Diverse and Relevant Visual Storytelling with Scene Graph Embeddings

Xudong Hong[12], **Rakshith Shetty**[1], **Asad Sayeed**[3],
Khushboo Mehra[2], **Vera Demberg**[2] **and Bernt Schiele**[1]

[1]Dept. of Computer Vision and Machine Learning, MPI Informatics
[2]Dept. of Language Science and Technology, Saarland University
[3]Dept. of Philosophy, Linguistics, and Theory of Science, University of Gothenburg

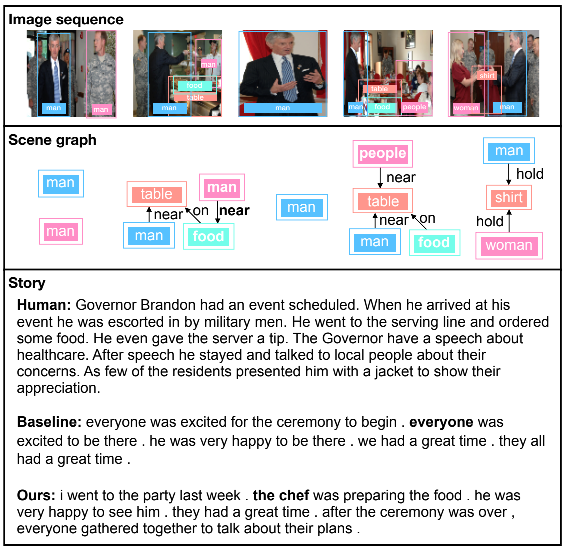
{xhong,kmehra,vera}@coli.uni-saarland.de
{rshetty,schiele}@mpg.mpi-inf.de, asad.sayeed@gu.se

Abstract

A problem in automatically generated stories for image sequences is that they use overly generic vocabulary and phrase structure and fail to match the distributional characteristics of human-generated text. We address this problem by introducing explicit representations for objects and their relations by extracting scene graphs from the images. Utilizing an embedding of this scene graph enables our model to more explicitly reason over objects and their relations during story generation, compared to the global features from an object classifier used in previous work. We apply metrics that account for the diversity of words and phrases of generated stories as well as for reference to narratively-salient image features and show that our approach outperforms previous systems. Our experiments also indicate that our models obtain competitive results on reference-based metrics.

1 Introduction

Visual storytelling is the generation of a coherent narrative from a series of images (Huang et al., 2016). In this paper, we address a particular challenge in visual storytelling: reflecting human preferences in narrative structure, especially the choice of content words and phrases that comprise a readable story. Humans prefer to use diverse words and phrases to construct the storyline to avoid repetitions within or across sentences. For example, in the human-written story in Fig. 1, very few content words are repeated. However, Modi and Parde (2019) have found that recent work often generate repetitive words and phrases which leads to repetitions across sentences and makes stories less diverse. For example, in the first story of Fig. 1, the model generates a verb phrase *had a great time* and then repeats it in the fifth sentence. These words and phrases are usually overly generic. We argue that this is because relations between objects in the last image are not well-represented in the image embedding, forcing the model to produce generic alternatives.



Human: Governor Brandon had an event scheduled. When he arrived at his event he was escorted in by military men. He went to the serving line and ordered some food. He even gave the server a tip. The Governor have a speech about healthcare. After speech he stayed and talked to local people about their concerns. As few of the residents presented him with a jacket to show their appreciation.

Baseline: everyone was excited for the ceremony to begin . **everyone** was excited to be there . he was very happy to be there . we had a great time . they all had a great time .

Ours: i went to the party last week . **the chef** was preparing the food . he was very happy to see him . they had a great time . after the ceremony was over , everyone gathered together to talk about their plans .

Figure 1: Example of extracting scene graphs from images and their relationship to content words and phrases in the stories. The first story (**Baseline**) is generated by AREL (Wang et al., 2018b). The second story (**Ours**) is generated by our proposed model. The **Human** story comes from the VIST dataset (Huang et al., 2016)[1].

We address this problem by employing a more explicit and structured representation of objects and their relations in form of *scene graphs* (Johnson et al., 2015). Scene graphs encode both spatial and predicate relations between objects in the images as well as semantic event relations (actions and their participants). Relations like *(man, near, food)* in the scene graph in Fig. 1 are essential to generate more specific noun phrases (e.g., *the chef*) instead of generic ones (e.g., *everyone*).

In our approach, we extract scene graphs from the images and then learn *scene graph embeddings*

[1] *Typo generated by human: "have" instead of "gave".*

Proceedings of the 24th Conference on Computational Natural Language Learning, pages 420–430
Online, November 19-20, 2020. ©2020 Association for Computational Linguistics
https://doi.org/10.18653/v1/P17

using graph neural networks (Marcheggiani and Perez-Beltrachini, 2018) for each image, which combine the visual features and the discrete semantic information from the scene graphs. A combination of story-wide and individual-image scene graph features is then decoded in the form of a story; parameter-sharing in the decoder encourages narrative coherence.

One difficulty in learning scene graph embeddings together with an end-to-end visual storytelling model is that they introduce a large number of parameters, increasing both computational and learning complexity and making them more difficult to integrate into larger, computationally-expensive learning approaches. We therefore break down the problem into a pipeline with three steps designed to be parameter-efficient and trained independently (Fig. 2): (1) the extraction and augmentation of scene graphs with an existing automatic tool; (2) the training of a graph encoder to obtain scene graph embeddings; and (3) the application of an attention-based visual storytelling model to these embeddings to generate stories. The first two steps establish that we can achieve competitive results without an end-to-end model that requires both story and image to be paired at all steps of training. The third step uses an attention mechanism to supplant a complex graph encoder in the second step, reducing the number of parameters in the story generation model.

