Image Captioning Through Various Deep Learning Architectures

Anonymous Authors[1]

Abstract

Automatically describing the content of an image is one of the primary goals in artificial intelligence. In this project, we reproduce four generative models under the encoder-decoder framework, three of which use attention mechanism, and propose two new models to accomplish image captioning task. Experiments on Flickr8k and Flickr30k datasets show the accuracy of six models and the fluency of the language they learn solely from image description. During evaluation, BLEU and METEOR scores show that AoA model and Top-Down model always perform the best, while the performance of other models changes with training datasets and model architectures. Code is available at https://github.com/lse-st456/project-2023-group-4.

1. Introduction

Image captioning is to automatically describe the content of an image using properly formed English sentences. This task is one of the primary goals of Computer Vision which aims to automatically generate natural descriptions for images. It requires to recognize salient objects in an image, to understand their interactions, and to verbalize them using natural language, which makes itself very challenging.

Researchers get inspiration from machine translation (He et al., 2021), where the task is to transform a sentence S written in a source language, into its translation T in the target language. Vinyals et al. (2015) follow the encoder-decoder architecture in machine translation and replace the encoder RNN by CNN to process image features. The attention mechanism is then introduced to the decoder (Bahdanau et al., 2014a; Anderson et al., 2018), which generate a weighted average over the extracted feature vectors for each time step. To capture global dependencies, Attention on Attention module (Huang et al., 2019) is proposed.

We reproduce four models from previous works, and then propose two models based on two of them. These six models are all under the encoder-decoder framework, where the encoder is based on CNN network, and decoder is based on RNN network. We train the above six models on Flickr8k and Flickr30k datasets and evaluate them via popular metrics in machine transkation field. Main contributions of this project include:

1) We reproduce four classic models in Tensorflow, namely Primary Model (Vinyals et al., 2015), Bahdanau Model (Bahdanau et al., 2014a), Top-Down Model (Anderson et al., 2018), and AoA Model (Huang et al., 2019).

2) We propose two new models, namely Top-Down-GRU model and Top-Down-Refiner model.

3) We apply Teacher Forcing in the training phase and Geedy Search in the evaluating phase.

4) We utilize BLEU and METEOR metrics to evaluate the captions generated by the above six models.

2. Related work

Ever since the advent and prevalence of deep neural network, deep learning based methods became the mainstream of image captioning, in place of the traditional methods that are mostly template-based (Mitchell et al., 2012; Kulkarni et al., 2013; Yang et al., 2011). Early applications, including Vinyals et al. (2015) and Donahue et al. (2015), share the common solution that employs a combination of CNN and RNN to generate textual descriptions of images, which can be then generalize into the encoder-decoder framework. After this, a large collections of studies attempted to enhance the components of the framework or introduce nested modules.

On the encoder side, one line of augmentation is to exploit the object relationships in the images with graph theory techniques (Yao et al., 2018; Yang et al., 2019), while Yao et al. (2017) explored how attributes and inter-attributes correlations are fed into subsequent LSTM can boost the captioning accuracy. Other attempts including adjustments of the CNN architectures (Chen et al., 2016; Feng et al., 2018).

On the decoder side, the researchers incorporated attention mechanism to selectively attend to different parts of the image and generate more informative and nuanced captions (Bahdanau et al., 2014b; You et al., 2016; Fu et al., 2016).

More elaborate designs that adding visual sentinels or extra gates to judge whether the attended features are suitable for the context even further improved the captioning performance (Lu et al., 2017; Huang et al., 2019). Modifications over the RNN architecture is another resort (Anderson et al., 2018; Zhou et al., 2017).

The encoder-decoder framework can be constructed with Transformer variants(Vaswani et al., 2017) in substitution of the CNN and RNN modules (Sharma et al., 2018; Li et al., 2019; Cornia et al., 2020). Moreover, Rennie et al. (2017) proposed self-critic sequence training (SCST) inspired by REINFORCE algorithm, which directly trains model with evaluation criteria like CIDEr.

More recently, image captioning is considered as a subtask of multi-modal, especially vision-language pre-training (Li et al., 2020; Wang et al., 2022; Hu et al., 2022; Li et al., 2022).

3. Model Architecture

In this section, we focus on four architectures, namely Primary Model (Vinyals et al., 2015), Bahdanau Model (Bahdanau et al., 2014a), Top-Down Model (Anderson et al., 2018), and AoA Model (Huang et al., 2019), and then propose two models by slightly changing Top-Down Model and combining Top-Down Model and AoA Model, respectively. These models are all under the encoder-decoder framework, in which the encoder extracts image features, and the decoder generates caption sequences with the encoder-processed features.

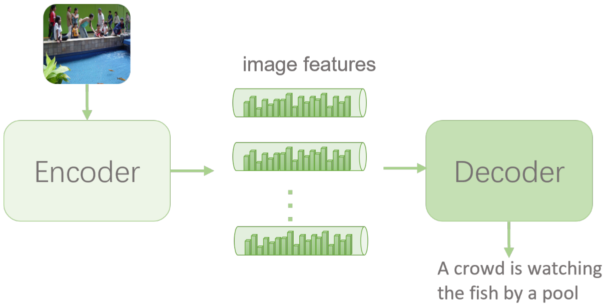


Figure 1. Overview of Encoder-Decoder Framework for Image Captioning. The encoder is usually a CNN or R-CNN based network that extracts image features, while the decoder is usually a well-designed RNN that generates caption sequences.

InceptionV3 (Szegedy et al., 2015) is employed as the encoder network in our experiment. Other options include ResNet and Fast R-CNN (Huang et al., 2019; Anderson et al., 2018). Compared with the encoder, the decoder plays a more salient role in image captioning, which distills the information of encoder-processed features and generates proper sequences.

