trained UHop finds a storyline in \mathcal{G}_{story} , where all entities are only the *objects* and *terms* from the given five images. The framework starts with $\langle s_0 \rangle$ to perform single-hop classification, where $h_1 = \langle s_0 \rangle$. Unlike training, it only selects r_i from R_i listed in \mathcal{G}_{story} . In the next hop, the previous predicted entities are used as the start entity: $h_0 = t_{i-1}$. It then continues to hop from entity to entity until it reaches the next token $\langle s_1 \rangle$. The path from $\langle s_{i-1} \rangle$ to $\langle s_i \rangle$ is called an event e_i . The path search from $\langle s_i \rangle$ to the next token $\langle s_{i+1} \rangle$ continues until the search is terminated by the termination decision described in UHop. Eventually, the model finds a storyline of arbitrary length L , that is, a storyline that contains any number of events: e_1, \dots, e_L .

4.2 Implementation Details

HR-BiLSTM (Yu et al., 2018) is adopted as the scoring model F , in which *objects* are converted to word embeddings via GloVe (Pennington et al., 2014) as $E(\text{object})$. All relation embeddings $E(r)$ are decomposed into graphical embedding E_{graph} and textual embedding E_{text} . E_{graph} transforms a verb frame v and a tail entity’s image position p^t into a one-hot vector, denoting the graphical and image positional information. E_{text} is composed of the verb frame and tail entity word embedding. Then, E_{graph} and E_{text} are concatenated into a unified representation $E(r)$. We formulate the repre-

sensation of relation r as

$$\begin{aligned} E_{graph}(r) &= [\mathbb{1}(v); \mathbb{1}(p^t)], \\ E_{text}(r) &= [E_w(f); E_w(t)], \\ E(r) &= [E_{graph}(r); E_{text}(r)], \end{aligned} \quad (2)$$

where $\mathbb{1}(\cdot)$ returns the one-hot representation, $E_w(\cdot)$ returns the word embeddings via GloVe, and $[\cdot]$ denotes concatenation.

A verb frame and tail entity are combined into r_i due to relational ambiguity issues among candidate relations. Using Figure 3 as an example, given a head entity *dog*, candidates *self_motion.park* and *self_motion.owner* represent different semantic meanings when tail entities *park* and *owner* are included. However, excluding tail entities results in identical relation candidates *self_motion* and thus ambiguity between two different candidates.

5 Stage 2: Story Reworking

In story reworking, the framework consists of two components: the story generator and the story evaluator. The story generator generates a story according to the storyline, and the story evaluator—a discriminator trained on the MTurk human ranking data to classify good and bad stories—outputs a story quality score and modifies the loss functions.

5.1 Story Generator

A storyline consists of a set of events $e_1 \dots e_L$ that are input to the story generator, which is based on the Transformer (Vaswani et al., 2017). Unlike most VIST models, the story generator is dynamic: the number of output sentences depends on the number of events. To manage a diverse number of events, the Transformer is designed as a sentence generator that iteratively generates one sentence per event until it generates L sentences. For each step i , event e_i and the previous predicted sentence y_{i-1} are used to predict the next sentence y_i . After L steps, the story generator outputs a story $s = y_1, \dots, y_L$.

5.2 Story Evaluator

Most VIST works use human evaluations to examine their work’s quality via crowdsourcing, comparing their generated stories with the baseline stories. In this paper, we use the first- and last-place stories in the MTurk human ranking data as positive and negative samples. The story evaluator, a discriminator trained on the MTurk human ranking data, learns to distinguish positive and negative samples.

It outputs a score for each story, and the scores are converted into rewards, as shown below:

$$p_{LM}(u|s) = \text{softmax}(\tanh(\mathbf{W}LM(s)) + \mathbf{b}), \quad (3)$$

$$\mathcal{R}(s) = -p_{LM}(u|s) + c, \quad (4)$$

where $LM(\cdot)$ is a GRU language model (Cho et al., 2014), $u = 1$ indicates story s is a positive sample, and $u = 0$ indicates s is a negative sample. Language model $p_{LM}(\cdot)$ returns a score from 0 to 1 to reflect story quality. The story evaluator $\mathcal{R}(\cdot)$ returns a reward, an inverse of $p_{LM}(\cdot)$ with coefficient $c = 1.5$. The reward later manipulates the loss, optimizing toward human preference. Note that the story evaluator is pre-trained.