Our results show that not only is this approach competitive with other recent work in terms of standard reference-based measures (e.g., BLEU), it has an addtional advantage: the distributional properties of the generated text are closer to human-generated stories than the output of competing systems. The improved quality of the stories and the finer control over the bias of the captioning model afforded by our approach is thus reflected in the outcome of our implementation and experiments.

The main contributions of this paper are:

(a) we introduce a pipeline method for visual storytelling that uses a graph-to-sequence model to learn embeddings for augmented scene graphs and an attention mechanism to combine the scene graph embeddings; (b) we perform the first fine-grained analysis of the diversity of visual stories by inspecting word and phrase distributions and show that machine generated stories from previous models are far less diverse than human-written stories; and (c) we show that the generated stories from our pipeline are not only more diverse than previous work but also more relevant to the images.

2 Related Work

Visual storytelling. Extracting a good representation of the information in the visual input is a key part of the visual storytelling task. Prior work in visual storytelling has typically opted for global features extracted from a pre-trained convolutional neural network (Liu et al., 2017; Yu et al., 2017; Wang et al., 2018a,b; Huang et al., 2019) and has focused on improving the language generation model. Wang et al. (2017) show that introducing regional features and implicit coreference relations of entities leads to more human-realistic word usage in generated stories. Only few prior works employ an intermediate structured representation on story telling task. Yang et al. (2019a) use an external database of knowledge graphs to enchance the visual representation and improve story telling performance. We use scene graphs extracted from images, which does not require an external knowledgebase. Wang et al. (2020) extract scene graphs from images and train an end-to-end model with a graph convolutional encoder directly on visual stories. We propose a pipeline method which first obtains scene graph embeddings from images then applies them to visual storytelling in order to reduce the difficulty of learning both the scene graph embeddings and the story generation model together. Our attention-based story generation model has less parameters while obtaining competitive results.

Scene graph representation. A scene graph is a symbolic representation of structural information where entities are nodes and their relations are edges (Johnson et al., 2015). The large scene-graph annotated Visual Genome (Krishna et al., 2017) dataset has enabled the development of models to extract scene graph representations from images (Zellers et al., 2018; Chen et al., 2019). These scene graph represenations have proven effective on various tasks like image retrieval (Johnson et al., 2015) and image generation (Johnson et al., 2018).

Scene graph based image captioning. A sequential scene graph representation is used to encode images in Gao et al. (2018) to improve image captioning. Yang et al. (2019b) propose auto-encoding text-based scene graphs to learn a shared dictionary between visual and text based graphs, achieving state-of-the-art image captioning performance. Wang et al. (2019b) show that image scene graphs extracted using a trained model can match the captioning performance of an oracle with access to ground-truth graphs. Aligning text- and image-based scene graphs has also been used to generate image captions without paired data (Gu et al., 2019).

3 Model Design

The task of visual storytelling can be decomposed into two distinct parts: (1) extracting relevant information from input images I into compact features and (2) generating stories using these visual features. We improve the visual feature representation by switching from commonly-used global feature vectors to a scene graph-based representation which explicitly encodes objects and their relations. We also reduce the number of parameters by taking a modular approach that separates learning scene graph embeddings from images and generating visual stories. This allows us to independently train the scene graph embedding model and to design a visual storytelling model with fewer parameters yet competitive performance.

Our full pipeline is shown in Fig. 2. We first apply a scene graph generator to extract scene graphs containing vertices for objects and edges for relations between two objects. We then augment the scene graph for each image by adding regional features (see section 3.1). A graph neural network embeds each graph node by aggregating information from across the graph. We propose a pre-training step to independently learn this graph embedding. To do this, we obtain the confidence of the object detector for each object in each image, termed as *visual saliency*, and construct a sequence of object labels ordered by their visual saliency for each image. Then we train a graph-to-sequence model to predict this object sequence given the scene graph embedding of the corresponding image (see section 3.2). To generate stories, we extract both global and regional features from the scene graph embedding for each image and feed them to an attention-based story generation model (see section 3.3).

3.1 Scene Graph Augmentation

Scene graphs can be extracted with the Knowledge-embedded Routing Network (KERN), a state-of-the-art scene graph generator (Chen et al., 2019) built on top of a Faster R-CNN object detector (Ren et al., 2015). KERN generates scene graphs $G = (G_1, G_2, ..., G_N)$ for all images, where each scene graph $G'_j = \{V_j, E_j\}$ contains a set of nodes V_j representing recognised entities with node labels $v_1, v_2, ..., v_M$ and a set of edges E_j with edge labels representing relations between entities.