Models in our experiment generates the next word s_t based on conditional probabilities

$$s_t = \arg\min_{s} \log p(s|I, s_0, \cdots, s_{t-1}) \tag{1}$$

Given the metrics of image features $I = [v_i]_{i \in [N]}$, where $v_i \in \mathbb{R}^D$ and D is the feature shape, we aim at maximizing the probability of the predicted caption with maximum length T

$$\log p(S|I) = \sum_{t=0}^{T} \log p(s_t|I, s_{t-1}, \cdots, s_0) \tag{2}$$

Researchers have designed various decoders with RNN cells to generate high-quality captions, such as incorporating attention mechanisms. In the following sections, we refer to the nonlinear operations in the RNN cells over a single time step by

$$h_t = f(\,\cdot\,, h_{t-1}) \tag{3}$$

Then we use softmax to normalize the linear combinations of hidden state $h_t \in \mathbb{R}^H$ into the probability

$$p(s_t|I, s_{1:t-1}) = \text{softmax}(W_{hs} h_t + b_s) \tag{4}$$

where $s_{1:t-1}$ is short for $\{s_1, \cdots, s_{t-1}\}$, $W_{hs} \in \mathbb{R}^{|\Sigma| \times H}$ and $b_s \in \mathbb{R}^{|\Sigma|}$ denotes the learned parameters that linearly transform the h_t to the space of vocabulary Σ.

3.1. Primary Model

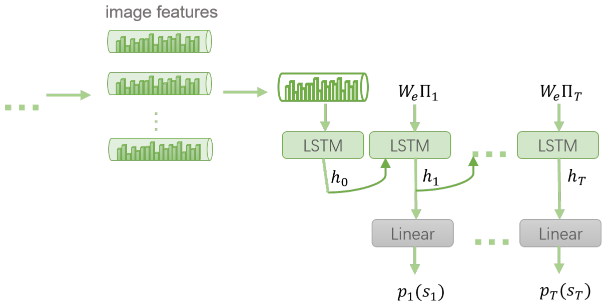


Figure 2. Overveiw of Primary Model. Pooled image feature is treated as the initial input of the LSTM once for all and at each time step afterwards the embedded words are fed into the LSTM.

Primary model (Vinyals et al., 2015) first flows the pooled image features $\tilde{v} = g(I)$ into the LSTM cell, and then input the embedded words in the ground truth $s = \{s_0, \cdots, s_T\}$ to the LSTM cell one by one. This can be formulated as:

$$h_0 = f_{LSTM}(\tilde{v}, h_{-1}) \tag{5}$$

$$h_t = f_{LSTM}(W_e \Pi_t, h_{t-1}),\ t \geq 1 \tag{6}$$

where $W_e \in \mathbb{R}^{E\times|\Sigma|}$ is a word embedding matrix, and Π_t is the one-hot encoding of the input word s_t at time step t.

3.2. Bahdanau Model

The architecture of Primary model is not enough to take full advantage of visual and language information. We thus introduce attention mechanism, which allows the decoder to selectively focus on different parts of the image during the caption generation.

Bahdanau model (Bahdanau et al., 2014a) calculates an attention score for each feature using additive attention mechanism:

$$\alpha_i = \alpha(v_i, h_{t-1}) = \text{softmax}(w_\alpha^\top \tanh(W_{ha}h_{t-1} + W_{va}v_i)) \tag{7}$$

where $W_{ha} \in \mathbb{R}^{M\times H}, W_{va} \in \mathbb{R}^{M\times D}, w_\alpha \in \mathbb{R}^D$ are learned parameters, and then re-weight the image features with the scores

$$\hat{v}_t = \sum_{i=1}^{n_v} \alpha_i v_i \tag{8}$$

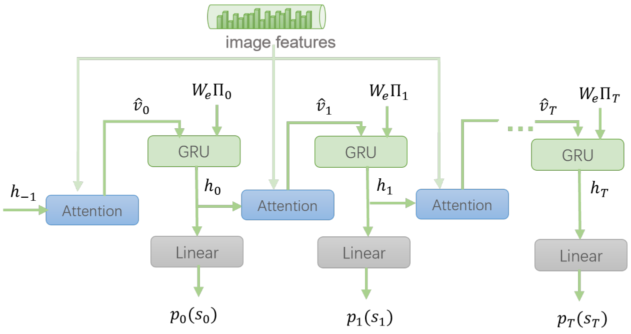


Figure 3. Overview of Decoder with Attention. The image features are attended at every time step and then concatenated with the embedded word to organize a multi-model input for the GRU.

The weighted image features are then concatenated with the embedded caption word to form the new input to the GRU cell.

$$x_t = [\hat{v}_t, W_e\Pi_t] \tag{9}$$

$$h_t = f_{GRU}(x_t, h_{t-1}) \tag{10}$$

3.3. Top-Down Model

Top-Down Model (Anderson et al., 2018) further combines two LSTM units in one cell. While one unit focus on attending the image features, the other one focus on generating the captions. This architecture assigns different tasks to separate units, allowing for more fine-grained control over the attention distribution.

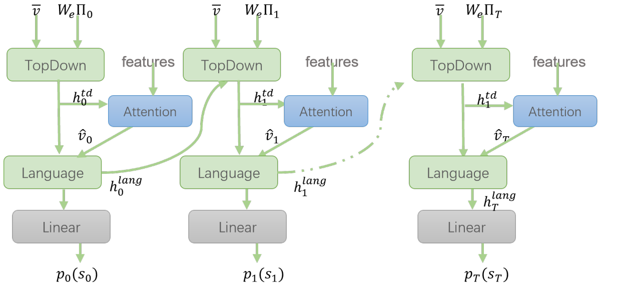


Figure 4. Overview of Top-Down Model. TopDown LSTM functions as a context understander and put forward ”queries”, while Language LSTM focuses on generating proper sequence given the attended features.

More specifically, at each time step, the model first inputs the average image feature $\bar{v}$ that captures the general characteristics of the picture, the previous hidden state of the language LSTM unit h_{t-1}^{lang}, and the embedded word $W_e\Pi_t$, all concatenated, to the top-down attention LSTM unit:

$$\bar{v} = \frac{1}{N}\sum_i^N v_i \tag{11}$$

$$x_t^{td} = [\bar{v}, h_{t-1}^{lang}, W_e\Pi_t] \tag{12}$$

$$h_t^{td} = f_{LSTM}(x_t^{td}, h_{t-1}^{td}) \tag{13}$$

The model then uses the hidden state of top-down LSTM unit h_t^{td} to compute the attention score $\alpha_i = \alpha(v_i, h_t^{td})$ and the corresponding attended image feature $\hat{v}_t$. The input of the language LSTM unit is thus the concatenation of the attended image feature $\hat{v}_t$ and the output of the top-down attention LSTM unit h_t^{td}.

$$x_t^{lang} = [\hat{v}_t, h_t^{td}] \tag{14}$$

$$h_t^{lang} = f_{LSTM}(x_t^{lang}, h_{t-1}^{lang}) \tag{15}$$

And finally we use h_t^{lang} to calculate the $p(s_t|I, s_{1:t-1})$.

