Show and Tell Project Report

Yifan Yin
Department of Computer Science & Engineering
University at Buffalo
Buffalo, NY 14228
yifanyin@buffalo.edu

Abstract

Reading an image and automatically describing the content of it is a fundamental and interesting work in both fields of computer vision and machine learning. Inspired by recent work in machine translation and object detection, the attention based model that automatically learns to process language has been more and more applied in image captioning. In this paper, we introduce an attention based model being used in in a deep recurrent architecture for describing the content of an image. Features of different parts of an image are fed into the model instead of directly feeding a single feature vector of an image, aiming to make the attention model focus on particular parts of an image to generate more accurate captions. The reccurent model is trained to maximize the likelihood of the target description sentence given the features of an image. Experiments on the MSCOCO dataset show this model is able to generate accurate descriptions. Our approach gets the BLEU-1, BLEU-2, BLEU-3, BLEU-4 score.

1 Introduction

Automatically generating descriptions for an image using properly formed sentences is a very attracting and challenging problem, and it can be widely used in applications. For examples, this technology will help visually impaired people better understand the content of a digital image. It's a significantly more challenging task because not only generating captions for an image needs to capture objects contained in an image, which is a major focus in the computer vision community, but it is also required to describe how these objects are related and interact with each other. What's more, the above semantic knowledge has to be expressed in a natural language like English and proper grammar and vocabulary.

Despite the challenging nature of this task, there has been a recent surge of research interest on the image captioning problem. Aided by advances in training neural networks [9] and large classification datasets [14], recent work using a combination of convolutional neural networks (convnet) as encoder and recurrent neural networks as decoder has significantly improved the accuracy of generated captions. The research by Vinyals et al. [18] combines the encoder and the decoder into one model and uses pre-trained convolutional neural network to improve image encoding performance.

Most previous attempts have proposed to use neural network to process the whole image directly. In contrast, we would like to accomplish this task in an indirect method. We divide an image into several parts and train the model to find a word which has maximum probability to fit each part. This method is called attention. One of the most curious facets of the human visual system is the presence of attention [13] [2]. Rather than compress an entire image into a static representation, attention allows for salient features to dynamically come to the forefront as needed. Implementing this mechanism of human visual system on image recognition is especially important when there is a lot of clutter in an image. To prevent from losing information which could be useful for richer and more descriptive captions when using representations of images distilled from the top layer of

a convnet, some models use low-level representation of images [19]. Working with these features necessitates a powerful mechanism to steer the model to information important to the tast at hand.

In this paper, we describe an approach to add attention model into the encoder and decoder framework. The model is based on a deterministic attention mechanism trainable by standard back-propagaion methods. We change activation functions in this model to test the impact on the performance. During the evaluation process, we use the visualization method proveded by [19] to gain insigt and interpret the results of this framework by visualizing "where" and "what" the atention focused on. Due to the limit of hardware capability, we reduce the size of MSCOCO dataset [11] to train and validate the usefulness of attention in caption geveration.

2 Related Work

A few methods have been proposed for the task of image captioning in recent years. Many of these methods are built on recurrent neural networks and inspired by the successful use of sequence to sequence training with neural networks for machine translation [1]. One of the reason why machine translation methods can be used in the task of image captioning is that the mechanism of image captioning is similar with that of machine translation that translate an image into features then learn to form a sentence.

The first approach to use neural networks for caption generation was [7], who proposed a multimodal log-bilinear model that was biased by features from the image. This work was later followed by [8] whose method was designed to explicitly allow a natural way of doing both ranking and generation. Mao et al. [12] took a similar approach to generation but replaced a feed-forward neural language model with a recurrent one. Both [18] and [3] use LSTM RNNs for their models. Unlike [8] and [12] whose models see the image at each time step of the output word sequence, [18] only show the image to the RNN at the beginning. Along with images, [3] also apply LSTMs to videos, allowing their model to generate video descriptions.

All of these works try to convert an image into a vector to represent the image by a single feature. In contrast, [6] proposed to learn a joint ecbedding space for ranking and generation whose model learns to score sentence and image similarity as a function of R-CNN object detections with outputs of a bidirectional RNN. [4] proposed a three-step pipeline for generation by incorporating object detections. The difference between our method and these above ones is that our method does not explicitly use object detectors but instead learns latent alignments from scratch.

