# Improvements to: Hierarchical LSTMs with Adaptive Attention for Visual Captioning

## 1. Introduction

Attention based encoder decoder frameworks are widely used for image and video captioning. Referenced article (Gao et al., 2019) is one of such papers with two contributions to the existing literature: one of which is the usage of natural language processing for non-visual words such as “the” and “a”, and the other is the use of hierarchical LSTM. The model proposed by the referenced article uses spatial or temporal attention to select specific frames to predict the related words whereas adaptive attention is utilized to decide whether to rely on the visual information. The paper includes connection between computer vision and natural language processing. Here, natural language processing enables to predict next words and therefore, better sentence for visual data can be obtained. This is one of the key points for image and video captioning because captions are suitable, clear and well for both given data and language rules. This framework is applied to both video and image captioning.

In this paper, referenced paper’s methodology and the related work in the literature are explained. Then, hypotheses are drawn with the reference from the similar works in the literature.

In the referenced paper, the goal is to generate a natural language sentence automatically for a given image or video data by using a hierarchical LSTMs with adaptive attention model. There are 5 different data sets in this paper:

1. The Microsoft Video Description Corpus (MSVD) - 1,970 short video clips
2. MSR Video to Text (MSR-VTT) - 10,000 web video clips
3. Large Scale Movie Description Challenge (LSMDC) - 118,081 video clips from 202 movies
4. COCO - 164,775 images
5. Flickr30K - 31,783 images

Let talk about what the hierarchical LSTMs with adaptive attention model for visual captioning is. Here, LSTM refers to Long Short-Term Memory. In this paper, LSTM is used for dealing with vanishing gradient problem because LSTM adds several hidden layers. In a recurrent neural network, the gradient increases or decreases exponentially over time, so the gradient disappeared for the weight of the next step (Sundermeyer et al., 2012). Therefore, LSTM is used for this purpose in this paper. Hierarchical LSTMs with adaptive attention model for visual captioning includes three components:

1. CNN Network as Encoder
2. Hierarchical LSTMs as Decoder
3. Maximum Likelihood Estimation as Losses

In addition, adaptive attention model is used both to get non-visual words such as “the”, “is” and “a” and to predict visual words for given image or video. Adaptive attention enables to update for hidden states in decoder part. Therefore, updated attention can be used as input and obtain captions for visual data by generating LSTMs for visual and language information.

## 2. Related Work

The paper deals with the image and video captioning problem by using hierarchical LSTMs with adaptive attention model. In literature, there have been many studies for this problem by using different methods and different models for captioning of both images and videos. Song et al. (2016) used optimized graph learning method for image and video annotation. This study focused on improvement of similarity graph for both labelled and un-labelled data by utilizing geometrical relationships; that is, it supported that similar data have similar labels by using optimized graph learning algorithm. Similar work (Song et al., 2018) improved self-supervised video hashing model by generating a binary auto-encoders. They argued for similarity of the data using neighborhood relations of the videos. On the other hand, “Translating videos to natural language using deep recurrent neural networks” paper (Venugopalan et al., 2015) and “Sequence to sequence-video to text” paper (Venugopalan et al., 2015) were based on images by video sequences and then, they generated a LSTM-based RNN model to translate videos to text. In other words, they actually used image-to-text model, but the data sets were obtained by video sequences.

Some researchers also focused on image captioning. Vinyals et al. (2015) improved a deep model based on LSTM to generate sentence for images with Flickr30k and MS COCO data sets. They relied on connection between computer vision and natural language processing. This is an important challenge for image and video captioning because captions are suitable for both given data and language rules. Similarly, “Show, attend and tell: Neural image caption generation with visual attention” paper (Xu et al., 2015) improved a LSTM based model by focusing standard backpropagation methods and tried to maximize BLEU scores. Differently, in this study, they used also Flickr9k data set and changed parameters and hyperparameters of the model. Ju et al. (2017) worked on an attention-based model for image captioning. Their model was to decide whether the visual data accompany to meaningful words and based on RNN with spatial and adaptive attention models. Therefore, they compared their results with previous works using BLEU, CIDEr, Meteor and Rouge-L scores by training the model with Flickr30k and MS COCO data sets.

For captioning of image and video, feature extraction is a very important task. For this purpose, CNN features were used by Krizhevsky et al. (2012), Szegedy et al. (2015), Simonyan & Zisserman (2014) and He et al. (2016). For these studies, there have been many different types of CNN features and they are based on extracting features like AlexNet (Krizhevsky et al., 2012), GoogLenet (Szegedy et al., 2015), VGG (Simonyan & Zisserman, 2014) and ResNet (He et al., 2016). Also, C3D model utilized to extract features for images and videos (Tran et al., 2015 and Wang et al., 2018). C3D helps to find temporal and motion information from the given visual data. In “Hierarchical LSTMs with Adaptive Attention for Visual Captioning” paper (Gao et al., 2019), that is, our paper we selected, utilized CNN features for appearance feature for image data and C3D model for extracting video motion information for video data.