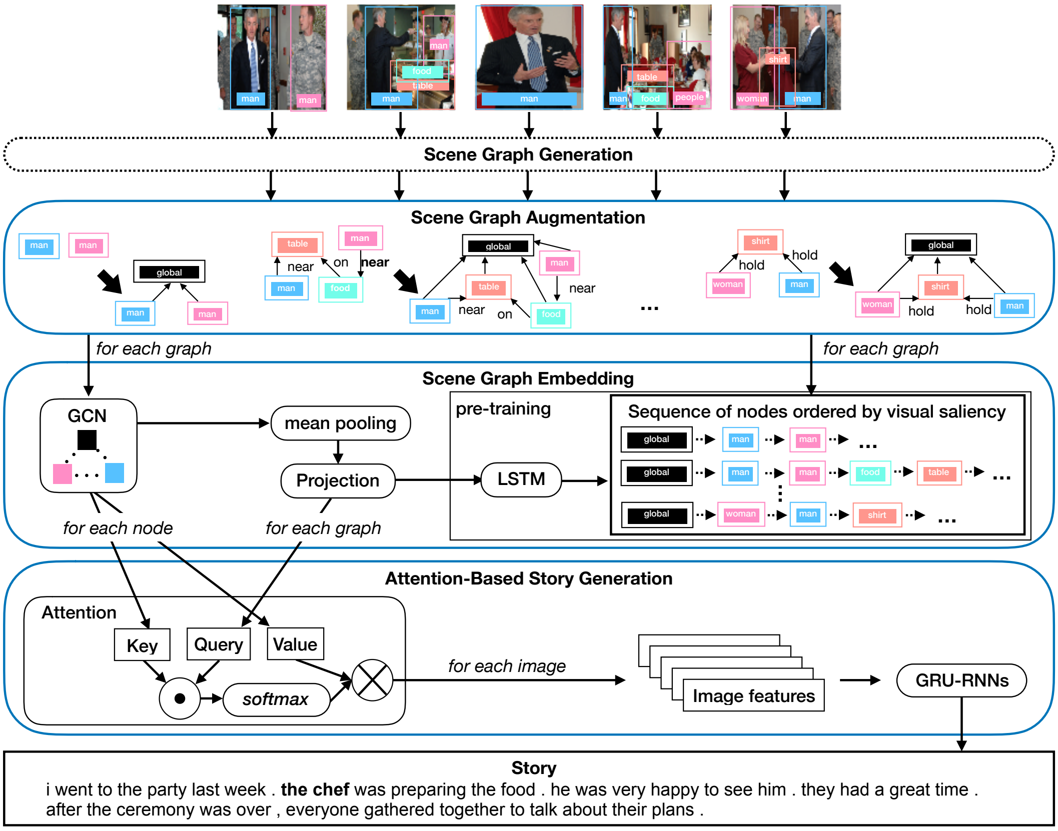
An issue here is that scene graphs are not always connected, but graph neural encoders require connected graphs as input (see the first scene graph in Fig. 2). To obtain a single connected graph for each image, we augment the scene graphs by introducing a *global* node in each graph G'_j, and connect it to all other nodes in the graph.

At this stage, the augmented scene graph contains discrete categorical triplets like *(man, near, table)* (see Fig. 2 for examples). It does not contain detailed visual appearance or shape information: e.g., the color of the man's suit. We address this by augmenting each node in the graph with a corresponding visual feature vector. This is done by extracting Regions of Interest (RoIs) of each object from the backbone Faster R-CNN model of KERN. Then we apply the RoI align algorithm (He et al., 2017) to extract visual features corresponding to each node. The *global* node is assigned the mean features of all the nodes in the graph.

3.2 Scene Graph Embedding

We employ graph convolution networks (GCN; Kipf and Welling, 2017) to encode our augmented scene graph, since they have been effective in learning representations with graph-like structures like parse trees (Du and Black, 2019) and knowledge graphs (Song et al., 2020).

When it comes to the learning of the scene graph encoders, we are inspired by human behaviours in image description task. Objects that appear earlier in image captions usually attract more human attention and are more visually salient to humans (Griffin and Bock, 2000; He et al., 2019). There is a large agreement between human attended regions and activation maps of the last convolutional layer of a VGG-16 network, even though the VGG-16 network is not fine-tuned for captioning (He et al., 2019). If a region of the feature maps is highly activated, it is very likely to be classified as an object with higher confidence. Therefore, we conclude that objects that appear earlier in captions should have a higher confidence when they are passed through a VGG-16 network. We make an assumption that it is the same in visual storytelling and leave the proof for future work due to space limitations.



i went to the party last week . **the chef** was preparing the food . he was very happy to see him . they had a great time . after the ceremony was over , everyone gathered together to talk about their plans .

Figure 2: Our pipeline for visual storytelling.

To simulate this phenomenon, we order the object labels in the scene graph of each image with their confidence and design a graph-to-sequence model to predict this sequence. The model contains two major components, a GCN which encodes the augmented scene graph and a recurrent neural network decoder which generates the sequence of object labels $(v_1, v_2, ..., v_M)$ ordered by their visual saliency, i.e. confidence from the object detector. This allows us to train the GCN in in a self-supervised manner without needing additional labels and keeps objects that tend to be more salient in similar sequence positions across images, giving them an advantage in training.

Graph encoder. We use a multiplicative Relational Graph Convolutional Network (mRGCN; Hong et al., 2019), a variant of GCN assigning parameters not only for nodes but also for edges in a graph, as the graph encoder to introduce explicit representations for edge labels. Given the augmented scene graph G, each node is represented with an regional visual feature vector $\mathbf{x}_v \in \mathbb{R}^d$ extracted from the object detector. For the first layer of the encoder, the hidden representation of the node $\mathbf{h}_v^1 = \mathbf{x}_v$. Then the l-th mRGCN layer computes the hidden representation for node v in $(l+1)$-th layer as follows:

$$\mathbf{h}_v^{l+1} = f(\mathbf{W}\mathbf{h}_v^l + \sum_{u \in N(v)} \mathbf{W}_{dir(e)}\mathbf{h}_u^l \circ \mathbf{r}_e) \quad (1)$$

where $\mathbf{W} \in \mathbb{R}^{d \times h}$ is a trainable parameter. $N(v)$ is the set of all neighbours of node v. f is the ReLU non-linearity. "$\circ$" is the Hadamard product, $\mathbf{W}_{dir(e)} \in \mathbb{R}^{d \times h}$, $dir(e) \in \{in, out\}$ is the direction of the edge $e_{u,v}$ connecting u and v. $\mathbf{r}_e \in \mathbb{R}^h$ is an embedding of the label of the edge $e_{u,v}$. Each layer aggregates the direct neighbours of each node. We stack L GCN layers to encode the full graph.

