Transtory-XL: Tell Stories from Image Streams

Hengjia Li, Tianji Liu, Junyu Mao, Jiaqi Wang
Department of Computer Science
University College London

March 26, 2020

Abstract

Beyond generating plain description of images, it is of great interests to generate a coherent story over a sequence of images (*visual storytelling*). Compared with image captioning, this task poses a higher-level challenge as the coherence need to be maintained within the sentence (locally) and across the entire story (globally), and also necessarily with a certain level of imagination. Current techniques still heavily rely on recurrent models (e.g., LSTM) to encode a fixed-size topic representation, which maintains the coherence in a fairly limited fashion. This project presents a two-fold deliverable: (1) an empirical study on introducing attention mechanisms to address this challenge; (2) a TransformerXL-based model, **Transtory-XL**, which dispenses recurrence in the encoding and dynamically updates memory which encodes the topic in a simple but flexible way. Experiments demonstrate that our model outperforms the previous technique in offering a longer dependence between sentences, much richer within-sentence information and remarkably more imagination generated while still maintaining competitive results in local coherence (e.g., perplexity, METEOR score). Furthermore, *Transtory-XL* achieves a more than 15% training boost over other experimented techniques.

1 Introduction

Visual storytelling (Ting-Hao et al., 2016) is essentially a *sequential* vision-to-language task, moving one step further from the main concern of static multi-model tasks (e.g., image captioning (Vinyals et al., 2015), visual question answering (Anderson et al., 2017)) - generate descriptive texts or answers for an image (captured at a certain moment), to focusing on reasoning about a potential story behind a stream of images which are structured with temporal information. In other words, visual storytelling extends the image captioning task with an additional temporal dimension which needs to be reasoned upon to reveal more information about the image stream. For a concrete instance, Figure 1 shows both the captioning and storytelling examples of a given image stream of size 3. It can be observed that those two tasks present fairly different literal styles, information levels and coherence.

Figure 1: Illustrated difference between image captioning targets (i.e., descriptions of images-in-isolation) and visual storytelling targets (i.e., descriptions of images-in-sequence), which comes from Visual Storytelling (VIST) Dataset.

For image captions, the text is aimed at capturing and summarizing the visual information in the image, as a result of which the captioning sentences are concrete, image-specific and literal. In contrast, the story presents continuous descriptions over images with a coherence across them. For example, the story setup in Figure 1 gives a specific event/topic "my sister is having a baby", and all following descriptions are centering around it instead of giving isolated and un-correlated image information. Furthermore, another feature of story is that there is strong but implicit dependence between entities (e.g., "my father", "grandfather" and "daddy") in different images, which

storytelling models are expected to be capable of reasoning about. With one level higher, reasonable imagination is equally highlighted in story-related tasks for making the generated more vivid, interesting and human-like, as shown by "is a grandfather for the first time" in the demonstrated example.

Our major technical contributions can be summarized as follows:

- We introduce attention mechanisms to visual storytelling task, constructing models relying on visual attention (*GLAC + non-local*), self-attention (*Transtory-decoder*, *Transtory-full* and *Transtory-XL*), with further evaluations of those models on VIST dataset.
- We empirically demonstrated that a vanilla transformer model *Transtory-full* (detailed in Section 3) failed to well maintain within-sentence (local) coherence and long-term dependence. To solve this issue, we propose a model based upon Transformer-XL, named as *Transtory-XL*, and we design a dynamic memory fusing technique to encode the topic information, to achieve a significantly improved story generation in terms of imagination, within-sentence information and reasoning outcome.

The code to reproduce our work is available at `https://github.com/junyumao1996/NLP_Project`.

