Temporally Steered Gaussian Attention for Video Understanding

Shagan Sah Thang Nguyen Miguel Dominguez
Felipe Petroski Such Raymond Ptucha
Rochester Institute of Technology
Rochester, New York, USA
email: sxs4337@rit.edu

Abstract

Recent advances in video understanding are enabling incredible developments in video search, summarization, automatic captioning and human computer interaction. Attention mechanisms are a powerful way to steer focus onto different sections of the video. Existing mechanisms are driven by prior training probabilities and require input instances of identical temporal duration. We introduce an intuitive video understanding framework which combines continuous attention mechanisms over a family of Gaussian distributions with a hierarchical based video representation. The hierarchical framework enables efficient abstract temporal representations of video. Video attributes steer the attention mechanism intelligently independent of video length. Our fully learnable end-to-end approach helps predict salient temporal regions of action/objects in the video. We demonstrate state-of-the-art captioning results on the popular MSVD, MSR-VTT and M-VAD video datasets.

1. Introduction

Automatically describing videos with natural language text enables more efficient search and retrieval. It can aid visual understanding in the medical, security, and military applications, and can even be used to describe pictorial content to the visually impaired. Recent advances in image classification [16, 10], object detection [26], semantic segmentation [18], image captioning [7, 13, 36], and localized image description [12] tasks have fostered dramatic improvements in image understanding. Spatial [36], temporal [38, 21, 40] and attribute [39] based attention models strive to localize objects in image frames, actions in videos or attend to specific word attributes. These attention mechanisms helped fuel recent progress, but our ability to understand how well temporal attention works on video is limited given that most datasets are comprised of short videos.

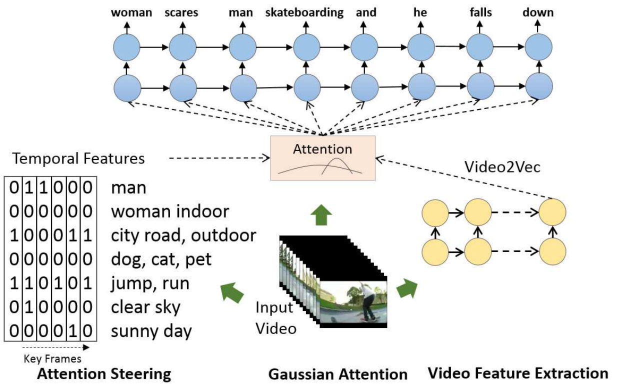


Figure 1. Overview of the Steered Gaussian Attention Model for video captioning. The attention filter is learned by training statistics (center), temporal features (left), and a video summary (right).

Current methods for generating attention weights are determined by temporal trends in the training data, not by visual concepts. For example, if training videos often end with a human falling down, the model may learn to associate end of video as it predicts the words "falls down". If a video and corresponding caption are "Woman scares man skateboarding and he falls down", the attention model would perform as expected. But if the caption is "man falls down and then does great skateboard trick", the trained attention parameters would perform poorly as the model would seek attention over the wrong temporal location in the video. Figure 1 shows a high level overview of our proposed model. To steer the attention mechanism to the proper region in the video, our model extracts frame-wise visual concepts across the length of the video. This enables the model to correlate specific concepts such as woman, man, and skateboarding, with region-specific locations across the video.

As attention weights are learned parameters, and parameters need to be fixed at train time, attention models are constrained such that all samples have equivalent number of regions. To enable the attention mechanism to be independent of video duration, we present a Gaussian attention model which learns a continuous function.

Our steered Gaussian attention model uses an intuitive video2vec latent encoding. When applied to variable length videos in a hierarchical fashion, we form an elegant architecture which can obtain state-of-the-art captioning results on the MSR-VTT [35], MSVD [6] and M-VAD [31] video captioning datasets. Our two main contributions are the introduction of length agnostic Gaussian attention models and temporal steering of these models:

Gaussian parametric attention. Soft attention models have an intrinsic limitation that all input buffers need to be of the same duration. This is because the attention vector is associated with a learnable, but fixed dimension weight matrix. For videos, this requires reducing longer videos or padding shorter videos. The proposed parametric Gaussian attention model removes this limitation by applying a continuous, rather than discrete weight distribution.