5.3 Optimization with Story Evaluator

For optimization, the story generator uses sentence-level and story-level loss functions. Given reference y_1^*, \dots, y_L^* and predicted story y_1, \dots, y_L , in the maximum likelihood estimation (MLE) optimization process, in each step from 1 to L , the model predicts a sentence y_i to calculate the loss between y_i and y_i^* and then backpropagates, as shown in Figure 4. After predicting L sentences, in story-level optimization, the model predicts y_1, \dots, y_L to calculate the negative log-likelihood to the reference y_1^*, \dots, y_L^* and then backpropagates. The sentence-level and story-level optimization by MLE on dataset \mathcal{D} are formulated as

$$J_{sen}^{MLE}(\theta, \mathcal{D}) = \sum_{Y \in \mathcal{D}} -\log p_\theta(y_i|e_i, y_{i-1}), \quad (5)$$

$$J_{story}^{MLE}(\theta, \mathcal{D}) = \sum_{Y \in \mathcal{D}} \frac{1}{L} \sum_{i=1}^L -\log p_\theta(y'_i|e_i, y'_{i-1}), \quad (6)$$

where e_i and y_i denote the i -th event and the sentence respectively, y'_i represents the updated sentence after sentence-level optimization, and θ represents the story generation model parameters, which are updated using Adam (Kingma and Ba, 2015).

After training for 30 epochs², the story evaluator begins to manipulate the story-level loss. Inspired by reinforcement learning (Williams, 1992), which utilizes rewards to guide the training process, we use the story evaluator $\mathcal{R}(\cdot)$ ³ to encourage the

²The generation model converges at around 20 epochs in our experiment, and we give it extra 10 epochs for precautions.

³The pre-trained LM’s weights are frozen to stabilize the training.

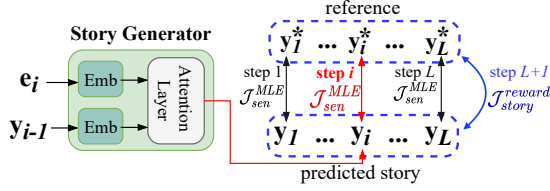


Figure 4: Optimization flowchart for story generator. For steps 1 to L , the model is optimized using sentence-level loss. In step $L+1$, all sentences are generated, and the model is optimized using story-level loss.

generation model to focus on stories preferred by humans. The reward directly multiplies the story-level loss as

$$J_{story}^{reward}(\theta, \mathcal{D}) = \begin{cases} J_{story}^{MLE} & \text{if epoch} \leq 30 \\ \mathcal{R}(s) J_{story}^{MLE} & \text{if } 30 < \text{epoch} \leq 60 \end{cases} \quad (7)$$

6 Experimental Results

6.1 Data Setups

We used four datasets in this paper: the VIST dataset, Visual Genome, ROCStories, and MTurk human ranking data. The VIST dataset and Visual Genome are used to construct the knowledge graphs, and ROCStories (Mostafazadeh et al., 2016) is a large quantity of pure textual stories used for pre-training the story generator. The VIST dataset is also used in story plotting to train the storyline predictor and in story reworking to fine-tune the story generator. Notably, we also collected MTurk human ranking data to train the story evaluator. We used the ranking results from KG-Story⁴ (Hsu et al., 2020). This data contains two experiments, each of which contains 500 distinct photo sequences. A photo sequence contains a set of machine-generated stories ranked by 5 MTurk workers. Thus we have 5000 rankings from MTurk workers. Specifically, MTurk workers were asked to rank AREL (Wang et al., 2018a), KG-Story, two KG-Story ablation models, and reference stories, using three different model settings in each experiment. We selected the rank-1 and rank-5 stories as positive and negative samples.

6.2 Baselines

We used several competitive baseline models. **AREL** (Wang et al., 2018a) and **GLAC** (Kim

et al., 2018) are end-to-end models with reinforcement learning and global-location attention mechanisms that achieved top ranks in the VIST Challenge (Mitchell et al., 2018). **KG-Story** (Hsu et al., 2020), the current state-of-the-art framework, utilizes a knowledge graph to enrich story contents and generates stories using Transformer.