2 Related Work

Many recent works have deployed multimodal network that combines deep covolutional neural network (CNN) (Krizhevsky et al., 2012) for interpreting the image content and recurrent neural network (RNN) (Uení et al., 2012) for retrieving sentences. Notable architectures in this category include the visual-attention model (Xu et al., 2015) and the bottom-up and top-down attended VQA model (Anderson et al., 2017), which apply attention mechanisms on image features to deliver appropriate embedding of visual content. (Wang et al., 2018a) showcases the effectiveness of a non-local neural network model in capturing the long-range dependences across different positions in the image. By defining a non-local self-attention module, the response at an image position is learnt as the weighted-average of features at all other positions. Thus, the model learns a coherent embedding with enhanced correlation between image features.

The key problem of the visual storytelling task is to ensure the smooth flow and the consistent context of the generated sentences. Some successful models in the past that addressed this problem include the coherence recurrent convolutional network (CRCN) (Park and Kim, 2015) and the skip gated recurrent unit (sGRU) module with the delay control (Liu et al., 2016). Recently, (Kim et al., 2018) proposes the GLAC network. Such a model incorporates two-level (global and local) attentions on the image features to maintain the coherence across different scenes, and it achieves the state-of-art performance in visual storytelling tasks.

The above described visual storytelling models all apply variants of the bidirectional RNN architecture as sentence generator or visual-feature encoder. However, the coherence of long sentences may be diminished due to the vanishing gradient problem. One intuitive solution is to apply attention-based mechanisms as an alternative of the RNN models. (Vaswani et al., 2017) proposes the purely attention-based Transformer model, which abandons recurrence and convolutions entirely to build long-term dependency. Such a model performs multi-head attention on information embeddings at different positions of input and output, and therefore avoids vanishing gradients. On top of this Transformer model, (Dai et al., 2019) introduces the Transformer-XL architecture that uses temporal memory to resolve the issue of information loss from context fragmentation. In our research, we adopt these Transformer architectures on top of the GLAC net model to prove their effectiveness in the storytelling tasks.

3 Models

Most competitive neural sequence transduction models have an encoder-decoder structure. In this storytelling task, the encoder takes five sequential images as input and encodes them into the global context vectors. Then based on these context vectors, the decoder translates each image into a sequence of the partial story.

3.1 GLAC – Baseline

In our experiment, we adopt the GLAC network (Kim et al., 2018) as a baseline model for the storytelling task. As shown in Figure 2, such a model