Attention model has been used in vision related tasks for a long time and some share the same spirit as our work such as [10] [17]. In particular, our work directly combines and extends the work of [18] and [19]

3 Model

In this section, we introduce our attention based encoder-decoder model. We use a low level representation of an image which is extracted from a convolutional neural network. Then the feature is fed into the attention based LSTM neural network.

3.1 Encoder: Convolutional Neural Network

Different from other encoders which take single feature vector as input of the model, our model takes multi-demension feature vectors to represent an image. We use a convolutional neural networt to extract features of an image which we refer to as annotation vectores. A feature set of each image is a $L \times D$ matrix which means that the encoder encodes the image into L vectores, the size of each vector is D:

$$\mathbf{A} = \{\mathbf{a}_1,\ldots,\mathbf{a}_L\}, \mathbf{a}_i \in \mathbb{R}^D$$

Directly using fully connected layer results of convolutional neural network does not satisfy our requirement towards the input of the encoder, so we use a lower convolutional layer to obtain a correspondence between feature vectors and a 2-D image. Based on this, the decoder is able to find relativity between captions and a portion of an image.

3.2 Decoder: Long Short-Term Memoty Network

The input of our model is a raw image and the model generates a caption of size C. The size of the vocabulary dictionary is K:

$$\mathbf{Y} = \{\mathbf{y}_1,\ldots,\mathbf{y}_C\}\ ,\ \mathbf{y}_i \in \mathbb{R}^K$$

Similar with machine translation and natural language processing, image captioning is analogous with translating an image into a sentence. Each word may be related to a certain part of an image and previous generated words. So we use a long short-term memory (LSTM) network [5] to generate captions. The mechanism is producing a caption by generating one word at every time step conditioned on a context vector, the prevous hidden state and the previously geverated words. Our implementation of LSTM follows the one used in [19] (see Figure 1).

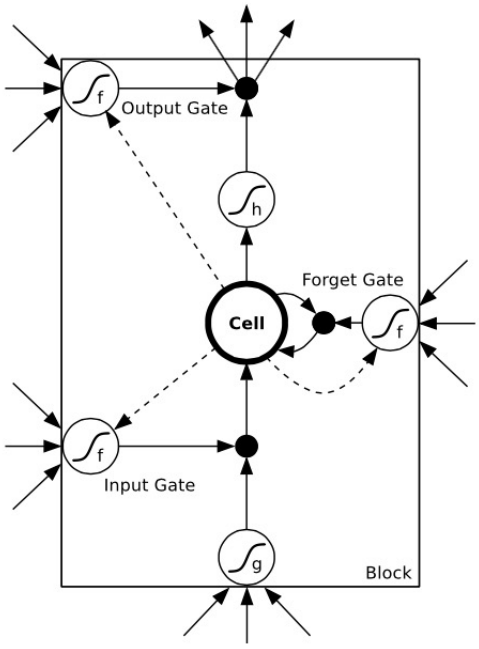

Figure 1: LSTM Cell.

We need to form a LSTM network, where the variable number of words we condition upon up to t-1 is expressed by a fixed length hidden state or memory h_t . This memory is updated after seeing a new input x_t by using a non-linear function f :

$$h_{t+1} = f(h_t, x_t)$$

The implementation of LSTM is:

$$\begin{Bmatrix} i_t \\ f_t \\ o_t \\ g_t \end{Bmatrix} = \begin{Bmatrix} \sigma \\ \sigma \\ \sigma \\ tanh \end{Bmatrix} T_{D+m+n,n} \begin{Bmatrix} \mathbf{E}y_{t-1} \\ h_{t-1} \\ \hat{z}_t \end{Bmatrix}$$

Here, i_t, f_t, c_t, o_t, h_t are the input, forget, memory, output and hidden state of the LSTM, respectively. The vector $\hat{z} \in \mathbb{R}^D$ is the context vector, capturing the visual information associated with a particular input location, as explained below. $\mathbf{E} \in \mathbb{R}^{m \times K}$ is an embedding matrix. Let m and n denote the embedding and LSTM dimensionality respectively and σ and $\odot$ be the logistic sigmoid activation and element-wise multiplication respectively.