Temporal steering. Existing attention models are guided by temporal features of the training data. For example, when the phrase “winning goal” occurs, the attention might jump towards the end of video. The introduced temporal attention steering mechanism uses frame level visual concepts to guide attention based on current video properties (detection of objects, activities, etc.) and not on training data trends.

2. Related Work

Success of deep learning in the still image domain has influenced research in the video understanding domain [15, 4]. Early work on video captioning relied on extracting semantic content such as subject, verb, object, and associating it with the visual elements [30, 37]. For instance, [30] form a Factor Graph Model to obtain the probability for the semantic content and then use a search based optimization to combine a subject, verb and object to fit in a sentence template. With availability of large video-sentence pair datasets with rich language information, recent studies [33, 7] have demonstrated use of neural networks to directly model language conditioned on video.

Initial works that introduced Recurrent Neural Networks (RNNs) for video captioning used a mean pooled feature as the video representation [33]. An alternate approach uses an encoder-decoder [29] framework that first encodes f frames, one at a time to the first layer of a two layer Long-Short-Term Memory (LSTM), where f can be of variable length. S2VT [34] encodes the entire video, then decodes one word at a time.

Attention mechanisms were initially proposed in [2] and used in video captioning context by [38]. They allow the focus of relevant temporal segments of a video conditioned on the text-generating recurrent network. Spatial attention over parts of an image is shown by [36]. They also present a hard-attention mechanism equivalent to reinforcement learning with the reward for selecting the image region proportional to the target sentence. Semantic attention over word attributes has been shown to enhance image captioning by [39]. Similarly, [8] and [41] included video attributes or tags to help generate improved captions. More recently, video captioning was extended to paragraph generation using independent recurrent networks at the word and sentence level [40]. Hierarchical recurrent networks have also been used to encode the video in an embedding before generating words [21].

Knowledge transfer from independent language and image data for image captioning was demonstrated by [11]. Our work is loosely inspired by this study because we want to use sentence independent visual features to improve the generated captions. Our work is additionally inspired by the soft attention model for video captioning presented in [38]. We augment it by parameterizing the attention mechanism with a Gaussian distribution over the video length and then further guide the attention using independent temporal “concepts” of the video inspired by the word attributes from [39]. Gaussian attention filters are discussed in [24] but the application is limited to activity classification and their equally spaced attention filters limit the use of attention for word generation. Our model is length agnostic since each Gaussian learns normalized mean and sigma values from the distribution.

2.1. Soft Attention

A simple way to encode video features is by averaging pixels or features across all frames in the video. Most commonly, features are the output of a frame passed into an ImageNet pre-trained CNN. Soft Attention (SA) uses a weighted combination of these frame-level features, where the weights are influenced by the word decoder. Soft attention was first used in the context of video captioning in [38]. They computed a frame relevance score $e_i^{(t)}$ for each frame i of video $v_1, v_2, ..., v_n$ at decoder time step t.

$$e_i^{(t)} = \mathsf{w}^\top tanh(W_a h_{t-1} + U_a v_i + b_a) \tag{1}$$

Where, h_{t-1} is the hidden state at the previous time step of the decoder, v_i is the frame feature vector representation of the i^{th} frame, and w, W_a, U_a, b_a are learned parameters. This can be interpreted as an alignment between the encoder and decoder sequence. It allows the video encoder to selectively emphasize relevant parts of the video. As the frame relevance score is computed using fixed dimension weight matrices, it restricts the exact number of frames in the video. Moreover, given that the average length of videos is a few seconds in most datasets, it seems counter intuitive to have strong localized attention in such a short duration.

3. The Steered Gaussian Attention Model

This section describes the main components of our model- Gaussian Attention, steering and Video2Vec representation.