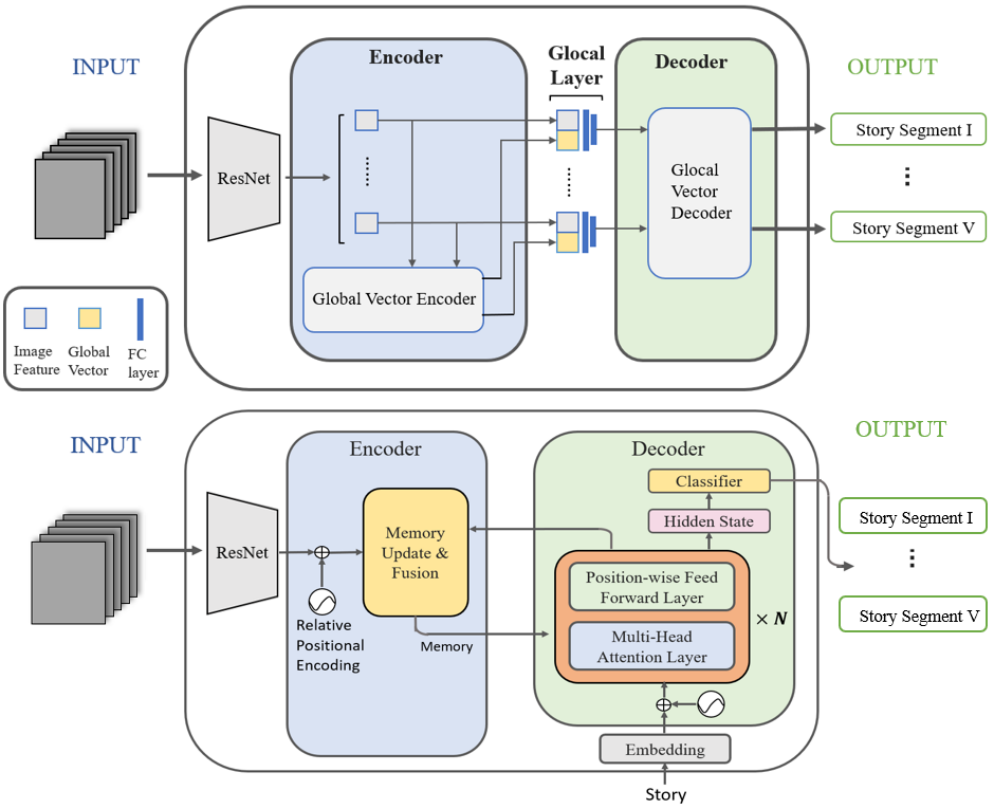

Figure 2: (Up) The structure of the baseline GLAC net model for visual story generation. In our experiment, the modules of the ResNet model, Glocal Vector Encoder and decoder are modified in order to achieve performance gain. (Bottom) Illustrated archetecture of *Transtory-XL* model.

follows the encoder-decoder architecture and it incorporates two-level attention (overall encoding level and image feature level) to construct sentences dependent with the overall image context. The features of each image are at first extracted by a pre-trained ResNet-152 model (He et al., 2016). In the encoder phase, the flattened features of all images are sequentially fed into a bi-LSTM model as the global vector encoder in Figure 2, and the global vectors made up of the bi-LSTM outputs reflects the overall coherent features of the image story. Then, these global vectors are concatenated with the image-specific feature maps and after going through the fully connected layers, these concatenations are encoded with not only the information of global image context, but also local image features. In the decoder phase, the output of fully connected layers, namely the glocal vectors, are fed into LSTMs for translating the feature embeddings into a sentence that descries each image's content. Moreover, a cascading mechanism is applied to convey the context of the previous sentence to next sentence, to ensure the flow of coherent information.

3.2 GLAC with Non-Local Blocks

Our first modification on the baseline GLAC net model is to add visual attention to enhance the important features of each image. We adopted the non-local neural network (Wang et al., 2018a) as an add-on building block after the last convolutional layer of ResNet. It computes the response at an image position as a weighted sum of features on all positions, and the output response can be defined as:

$$\mathbf{y}_i = \frac{1}{C(\mathbf{x})} \sum_{\forall j} f(\mathbf{x}_i, \mathbf{x}_j) g(\mathbf{x}_j). \quad (1)$$

Here j denotes all possible positions in the feature map and i is the position of the output response. Let $\mathbf{x}$ be the input feature map. The unary function g computes a representation of the input feature at position j. Here, the function g is only considered as a form of linear embedding:

$$g(\mathbf{x}_j) = W_g \mathbf{x}_j, \quad (2)$$

The pairwise function f computes a scalar which represents the relationship between the position i

and j. Meanwhile, we defined f explicitly as a dot product similarity:

$$f(\mathbf{x}_i, \mathbf{x}_j) = \theta(\mathbf{x}_i)^\top \phi(\mathbf{x}_j), \tag{3}$$

where $\theta(\mathbf{x}_i) = W_\theta \mathbf{x}_i$ and $\phi(\mathbf{x}_j) = W_\phi \mathbf{x}_j$ are two embeddings to be learnt. We set the normalisation factor $C(\mathbf{x})$ in eq (1) as the total number of positions on $\mathbf{x}$. The final output of the non-local block also considers a residual connection, defined as

$$\mathbf{z}_i = W_z \mathbf{y}_i + \mathbf{x}_i. \tag{4}$$

Therefore, this add-on building block of non-local neural network takes the extracted features as input and then output the attended feature maps with the local features enhanced.

3.3 GLAC with Decoder Attention

The second modification of GLAC is to exploit attention mechanism on the decoder LSTM, inspired by (Xu et al., 2015). The motivation is that we would like the network to attend on different parts of the image features and topic representations based on previous generated tokens.