Object label generator. We use a two-layer LSTM (Hochreiter and Schmidhuber, 1997) to merge the node representations and generate the sequence of object labels. We apply global attention (Luong et al., 2015) to re-weight the hidden representations from the first layer and merge them into a global hidden vector $\mathbf{h}'_G$. The we feed the global hidden vector into two-layer feed-forward networks to get the global encoder output $\mathbf{h}_G$. The probability of node label y_t conditioned on input G and previous node label $y_{1:t-1}$ is obtained by applying a softmax layer on the decoder output as $P(y_t|y_{1:t-1}, G) = softmax(g(\mathbf{h}_G, \mathbf{h}_C))$, where g is a perceptron.

Pre-training. The graph-to-sequence model is trained to maximize the likelihood function $ll = \prod_{|Y|}^{t=1} P(y_t|y_{1:t-1}, G)$. We use extracted visual features as node embeddings and randomly initialise edge embeddings in the encoder. We tune three hyper-parameters on a validation set to minimise the loss, namely the number of hidden units in mRGCN encoder, the number of hidden units in LSTM, and the number of GCN layers. Then we extract augmented scene graph embeddings for the target dataset. After the pre-training of the graph embeddings, each node representation should contain not only node-specific information but also the information from neighbours up to a distance of L.

3.3 Attention-Based Story Generation

The pre-trained graph embeddings serve as input to the story generation model. Instead of using a full graph encoder as Wang et al. (2020), we use the global representations of each image and the local representations of each entity extracted from the pre-trained mRGCN scene graph encoder. This allows us to encode both object-specific information and the relations between each object and the whole image.

We use a dot product attention mechanism to merge all the entities into one hidden vector for each image as follows:

$$\mathbf{a} = \frac{exp(\mathbf{Kq})}{\sum_{j=1}^{M} exp(\mathbf{K}_j\mathbf{q})} \tag{2}$$

$$\mathbf{h} = \mathbf{V}^T\mathbf{a} \tag{3}$$

where we use the global image representations as the query $\mathbf{q} \in \mathbb{R}^d$ and local object representations as the keys $\mathbf{K} \in \mathbb{R}^{M \times d}$ and values $\mathbf{V} \in \mathbb{R}^{M \times d}$.

We follow Wang et al. (2018b) in using a GRU to encode the hidden vectors of all images in a sequential manner and to generate the story. The model is optimised using maximum likelihood estimation with backpropagation.

4 Experiment and Evaluation

Now we show that using pre-trained scene graph embeddings yields competitive results as compared to state-of-the-art approaches on reference-based metrics while using fewer parameters in the image encoder. We also perform an ablation study to show that all proposed components contribute to the performance of the full model and that scene graph embeddings are effective across different attention mechanisms. While the reference-based metrics are useful, they do not always correlate with better story quality as perceived by humans (Wang et al., 2018b). Hence, we also evaluate our model in terms of diversity of word and phrase structure and propose metrics to explicity measure the correctness of object references in section 5. Results show that our scene graph-based model uses more diverse/relevant words and phrases compared to prior work.

4.1 Experiment Design

We train and evaluate our storytelling model on the VIST dataset (Huang et al., 2016), containing 50K visual stories of 10K Flickr albums with 210K images. Each story is based on a 5-image sequence. We follow Wang et al. (2018b) and split the data into 40K training, 10K validation, and 10K test set. We extract scene graphs (including node and edge labels) with the state-of-the-art scene graph generator, KERN, mentioned above.

For neural architecture like GCN in scene graph embeddings, we need to select one important hyperparameter, the number of layers in the GCN encoder. We therefore perform a grid seach from 1 to the maximal diameter in all augmented scene graph. The number of GCN layers is also bounded by the memory size of our GPU cards. So we choose a maximum of 6. We train the scene graph embedding on the VIST dataset and select the optimal setting by validation loss.

We compare our models with previous baselines: **Contextual Attention** (CA; Wang et al., 2017) uses local features from an object detector and a contexual attention layer to intergrate features from different images.

Hierarchically Structured Reinforcement Learning (HSRL; Huang et al., 2019) proposed a hierarchical RNN trained to generate stories by reinforcement learning, with two critics including a multi-modal and a language-style discriminator.

Adversarial Reward Learning (AREL; Wang et al., 2018b) is an Adversarial REward Learning framework to learn an implicit reward function from human demonstrations and then optimize policy search with the learned reward.

Hierarchical Photo-Scene Encoder (HPSR; Wang et al., 2019a) applied hierarchically struc-

Models	# para	B-1	B-2	B-3	B-4	M	R-L	C
CA (Wang et al., 2017)	3.36 M	-	-	-	-	31.73	-	-
HSRL (Huang et al., 2019)	1.05 M	-	-	-	12.3	35.2	30.8	10.7
AREL (Wang et al., 2018b)	1.05 M	63.7	39	23.1	14	35	29.6	9.5
HPSR (Wang et al., 2019a)	1.05 M	61.9	37.8	21.5	12.2	34.4	**31.2**	8
KS (Yang et al., 2019a)	1.05 M	**66.4**	39.2	23.1	12.8	35.2	29.9	**12.1**
SGVST (Wang et al., 2020)	3.41 M	65.1	**40.1**	**23.8**	14.7	**35.8**	29.9	9.8
Ours: SGEmb, attn	2.10 M	62.2	38.7	23.5	**14.8**	35.6	30.2	8.6

Table 1: Results of proposed model on test set compared to previous work using reference-based metrics including BLEU (B), METEOR (M), ROUGE-L (R-L), and CIDEr-D (C). # para is the number of parameters in the image encoder to obtain one vector representation for each image. Parameters in pre-trained components are not counted.