3.4. AoA Model

Attention-on-Attention model (Huang et al., 2019), i.e. AoA model, is the most complicated model in our experiment. This is because it not only employs Multi-head Attention and Attention-on-Attention module, but also adds a Refiner into the encoder, which is inspired by Transformer.

We begin by explaining AoA module, particularly how it is derived from ordinary attention mechanisms such as multi-head attention. The multi-head attention is based on scaled dot product attention which begets the re-weighted values $\hat{V}$ by

$$\hat{V} = A_{dot}(Q, K, V) = \text{softmax}(\frac{QK^\top}{\sqrt{d}})V \tag{16}$$

where d is the scaler and Q, K, V are matrices of queries, keys and values. So multi-head attention is to slice Q, K, V into several pieces and apply scaled dot product attetion on each piece respectively, and then concatenated the reweighted pieces of values:

$$\hat{V} = [\hat{V}_1, \cdots, \hat{V}_M],\ \hat{V}_m = A_{dot}(Q_m, K_m, V_m), m \in [M] \tag{17}$$

where M is the number of the "heads" (pieces).

The above attention mechanism can produce the query and the corresponding attended vector, but does not indicate to what extent they are matched or relevant. AoA module is able to address this limitation, serving as a critic to evaluate the quality of attended vectors and identify the useful information out of the attended vector for generating accurate captions.

Given the query and the attended vector pair $(q, \hat{v})$, AoA module generates an information vector i and a tributary attention gate g by

$$i = W_{qi}q + W_{vi}\hat{v} + b_i \tag{18}$$

$$g = \sigma(W_{qg}q + W_{vg}\hat{v} + b_g) \tag{19}$$

where $W_{qi}, W_{vi}, W_{qg}, W_{vg} \in \mathbb{R}^{D\times D}, b_i, b_g \in, \mathbb{R}^D$ are learned parameters, and σ denotes the sigmoid activation function. The final step of AoA module is the element-wise multiplication of i and g that derives the final attended information $\hat{i}$,

$$\hat{i} = g \odot i \tag{20}$$

The whole pipeline of getting attend information $\hat{i}$ then can be formalised as:

$$\hat{i} = \mathrm{AoA}(Q, K, V; A) \tag{21}$$

where A is a first-order attention (e.g. multi-head) operator.

With well-defined AoA module, the refiner can be assembled to further rectify the image features processed by the original encoder. To be more specific, the refiner applies AoA module on three linear projections of the feature matrix I, representing Q, K and V, add up an identity metric I, and finally normalize the sum.

$$I' = \mathrm{Norm}(I + \mathrm{AoA}(W_q^r I, W_k^r I, W_v^r I; A_{mh})) \tag{22}$$

where $W_q^r, W_k^r, W_v^r \in \mathbb{R}^{D\times D}$ are learned parameters, and A_{mh} denotes multi-head attention.

The refiner maintains the shape of the feature matrix I, so it can be stacked for several times and will not affect the design of the subsequent decoder. While Huang et al. (2019) stacked the refiner for 6 times, our experiment only applies the refiner once to reduce computational cost.

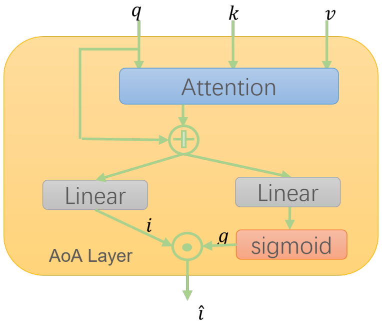


Figure 5. Overview of Attention-on-Attention Module. AoA generates information vector and attention gate based on some first-order attention (in this paper, multi-head attention), and then multiplicate them element-wisely to give out the final attended information (context vextor).

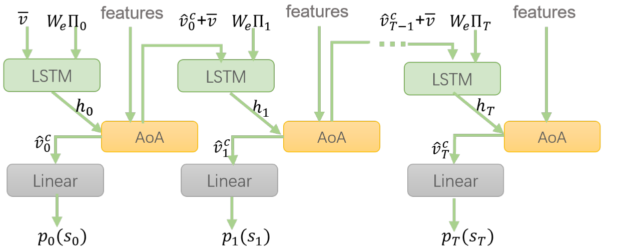


Figure 6. Overview of AoA Model. The AoA attended context vectors replace hidden states, to predict the probability distribution over the vocabulary.

The decoder utilized in AoA model is composed of only one LSTM unit. Instead of directly feeding the hidden state h_t to the fully connected layer, the AoA model uses the context vector $\hat{v}_t^c$ to generate the probabilty distribution, which is calculated by applying AoA module on h_t and the refined image features, where h_t serves as a query:

$$\hat{v}_t^c = \mathrm{AoA}(W_q^d h_t, W_k^d I', W_v^d I', A_{mh}) \tag{23}$$

Here $W_q^d \in \mathbb{R}^{D\times H}, W_k^d, W_v^d \in \mathbb{R}^{D\times D}$ are learned parameters. The decoder integrates the embedded word $W_e\Pi_t$, the average refined feature $\bar{v}$, and the previous context vector $\hat{v}_{t-1}^c$ to input to the LSTM cell. This operation is similar to Top-Down model, except that AoA model add the average feature and the context vector together because essentially they both captures the image characteristics, one globally, the other discriminately.

$$x_t = [\hat{v}_{t-1}^c + \bar{v}, W_e\Pi_t] \tag{24}$$

$$h_t = f_{LSTM}(x_t, h_{t-1}) \tag{25}$$

3.5. Modified Models

Considering both the architectures and performance of above four models in our experiment, we propose two modified model, namely Top-Down-GRU model and Top-Down-Refiner model.

Top-Down-GRU model substitutes the two LSTM cells in Top-Down model with GRU cells. The basic idea of learning long term dependencies in GRU is the same as in LSTM. They are well-matched and share a similar architecture. The key difference is that a GRU has two gates, namely reset and update gates, whereas an LSTM has three gates, namely input, output and forget gates. We propose Top-Down-GRU model to see if Top-Down model with GRU cells outperforms that with LSTM cells.