Formally, taking the output vector $\mathbf{f}_{gl}^{(i)}$ of the GLAC encoder for image i, we first pass it through a fully connected layer $\{W_{slice}, \mathbf{b}_{slice}\}$ to get a list of sliced features

$$\left[\mathbf{f}_{slice,1}^{(i)}; \cdots; \mathbf{f}_{slice,s}^{(i)}\right] = W_{slice}\mathbf{f}_{gl}^{(i)} + \mathbf{b}_{slice}, \tag{5}$$

where s is the number of slices of glocal features and $[;]$ denotes concatenation operation. The dimensions of $\mathbf{f}_{slice,k}^{(i)}$ are set identical to that of the hidden state of decoder LSTM.

For each time step t, taking the previous hidden state of decoder LSTM $\mathbf{h}_{t-1}^{(i)}$ as a query vector, and all $\mathbf{f}_{slice,k}^{(i)}$ as key and value vectors, we apply the following attention mechanism to obtain weighted sum of glocal features $\hat{\mathbf{f}}_{gl,t}^{(i)}$:

$$e_{tk}^{(i)} = {\mathbf{h}_{t-1}^{(i)}}^\top \mathbf{f}_{slice,k}^{(i)} \tag{6}$$

$$\alpha_{tk}^{(i)} = \frac{\exp\left(e_{tk}^{(i)}\right)}{\sum_{m=1}^{s} \exp\left(e_{tm}^{(i)}\right)} \tag{7}$$

$$\hat{\mathbf{f}}_{gl,t}^{(i)} = \sum_{k=1}^{s} \left(\alpha_{tk}^{(i)} \mathbf{f}_{slice,k}^{(i)}\right) \tag{8}$$

We replace $\mathbf{f}_{gl}^{(i)}$ with $\hat{\mathbf{f}}_{gl,t}^{(i)}$ in the GLAC pipeline (concatenate it to the previous output) as the input at step t of decoder LSTM.

3.4 Vanilla transformer model

Transformers entirely rely on self-attention mechanism and can reduce number of operations between two positions to a constant. We follow the general transformer structure as proposed by (Vaswani et al., 2017). In the transformer encoder, extracted image features go through stacks of multi-head self-attention layer and position-wise fully connected layer to output a memory matrix, which is subsequently shared with the decoder. Except for the self-attention previously used in the encoder, decoder also performs multi-head attention over the output of the encoder stack and the previously generated text. To make up for the lack of position information, we add positional encoding to the input image features and word embedding at the bottom stack of the encoder and decoder.

3.5 Transtory-XL Model

Recall that in the vanilla transformer model, we feed all five image features from the encoder output into the decoder memory. Although attending to all the images allow transformer to capture global coherence, when generating predictions for a specific image, the attention weight of the image feature is low because other 4 images occupy some weights, naturally degrading the local coherence, which should also be highlighted. In addition, since we independently process each segment, the model cannot establish dependencies relationship exceeding the segment length. Another existing drawback of vanilla transformer is that, in general, given a long sentence, the vanilla transformer may split it into several segments, leading to the context fragmentation problem. To resolve above issues, we propose the *Transtory-XL* model, which is adapted from Transformer XL model (Dai et al., 2019). The biggest difference compared to the vanilla transformer model is that we introduce hidden states to retain information of segments processed previously. By the mechanism of segment-level recurrence and incorporating hidden states with corresponding image feature information, we can reuse the previous state information when processing a new segment so as to extend the dependencies distance the model can capture. Furthermore, it resorts the *relative* positional encoding to inject the relative distance between any two positions to avoid the indistinguishability. Figure 2 (Bottom) shows the architecture of *Transtory-XL*