The initial memory state and hidden state of the LSTM network are computed by an average of the annotation vectors fed through two separate MLPs:

$$c_0 = f_{init,c}(\frac{1}{L}\Sigma a_i)$$

$$h_0 = f_{init,h}(\frac{1}{L}\Sigma a_i)$$

The probability given the LSTM state, the context vector and previous word:

$$\mathbf{p}(y_t|a, y_1^{t-1}) \propto \mathbf{exp}(L_o(Ey_{t-1} + L_h h_t + L_z \hat{z}_t))$$

3.3 Attention Model

The macroscopical image of the system is showed as Figure 2. If we have predicted i words, the hidden state of the LSTM is h_i. We select the relevant part of the image by using h_i as the context. Then, the output of the attention model z_i, which is the representation of the image filtered such that only the relevant parts of the image remains, is used as an input for the LSTM. Then, the LSTM predicts a new word, and returns a new hidden state h_{i+1}.

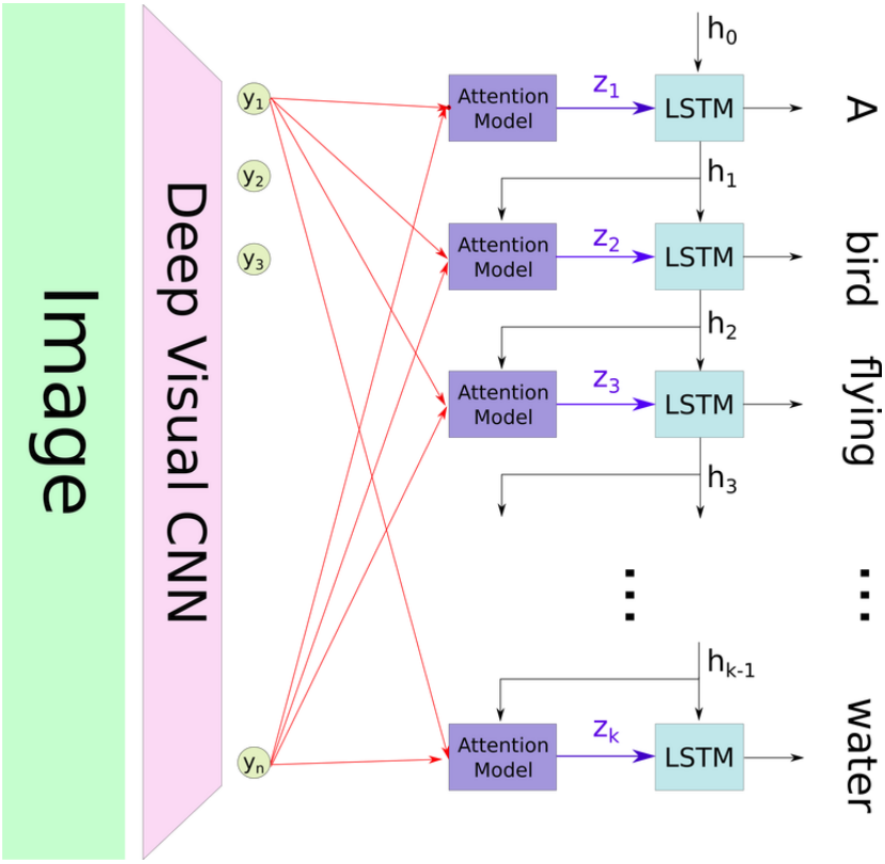

Figure 2: Attention-based Network.

The attention block is showed in Figure 3. $\mathbf{c}$ is the context, and the y_i are the part of the data we are looking at. the network computes $m_1, \dots m_n$ with a $\tanh$ layer. It means that we compute an aggregation of the values of y_i and $\mathbf{c}$. An important remark here is that each m_i is computed without looking at the other y_j for $j \neq i$. They are computed independently. The model presented above of an attentive model can be modified. For example, the $\tanh$ layer can be replaced by any

other network. The only important thing is that this function mixes up **c** and y_i. Here we use $\tanh$ function.