3.1. Gaussian Attention

We define the Gaussian Attention (GA) to remove restrictions with the generic soft attention mechanism. The relevance score that weighs the input sequence is modeled with a Gaussian distribution. At each time step, the decoder observes a filtered/weighted encoder sequence. GA weights the input sequence based on the temporal location and the shape of the distribution modeled by the mean and standard deviation, respectively. We adapt the function to compute a continuous relevance score e^t across the entire input sequence $X = (x_1, x_2, ..., x_F)$ at decoder time step t as-

$$e^t = \sum_{k=1}^{N} \pi_k \mathcal{N}(X|\mu_k^t, \Sigma_k^t) \tag{2}$$

Where, each GA $\mathcal{N}(X|\mu_k^t, \Sigma_k^t)$ is a Gaussian distribution with its unique mean μ_k^t and covariance matrix Σ_k^t at time t, N is the number of Gaussians and π_k is the mixing coefficient. The mixing coefficients are normalized to sum to one. The input features $X \in \mathbb{R}^{D \times F \times M}$, where D is the number of input modalities, F is the length of the each sequence, and M is the dimension of each feature. For example, if the two input modalities of spatial domain and temporal domain are used, we can learn a unique set of Guassians for each modality by setting $D = 2$. By varying mixing coefficients, mean and covariance of basic Gaussians, the superposition can approximate any continuous function by using sufficient number of Gaussians. Hence with correct parameters, a GA model can achieve the same function as soft attention. We choose to model independent Gaussians, and replace Σ_k^t with a scalar standard deviation, σ_k^t at each time t.

Computing the parameters allows the filter to temporally adapt to decoder decisions. With loss back-propagated at each time step, the mean value of the Gaussian learns to control focus on relevant locations of the sequence. Similarly, the standard deviation can learn to extract information from a longer or shorter segment. Thus, the GA formulation makes it adaptive both in terms of location and range. Resource utilization can be optimized as the decoder need not necessarily compute attention over the entire input sequence. The mean and standard deviation are computed as:

$$\mu^t = \wp(W_\mu h_{t-1} + U_\mu X + b_\mu) \tag{3}$$

$$\sigma^t = |W_\sigma h_{t-1} + U_\sigma X + b_\sigma| \tag{4}$$

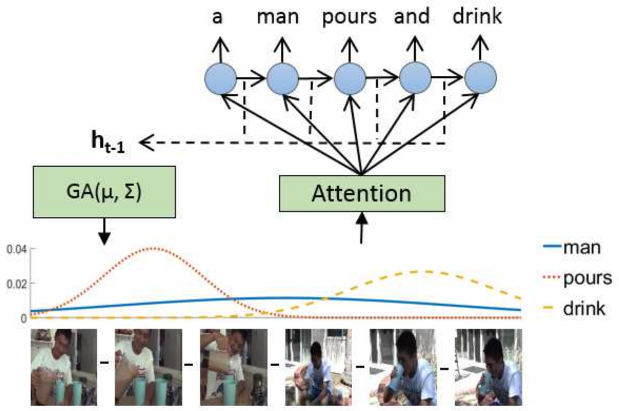


Figure 2. Illustration of the parameterized Gaussian attention model for steering the temporal alignment between the video and word sequence. The caption is generated using a recurrent neural network. For a video, the mean and standard deviation of the distribution is computed based on the outputs of the previous time steps (dotted lines). The curves depict change in the attention over the video based on the word generated in the caption generator.

Where, W_μ, W_σ, U_μ, U_σ, b_μ, b_σ are learned weights. We use the activation $\wp(s) = |s|/(|s| + c)$ for the mean values to scale to range [0,1] as the input sequence is normalized temporally. The normalization allows the model to compute attention over sequences of varying length. It also reduces the number of learnable weights from $\mathbb{R}^{h \times h}$ to $\mathbb{R}^{h \times N}$, where h is hidden dimension size of decoder and $N \ll h$. Similar to soft attention, the attention weights α_i^t at time t for input X are obtained by normalizing the relevance scores. The input to the decoder is a weighted sum of the input X using the attention at time t.

$$\alpha_i^t = \frac{exp(e_i^t)}{\sum_{j=1}^{F} exp(e_j^t)} \tag{5}$$

$$\Phi_t(X) = \sum_{i=1}^{F} \alpha_i^t x_i \tag{6}$$

Modeling the attention filter with a parametric distribution allows the decoder to view inputs with varying duration and hence it is better at exploiting the temporal structure of an input sequence. The parametric attention has the capability to sense the complete encoder sequence if required. This is important in a translation like task where the generated word may hold relevance throughout the video. For example, after the word *man* in Figure 2, the model learns to expand the attention to allow the caption generator to view the entire input as the associated visual feature of *man* appears in the entire video.