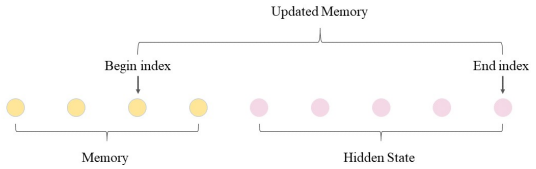

Figure 3: Memory updating process.

model: firstly segment captions are padded to produce the target, which is then fed into the decoder block constructed by stacks of multi-head attention layer and position-wise feed forward layer along with memory. The hidden states go through a classifier to generate a prediction as the output of the decoder. Meanwhile, the decoder updates the stored memory, which is used to fuse with corresponding image feature and then be served as the memory for the next decoder input.

3.5.1 Segment-level recurrence

Segment-level recurrence is the core mechanism of the *Transtory-XL* model. The recurrence means that at time step t, the model reuses previous memory and hidden states by updating memory at time step $t-1$. As shown in Figure 3, for each layer (a layer contains two sub-layers: multi-head attention layer and position-wise feed forward layer) at time step $t-1$, considering the current memory input $\mathbf{m}_{t-1}^{i}$ with length l_m and hidden state output $\mathbf{h}_{t-1}$ with length l_h, there are $l_m + l_h$ steps in total can be cached into the updated memory. We will then compute the begin index and end index to determine which steps would be used for memory updates. Relevant calculations are as follows:

$$idx_{end} = l_m + l_h \tag{9}$$

$$idx_{beg} = \max(0, idx_{end} - l_M) \tag{10}$$

where l_M is the expected length of the updated memory, which is tun-able and should be fixed during training. Then we can update memory by

$$\mathbf{m}_{new} = \left[\mathbf{m}_{t-1}^{i}; \mathbf{h}_{t-1}\right][idx_{beg} : idx_{end}], \tag{11}$$

where $[:]$ denotes between-slicing of the target vector. This mechanism allows the model to break the limitation of context length so that *Transtory-XL* can model longer-term dependencies because the model can learn from previous hidden states containing information from both images features and known token output. In addition, reusing previous state information makes the evaluation more efficient without repeating computation.

3.5.2 Memory fusion

Back to the storytelling task, two critical factors for a good story is relevance and coherence. Relevance is provided by image features while coherence depends on how we take known output and other images' information into consideration. Unlike the case in vanilla transformer, at each time we make only one image feature involved and other coherence-related information is given by the output memory (which is derived hidden states). Memory fusion module aims to fuse these two into a memory input to the decoder. The fusion process can be represented as:

$$\mathbf{m}_{t}^{i} = W_{fuse}\left[\mathbf{m}_{t-1}^{o}; \mathbf{f}_t\right] + \mathbf{b}_{fuse}, \tag{12}$$

where the $\mathbf{m}_{t}^{i}$ denotes memory input at time step $t+1$, $\mathbf{m}_{t-1}^{o}$ denotes memory output at time step t, $\mathbf{f}_t$ is the corresponding image features and $\{W_{fuse}, \mathbf{b}_{fuse}\}$ forms a linear transformation to project concatenation of $\mathbf{m}_{t-1}^{o}$ and $\mathbf{f}_t$ to the dimension of $\mathbf{m}_{t}^{i}$.

3.5.3 Segment padding

We usually set a fixed length (a.k.a., memory length) to specify the length of extended context cached to memory when reusing representation of previous segments. However, if sometimes this fixed memory length is longer than the last sentence for the last image, the model will refer to the more previous segments and relevant image features. In some sense it leads back to context fragmentation again. To resolve this, we pad each segment to a fixed length (a.k.a., padding length). Now each segments share the same length as long as we set the memory length equal to the multiples of padding length, we can fully recall the representation of the last one or more segments without the risk of reusing fragmentary memory information. We further apply a padding mask to remove the side effect of padding on attention computation.