Back to the LSTM model. In simple terms, the context vector $\hat{z}_t$ is a dynamic representation of the relevant part of the image input at time t. [19] define a mechanism ϕ that computes $\hat{z}_t$ from the annotation vectors a_i, $i = 1, \ldots, L$ corresponding to the features extracted at different image locations. The model creates a positive weight α_i, which can be used as the probability that location i is the correct region to prduce the next word. We compute the weight α_i for each annotation a_i using attention layer $fatt$ conditioned on the previous hidden state h_{t-1}.

$$e_{ti} = f_{att}(a_i, h_{t-1})$$

$$\alpha_{ti} = softmax(e_{ti})$$

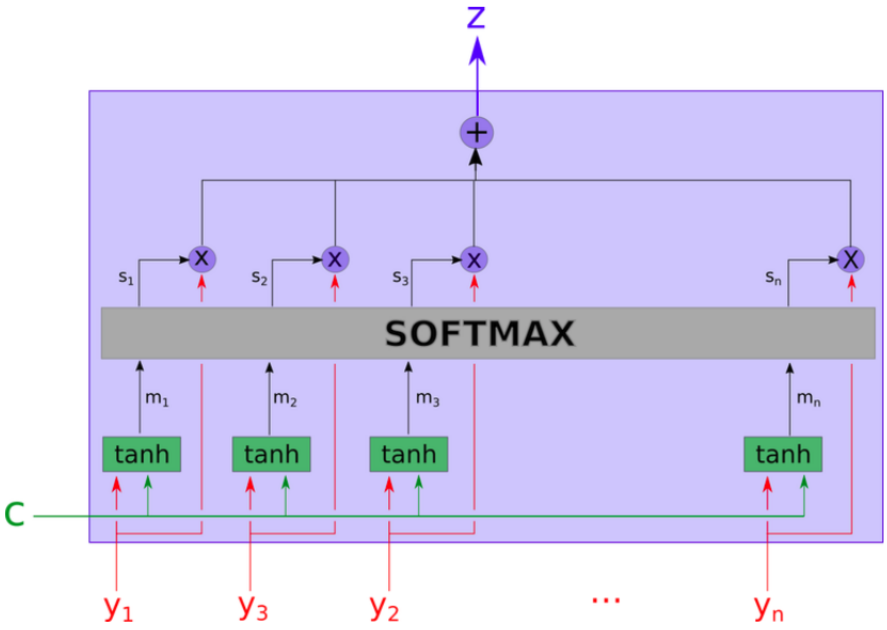

Figure 3: Attention Model.

Once the weights are computed, the context vector $\hat{z}_t$ is computed by

$$\hat{z}_t = \phi(\{a_i\}, \{\alpha_i\})$$

3.4 Regularization

By construction, $\Sigma\alpha_{ti} = 1$ as they are the output of a softmax. [19] introduces a form of doublystochastic regularization. This can be interpreted as encouraging the model to pay equal attention to every part of the image over the course of generation. In addition, the model also predicts a gating scalar β from previous hidden state h_{t-1} at each time step t, such that,

$$\phi(\{a_i\}, \{a_i\}) = \beta\Sigma_i^L \alpha_i a_i$$

, where

$$\beta_t = \sigma(f_\beta(h_{t-1}))$$

Finally, the model is trained end-to-end by minimizing the following penalized negative log-likelihood:

$$L_d = -log(P(\mathbf{y}|\mathbf{x})) + \lambda\Sigma_i^L(v - \Sigma_t^C \alpha_{ti})^2$$

4 Training Setting

The attention based model was trained with stochastic gradient descent optimizer using exponential adaptive learning rate algorithm. We adopted the encoder of [19], using the Oxford VGGnet [15] pre-trained on ImageNet without finetuning to generate annotations. We used the conv5_3 layer to extract low level features which was converted into size of flattened 196 $\times$ 512 to feed into the decoder. We didn't use the inception network [16] which is used in [18] as encoder model because the inception is too deep to costomize.

To prevent computational waste from training on a random group of captions, we preprocess the caption file to build a dictionary mapping the length of a sentence to the corresponding subset of captions. Then, we randomly sample a length and retrieve a mini-batch of size 128 during training.