6.3 Evaluation Methods

Per the literature (Wang et al., 2018a), human evaluation is the most reliable way to evaluate the quality of visual stories; automatic metrics often do not align faithfully to human judgment (Hsu et al., 2019). Therefore, in this paper, we prioritize human evaluation over automatic evaluations.

Human Evaluation: Ranking Stories and Filling a Questionnaire

We recruited crowd workers from Amazon Mechanical Turk (MTurk) to assess the quality of the generated stories. For each experiment, we randomly selected 250 stories, each of which was evaluated by five different workers in the US. The experiment includes a comparison study with three baseline models and three ablation studies, and each annotator was compensated \$0.10 for each experiment. Workers were asked to rank the stories (*e.g.*, ours and those of the baseline/ablation models) based on their overall quality.

In addition, the user interface also provides a questionnaire to collect in-depth feedback from MTurk workers. The questions include “What do you like about the best stories” and “What do you dislike about the worst stories” for workers to select aspects that affect overall story quality. These aspects are provided by Huang et al. (2016): they include focus, coherence, shareability, humanness, grounding, and detail. We calculated the average rank and the majority rank among five workers for each story, as well as total votes for each model’s best and worst aspects.

Non-Classic Automatic Evaluation: BLEURT, voc-d, and MLTD

Many VIST studies have shown that classic automatic evaluation scores like BLEU and METEOR correlate poorly with human judgment (Hsu et al., 2020; Hu et al., 2019; Wang et al., 2020; Li et al., 2020; Yang et al., 2019; Hsu et al., 2019; Wang et al., 2018a; Modi and Parde, 2019). These n-gram matching metrics fail to account for the semantic similarity to the reference stories and lexical richness in the generated stories.

Therefore, we adopted BLEURT (Sellam et al., 2020), a state-of-the-art BERT-based evaluation

⁴Data obtained from the authors of KG-Story.



AREL 🧑: avg rank #3.6

the parade started with a lot of people in the parade. there were a lot of people there. there was a lot of people there. there were a lot of people there. there were a lot of cars in the parade.

GLAC 🧑: avg rank #2.4

the marathon was about to begin. there were many people there. it was a great day. everyone was very excited. they were all very fast.

KG-Story 🧑: avg rank #2.2

the parade started with a beautiful day. many people showed up. there were runners. everyone was watching the parade. one woman stood in a car to get everyone involved. it was a nice car.

PR-VIST 🧑: avg rank #1.8

we went to see the parade. there was a band guard. many people showed up to watch the parade runners. the runners started to gather in line. there were many cars at the show. the family decided to walk around town.

Figure 5: Generated stories for PR-VIST and baseline models. MTurk workers were asked to rank the stories.

metric, to further correlate generated stories and reference stories based on their semantic meaning. We also adopted lexical diversity metrics voc-d and MLTD (McCarthy and Jarvis, 2010) to quantify story lexical richness. Several works have shown that lexical diversity is positive correlated to story quality (Liu et al., 2019; Dai et al., 2017).

6.4 Results

In our experiments, the stories generated by PR-VIST have an average of 5.96 sentences. 57.3% of these stories contain at least one event (sentence) that uses story elements extracted from two (or more) images, showing PR-VIST’s ability to utilize intra-image entities.

Human Evaluation We asked MTurk workers to rank four stories: those of PR-VIST, the three baseline models, and the state-of-the-art KG-story. Table 2 shows the results. PR-VIST outperforms other models in average ranking: it outranks AREL by 0.24 and KG-Story by 0.16. As for the percentage of 1st-rank stories, PR-VIST produces 12.0% more than AREL and 7.5% more than KG-Story. Figure 5 shows a representative example. Compared with end-to-end models (*i.e.*, AREL and GLAC), graph-based methods (*i.e.*, KG-Story and PR-VIST) generate more diverse stories. Compared with KG-Story, whose sentences are relatively simple and plain, generating sentences such as “Many people showed up”, our model reuses entities such as “parade” in the first sentence and associates relations with other entities, *e.g.*, “people” and “runners”, to compose “many people showed up to watch the parade runners”.