4 Experiments

4.1 Data and Metrics

Visual Storytelling (VIST) dataset (Ting-Hao et al., 2016) is composed of sequential images with corresponding descriptions in three tiers: (1) Descriptions of images-in-isolation (DII); (2) Descriptions of images-in-sequence (DIS); and (3) Stories for images-in-sequence (SIS). It consists of 50,200 sequences (stories) using 209,651 images (train: 40,155, validation: 4,990, test: 5,055).

So far, the best and most reliable evaluation for natural language generation tasks is still human judgment, especially for such complex tasks as storytelling, where basically no metric is perfect (Wang et al., 2018b). However, it is not realistic to ask human experts to rate each story generated by a model for evaluation. It has been proved that METEOR incorporates paraphrasing correlates best with human judgement on this task. We will take METEOR as our metric to evaluate models' performance.

4.2 Empirical Study on Attention Mechanisms

We conducted an empirical study based on the CNN, RNN (encoder) + RNN (decoder) structure of GLAC model as described in Section 3.2. Results are provided in Table 1.

Model	Perplexity ↓	METEOR Score ↑
GLAC (baseline)	16.69	0.3020
GLAC + non local	**16.59**	0.3028
GLAC + decoder attn	17.05	**0.3042**

Table 1: Empirical study on VIST test dataset. Direction of an arrow indicate either a high or low value is preferred for the corresponding metric.

The METEOR score of the decoder attention model is the best among the three. As shown in Figure 6, the coherence between generated sentences is better than the baseline (note the blue tokens), which is reasonable since the exploited attention mechanism enabled the model to use as inputs refined glocal features based on previous outputs. The difference between our method and (Xu et al., 2015) is that they used image features extracted directly from a CNN, while our model utilizes glocal features which is concatenated and linearly transformed image features. Our method could not exploit spacial information to some extent, and that could contribute to the worsening of performance on perplexity after decoder attention was used.

Non-local self-attention blocks offers boosts on both scores. According to the qualitative results in Figure 6, the non-local model successfully identified entities, e.g. "kids", while the baseline model failed to generate concrete tokens like these. This is a strong evidence that our non-local self-attention module can learn and capture spacial relations and importance of different locations within an image, and then force the model to attend on appearance features that are more important. To our surprise the non-local model seems able to generate coherent results as well, even though it has no attention modules in decoder.

Implementation Details. The baseline GLAC results were obtained by re-training the model using the code made available by (Kim et al., 2018). For the non-local self-attention module, we set the number of intermediate channels to be half of the number of input channels, and we did not utilized the subsampling trick proposed by (Wang et al., 2018a). For LSTM (decoder) attention, we set the number of slices of glocal features to be 8. Other hyperparameters (if mutual), training methods and validation strategies were same as the original implementation.

4.3 Transtory-XL and Other Transformer Based Models

As described in Section 3.4, we designed two kinds of transformer based models, namely a GLAC network whose decoder is substituted by Transformer decoder, and a full Transformer model. We will refer to these two models as *Transtory-decoder* and *Transtory-full*, respectively. In Table 2 results and structural details are listed for further reference.

As shown in Figure 6, for the two Transformer based models the generated stories are more informative between sentences, compared with three models in Section 4.2. However, we observed that Transtory-full usually generates stories with inaccurate topics, e.g., wedding, concert for the image sequence in Figure 6, which should be the direct cause of its memory strategy: it forces words to attend on different images which might not favor the model to encode a uniform topic which is crucial for a coherent story, since different image contains different elements.

By combining the advantages of recurrence model (coherent) and Transformer (informative), *Transtory-XL* (see Section 3.5) successfully encodes long-term dependencies and generates results with much higher qualities. For instance, in results shown in Figure 6 there are relations of higher-levels like "girl" and "her best friend", "kids" and "school" that are never observed in all other models, which shows that *Transtory-XL* presents a superb **reasoning capability** over en-

Figure 4: Visualization of attention weights in non-local blocks on selected images. The tail of arrows corresponds to source location i. Arrows point to top-5 locations j according to values of $f(\mathbf{x}_i, \mathbf{x}_j)$ (see eq 3).

tity relation across images.