Top-Down-Refiner model combines the refiner in AoA model and the decoder in Top-Down model. In our experiment, AoA model and Top-Down model are the top two models of performing image captioning task. The refiner in AoA model utilizes AoA module to process image features, and the decoder in Top-Down model organizes two LSTM cells, which contribute to the models' high performance. Combining these two components may lead to considerably good results.

4. Training Methods

In this section, we illustrate the hyperparameter settings and the training objective in our experiment. We will also introduce a popular technique for training sequence-to-sequence models, Teacher Forcing, and our way of generating image captions, Greedy Search.

4.1. Hyperparameter Setting

Pretrained InceptionV3 model without the fully-connected layer at the top is used to preprocess the images in the datasets. The images is RGB and resized into $(299, 299)$ before flowing into InceptionV3. The image features processed by InceptionV3 then expand to 2048, which is named feature shape in the hyperparameters. For captions, the embedding dimension of each word is 256, which is achieved by the embedding layer. The units in LSTM and GRU cells is set to be 512.

Adam is used as the optimizer in the experiment. For most models, the learning rate and epsilon is 0.001 and 0.0001 respectively. However, this learning rate is too large for models with AoA module, which will be discussed later.

The batch size and buffer size when loading the training dataset is 64 and $1,000$ respectively. The models are trained on 40 epochs. Moreover, in Flickr8k dataset, the vocabulary size and maximum words of the captions is 8425 and 35 respectively, whereas those in Flickr30k dataset is 18052 and 74.

Hyperparameter setting is slightly different when training models with Attention on Attention, i.e. AoA module. The number of trainable variables in AoA model is the larget among all models, leading to a high demand of GPU memory. Moreover, the final step of AoA model is an element-multiplication of two high-dimensional vectors, whose dimensions are the feature shape 2048. This also leads to the out-of-memory error of GPUs.

After several experiments with different combination of hyperparameters, the feature shape of models with AoA module is set to be 1024, and Flickr8k's tokenizer is used when training these models on Flickr30k. This is becaure the vocabulary size of Flickr8k's tokenizer is twice smaller than that of Flickr30k, allowing the last dense layer to be much smaller. Additionally, the learning rate is adjusted to 0.0001 for models with AoA module, otherwise they suffer from gradient explosion problem.

Please see https://github.com/lse-st456/project-2023-group-4 for more details of hyperparameter settings. The whole experiment runs on 4 Geforce RTX 2080 Ti GPUs.

4.2. Training Objective

The loss function during training is Sparse Categorical Cross-entropy, which computes the cross-entropy loss between the ground truths and predicted words.

Predicting the next word is essentially a classification task, where the number of classes is the vocabulary size. The cross-entropy function is defined as

$$L = -\sum_{i=1}^{n} t_i \log(p_i) \tag{26}$$

where t_i is the ground truth, p_i is the Softmax probability for the i^{th} class, and n is the number of classes, i.e. the vocabulary size.

4.3. Teacher Forcing

Teacher forcing (Lamb et al., 2016) is a popular technique of training neural machine translation models. During training, the model's input is always the ground truth of previous timestamps, instead of its own previous predictions. Teacher forcing has several advantages (Zhang et al., 2019). It helps the model converge fast, and keeps the training process stable. However, its drawback is making the model too reliant on the ground truth, leading to poor inference performance when faced with unseen data.

4.4. Inference

During evaluation, Greedy Search is used to generate a sentence given an image. Greedy Search just samples the first word according to p_1, then provide the corresponding embedding as input and sample p_2, continuing like this until sampling the special end-of-sentence token or some maximum length. There is also another complex way of generating image caption. Beam Search (Vinyals et al., 2015) iteratively considers the set of the k best sentences up to time t as candidates to generate sentences of size $t + 1$, and keep only the resulting best k of them. Greedy Search is applied in our experiment, since it helps to reduce the computational burden and concentrates in comparisons among different network designs.

5. Experiment

In this section, we will present the numerical results in our experiment which validate the effectiveness of the models described in Section 3 for image captioning task.

5.1. Datasets

We choose Flickr8k (Rashtchian et al., 2010) and Flickr30k (Young et al., 2014) as the datasets in our experiment. Flickr8k contains 8,000, while Flickr30k contains 30,000 images. They both include images with corresponding human-annotated captions that describe the scene in the image. In both datasets, each image comes with up to five captions.

Flickr8k and Flickr30k are both split into training, validation, and testing sets. Flickr8k consists of 6,000 training images, 1,000 validation images, and 1,000 test images, while Flickr30k consists of 28,000 training images, 1,000 validation images, and 1,000 test images. We use the training sets during training our models, and use the testing sets during evaluating.

5.2. Training Results

In Figure 7, we can observe how the training losses for six models decrease as the number of epochs increases in Flickr8k. The loss of AoA model, Top-Down model and Top-Down-GRU model in epoch 40 lies around 0.15, achieving high performance. Although Bahdanau model shows a high rate of convergence at the first 10 epochs, its loss in epoch 40 is 0.23, much higher than 0.15. Primary model's loss decrease to 0.22 in epoch 40, with a relatively low rate of convergence.

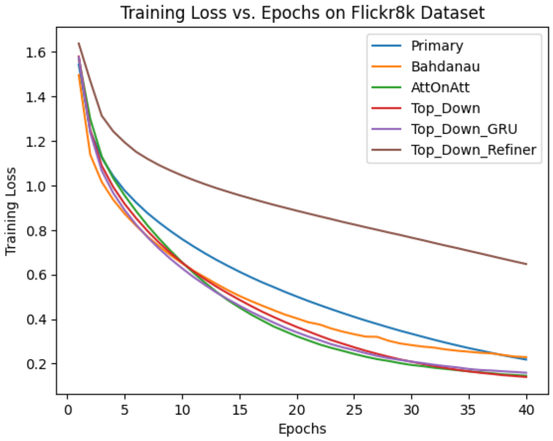


Figure 7. Training Loss vs. Epochs for Six Models in Flickr8k. This figure presents a comparative plot of training loss against the number of epochs for six models in Flickr8k.

Top-Down-Refiner model shows the worst performance in FLickr8k, whose loss in epoch 40 only lies around 0.65. We propose Top-Down-Refiner model under the assumption that the combination of the top two models, AoA model and Top-Down model, could generate a model with even better performance. However, it turns out that using refiner to process image features again is not as helpful for Top-Down model's decoder as we think. The reason may be that, the key component of AoA model's high performance is its decoder, not just the AoA module. We shall try to modify the decoder that involves words, instead of processing image features.