Model variations	B-4	M	R-L
Visual features			
VGG global	13	34.4	29.7
ResNet global	13.6	34.9	29.5
SGEmb global	12	33.8	28.8
VGG, attn	13.5	35.5	30.1
Attention types			
VGG, add attn	12.6	34.2	29.5
VGG, location attn	13.8	35.1	29.8
VGG, simple attn	13.9	35.1	29.7
SGEmb, add attn	13.6	35.5	30.1
SGEmb, location attn	14.1	35.5	30.1
SGEmb, simple attn	14	35.5	**30.2**
Our full model			
SGEmb, attn	**14.8**	**35.6**	**30.2**

Table 2: Ablation study of our full model versus different variants using reference-based metrics including BLEU-4 (B-4), METEOR (M), and ROUGE-L (R-L).

tured reinforcement learning to generate topically coherent multi-sentence stories.

Knowledgeable Storyteller (KS; Yang et al., 2019a) extract objects with an object detector, infer relations between objects with an external knowledge base, and train a knowledge-augmented story generation model.

SGVST (Wang et al., 2020) extract scene graphs from the image sequence and use GCN with temporal convolutionals to merge features across images.

The ablation study we performed over our full model is intended to demonstrate whether the scene graph embedding and the attention mechanism contribute to the final results. We compare the full model with the following simplified models:

VGG global is an seq2seq model using VGG16 (Simonyan and Zisserman, 2015) global features.

ResNet global is a seq2seq model using ResNet-152 (He et al., 2016) global features.

SGEmb global is a seq2seq model which uses only global features from the scene graph embedding.

VGG, attn is an attention-based model which uses regional features directly from the object detector instead of the scene graph embedding.

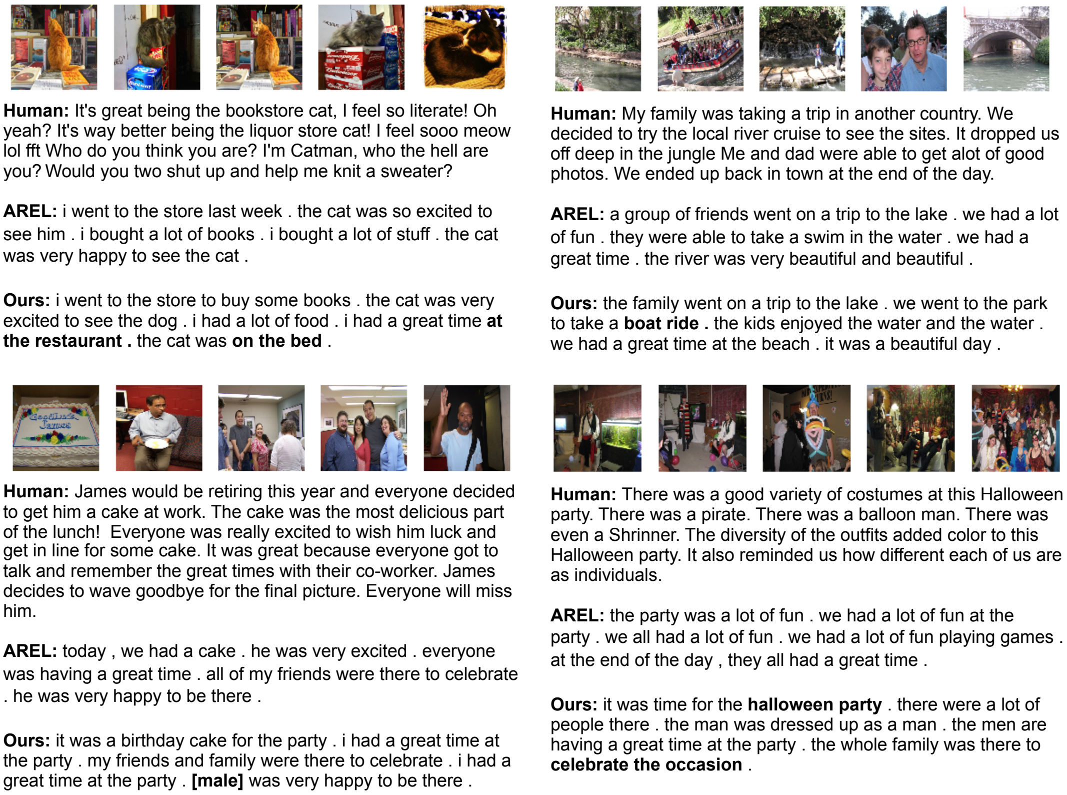
SGEmb, attn is our full model with scene graph embedding and attention mechanism.

4.2 Reference-Based Evaluation

We first evaluate our model and ablations using automatic reference-based metrics on the test set to quantify the similarity between the generated stories w.r.t. human-written ones. We use metrics including unigram (B-1), bigram (B-2), trigram (B-3), and 4-gram (B-4) BLEU scores (Papineni et al., 2002), METEOR (M; Banerjee and Lavie, 2005), ROUGE-L (R; Lin, 2004), and CIDEr (C; Vedantam et al., 2015), based on Wang et al. (2018b)'s evaluation code.