3.2. Attention Steering

Traditional attention models are associated with a set of weight matrices that are learned during training. During test time, the weight matrices guide the attention and hence limit the attention mechanisms by prior temporal statistics. We introduce temporal attention steering that guides the attention based on the visual features of a test video. The temporal features across the video are normalized over all frames. The resulting matrix is a temporal map that translates feature relevance to frame relevance. At each LSTM time step, the model computes an updated frame relevance vector. For example, if the network computes that "apple" is an important feature for the next word prediction, the relevance factor of the feature "apple" will be higher. The temporal feature map in Figure 3 would then translate the relevance factor of "apple" to the center/end of the video. This provides a way to steer the attention without increasing the number of inputs to the system.

We further investigate the use of word label embeddings of objects present in video frames as temporal visual features. We use an ImageNet classifier trained on 4k classes [20] represented using a Glove [23] word embedding. Representing a large number of objects is important for "in-the-wild" videos. A bottom-up grouping strategy [20] is applied to the categories to deal with the problems of over-specific classes. In reality, a sentence is described by both the objects and the whole scene as the context. Distinguishing individual objects from others in a scene, especially when there are multiple objects of different categories, can be highly challenging. Hence, EdgeBox is used [42] to obtain proposal bounding box regions within each frame of a video. The Glove word embeddings of bounding box class labels are mean pooled to obtain a frame-level representation. We discover that the mean pooled class label embedding is rich in semantic information and is closer to the words in the ground truth sentence. As a complementary or alternative approach to temporal word embeddings, one could use frame CNN features directly.

3.3. Video2Vec Representation

In addition to the steering mechanism, an embedded vector representation of the entire video is input into the captioning model (right input in Figure 1). To learn powerful action and motion representations, we use a recent activity classification dataset- ActivityNet [5], on human activity understanding that covers a wide range of complex daily activities. It is comprised of 849 video hours in over 200 activity classes. As these videos were collected from online video sharing sites they are excellent to transfer learned features for MSVD and MSR-VTT datasets which are also based on Youtube videos. The labeled videos are used to train a standard video-based activity classifier. We utilize two independent models with RGB (3- color channels) and Optical Flow (OF) inputs. Features before the loss layer are used as *Video2Vec-Activity* representation.

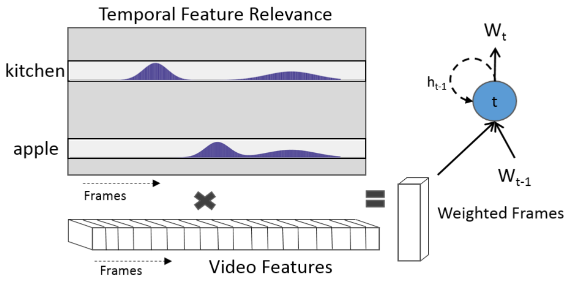


Figure 3. Attention steering using normalized temporal feature relevance. Frame level features are weighted based on the relevance map and assists in guiding attention to video regions. W_t and W_{t-1} are words at times t and $t-1$, h_{t-1} is RNN hidden state.

4. Captioning Framework

The video captioning framework has three main components- Attention Steering, Video2Vec encoder and Gaussian attention based sentence generator as shown in Figure 1 (left, right and center). The recently proposed hierarchical neural encoder [21] technique efficiently captures temporal dependencies in videos. Hence, we integrate it by replacing soft attention with Gaussian attention between all recurrent layers and term it as Hierarchy over Gaussian Attention (HGA). The sentence generation engine takes in input from all three to generate word sequences. Recurrent Neural Networks (RNN) are a natural choice for generating sequences such as natural language sentences. However, RNNs suffer from vanishing and exploding gradient problems when learning long sequences. To solve this, we use the LSTM variant of RNNs to learn sentence generation as it is known to learn sequences with both short and long temporal dependencies [7].