Beyond the expectation, it can be observed *Transtory-XL* composes some fairly imaginative and creative sentences, which are barely generated in other models. One strong example is given in Figure 6, second caption of *Transtory-XL*'s output - "she was a little nervous", which reasons about the **emotional activity** of the human entity. This feature considerably pushes the generated text to be more story-like and improve the vividness. In other words, this model captures higher-level abstract elements in story composition.

Implementation Details. Learning rates were modified to be 10^{-4}. For Transtory-decoder and Transtory-full we used four heads and one layer for both Transformer decoder and encoder, because with more layers the models could not converge to a comparably low loss. However we did not observe the same issue on *Transtory-XL* so two layers were used for the final model.]Non-local module is discarded for all these three models.

5 Analysis

5.1 Attention Visualization in Non-local Modules

We visualized the self-attention weights (eq (3)) on selected images in Figure 4. In these examples our non-local module successfully discovered objects that are potentially related, e.g., adults and kids, gorilla and person. The features at corresponding locations is enhanced and thereby giving downstream networks useful hints to generate higher-quality results. It worth pointing out that the self-attention weights are not explainable on all input images. Moreover, the localization of objects in Figure 4 is fairly accurate, of which one possible explanation could be that the non-local module directly takes features from the last layer of the ResNet CNN, which are high-level semantic, low-resolution and thus contain vaguer spacial information than shallower features.

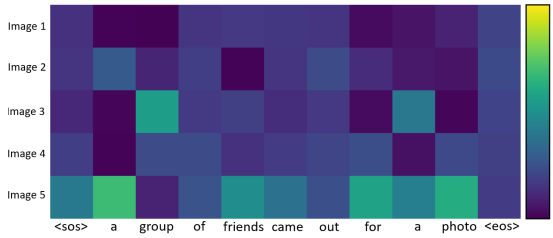

Figure 5: Visualization of attention weights in the decoder layer 1, which are averaged over heads. Only image attentions are shown here by re-scaling weights after dropping attentions on previously generated tokens. Images 1-5 are consistent with those given in Figure 6.

5.2 Attention Visualization in Transtory-full

As detailed in Section 3.4, *Transtory-full* takes 5 extracted image feature vectors into the encoder, generating a static memory which is jointly attended in decoder with the target story. In Figure 5, we attempt to visualize how tokens in a story segment attend to different images to maintain that global coherence. It can be observed that every generated token does attend to other images to a certain degree. For example, the word "group" highly attends to Image 3, where a group of kids gather. And "friends" also looks into other images involved with kids. This attention approach enable *Transtory-full* to generate remarkably coherent story with improved imagination since correlation between images are heavier captured. However, as aforementioned, this architecture also brings out the weak alignment between generated story segment and image, which is further optimized and resolved by the dynamic memory fusion and update (detailed in Section 3.5) in *Transtory-XL* model .

5.3 Generated Qualitative Results

Figure 6 listed generated story examples of the models we implemented. All the results have higher quality than baseline if judged by human. For example, all the generated stories by our models contains more specific tokens, and for the three Transtory models coherence between sentences is greatly improved, as discussed in Section 4.2 and 4.3. The qualitative results of *Transtory-XL* model surpasses any other models without a doubt, and we believe that if properly tuned and trained, the network could be more expressive when applied to more complicated tasks like video-based language generation.

Model	Encoder	Decoder	Positional Embedding	Temporal Memory	Perplexity ↓	METEOR Score ↑
Transtory (decoder)	CNN + BiLSTM	Transformer	absolute	image seq.	21.69	**0.3099**
Transtory (full)	CNN + Transformer	Transformer	absolute	none	27.72	0.2918
Transtory XL	CNN	Transformer-XL	relative	image and language seq.	**18.54**	0.3025

Table 2: Results of *Transtory-XL* and other transformer based models on VIST test dataset. Direction of an arrow indicates either a higher or a lower value is preferred for the corresponding metric.