In terms of regularization, we use dropout with 0.5 keep probability. The parameter in regularization of the final log-likelihood is set to 16 / 196.

The code for the model is based on Tensorflow, Python 2.7.

5 Experiments

In this section, we describe our experimental methodology and quantitative results which validate the effectiveness of our model for caption generation.

5.1 Data

We train and validate our model on Micosoft COCO dataset. Due to hardware capacity, we modify the original dataset which has 82783 images into a small one which only has 5033 images. To feed these images into VGGnet, these images are resized into size of 224 $\times$ 224.

Results for our attention-based architecture are reported in Table . We report results with the frequently used BLEU metric 2 which is the standard in the caption generation literature. We report BLEU from 1 to 4 without a brevity penalty.

5.2 Evaluation

Part of the computational graph is showed as Figure 4, which displays one unit of the decoder structure.

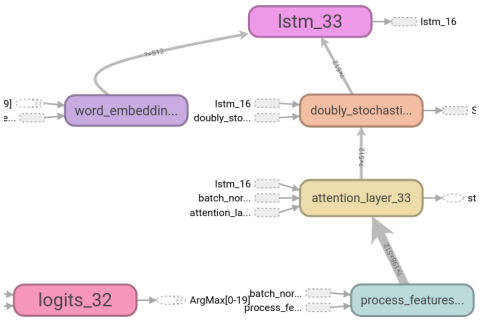

Figure 4: Part of the Computational Graph.

The parameter diagrams are showed as Figure 5, 6, which are the loss decreasing procedure and corresponding learning rate decay procedure. According to the loss diagram, we can see before around 1500 iterations, the loss decreases rapidly. However, after that, the loss seems to be converged at a value aroud 48. Due to the limit of hardware capacity and limit of time, we cannot proceed this training process more.

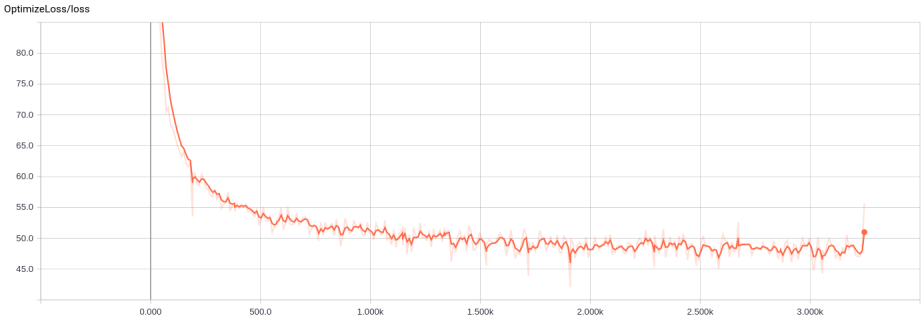

Figure 5: Part of the Computational Graph.

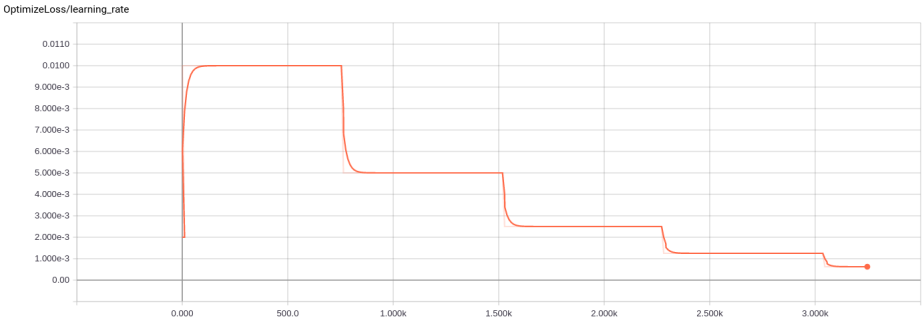

Figure 6: Part of the Computational Graph.

In Table 1, we provide a summary of the experiment validating the quantitative effectiveness of attetion. We show the BLEU score obtained based on our attention model on MSCOCO dataset. The result is not good enough compared with other mature methods of image captioning.