The model is trained using stochastic gradient descent by learning parameters θ for the sentence $w_1, w_2..., w_\tau$. The words loss for a video is $loss_{caption}$. The log-likelihood is optimized by minimizing the loss for video V with word embeddings V_c and Video2Vec embedding S_v.

$$\theta^* = \max_{\theta} \sum_{t=0}^{\tau+1} log(w_t|V, S_v, V_c, w_{t-1}; \theta) \tag{7}$$

where w_0 and $w_{\tau+1}$ are special tokens for start and end of sentence. During testing, the model is input with the token for beginning of sentence and it generates words until the end of sentence token is generated.

Inspired by the work in [9] on multi-modal embedding between text and visual inputs, we compute the cosine similarity between the mean pooled video level word embedding

(V_c) and Gaussian attention weighted video vector (Φ_V). This similarity measure is added to the caption generation loss for the entire video:

$$loss_{video} = loss_{caption} + \frac{V_c^T \Phi_V}{||V_c||\,||\Phi_V||} \tag{8}$$

5. Results and Discussion

5.1. Training Details

Each video frame is passed through the 152-layer ResNet CNN model [10] pre-trained on ImageNet data, where the $[1 \times 2048]$ vector from the last pooling layer-*pool5* is used as frame feature representation. In our HGA model, the inputs to the first hierarchical layer are 12 frame clips and the output at the last time step is the input to the second hierarchy layer. We use the PTBTokenizer in the Stanford CoreNLP tools [19] to pre-process all words in captions. This involved converting all text to lower case, removing punctuation and tokenizing the sentences. We use captions only from the training and validation set to generate the vocabulary. A one-hot vector encoding of the vocabulary is used to represent each word as a vector. For MSR-VTT video categories, we use 300-dimension Glove embedding [23] to obtain word vector representations.

During training, ADAM optimization is used to minimize the negative log likelihood loss. The learning rate is 2×10^{-4} and we use decay parameters ($\beta_1 = 0.9$, $\beta_2 = 0.999$) as reported in [14]. The hidden dimension of LSTM layers in HGA is 1024 and for the sentence generation layer is 384. We employ a Dropout [28] of 0.5 on the output of all LSTM layers. The mini-batch size is 100 and all models are trained for 40 epochs. Hyperparameters are selected by running tests on the validation set.

5.2. Dataset Description

We choose the Microsoft Video Description Dataset (MSVD) [6], the newly released Microsoft Research - Video to Text (MSR-VTT) [35] and the movie description datset M-VAD [31], to evaluate the proposed model. Standard train, validation and test splits were used for all datasets. Detailed statistics are listed in Table 1.

Table 1. Video-sentence pair dataset statistics.

	MSVD	MSR-VTT	M-VAD
#sentences	80,827	200,000	54,997
#sent. per video	∼42	20	∼1-2
vocab. size	9,729	24,282	16,307
avg. length	10.2s	14.8s	5.8s
#train video	1,200	6,513	36,921
#val. video	100	497	4,651
#test video	670	2,990	4,951

5.3. Evaluation Metrics

Quantitative evaluation was performed using the Microsoft COCO caption evaluation tool [1] to make our results directly comparable with other studies. For evaluation, we use standard metrics- BLEU [22], METEOR [3] CIDEr [32] and ROUGE [17] to score a predicted sentence against all ground truth sentences. Typically, the generated sentence correlates well with a human judgment when the metrics are high as they measure the overall sentence meaning and fluency. We report all scores as percentages (MET is METEOR and B-n is n-gram BLEU).

5.4. Performance on MSVD

Table 2 reports current captioning results (top half) vs. variations on our model (bottom half) on the MSVD dataset. Our baseline model (Baseline GA-5) is a Gaussian attention with five Gaussians. The addition of a hierarchical model (+HGA) shows significant improvement. The HGA model learns powerful motion features that a simple attention is unable to capture. As recommended in [40], we test a HGA variant with BLEU-4 score included in the caption loss (BLEU reg). The BLEU score is computed on the validation set and regularized with the loss after each mini-batch. Though it significantly improves BLEU scores, other scores are not much affected and we notice that sentence fluency degrades as well.