Comparison with baselines. We compare our model with baselines on reference-based metrics in table 1. Our model outperforms all previous methods which do not utilize scene graphs (except SGVST) on BLEU-4 and METEOR. Compared to the recent work using scene graphs, SGVST, we obtain a better score on BLEU-4 and competitive results on BLEU-3, METEOR, and ROUGE-L, although we perform with lower scores on BLEU-1, BLEU-2, and CIDEr. This indicates that the relations between objects in scene graph embeddings empower our model to generate long phrases that are more similar to human text. However, the similarity of shorter grammatical units is sacrificed.

Ablation study. We also report the results of our ablated models to show the importance of the scene graph embedding and the attention mechanism in table 1. Removing the scene graph embedding from our final model and using VGG features in-



Human: It's great being the bookstore cat, I feel so literate! Oh yeah? It's way better being the liquor store cat! I feel sooo meow lol fft Who do you think you are? I'm Catman, who the hell are you? Would you two shut up and help me knit a sweater?

AREL: i went to the store last week . the cat was so excited to see him . i bought a lot of books . i bought a lot of stuff . the cat was very happy to see the cat .

Ours: i went to the store to buy some books . the cat was very excited to see the dog . i had a lot of food . i had a great time **at the restaurant .** the cat was **on the bed** .

Human: My family was taking a trip in another country. We decided to try the local river cruise to see the sites. It dropped us off deep in the jungle Me and dad were able to get alot of good photos. We ended up back in town at the end of the day.

AREL: a group of friends went on a trip to the lake . we had a lot of fun . they were able to take a swim in the water . we had a great time . the river was very beautiful and beautiful .

Ours: the family went on a trip to the lake . we went to the park to take a **boat ride .** the kids enjoyed the water and the water . we had a great time at the beach . it was a beautiful day .

Human: James would be retiring this year and everyone decided to get him a cake at work. The cake was the most delicious part of the lunch! Everyone was really excited to wish him luck and get in line for some cake. It was great because everyone got to talk and remember the great times with their co-worker. James decides to wave goodbye for the final picture. Everyone will miss him.

AREL: today , we had a cake . he was very excited . everyone was having a great time . all of my friends were there to celebrate . he was very happy to be there .

Ours: it was a birthday cake for the party . i had a great time at the party . my friends and family were there to celebrate . i had a great time at the party . **[male]** was very happy to be there .

Human: There was a good variety of costumes at this Halloween party. There was a pirate. There was a balloon man. There was even a Shrinner. The diversity of the outfits added color to this Halloween party. It also reminded us how different each of us are as individuals.

AREL: the party was a lot of fun . we had a lot of fun at the party . we all had a lot of fun . we had a lot of fun playing games . at the end of the day , they all had a great time .

Ours: it was time for the **halloween party** . there were a lot of people there . the man was dressed up as a man . the men are having a great time at the party . the whole family was there to **celebrate the occasion** .

Figure 3: Qualitative results of our model versus AREL and human-written stories.

stead (VGG, attn) decreases BLEU-4 by 1.3 (-8.8%). Using global features from the object detector (VGG global, ResNet global) or global scene graph embeddings (SGEmb global) without the attention mechanism harms performance across all metrics significantly. We further compare models using regional features from scene graph embeddings and from the VGG object detector across different attention mechanisms, like additive attention (add attn; Bahdanau et al., 2014), location-based attention (location attn; Luong et al., 2015), simple attention (simple attn, computing coefficient with keys only) and dot product attention (attn, i.e., the one we use in the full model). Results show that scene graph embeddings boost performance of models across all types of attention mechanisms on all three metrics.

4.3 Qualitative Results

We perform a qualitative comparison to identify what is different in generated stories when we introduce scene graph embeddings and the attention mechanism, as in Figure 1. AREL generates *everyone*, a very generic expression referring to all *man* objects in the image. After introducing scene graph embeddings, our model generates a more specific term *chef* which can be inferred from the sub-graph *(man, near, food)* of the second image. More examples can be found in Figure 3.

5 Evaluating Diversity and Relevance

To get an in-depth understanding of the diversity of different types of words or phrases in generated stories, we perform the first fine-grained analysis of the distributions of words by different Part-of-Speech (POS) tags and phrases by different constituent tags. We first process the generated stories with a state-of-the-art POS tagger and constituency parser (Joshi et al., 2018). Then we plot the frequency vs. rank distributions following Zipf's Law for each POS tag and each constituent tag. We follow Holtzman et al. (2019) to compute the Zipf's coefficient to check how similar the distributions of generated stories are to human-written stories. Using this metric, we compare the diversity of output stories from our model to the baselines and to the best-available prior work, AREL[2].

[2] Despite our best efforts, we could not get access to the code or stories generated by the SGVST model.

Baselines	**Noun**	**Verb**	**Adj.**	**Adv.**	**Pronoun**	**all**
VGG global	1.141	1.592	1.67	2.06	2.186	1.195
ResNet global	1.107	**1.428**	1.461	**1.843**	**1.545**	1.172
AREL**	1.101	1.433	**1.367**	1.928	1.791	1.185
SGEmb global	1.14	1.505	1.645	2.127	1.859	1.185
VGG, attn	1.193	1.541	1.669	2.032	2.055	1.203
Ours: SGEmb, attn	**1.092**	1.439	1.495	1.995	1.617	**1.165**
Human	0.795	1.088	0.965	1.083	1.118	1.011

Table 3: Zipf's coefficient of the word distribution on test set compared to baselines. The score of generated stories should be as close to the human scores as possible, so the **smaller** numbers are better.