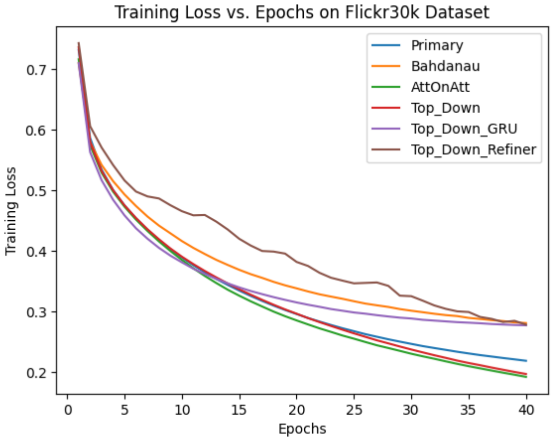


Figure 8. Training Loss vs. Epochs for Six Models in Flickr30k. This figure presents a comparative plot of training loss against the number of epochs for six models in Flickr30k.

In Figure 8, we can observe how the training losses decrease as epochs goes on in Flickr30k. The loss of AoA model, Top-Down model and Primary model in epoch 40 lies around 0.20, achieving high performance, while the rest three models lies around 0.28. There are some differences between the results in Flickr8k and Flickr30k. First, Top-Down-GRU model shows well-matched performance compared with AoA and Top-Down model in Flickr8k, while it is far behind them in Flickr30k. Second, Primary model narrows the gap between its own performance and AoA model's performance in Flickr30k. Last, Top-Down-Refiner model shows the worst performance in Flickr8k, while it achieves comparable performance with Bahdanau model and Top-Down-GRU model.

In a word, AoA model and Top-Down model are always the winners. Primary model and Top-Down-Refiner model show higher performance if the size of training set is over four times larger, while Top-Down-GRU model shows an opposite trend. The reason for Primary model and Top-Down-Refiner model is that, a larger amount of training images give more space to the models to update their parameters, i.e., making up for its architecture weakness by training on large datasets. However, the reason for Top-Down-GRU model is hard to tell, since its only difference with Top-Down model is that its uses GRU cells instead of LSTM cells, while Top-Down model perform much better on FLickr30k. Past works have concluded that there is not a clear winner between GRU and LSTM among various empirical evaluation. We shall leave this question to the future.

The above observations highlight the importance of choosing the most suitable model architecture for a given dataset and task. The evaluation of these six models will be discussed later, which provides a more comprehensive understanding of the models in the next section.

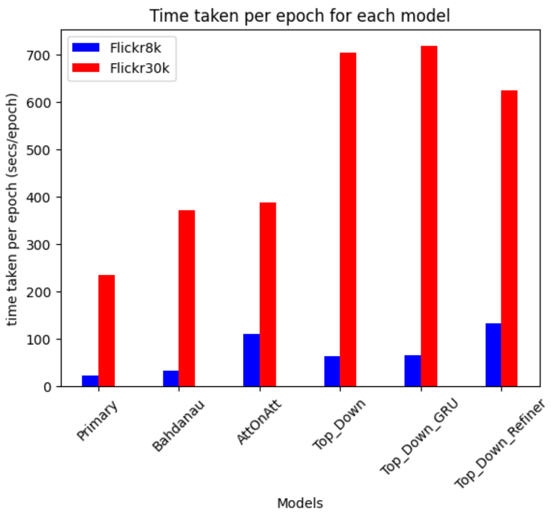


Figure 9. Training Loss vs. Epochs for Six Models in Flickr30k. This figure presents a comparative plot of time taken per epochs for six models.

The bar plot in Figure 9 shows the training time per epoch by each model on Flickr8k and Flickr30k. Training time for each model is much longer in Flickr30k compared with Flickr8k. In Flickr8k, models with AoA module, namely AoA model and Top-Down-Refiner model requires longer training time. However, in Flickr30k, models under Top-Down architecture, namely Top-Down model, Top-Down-GRU model and Top-Down-Refiner model, requires nearly twice longer time than the rest three models. We can conclude that larger dataset requires more computational resources and time, and that the complexity of the models, i.e. the number of parameters in the models, also affects the training time.

5.3. Evaluation

We evaluated the performance of six models on the testing sets of Flickr8k and Flickr30k datasets via BLEU-1, BLEU-2, BLEU-3, BLEU-4, and METEOR metrics.

The Bilingual Evaluation Understudy (BLEU) is a commonly used metric to evaluate the quality of the generated text. It measures the similarity between the generated text and the ground truth text based on n-grams, where n is the length of the n-gram. The BLEU-1 counts the number of times that each unigram in the generated caption appears in the reference captions, and then calculates a weighted geometric mean of the precision scores. The weights encourage generating longer captions that still match the reference captions. Instead of counting the unigram, BLEU-2 counts the number of times that each bigram (pair of adjacent words) in the generated caption appears in the reference captions. SImilarly, for BLEU-3 and BLEU-4, the metrics measure the overlap between the generated and reference captions using trigram and 4-gram precision respectively. The BLEU-Average is the mean of BLEUs' scores, which are based on the n-gram precision of the generated text compared to reference text. BLEU-Average provides a single score that reflects the overall quality of the generated text.

The Metric for Evaluation of Translation with Explicit ORdering (METEOR), on the other hand, evaluates the generated captions by comparing them with the reference captions based on both precision and recall, and takes into account the synonyms and paraphrases. It uses a weighted harmonic mean of precision and recall scores, with the recall score based on the overlap between the generated caption and the reference captions, and the precision score based on the alignment of the words in the generated and reference captions.