## 3. Method

Although the performance of the referenced paper’s approach is very good, there are some limitations to the task of video and image captioning. In this section, these limitations are discussed, and solutions are proposed making use of the existing literature.

Before talking about limitations and improvements for them, we explain what we do during do project and how we try to improve it. We try to run codes which are supplied for video captioning. However, their codes are compatible with python 2 although libraries in the codes are compatible with python 3. For example, they use matplotlib library in some parts, but codes have some errors since mismatching. Also, we cannot write codes from scratch due to the complexity of the model and not having enough instructions to extract features for videos or images. The paper says CNN encoder for feature extraction, but we do not have enough information about how to use it and how to apply it from scratch. We think we change hyperparameters in the model. Model has many parameters and hyperparameters such as encoder dimension, video feature like GoogLenet, number of words. This is also a disadvantage in terms of controlling them and selecting the best.

When we run some codes in this mode, we face word dictionary to convert string captions to numbers. We think this is a disadvantage for new words; however, we cannot improve it since this way is used generally in the literature. Also, shape of the first video is (30, 4096) and second video shape (4, 4096). For each video, there are some captions which are said by people. For the labels, first video has 29 different captions while second video has 43 different captions. We did not decide how to use them because all shapes are different and how to deal with them. So, we cannot make a similar model from scratch. Therefore, we focus on limitations and try to compose a hypothesis by using literature.

The first limitation to the hLSTMat model is that it uses only 2 layers of LSTM, where each layer deals with a part of the task before passing it on to the next until the creation of the output. Specifically, the first layer decodes visual features whereas the second one mines deep language context information. This way, the LSTM network improves the quality of the captions generated. However, the usage of only two LSTMs may not be enough for such a complicated task, and we know that deep RNNs work better than shallower ones (Nabati & Behrad, 2020). That is why we propose the usage of more LSTM layers.

Another limitation of the referenced paper's model is about the spatio-temporal object interaction. Humans recognize actions by noticing the shape of an object and observing how it changes over time. In such process, they are aware of the human-to-object and object-to-object interactions. However, in the referenced paper's approach only models the temporal dynamics of objects. That is, it classifies actions based on individual video frames. Although it uses recurrent neural networks, it still focuses on the features extracted from the scenes and cannot capture the long-range temporal dependencies. To tackle this issue, we propose a method by Wang et al. (2017). They built a graph-based reasoning framework, where they represent videos as graph of objects. In this framework, input videos are represented as a space time region graph where each node in the graph corresponds to the region of interest in the video. These nodes are connected as follows: Regions with similar appearance are connected, this way the change in the object states can be modeled. Also, object that overlap in space and close in time are joined together, this way interactions between nearby objects and ordering of object changes are captured.

Specifically, this model takes more than 100 equally spaced frames from a video whereas the referenced paper's model takes only 28 frames. After the feature extraction, similarity between objects in the feature space is measured to construct the similarity graph. Pairwise similarity every two proposals can be represented as:

$$F(\mathbf{x}_i, \mathbf{x}_j) = \phi(\mathbf{x}_i)^T \phi'(\mathbf{x}_j)$$

where X = {$x_1$, $x_2$, …. $x_N$} which is the feature space, N is the # of object proposals, each object proposal feature $x_i$ is a d-dimensional vector, $\phi(\mathbf{x}) = \mathbf{w}\mathbf{x}$ ,$\phi'(\mathbf{x}) = \mathbf{w}'\mathbf{x}$. and w and w' are d x d weights that are learned with back propagation.

By adding the w and w', we can learn the correlations between different states of the same object across frames and the relation between different objects. Then, a spatial-temporal graph is constructed in which the object close to each other in space and time are connected together. Given an object proposal in frame t, the Intersection Over Unions (IoUs) value between this and the other object bounding boxes in frame t+1 are calculated. IoU between object i in frame t and object j in frame is denoted by $\sigma_{ij}$. If IoU is greater than 0, object i and object j are connected. After the construction of the similarity and spatial-temporal graph, Graph Convolutional Networks are used to make use of them. Via GCNs, features belonging to each object are updated by inspecting neighboring nodes in the graphs.

We propose that the discussed solution to the representation of the spatio-temporal object interaction would increase the performance of hLSMat when it is located between the CNN and LSTMs. With this method, the referenced paper's approach would make use of the interaction between each frame, which is closer to the human perception. Also, in the paper by Wang et al. (2017), we see that their proposed approach improved the performance of a model without GCNs.