Baselines	**NP**	**VP**	**PP**	**Adj. P**	**Adv. P**	**all**
VGG global	1.191	1.208	1.148	**1.023**	3.043	1.067
ResNet global	1.128	1.054	1.087	1.215	2.424	1.013
AREL**	1.117	1.043	1.035	1.11	2.953	1
SGEmb global	1.164	1.137	1.119	1.309	3.456	1.046
VGG, attn	1.23	1.245	1.183	1.227	3.273	1.093
Ours: SGEmb, attn	**1.101**	**1.037**	**1.007**	1.057	**1.959**	**0.987**
Human	0.794	0.563	0.583	0.703	0.983	0.723

Table 4: Zipf's coefficient of the phrase distribution on test set compared to baselines. The score of generated stories should be as close to the human scores as possible, so the **smaller** numbers are better.

5.1 Word Diversity

In table 3, our model obtains the lowest Zipf's coefficient, closest to the human score, which shows that our model generates more diverse words than the baselines. By POS tag, our model generates the most diverse nouns. The ResNet global baseline generate more diverse verbs, adverbs and pronouns by using a stronger image feature extraction backbone. Generating diverse adjectives requires accurate visual features. The performance of our model is bounded by the VGG object detector. Producing pronouns requires cross-image coreference resolutions for objects. Handling this implicitly leads to sub-optimal results of our model diversity in pronouns. However, our proposed architecture is independent of the backbone network and can be upgraded to the stronger ResNet backbone in future work.

5.2 Phrase Diversity

From Table 4, we see that the phrase diversity scores are similar to word diversity, with our model achieving lowest Zipf's coefficient overall and across all tags except on adjective phrases. This indicates that our stories are also more diverse on the phrase level than the baselines. Suprisingly, the VGG global obtains the lowest score on adjective phrases. We thus counted the unique adjective phrases generated by VGG global (31) and by our model (65). We can conclude that the VGG global model generates less unique adjective phrases but with a distribution closer to that of humans.

5.3 Relevance

Models	**match/story**	**# matches**
VGG global	1.62	1579
ResNet global	1.90	1859
AREL**	1.94	1896
SGEmb global	1.58	1542
VGG, attn	1.65	1613
Ours: SGEmb, attn	**1.99**	**1946**
Human	3.01	2939

Table 5: Relevance metric evaluation on the test set.

We show in previous sections that our model generates more diverse nouns and noun phrases. However, do these diverse nouns actually appear in the corresponding images? To explicitly measure this, we utilize the ground truth image captions also available in VIST. Since human written captions refer to salient objects appearing the image, we posit that a relevant story should also refer to these objects as much as possible. Based on this we can quantify the relevance of the generated stories. First, we automatically match the noun phrases in the generated stories with the noun phrases in the corresponding human image captions. The matching is based on the head noun in the noun phrase. We experimented with Lin's similarity on WordNet synsets (Lin, 1998) and cosine similarity using GloVe and BERT embeddings (Pennington et al., 2014; Devlin et al., 2019). The threshold value for counting a match was optimised to minimise false positives on a set of human annotated matches (number=194) from 10 stories in the validation set. We obtained the highest precision using GloVe embeddings, with a threshold of 0.85 (precision=0.82, recall=0.11). This metric is then computed on our model as well as the baselines. The results in table 5 show that the stories generated by our model have higher matches with entities in human-generated captions. Our scene graph embedding model also outperforms the model using the stronger ResNet features, showing that explicitly representing objects and relations in the form of scene graphs helps the model correctly refer to salient objects.

6 Conclusions

We show that introducing scene graph embeddings into visual storytelling with a pipeline method can obtain competitive results while reducing the number of parameters in the storytelling model. We also perform the first fine-grained analysis on the distributions of words and phrases in generated stories which shows that scene graph embeddings increase word and phrase diversities and bring the distributions closer to that of humans. We finally show that the diverse noun phrases we generate are more relevant to the objects in the images.

Future work One benefit of this work is that it provides a baseline for the pre-training of images in visual storytelling, allowing for any images to be used to augment the model without requiring story text; in future work, we will show that this mitigates the limitation of data size. We are currently working on how to merge regional representations for each graph effectively in pre-training and storytelling. GCN is a powerful method for pre-training, but the number of layers is strongly related to the diameter of the graph which is highly variable. A solution is to use Graph Transformer (Cai and Lam, 2020) which learns global attentions across the whole graph.

Moreover, we would like to explore how to extract features from images more accurately for storytelling. The edges of scene graphs in the Visual Genome dataset only contain spatio-temporal relations and limited numbers of general actions like 'holding' as in Fig. 1. We need to extract more common-sense directed events like 'giving' from a sub-graph of the scene graph. This requires implicit graph induction in the current model; we will test an explicit component.

Acknowledgement This research was funded in part by the German Research Foundation (DFG) as part of SFB 1102 "Information Density and Linguistic Encoding" and a Swedish Research Council (VR) grant (2014-39) for the Centre for Linguistic Theory and Studies in Probability (CLASP). Xudong Hong is supported by International Max Planck Research School for Computer Science (IMPRS-CS) of Max-Planck Institute for Informatics (MPI-INF). We sincerely thank the anonymous reviewers for their insightful comments that helped us to improve this paper.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, Michigan. Association for Computational Linguistics.

Deng Cai and Wai Lam. 2020. Graph transformer for graph-to-sequence learning. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 7464–7471. AAAI Press.

Tianshui Chen, Weihao Yu, Riquan Chen, and Liang Lin. 2019. Knowledge-embedded routing network for scene graph generation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6163–6171.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. 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 4171–4186.