Moreover, Figure 6 shows the questionnaire (see

| Method | 1st | 2nd | 3rd | 4th | Avg | Major |
|----------|--------------------|-------|-------|--------------|-------------|-------------|
| AREL | 20.6% (258) | 26.8% | 27.2% | 25.4% | 2.57 | 2.56 |
| GLAC | 21.7% (271) | 24.2% | 25.5% | 28.6% | 2.61 | 2.73 |
| KG-Story | 25.1% (314) | 25.2% | 25.7% | 24.0% | 2.49 | 2.53 |
| PR-VIST | 32.6% (407) | 23.7% | 21.7% | 22.0% | 2.33 | 2.28 |

Table 2: Human rankings between PR-VIST and three methods. The first four columns indicate the percentage of worker rankings for each method, and the fifth and the last column denote the average and majority ranks (1 to 4, lower is better). PR-VIST outperforms other models in average ranks ($p < 0.05$, $N=250$), majority ranks, and also the percentage of 1st-rank stories.

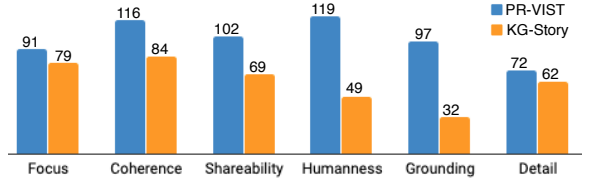


Figure 6: Aspect-wise votes for PR-VIST and KG-Story’s first-place stories collected via the questionnaire (see Section 6.3). PR-VIST outperforms drastically in coherence, humanness, and grounding.

Section 6.3) result for the best-ranked stories. For PR-VIST and KG-Story’s best-ranked stories, the PR-VIST story count is significantly higher in all aspects; specifically, coherence, shareability, and humanness are higher than other categories.

Automatic Evaluations Table 3 shows that the proposed method outperforms all the baselines in BLEURT, voc-d, and MLTD. Although n-gram-based automatic metrics are known to correlate poorly with human judgment in VIST (see Section 6.3), it is still noteworthy that PR-VIST results in significantly lower BLEU-4 scores. This might be caused by the fact that PR-VIST uses knowledge to enrich the story content and increase lexical diversity, but could lower the performance in n-gram matching.

| Method | BLEU-4 | METEOR | BLEURT | MLTD | voc-d |
|----------|-------------|-------------|-------------|--------------|-------------|
| AREL | 14.4 | 35.4 | 0.52 | 22.45 | 0.53 |
| GLAC | 10.7 | 33.7 | 0.71 | 32.87 | 0.67 |
| KG-Story | 9.93 | 32.2 | 0.72 | 40.52 | 0.71 |
| PR-VIST | 7.65 | 31.6 | 1.37 | 45.79 | 0.73 |

Table 3: The first two columns show the results of classic n-gram based metrics. The third column shows BLEURT, a BERT-based metric. The last two columns show the lexical diversity evaluation results (MLTD and voc-d). High lexical diversity corresponds to low scores for n-gram metrics.

7 Ablation Study

Three factors contribute to PR-VIST’s superior performance: story elements, knowledge graphs, and plot reworking. To evaluate the effectiveness of each factor in our framework, we conducted three ablation studies using human evaluations. The evaluation results are shown in Table 4. All three experiments use the same qualitative analysis, and each experiment ranks PR-VIST and two settings with certain components removed.

Story Elements PR-VIST is compared to two models, each of which uses only *objects* or *terms* for the storyline predictor to plot storylines.

Knowledge Graphs PR-VIST is compared to two models, each of which uses only \mathcal{G}_{vist} or \mathcal{G}_{vg} for the storyline predictor to plot storylines.

Plot and Rework PR-VIST is compared to two models: one without reworking and one without plotting or reworking. Without-reworking means the discriminator is removed, that is, the story generator uses Equation 6 for all epochs. Without-plotting-reworking means that the storyline predictor is additionally removed, so no frames are included; *terms* are used directly as the story generator’s input.

Table 4 shows that PR-VIST outperforms all the ablation models. Furthermore, the first and second experiments show that MTurkers prefer story-like storylines to image-like storylines. That is, *terms* and \mathcal{G}_{vist} are better than *objects* and \mathcal{G}_{vg} . For the third experiment, we note a steady improvement from without-plotting-reworking to PR-VIST, showing the effectiveness of the proposed method. An example is shown in Figure 7. The model cannot manage the abundant story elements without the guidance of story plotting. Comparing PR-VIST with PR-VIST w/o R, we see that reworking revises and enlivens (e.g., “[organization] in [location]”) the stories.