In addition to above limitations, in referenced paper, CNN networks are used to extract features for images and videos. The goal of this is to get compact features and then make visual data suitable for decoding part. However, CNN with LSTM based model to extract features can be trained. Wu et al. (2017), Gao et al. (2015) and Ren et al. (2015) used CNN architectures which fed with LSTM to extract features for visual data. VIS+LSTM and VIS+BLSTM is for dimensionality reduction and linear transformations respectively (Ren et al., 2015) These models enable to get better weight matrix and input features as at start and end by transforming linearly. In “Are you talking to a machine? dataset and methods for multilingual image question answering” paper (Gao et al., 2015), there is a multimodal question answering model. The components of this model enable to extract better features in terms of question representation, visual representation and language context by using LSTM. Similarly, Wu et al. (2017) support CNN with LSTM based model for feature extraction contains higher level concepts to look for visual features to text directly.

Using only CNN as encoder to extract features leads to a problem. For example, in one of the studies (Srivastava, 2019), similarly CNN is used as encoder and LSTMs are used in decoder part. In the image, there is a woman who is sitting on a toilet seat cover and wearing a t-shirt. However, the model says a man in a darth vader shirt and tie. This is a failure for CNN encoder. Therefore, as a solution for only using CNN for feature extraction part, we assume that CNN+LSTM model can be worked better.

To sum up, our hypothesis is to use an improved hierarchical LSTMs with adaptive attention model for visual captioning. This model has improved parts in terms of using more layers LSTMs, spatio-temporal object interaction and CNN+LSTM as encoder while extracting features. This improved model has four components:

1. CNN+LSTM Network as Encoder
2. Spatio-Temporal Object Interaction as Decoder
3. Hierarchical LSTMs as Decoder
4. Maximum Likelihood Estimation as Losses

That is, it is actually composed of extra parts which can be achieved by developing the limited constraints we talked above discussion.

# REFERENCE

- Gao, H., Mao, J., Zhou, J., Huang, Z., Wang, L., & Xu, W. (2015). Are you talking to a machine? dataset and methods for multilingual image question answering. *arXiv preprint arXiv:1505.05612*.
- Gao, L., Li, X., Song, J., & Shen, H. T. (2019). Hierarchical LSTMs with adaptive attention for visual captioning. *IEEE transactions on pattern analysis and machine intelligence*, *42*(5), 1112-1131.*
- He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770-778.
- Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, *25*, 1097-1105.
- Lu, J., Xiong, C., Parikh, D., & Socher, R. (2017). 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*, 375-383.
- Nabati, M., & Behrad, A. (2020). Video captioning using boosted and parallel Long Short-Term Memory networks. *Computer Vision and Image Understanding*, *190*, 102840.
- Ren, M., Kiros, R., & Zemel, R. (2015). Exploring models and data for image question answering. *arXiv preprint arXiv:1505.02074*.
- Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.
- Srivastava, S. (2019). Using a CNN-LSTM hybrid network to generate captions for images. Retrived from https://github.com/siddsrivastava/Image-captioning
- Song, J., Gao, L., Nie, F., Shen, H. T., Yan, Y., & Sebe, N. (2016). Optimized graph learning using partial tags and multiple features for image and video annotation. *IEEE Transactions on Image Processing*, *25*(11), 4999-5011.
- Song, J., Zhang, H., Li, X., Gao, L., Wang, M., & Hong, R. (2018). Self-supervised video hashing with hierarchical binary auto-encoder. *IEEE Transactions on Image Processing*, *27*(7), 3210-3221.

- Sundermeyer, M., Schlüter, R., & Ney, H. (2012). LSTM neural networks for language modeling. In *Thirteenth annual conference of the international speech communication association*.
- Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., ... & Rabinovich, A. (2015). Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1-9.
- Tran, D., Bourdev, L., Fergus, R., Torresani, L., & Paluri, M. (2015). Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, 4489-4497.
- Venugopalan, S., Xu, H., Donahue, J., Rohrbach, M., Mooney, R., & Saenko, K. (2015). Translating videos to natural language using deep recurrent neural networks. *NAACL HLT,* 1494-1504.
- Venugopalan, S., Rohrbach, M., Donahue, J., Mooney, R., Darrell, T., & Saenko, K. (2015). Sequence to sequence-video to text. In *Proceedings of the IEEE international conference on computer vision*, 4534-4542.
- Vinyals, O., Toshev, A., Bengio, S., & Erhan, D. (2015). Show and tell: A neural image caption generator. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 3156-3164.
- Wang, X., Gao, L., Wang, P., Sun, X., & Liu, X. (2017). Two-stream 3-d convnet fusion for action recognition in videos with arbitrary size and length. *IEEE Transactions on Multimedia*, *20*(3), 634-644.
- Wu, Q., Shen, C., Wang, P., Dick, A., & Van Den Hengel, A. (2017). Image captioning and visual question answering based on attributes and external knowledge. *IEEE transactions on pattern analysis and machine intelligence*, *40*(6), 1367-1381.
- Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhudinov, R., Zemel, R., & Bengio, Y. (2015). Show, attend and tell: Neural image caption generation with visual attention. In *International conference on machine learning*, 2048-2057.