The addition of Video2Vec-Activity (+RGB,OF) further helps with METEOR scores due to importance of motion features. The highest METEOR score that we achieve is 33.1% which matches the state-of-the-art. We achieve a high BLEU-1 score with the baseline model and the hierarchy seems to hurt the *n*-gram metric. Our implementation of HRNE [21] yields a 31.7 Meteor indicating that more frames (we use 120 frames with hierarchy 8×15 in place of 160 frames with hierarchy 8×20), addition of Maxout in word prediction, and inclusion of test captions in vocabulary may be beneficial.

Table 2. MSVD caption evaluation results on the held out test set.

Method	MET	B-1	B-2	B-3	B-4
MP [33]	29.1	-	-	-	33.3
S2VT [34]	29.8	-	-	-	-
SA [38]	29.6	-	-	-	41.9
p-RNN [40]	32.6	81.5	**70.4**	**60.4**	**49.9**
HRNE Att [21]	**33.1**	79.2	66.3	55.1	43.8
Baseline GA-5	31.5	80.4	66.6	54.5	42.8
+HGA	32.8	79.1	65.8	54.8	43.9
BLEU reg	30.8	**81.6**	68.3	55.2	42.4
+ RGB, OF	**33.1**	77.5	64.1	53.4	43.0

Table 3. Performance evaluation with number of Gaussian filters for attention on the MSVD test set.

# Gaussians	MET	B-1	B-2	B-3	B-4
1	30.7	76.3	62.3	50.3	39.0
3	31.2	77.6	64.1	53.0	42.1
5	**31.5**	**80.4**	**66.6**	**54.5**	**42.8**

5.4.1 Analysis of Gaussian Attention

Since GA allows the model to focus on segments of the input, we train baseline GA models with 1, 3 and 5 Gaussians. Results are reported in Table 3. More attention curves allow the model to view specific but multiple regions of the input by increasing the number of learnable parameters. We observed exploding gradient problems with higher number of Gaussians as standard deviation starts to approach zero.

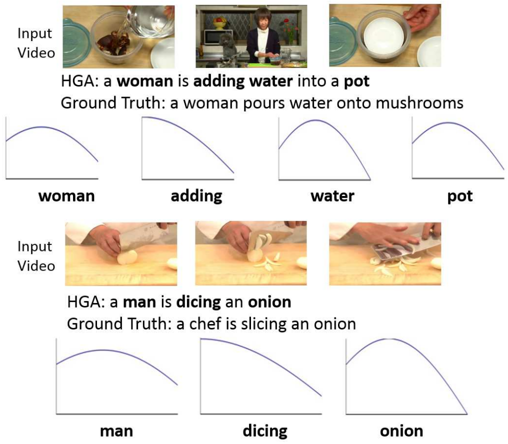


Figure 4. Gaussian attention visualization for sample videos from MSVD. Distribution focuses on relevant video segment based on key words (bold) in the sentence. For the word "adding", relevant activity is in the starting of video, hence the mean of the distribution is close to 0. X-axis ranges from $0 - 1$ normalized temporal video location and Y-axis is normalized attention weight α_i^t.

Figure 4 shows words from generated sentences along with the temporal Gaussian attention distributions generated on sample test MSVD videos. The distribution shows the adaptable nature of Gaussian attention. Even though the videos are short, at certain times the model needs to attend to different parts of the video. We anticipate that with longer and more complex videos, a higher number of Gaussians would be required. More broadly, these results indicate that the Gaussian attention is not restricted by the video duration since the videos are normalized temporally.

5.5. Performance on MSR-VTT

Caption evaluation scores for our models on the MSR-VTT dataset are reported in Tables 4 and 5. Each model is trained end-to-end and we compare our approach with recent results. A single layer GA performs better than the mean pooled video frame input features. The HGA model adds hierarchy features to a single layer GA model and hence is better at learning temporal dependencies. The significance of Gaussian attention is shown by comparison of HGA with and without attention. This has better performance than a weighted average through an attention mechanism. To study the importance of temporal steering (STE) and Video2Vec-Activity (RGB and OF) features, we also input these as features to the captioning model. All of these inputs have positive impacts on the evaluation metrics. The addition of activity features show clear improvement over the baseline HGA. The OF features yield slightly improved scores over RGB. This indicates that motion/activity features from the ActivityNet dataset generalize well to other datasets.