8 Discussion

To understand areas for improvement, in the human ranking evaluation, we asked crowd workers to select the aspect (out of six) they disliked about the worst story (see Section 6.3.) Of the negative votes, 24.6% were for “grounding.” Namely, lower-ranked stories are often not visually grounded. We examined the outputs and found that Faster-RCNN

| | <i>objects</i> | <i>terms</i> | \mathcal{G}_{vg} | \mathcal{G}_{vist} | Plot | Rework | Avg | Major |
|---|----------------|--------------|--------------------|----------------------|------|--------|------|-------|
| 1 | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | 1.89 | 1.87 |
| | | ✓ | ✓ | ✓ | ✓ | ✓ | 1.98 | 2.00 |
| | ✓ | | ✓ | ✓ | ✓ | ✓ | 2.12 | 2.13 |
| 2 | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | 1.97 | 1.94 |
| | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | 1.98 | 1.99 |
| | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | 2.00 | 2.06 |
| 3 | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | 1.95 | 1.93 |
| | ✓ | ✓ | ✓ | ✓ | ✓ | | 2.00 | 2.02 |
| | ✓ | ✓ | ✓ | ✓ | | | 2.03 | 2.08 |

Table 4: Human evaluation results for ablation studies: 1. Story elements 2. Knowledge graph 3. Plot and Rework. PR-VIST outperforms in all ablation settings, indicating the importance of using all components.

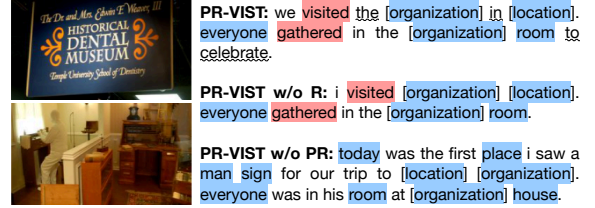


Figure 7: Snippet of stories generated by the proposed method and two configurations: without reworking (R) and without plotting or reworking (PR). Nouns and verb frames are denoted in blue and red. Reworked parts are marked with wavy underlines.

in Stage 1 sometimes predicts objects that are inaccurate but semantically related to the context. Figure 8 shows a typical example, where the soccer ball is identified as a frisbee, which is incorrect but still fits the “sports” theme. When the storyline predictor is unable to distinguish such mistaken objects from appropriate objects, grounding errors occur. A better object detector would mitigate this problem, or we could jointly optimize plotting and generation, for instance by including reworking within storyline plotting.

9 Conclusion

We propose a novel story plotting and reworking framework to mimic the human story-writing process. To the best of our knowledge, no study has integrated knowledge graph and story plotting to utilize visual elements in VIST. Also novel is our approximation of human-preferred stories by reusing and aggregating story generation using the results of human-annotated story ranking evaluations, e.g., human evaluation results from MTurk. We also propose a novel questionnaire embedded in the comparative study to collect detailed, meaningful human-annotated data from MTurk. Experiments attest PR-VIST’s strong performance in diversity, coherence, and humanness.

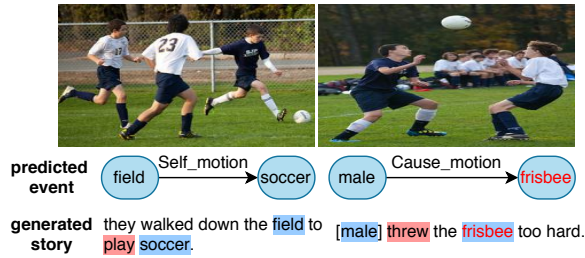


Figure 8: PR-VIST grounding error

10 Ethical Considerations

Although our research aims to produce short stories that are vivid, engaging, and innocent, we are aware of the possibilities of utilizing a similar approach to generate inappropriate text (e.g., violent, racial, or gender-insensitive stories). The proposed visual storytelling technology enables people to generate stories rapidly based on photo sequences at scale, which could also be used with malicious intent, for example, to concoct fake stories using real images. Finally, as the proposed methods use external knowledge graphs, they reflect the issues, risks, and biases of such information sources. Mitigating these potential risks will require continued research.