Wenchao Du and Alan W Black. 2019. Learning to order graph elements with application to multilingual surface realization. In *Proceedings of the 2nd Workshop on Multilingual Surface Realisation (MSR 2019)*, pages 18–24.

Lizhao Gao, Bo Wang, and Wenmin Wang. 2018. Image captioning with scene-graph based semantic concepts. In *Proceedings of the 2018 10th International Conference on Machine Learning and Computing*, pages 225–229.

Zenzi M Griffin and Kathryn Bock. 2000. What the eyes say about speaking. *Psychological science*, 11(4):274–279.

Jiuxiang Gu, Shafiq Joty, Jianfei Cai, Handong Zhao, Xu Yang, and Gang Wang. 2019. Unpaired image captioning via scene graph alignments. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 10323–10332.

Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. 2017. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

Sen He, Hamed R Tavakoli, Ali Borji, and Nicolas Pugeault. 2019. Human attention in image captioning: Dataset and analysis. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 8529–8538.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2019. The curious case of neural text degeneration. In *International Conference on Learning Representations*.

Xudong Hong, Ernie Chang, and Vera Demberg. 2019. Improving language generation from feature-rich tree-structured data with relational graph convolutional encoders. In *Proceedings of the 2nd Workshop on Multilingual Surface Realisation (MSR 2019)*, pages 75–80.

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 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.

Justin Johnson, Agrim Gupta, and Li Fei-Fei. 2018. Image generation from scene graphs. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1219–1228.

Justin Johnson, Ranjay Krishna, Michael Stark, Li-Jia Li, David Shamma, Michael Bernstein, and Li Fei-Fei. 2015. Image retrieval using scene graphs. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3668–3678.

Vidur Joshi, Matthew Peters, and Mark Hopkins. 2018. Extending a parser to distant domains using a few dozen partially annotated examples. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1190–1199.

Thomas N. Kipf and Max Welling. 2017. Semi-supervised classification with graph convolutional networks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.

Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li Jia Li, David A. Shamma, Michael S. Bernstein, and Li Fei-Fei. 2017. Visual Genome: Connecting Language and Vision Using Crowdsourced Dense Image Annotations. *International Journal of Computer Vision*, 123(1):32–73.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Dekang Lin. 1998. An information-theoretic definition of similarity. In *Proceedings of the Fifteenth International Conference on Machine Learning*, pages 296–304.

Yu Liu, Jianlong Fu, Tao Mei, and Chang Wen Chen. 2017. Let your photos talk: Generating narrative paragraph for photo stream via bidirectional attention recurrent neural networks. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, pages 1445–1452.

Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Effective approaches to attention-based neural machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1412–1421, Lisbon, Portugal. Association for Computational Linguistics.

Diego Marcheggiani and Laura Perez-Beltrachini. 2018. Deep graph convolutional encoders for structured data to text generation. In *Proceedings of the 11th International Conference on Natural Language Generation*, pages 1–9, Tilburg University, The Netherlands. Association for Computational Linguistics.

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.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Jeffrey Pennington, Richard Socher, and Christopher 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.

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.

Karen Simonyan and Andrew Zisserman. 2015. Very deep convolutional networks for large-scale image recognition. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA*.

Linfeng Song, Ante Wang, Jinsong Su, Yue Zhang, Kun Xu, Yubin Ge, and Dong Yu. 2020. Structural information preserving for graph-to-text generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7987–7998.

Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh. 2015. Cider: Consensus-based image description evaluation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4566–4575.

Bairui Wang, Lin Ma, Wei Zhang, Wenhao Jiang, and Feng Zhang. 2019a. Hierarchical Photo-Scene Encoder for Album Storytelling. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33:8909–8916.

Dalin Wang, Daniel Beck, and Trevor Cohn. 2019b. On the role of scene graphs in image captioning. In *Proceedings of the Beyond Vision and LANguage: inTEgrating Real-world kNowledge (LANTERN)*, pages 29–34, Hong Kong, China. Association for Computational Linguistics.

Hanqi Wang, Siliang Tang, Yin Zhang, Tao Mei, Yueting Zhuang, and Fei Wu. 2017. Learning deep contextual attention network for narrative photo stream captioning. In *Proceedings of the on Thematic Workshops of ACM Multimedia 2017*, pages 271–279.

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. *The AAAI Conference on Artificial Intelligence (AAAI), 2018.*, pages 7396–7403.

Ruize Wang, Zhongyu Wei, Piji Li, Qi Zhang, and Xuanjing Huang. 2020. Storytelling from an image stream using scene graphs. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 9185–9192. AAAI Press.

Xin Wang, Wenhu Chen, Yuan-Fang Wang, and William Yang Wang. 2018b. No metrics are perfect: Adversarial reward learning for visual storytelling. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 899–909, Melbourne, Australia. Association for Computational Linguistics.

Pengcheng Yang, Fuli Luo, Peng Chen, Lei Li, Zhiyi Yin, Xiaodong He, and Xu Sun. 2019a. Knowledgeable storyteller: A commonsense-driven generative model for visual storytelling. *IJCAI International Joint Conference on Artificial Intelligence*, 2019-Augus:5356–5362.

Xu Yang, Kaihua Tang, Hanwang Zhang, and Jianfei Cai. 2019b. Auto-encoding scene graphs for image captioning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10685–10694.

Licheng Yu, Mohit Bansal, and Tamara Berg. 2017. Hierarchically-attentive RNN for album summarization and storytelling. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 966–971, Copenhagen, Denmark. Association for Computational Linguistics.

Rowan Zellers, Mark Yatskar, Sam Thomson, and Yejin Choi. 2018. Neural Motifs: Scene Graph Parsing with Global Context. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 5831–5840.