In Table 1 and 2, we can be observed that AoA model generates captions with the hightest quality in all metrics and both two datasets, followed by Top-Down model and Top-Down-GRU model. Bahdanau model also displays a good caption-generating ability in both Fliackr8k and Flickr30k, just a little behind the above three models. While the results of Top-Down-Refiner model is second to last in all metrics in Flickr8k, it achieves well-matched performance in Flickr30k compared with Bahdanau model. Primary model generates

Model	BLEU-Ave	BLEU-1	BLEU-2	BLEU-3	BLEU-4	METEOR
Primary	0.038808	0.165608	0.061826	0.024089	0.009287	0.048512
Bahdanau	0.075121	0.301000	0.118599	0.048263	0.019246	0.131531
Att-on-Att	**0.088356**	**0.347902**	**0.139157**	**0.057328**	**0.023023**	**0.147104**
Top-Down	0.080456	0.322080	0.127196	0.051716	0.020487	0.133066
Top-Down-Refiner	0.045868	0.196815	0.073280	0.028370	0.010892	0.053989
Top-Down-GRU	0.082186	0.328754	0.129927	0.052832	0.020979	0.137004

Table 1. Evaluation Metrics Results of Six Models in Testing Set of Flickr8k

Model	BLEU-Ave	BLEU-1	BLEU-2	BLEU-3	BLEU-4	METEOR
Primary	0.041605	0.182272	0.067039	0.025533	0.009653	0.053521
Bahdanau	0.062285	0.261866	0.099664	0.038987	0.014993	0.097822
Att-on-Att	**0.069778**	**0.291746**	**0.111523**	**0.043867**	**0.016950**	**0.108806**
Top-Down	0.069165	0.287907	0.110484	0.043588	0.016907	0.106722
Top-Down-Refiner	0.057397	0.244189	0.091677	0.035723	0.013800	0.077356
Top-Down-GRU	0.066128	0.277876	0.105744	0.041455	0.016010	0.100889

Table 2. Evaluation Metrics Results of Six Models in Testing Set of Flickr30k

captions with the lowest quality in all metrics and both two datasets.

Top-Down-Refiner model achieves both lower training loss and higher evaluation results in Flickr30k, compard with those in Flickr8k. Though the training loss of Top-Down-GRU model in Flickr30k is not as low as Top-Down model, its evaluation results are as good as Top-Down model. Moreover, Primary model gets the worst evaluation results, despite that it achieves lower training loss than Bahdanau model, Top-Down-GRU model and Top-Down-Refiner model.

We choose some examples of captions generated by six models in Appendix A and B. Please check them if interested.

It is worth noting that while BLEU and METEOR scores provide a quantitative measure of performance, they should be interpreted with caution and should not be the only metric used to evaluate the models.

6. Conclusion

We have compared six deep learning architectures on image captioning task, namely Primary model, Bahdanau model, Top-Down model, AoA model, Top-Down-GRU model and Top-Down-Refiner model. We first introduce the architecture and mathematic methods of the six models and then illustrate the hyperparameter settings and training objective. Teacher forcing and Greedy Search are applied during training and evaluating phases respectively. We choose Flickr8k and Flickr30k as the datasets. The traning losses and evaluation metrics shows that AoA model and Top-Down model are always the winners, achieving the lowest traning loss and generating captions with the highest quality.

There are some limitations in our work. First, there are many other methods that we have not explored, such as introducing Vison Transformer (Dosovitskiy et al., 2021) into the encoder-decoder framework. Second, MSCOCO dataset (Lin et al., 2015), containing over 160K images, is not used in our experiment due to the time and disk memory limits. Therefore, we believe future work for image captioning task should focus on utilizing state-of-the-art techniques in both Computer Vision and Nature Language Processing, and try to train a large-scale model with over 1 million parameters on large datasets like MSCOCO.

Statement

Zhihan Zhang, whose Candidate ID is 56111, writes the codes for preprocessing images and captions, generating tokenizers, AoA model, training models and evaluating. She also writes Section 1 Introduction, Section 4 Training Methods, and Section 6 Conclusion.

Erhan Xu, whose Candidate ID is 54751, writes the codes for Top-Down model, Top-Down-GRU model and Top-Down-Refiner model. He also writes Section 2 Related Work and Section 3 Model Architecture.

Tiam Tee, whose Candidate ID is 48119, writes the codes for Primary model and Bahdanau model. He also writes Section 5 Experiment and Appendix.

References

Anderson, P., He, X., Buehler, C., Teney, D., Johnson, M., Gould, S., and Zhang, L. Bottom-up and top-down attention for image captioning and visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 6077–6086, 2018.

Bahdanau, D., Cho, K., and Bengio, Y. Neural machine translation by jointly learning to align and translate. *CoRR*, abs/1409.0473, 2014a.

Bahdanau, D., Cho, K., and Bengio, Y. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014b.

Chen, L., Zhang, H., Xiao, J., Nie, L., Shao, J., Liu, W., and Chua, T.-S. Sca-cnn: Spatial and channel-wise attention in convolutional networks for image captioning. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6298–6306, 2016.

Cornia, M., Stefanini, M., Baraldi, L., and Cucchiara, R. Meshed-memory transformer for image captioning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10578–10587, 2020.

Donahue, J., Anne Hendricks, L., Guadarrama, S., Rohrbach, M., Venugopalan, S., Saenko, K., and Darrell, T. Long-term recurrent convolutional networks for visual recognition and description. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=YicbFdNTTy.

Feng, Y., Lan, L., Zhang, X., Xu, C., Wang, Z., and Luo, Z. Attresnet: Attention-based resnet for image captioning. *Proceedings of the 2018 International Conference on Algorithms, Computing and Artificial Intelligence*, 2018.

Fu, K., Jin, J., Cui, R., Sha, F., and Zhang, C. Aligning where to see and what to tell: Image captioning with region-based attention and scene-specific contexts. *IEEE transactions on pattern analysis and machine intelligence*, 39(12):2321–2334, 2016.

He, Q., Huang, G., Cui, Q., Li, L., and Liu, L. Fast and accurate neural machine translation with translation memory. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 3170–3180, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.246. URL https://aclanthology.org/2021.acl-long.246.

Hu, X., Gan, Z., Wang, J., Yang, Z., Liu, Z., Lu, Y., and Wang, L. Scaling up vision-language pre-training for image captioning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 17980–17989, 2022.

Huang, L., Wang, W., Chen, J., and Wei, X.-Y. Attention on attention for image captioning. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 4634–4643, 2019.

Kulkarni, G., Premraj, V., Ordonez, V., Dhar, S., Li, S., Choi, Y., Berg, A. C., and Berg, T. L. Babytalk: Understanding and generating simple image descriptions. *IEEE transactions on pattern analysis and machine intelligence*, 35 (12):2891–2903, 2013.

Lamb, A., Goyal, A., Zhang, Y., Zhang, S., Courville, A., and Bengio, Y. Professor forcing: A new algorithm for training recurrent networks. 10 2016.