Finally, some visualizations of evaluation procedure which can show which part of an image the attention focuses on are presented. In Figures 7 ˜ 12, the parts in white show the location of the image the attention is focusing on. We can see that althogh this model hasn't been trained completely, it is able to produce basic grammar and recognize objects. However, it is unable to produce a piece of caption describing the main subject in a sentence except "man".

6 Conclusion

We propose an attention model which can be added between encoder and decoder of an image captioning system. This approach can produce basic sentence describing what objects are in an image, where are these objects and how are they interacting. We evaluate its performance using BLEU matric. The attention methodology is very heuristic due to its similarity with functions of human eyes. During this project, I got to know convolutional neural network and recurrent neural network in more depth and enriched my research experience. For the future, I may focus more on inner structure of attention model to obtain more accuracy of generating captions.

Table 1: BLEU-1,2,3,4 Metrics

Dataset	Model	BLEU-1	BLEU-2	BLEU-3	BLEU-4
COCO	Attention	0.572021	0.322533	0.172477	0.090547

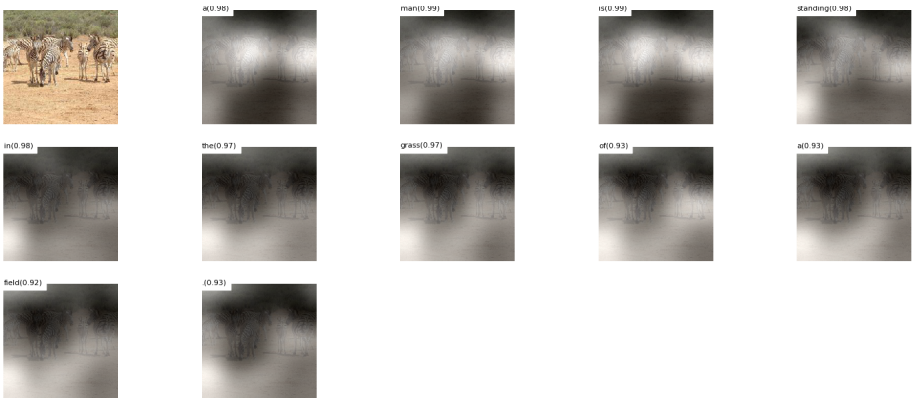

Figure 7: Caption: a man is standing in the grass of a field .

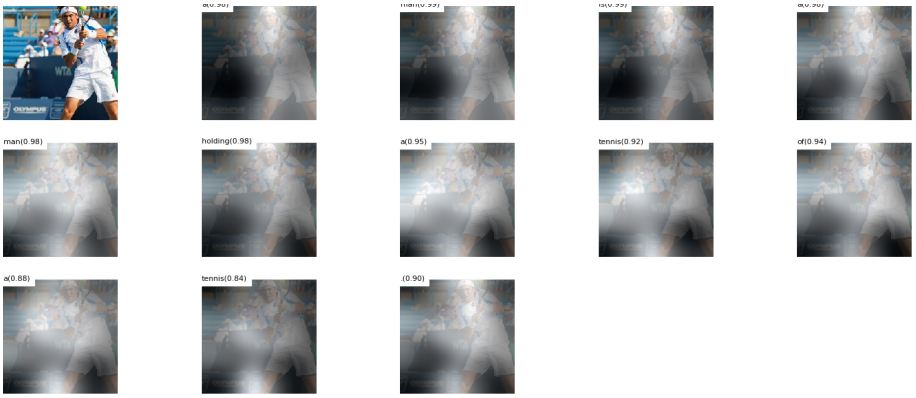

Figure 8: Caption: a man is a man holding a tennis of a tennis .