Table 4. HGA results on the held out MSR-VTT test set.

Method	MET	B-4	CIDEr	ROUGE-L
Only GA 1-layer	25.6	34.6	37.4	57.4
HGA (w/o att)	26.6	36.0	38.9	58.4
HGA	**27.4**	**38.8**	**43.4**	**59.1**

Across all features we observe that the scores did not change significantly when trained without word features loss from (8). However, it helped the model to converge faster. While generating the vocabulary from the training captions, we note that out of total 24,282 words, 10,155 words appear just once and 3,211 words twice. From the vocabulary, 4,716 words were not part of the Glove 400K dictionary. Such issues add to challenges of the language model. Similar trends appear in other datasets as well.

Fusion based models – Although the METEOR score does not improve with a combination of RGB and optical flow features, all other metrics show improvement. It also indicates that either of the features are sufficient to capture the activity information. We also use the Glove embedded video category label (CAT) available for all videos. The combined model is trained by concatenating the features before input to the LSTM. We note that the categories are the ground truth labels that are part of the original dataset and hence are better than any features generalized from an another dataset.

5.5.1 Gaussian Attention in Different HGA Layers

Experiments are run on the HGA model to compare soft and Gaussian attentions. The HGA-only model can be interpreted as a three layer LSTM with the first two hierarchical layers as the video encoder and the last layer as the sentence generator or word decoder. We replace soft attention with GA at multiple layer combinations. Results are reported in Table 6. Adding GA at more layers seem to help focus on relevant inputs and features. Attention on the middle HGA

Table 5. MSR-VTT results on the held out test set. We compare with recent entries in the MSR Video to Language Challenge.

Method		METEOR	BLEU-1	BLEU-2	BLEU-3	BLEU-4	CIDEr	ROUGE-L
Dong et al. [8]		26.9	-	-	-	39.3	45.9	58.3
Multimodal (only visual input) [25]		27.0	-	-	-	38.3	41.8	59.7
Shetty and Laaksonen [27]		27.7	-	-	-	**41.1**	**46.4**	59.6
Mean pool		25.4	75.3	60.4	46.4	34.1	35.8	57.7
Ours	HGA	27.4	79.7	64.8	51.1	38.8	43.4	59.1
	+ STE	27.6	79.8	64.4	50.1	37.9	43.4	59.2
	+RGB	27.6	79.7	64.6	50.8	38.6	42.8	58.9
	+OF	27.7	79.5	64.8	50.9	39.0	43.8	59.6
	+RGB, OF	27.7	79.9	64.9	51.0	39.2	43.5	59.2
	+RGB, OF, CAT	**28.2**	**80.3**	**66.1**	**52.5**	40.5	45.3	**60.4**

layers can be viewed as the weighted sum of the encoded outputs of video clips input to the first layer. Attention is most important at the word decoder (layer 3) as it not only finds relevant segments in the video but also relevant HGA encoded features based on generated words.

Table 6. Comparing Gaussian attention at different layers for MSR-VTT test set. Adding GA show clear improvement over SA and attention is most important at the word generation layer. MET is METEOR and B-1 to B-4 are n-gram BLEU scores.

Layer replacing SA with GA	MET	B-1	B-2	B-3	B-4
None	26.7	77.7	62.6	48.4	36.1
3	27.3	78.9	63.6	49.8	38.1
3,2	**27.4**	79.3	64.5	50.8	**38.8**
3,2,1 (HGA)	**27.4**	**79.7**	**64.8**	**51.1**	**38.8**

5.6. Performance on Movie Description Dataset

We present results of the HGA model on the M-VAD movie description dataset. This is a very challenging dataset as the videos are not specific activities but are movie scenes with complex sentences. We obtain a METEOR score of 6.9%, which is an improvement over the HRNE (6.8%) [21] and S2VT (6.7%) [34] models. The Bleu scores are 17.3%, 6.0%, 2.7%, 1.0% for 1,2,3,4 − grams, respectively.