Li, C., Xu, H., Tian, J., Wang, W., Yan, M., Bi, B., Ye, J., Chen, H., Xu, G., Cao, Z., et al. mplug: Effective and efficient vision-language learning by cross-modal skip-connections. *arXiv preprint arXiv:2205.12005*, 2022.

Li, G., Zhu, L., Liu, P., and Yang, Y. Entangled transformer for image captioning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.

Li, X., Yin, X., Li, C., Zhang, P., Hu, X., Zhang, L., Wang, L., Hu, H., Dong, L., Wei, F., et al. Oscar: Object-semantics aligned pre-training for vision-language tasks. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXX 16*, pp. 121–137. Springer, 2020.

Lin, T.-Y., Maire, M., Belongie, S., Bourdev, L., Girshick, R., Hays, J., Perona, P., Ramanan, D., Zitnick, C. L., and Dollár, P. Microsoft coco: Common objects in context, 2015.

Lu, J., Xiong, C., Parikh, D., and Socher, R. Knowing when to look: Adaptive attention via a visual sentinel for image captioning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 375–383, 2017.

Mitchell, M., Dodge, J., Goyal, A., Yamaguchi, K., Stratos, K., Han, X., Mensch, A., Berg, A., Berg, T., and Daumé III, H. Midge: Generating image descriptions from computer vision detections. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pp. 747–756, 2012.

Rashtchian, C., Young, P., Hodosh, M., and Hockenmaier, J. Collecting image annotations using Amazon's Mechanical Turk. In *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*, pp. 139–147, Los Angeles, June 2010. Association for Computational Linguistics. URL https://aclanthology.org/W10-0721.

Rennie, S. J., Marcheret, E., Mroueh, Y., Ross, J., and Goel, V. Self-critical sequence training for image captioning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7008–7024, 2017.

Sharma, P., Ding, N., Goodman, S., and Soricut, R. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 2556–2565, 2018.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A. Going deeper with convolutions. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1–9, 2015. doi: 10.1109/CVPR.2015.7298594.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Vinyals, O., Toshev, A., Bengio, S., and Erhan, D. Show and tell: A neural image caption generator. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3156–3164, 2015.

Wang, P., Yang, A., Men, R., Lin, J., Bai, S., Li, Z., Ma, J., Zhou, C., Zhou, J., and Yang, H. Ofa: Unifying architectures, tasks, and modalities through a simple sequence-to-sequence learning framework. In *International Conference on Machine Learning*, pp. 23318–23340. PMLR, 2022.

Yang, X., Tang, K., Zhang, H., and Cai, J. Auto-encoding scene graphs for image captioning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

Yang, Y., Teo, C., Daumé III, H., and Aloimonos, Y. Corpus-guided sentence generation of natural images. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pp. 444–454, 2011.

Yao, T., Pan, Y., Li, Y., Qiu, Z., and Mei, T. Boosting image captioning with attributes. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

Yao, T., Pan, Y., Li, Y., and Mei, T. Exploring visual relationship for image captioning. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018.

You, Q., Jin, H., Wang, Z., Fang, C., and Luo, J. Image captioning with semantic attention. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

Young, P., Lai, A., Hodosh, M., and Hockenmaier, J. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics*, 2:67–78, 2014. doi: 10.1162/tacl_a_00166. URL https://aclanthology.org/Q14-1006.

Zhang, W., Feng, Y., Meng, F., You, D., and Liu, Q. Bridging the gap between training and inference for neural machine translation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 4334–4343, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1426. URL https://aclanthology.org/P19-1426.

Zhou, L., Xu, C., Koch, P., and Corso, J. J. Watch what you just said: Image captioning with text-conditional attention. In *Proceedings of the on Thematic Workshops of ACM Multimedia 2017*, pp. 305–313, 2017.

A. Examples of models' predictions

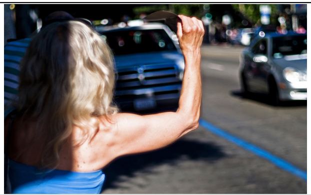
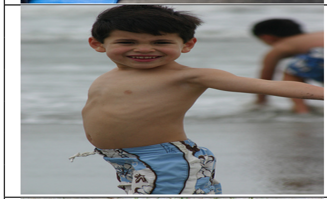
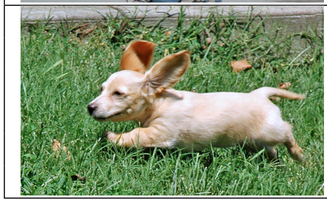
Image	Predicted Caption
	**Att-on-Att**: blond boy is riding next to street in city street
	Bahdanau: little girl in the blue shirt looks on his face up on the city
	Primary: farris tube with white and orange striped helmet
	Top-Down: boy in blue cap is hugging black and riding on the street
	Top-Down-GRU: boy in blue with dark hands
	Top-Down-Refiner: girl in pink and blue along boat
	**Att-on-Att**: young boy wearing blue swim trunks and blue shorts walking on the sand
	Bahdanau: young boy is holding up
	Primary: trows dog
	Top-Down: young boy walking barefoot in shallow water
	Top-Down-GRU: boy with blue blue shirt and red pants is standing on the beach
	Top-Down-Refiner: three boys and dog are interested in front of canoe
	**Att-on-Att**: two dogs fight together and playing together of green green grass outdoors
	Bahdanau: the little dog is running in the woods with toy in field of brown and grass
	Primary: inscribed with black and white dog is red and white fur and black shirt
	Top-Down: small dog with red hair is running past white and green ball in the grass
	Top-Down-GRU: white dog is playing in the grass with tennis ball in its mouth
	Top-Down-Refiner: small child runs through grass