11 Acknowledgements

This research is supported by Ministry of Science and Technology, Taiwan under the project contract 108-2221-E-001-012-MY3 and 108-2923-E-001-001-MY2 and the Seed Grant from the College of Information Sciences and Technology (IST), Pennsylvania State University. We also thank the crowd workers for participating in this project.

References

- Collin F Baker, Charles J Fillmore, and John B Lowe. 1998. The Berkeley FrameNet Project. In *Proceedings of the 17th International Conference on Computational Linguistics-Volume 1*, pages 86–90. Association for Computational Linguistics.
- Zhiqian Chen, Xuchao Zhang, Arnold P Boedihardjo, Jing Dai, and Chang-Tien Lu. 2017. Multimodal storytelling via generative adversarial imitation learning. *arXiv preprint arXiv:1712.01455*.
- Zi-Yuan Chen, Chih-Hung Chang, Yi-Pei Chen, Jijnasa Nayak, and Lun-Wei Ku. 2019. [UHop: An unrestricted-hop relation extraction framework for knowledge-based question answering](#).
- Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*.
- Bo Dai, Sanja Fidler, Raquel Urtasun, and Dahua Lin. 2017. Towards diverse and natural image descriptions via a conditional GAN. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2970–2979.
- Angela Fan, M. Lewis, and Yann Dauphin. 2019. Strategies for structuring story generation. In *ACL*.
- Linda Flower and John R Hayes. 1981. A cognitive process theory of writing. *College Composition and Communication*, 32(4):365–387.
- Diana Gonzalez-Rico and Gibran Fuentes-Pineda. 2018. Contextualize, show and tell: A neural visual storyteller. *arXiv preprint arXiv:1806.00738*.
- Chao-Chun Hsu, Szu-Min Chen, Ming-Hsun Hsieh, and Lun-Wei Ku. 2018. Using inter-sentence diverse beam search to reduce redundancy in visual storytelling. *arXiv preprint arXiv:1805.11867*.
- Chao-Chun Hsu, Zi-Yuan Chen, Chi-Yang Hsu, Chih-Chia Li, Tzu-Yuan Lin, Ting-Hao (Kenneth) Huang, and Lun-Wei Ku. 2020. Knowledge-enriched visual storytelling. In *Proceedings of Thirty-Fourth AAAI Conference on Artificial Intelligence*.
- Ting-Yao Hsu, Huang Chieh-Yang, Yen-Chia Hsu, and Ting-Hao Kenneth Huang. 2019. Visual story post-editing. In *ACL*.
- Junjie Hu, Yu Cheng, Zhe Gan, Jingjing Liu, Jianfeng Gao, and Graham Neubig. 2019. [What makes a good story? designing composite rewards for visual storytelling](#).
- Qiuyuan Huang, Zhe Gan, Asli Celikyilmaz, Dapeng Wu, Jianfeng Wang, and Xiaodong He. 2019. Hierarchically structured reinforcement learning for topically coherent visual story generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 8465–8472.
- Ting-Hao Kenneth Huang, Francis Ferraro, Nasrin Mostafazadeh, Ishan Misra, Aishwarya Agrawal, Jacob Devlin, Ross Girshick, Xiaodong He, Pushmeet Kohli, Dhruv Batra, et al. 2016. Visual storytelling. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1233–1239.
- J. Johnson, Agrim Gupta, and Li Fei-Fei. 2018. Image generation from scene graphs. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1219–1228.
- Yunjae Jung, Dahun Kim, Sanghyun Woo, Kyungsu Kim, Sungjin Kim, and In So Kweon. 2020. Hide-and-tell: Learning to bridge photo streams for visual storytelling. *ArXiv*, abs/2002.00774.