Ground Truth	there were a lot of children that came by my house last halloween .	there were so many pretty costumes .	some of them were scary too .	i had to hand out a lot of candy .	this year i 'm going to be prepared and buy a lot more candy .
GLAC Baseline	the halloween party was a lot of fun .	there were many different costumes .	everyone was dressed up .	some people were very creative .	i had a great time .
GLAC + decoder attn	the halloween party was a lot of fun .	there were many costumes .	some people dressed up as pirates .	others were very creative .	everyone had a great time .
GLAC + non local	the kids were having a great time at the halloween party .	they were dressed up like a devil .	there were many people .	some of them were very scary .	others were very unique .
Transtory (decoder)	the halloween party was a lot of fun .	we had a great time .	everyone was dressed up as a vampire .	some people were very happy to be there .	and then they all got together for a group photo .
Transtory (full)	the kids were excited to be there .	we had a great time at the party .	they were all very happy .	i love this place .	a group of friends came out for a photo .
Transtory XL	the girl was dressed in a costume .	she was a little nervous .	her best friend came to celebrate .	all of the kids were excited for their big day .	they had a great time at the school .

Figure 6: Qualitative results of our models. Tokens in orange are concrete entities or concepts that are crucial to informative stories, and blue denotes words that are coherent to contents in the previous sentences.

6 Conclusion and Future Work

Attention-based models obtain improved results compared with the previous RNN-based baseline in visual storytelling task. Among them, the final model, *Transtory-XL* strongly outperforms the others by demonstrating remarkable story generation capability in both story (global) coherence, entity relation reasoning and imaginative output. On top of this work, there are a few future research moves can be followed up:

- A **bi-directional** decoder and memory fusion can be developed to propagate the topic information in both forwards and backwards fashion.
- We envision the transfer of our work in more sequential multi-modal tasks, which largely shares the same characteristics with visual storytelling, e.g., video captioning.

References

Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. 2017. Bottom-up and top-down attention for image captioning and visual question answering. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6077–6086.

Zihang Dai, Zhilin Yang, Yiming Yang, Jaime G. Carbonell, Quoc V. Le, and Ruslan Salakhutdinov. 2019. Transformer-xl: Attentive language models beyond a fixed-length context. In *ACL*.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778.

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*, abs/1805.10973.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. 2012. Imagenet classification with deep convolutional neural networks. In *NIPS*.

Yu Liu, Jianlong Fu, Tao Mei, and Chang Wen Chen. 2016. Storytelling of photo stream with bidirectional multi-thread recurrent neural network. *ArXiv*, abs/1606.00625.

Cesc C. Park and Gunhee Kim. 2015. Expressing an image stream with a sequence of natural sentences. In *NIPS*.

Ting-Hao, Huang, Francis Ferraro, Nasrin Mostafazadeh, Ishan Misra, Aishwarya Agrawal, Jacob Devlin, Ross Girshick, Xiaodong He, Pushmeet Kohli, and et al. 2016. Visual storytelling. *arXiv:1604.03968 [cs]*. ArXiv: 1604.03968.

Vysoké Uení, Technické V Brně, Grafiky A Multimédií, and Disertaní Práce. 2012. Statistical language models based on neural networks.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *NIPS*.

Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. 2015. Show and tell: A neural image caption generator. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3156–3164.

Xiaolong Wang, Ross B. Girshick, Abhinav Gupta, and Kaiming He. 2018a. Non-local neural networks. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7794–7803.

Xin Wang, Wenhu Chen, Yuan-Fang Wang, and William Yang Wang. 2018b. No metrics are perfect: Adversarial reward learning for visual storytelling. *arXiv:1804.09160 [cs]*. ArXiv: 1804.09160.

Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron C. Courville, Ruslan Salakhutdinov, Richard S. Zemel, and Yoshua Bengio. 2015. Show, attend and tell: Neural image caption generation with visual attention. In *ICML*.