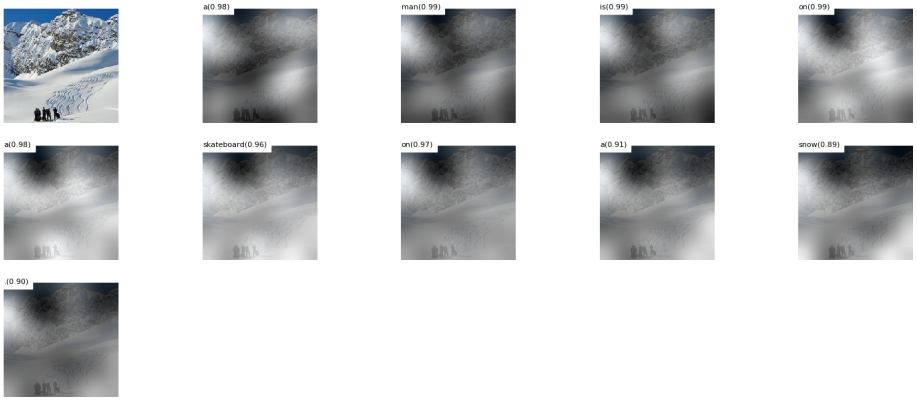

Figure 9: Caption: a man is on a skatorboard on a snow .

Figure 10: Caption: a man is a man in a room with a table.

Figure 11: Caption: a man is a man on a skatorboard .

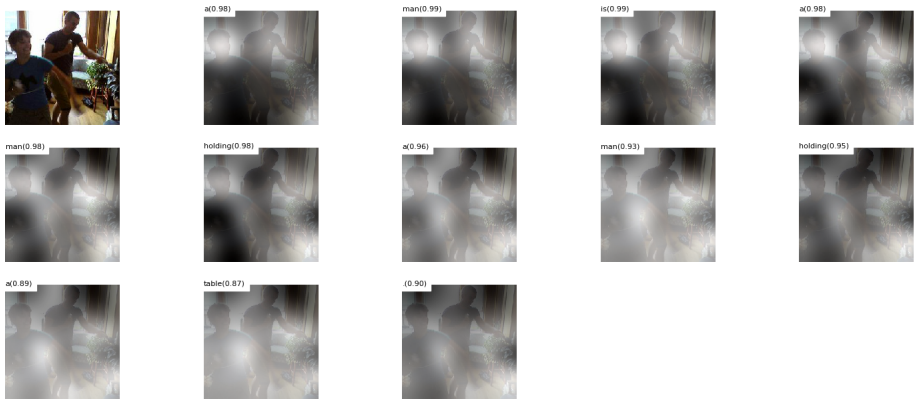

Figure 12: Caption: a man is a man holding a man holding a table .

References

[1] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*, 2014.

[2] Maurizio Corbetta and Gordon L Shulman. Control of goal-directed and stimulus-driven attention in the brain. *Nature reviews neuroscience*, 3(3):201–215, 2002.

[3] Jeffrey Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, and Trevor Darrell. Long-term recurrent convolutional networks for visual recognition and description. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2625–2634, 2015.

[4] Hao Fang, Saurabh Gupta, Forrest Iandola, Rupesh K Srivastava, Li Deng, Piotr Dollár, Jianfeng Gao, Xiaodong He, Margaret Mitchell, John C Platt, et al. From captions to visual concepts and back. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1473–1482, 2015.

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

[6] Andrej Karpathy and Li Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3128–3137, 2015.

[7] Ryan Kiros, Ruslan Salakhutdinov, and Rich Zemel. Multimodal neural language models. In *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pages 595–603, 2014.

[8] Ryan Kiros, Ruslan Salakhutdinov, and Richard S Zemel. Unifying visual-semantic embeddings with multimodal neural language models. *arXiv preprint arXiv:1411.2539*, 2014.

[9] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[10] Hugo Larochelle and Geoffrey E Hinton. Learning to combine foveal glimpses with a third-order boltzmann machine. In *Advances in neural information processing systems*, pages 1243–1251, 2010.

[11] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[12] Junhua Mao, Wei Xu, Yi Yang, Jiang Wang, Zhiheng Huang, and Alan Yuille. Deep captioning with multimodal recurrent neural networks (m-rnn). *arXiv preprint arXiv:1412.6632*, 2014.

[13] Ronald A Rensink. The dynamic representation of scenes. *Visual cognition*, 7(1-3):17–42, 2000.

[14] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015.

[15] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[16] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2818–2826, 2016.

[17] Yichuan Tang, Nitish Srivastava, and Ruslan R Salakhutdinov. Learning generative models with visual attention. In *Advances in Neural Information Processing Systems*, pages 1808–1816, 2014.

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

[19] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *International Conference on Machine Learning*, pages 2048–2057, 2015.