- Taehyeong Kim, Min-Oh Heo, Seonil Son, Kyoung-Wha Park, and Byoung-Tak Zhang. 2018. GLAC Net: GLocal Attention Cascading Networks for multi-image cued story generation. *arXiv preprint arXiv:1805.10973*.
- Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980.
- Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A. Shamma, and et al. 2017. Visual Genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision*, 123(1):32–73.
- Jiacheng Li, Siliang Tang, Juncheng Li, Jun Xiao, Fei Wu, Shiliang Pu, and Yueting Zhuang. 2020. Topic adaptation and prototype encoding for few-shot visual storytelling. *Proceedings of the 28th ACM International Conference on Multimedia*.
- Nanxing Li, Bei Liu, Zhizhong Han, Yu-Shen Liu, and Jianlong Fu. 2019. Emotion reinforced visual storytelling. In *Proceedings of the 2019 on International Conference on Multimedia Retrieval, ICMR '19*, page 297–305, New York, NY, USA. Association for Computing Machinery.
- Hugo Liu and Push Singh. 2004. ConceptNet—a practical commonsense reasoning tool-kit. *BT Technology Journal*, 22(4):211–226.
- Lixin Liu, Jiajun Tang, Xiaojun Wan, and Zongming Guo. 2019. Generating diverse and descriptive image captions using visual paraphrases. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.
- McCarthy and S. Jarvis. 2010. MTL-D, vocd-D, and HD-D: A validation study of sophisticated approaches to lexical diversity assessment. *Behavior Research Methods*, page 381–392.
- Margaret Mitchell, Ting-Hao Huang, Francis Ferraro, and Ishan Misra. 2018. Proceedings of the first workshop on storytelling. In *Proceedings of the First Workshop on Storytelling*.
- Yatri Modi and Natalie Parde. 2019. The steep road to happily ever after: An analysis of current visual storytelling models. In *Proceedings of the Second Workshop on Shortcomings in Vision and Language*, pages 47–57, Minneapolis, Minnesota. Association for Computational Linguistics.
- N. Mostafazadeh, Nathanael Chambers, Xiaodong He, Devi Parikh, Dhruv Batra, Lucy Vanderwende, P. Kohli, and James F. Allen. 2016. A corpus and cloze evaluation for deeper understanding of commonsense stories. In *NAACL*.
- Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. GloVe: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.
- Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2015. Faster R-CNN: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems*, pages 91–99.
- Thibault Sellam, Dipanjan Das, and Ankur P. Parikh. 2020. BLEURT: Learning robust metrics for text generation.
- Swabha Swayamdipta, Sam Thomson, Chris Dyer, and Noah A. Smith. 2017. Frame-Semantic Parsing with Softmax-Margin Segmental RNNs and a Syntactic Scaffold. *arXiv preprint arXiv:1706.09528*.
- 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 *Advances in Neural Information Processing Systems*, pages 5998–6008.
- Jing Wang, Jianlong Fu, Jinhui Tang, Zechao Li, and Tao Mei. 2018a. Show, reward and tell: Automatic generation of narrative paragraph from photo stream by adversarial training. In *Thirty-Second AAAI Conference on Artificial Intelligence*.
- Ruize Wang, Zhongyu Wei, Piji Li, Haijun Shan, Ji Zhang, Qi Zhang, and Xuanjing Huang. 2019. Keep it consistent: Topic-aware storytelling from an image stream via iterative multi-agent communication.
- Ruize Wang, Zhongyu Wei, Piji Li, Qi Zhang, and Xuanjing Huang. 2020. Storytelling from an image stream using scene graphs. In *AAAI 2020*.
- Xin Wang, Wenhui Chen, Yuan-Fang Wang, and William Yang Wang. 2018b. No metrics are perfect: Adversarial reward learning for visual storytelling. *CoRR*, abs/1804.09160.
- Ronald J. Williams. 1992. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Mach. Learn.*, 8(3–4):229–256.
- Pengcheng Yang, Fuli Luo, Peng Chen, Lei Li, Zhiyi Yin, Xiaodong He, and Xu Sun. 2019. Knowledgeable Storyteller: A commonsense-driven generative model for visual storytelling. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*, pages 5356–5362. International Joint Conferences on Artificial Intelligence Organization.
- Lili Yao, Nanyun Peng, Ralph M. Weischedel, Kevin Knight, Dongyan Zhao, and Rui Yan. 2018. Plan-and-write: Towards better automatic storytelling. *CoRR*, abs/1811.05701.

Yang Yu, Kazi Saidul Hasan, Mo Yu, Wei Zhang, and Zhiguo Wang. 2018. [Knowledge base relation detection via multi-view matching](#). *New Trends in Databases and Information Systems*, page 286–294.