Table 3. Examples of generated captions of six models in Flickr8k

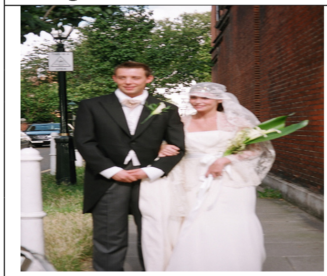
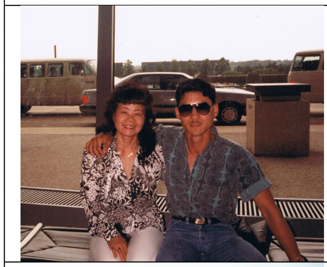
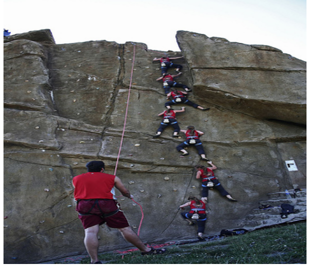
Image	Predicted Caption
	**Att-on-Att**: man wearing tuxedo and woman wearing suit and holding gold flower laugh
	Bahdanau: man and women smile for stand at wedding
	Primary:blue and white enjoy some type of shallow metal space
	Top-Down: african old woman with flowers is leaving the entrance of young man in wedding gown
	Top-Down-GRU: wedding party man smiles as young student stands on an outdoor wedding balloon
	Top-Down-Refiner: woman in wedding dress with fur hat reading the paper
	**AttOnAtt**: people are sitting on bench and one sitting or drinking
	Bahdanau:woman sitting with wood bench
	Primary: people in sunglasses with decorated clothes are outside
	Top-Down: woman in floral dress sits next to young men
	Top-Down-GRU: man is wearing glasses and glasses poses for picture in front of three people with stone shirts next to him
	Top-Down-Refiner: man on cellphone wearing tie jacket scarf and tie looks down while eating ice cream cones down busy street
	**AttOnAtt**: man in red and black checkered shirt is climbing in an outdoor cement area whilst man in red and blue shirt holds rope from the camera
	Bahdanau:two men rock climbs up rock face
	Primary: people are moving around in fenced area and in front of small crowd
	Top-Down: man is hanging out structure that is attached to ropes and half built above the opening of frame
	Top-Down-GRU:people climbing rock dome in warm safety day
	Top-Down-Refiner:runner with the number on his shirt is appearing to be throwing open for the day of himself

Table 4. Examples of generated captions of six models in Flickr30k

B. Under-performance Examples

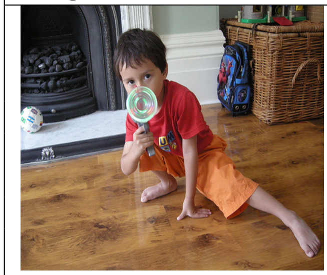
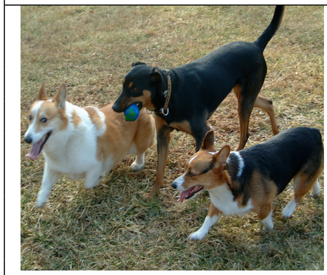
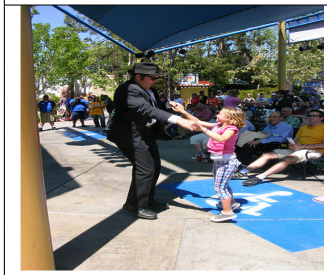
Image	Predicted Caption
	**Att-on-Att**: the little boy is playing with colorful ball
	Bahdanau: young boy is wearing red hat and holding the carrying red guitar while holding piece
	Primary: chainsaw black and white dog in green field
	Top-Down: young boy wearing colapsable brown shirt holds his arms out of kitchen
	Top-Down-GRU: the little boy is climbing with blue and white dog
	Top-Down-Refiner: girl up her foot on the grass
	**Att-on-Att**: brown and brown dog are fighting with war with large brown and brown dog
	Bahdanau: the dogs are playing
	Primary: wheeled shoulder of the brown and white dog is covered in green grass
	Top-Down: two dogs are standing together in the grass beside the road and two other dogs
	Top-Down-GRU: two brown and white dogs are playing on harness through grassy area
	Top-Down-Refiner: young boy in red and white wet with smile
	**Att-on-Att**: man in green jacket is balancing while another man is sitting in chairs
	Bahdanau: two men rock climbs up rock face
	Primary: the man is wearing orange shirt is doing skateboarding
	Top-Down: group of people are jumping over an outdoor park bar
	Top-Down-GRU: large group of people wait in athletic setting
	Top-Down-Refiner: man in white shirt and pink hat holding cardboard box

Table 5. This table consists of examples of generated captions from the six models that are considered as under-performed relative to others in the Flickr8k dataset.

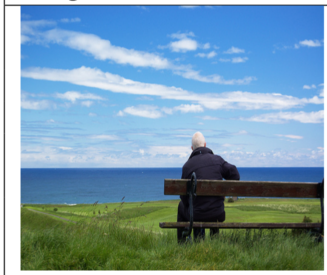
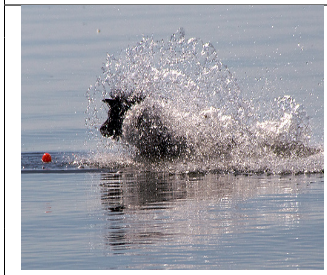
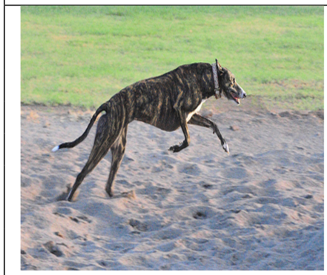
Image	Predicted Caption
	**Att-on-Att**: two adults sitting on bench by lake
	Bahdanau: little girl sits in hand
	Primary: man is seen watching his instrument at festival
	Top-Down: an dark haired man takes out of very brown hat on bench
	Top-Down-GRU: an older woman carrying pitcher hand out of stone beach
	Top-Down-Refiner: woman and child in winter jackets sitting beside the water
	**Att-on-Att**: tan sun in the ocean
	Bahdanau: player in bright blue swim cap blue water fishes
	Primary: big blue ball above the wooden ground while people watch
	Top-Down: on the same long and silver shadow is flying to the finish line
	Top-Down-GRU: green and white dog surveying in doll
	Top-Down-Refiner: rowing full boat with speed in the water
	**Att-on-Att**: the large greyhound dog works into the air behind the muzzle of the sand
	Bahdanau: machine trying to bite test
	Primary:young man is doing stunt on white and pink bicycle
	Top-Down: large brown dog is playing on the green grass
	Top-Down-GRU:the brown and brown cat jumps in the sand
	Top-Down-Refiner:brown dog is playing with ball in still in grassy field

Table 6. This table consists of examples of generated captions from the six models that are considered as under-performed relative to others in the Flickr30k dataset.