6. Conclusion

We introduce a general purpose Steered Gaussian Attention Model for video understanding. Rather than use fixed training priors, we use video attributes as features along the length of the video to smartly steer the attention. When these temporal video features are bundled with a video summary vector, a semantically rich latent representation continuously feeds the captioning engine. A Gaussian parametric descriptor adds a degree of freedom to the input videos. The usage of hierarchical recurrent models are both efficient and robust. We demonstrate state-of-the-art captioning results on multiple video datasets.

Acknowledgement

We would like to thank NVIDIA for donating some of the GPUs used in this research.

References

[1] *Microsoft COCO Caption Evaluation*, (accessed October 3, 2016). `https://github.com/tylin/coco-caption`.

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

[3] S. Banerjee and A. Lavie. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *ACL workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, volume 29, pages 65–72, 2005.

[4] A. Barbu et al. Video in sentences out. *arXiv preprint arXiv:1204.2742*, 2012.

[5] F. Caba Heilbron et al. Activitynet: A large-scale video benchmark for human activity understanding. In *CVPR*, pages 961–970, 2015.

[6] D. L. Chen and W. B. Dolan. Collecting highly parallel data for paraphrase evaluation. In *Proceedings of the 49th Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 190–200. ACL, 2011.

[7] J. Donahue et al. Long-term recurrent convolutional networks for visual recognition and description. In *CVPR*, pages 2625–2634, 2015.

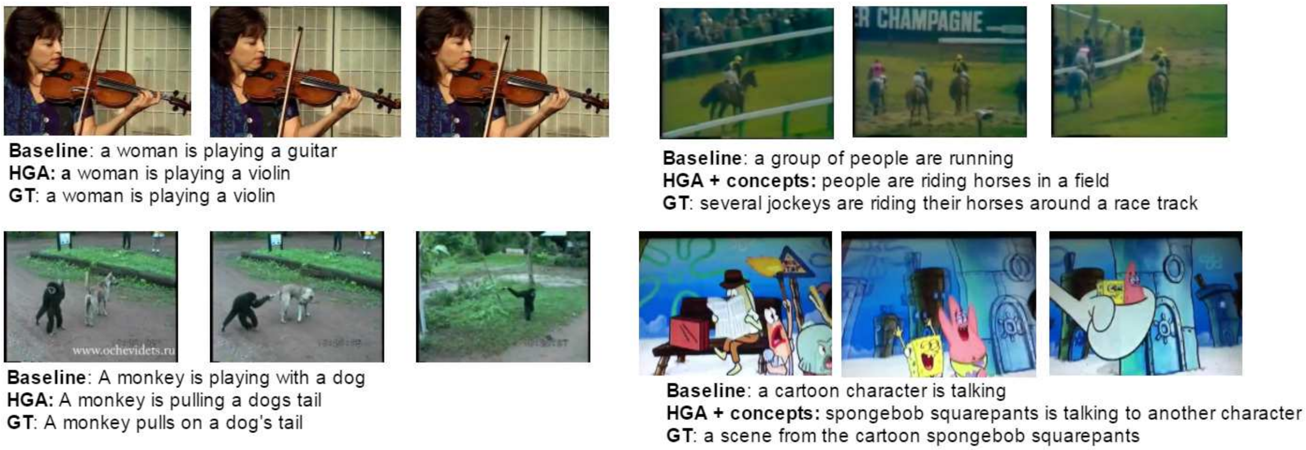
[8] J. Dong et al. Early embedding and late reranking for video captioning. In *Proceedings of the 2016 ACM on Multimedia Conference*, pages 1082–1086. ACM, 2016.

[9] J. Dong, X. Li, and C. G. Snoek. Word2visualvec: Cross-media retrieval by visual feature prediction. *arXiv preprint arXiv:1604.06838*, 2016.

[10] K. He et al. Deep residual learning for image recognition. *arXiv preprint arXiv:1512.03385*, 2015.

[11] L. A. Hendricks et al. Deep compositional captioning: Describing novel object categories without paired training data. *arXiv preprint arXiv:1511.05284*, 2015.

[12] J. Johnson et al. Densecap: Fully convolutional localization networks for dense captioning. In *CVPR*, 2016.



Baseline: a woman is playing a guitar
HGA: a woman is playing a violin
GT: a woman is playing a violin

Baseline: a group of people are running
HGA + concepts: people are riding horses in a field
GT: several jockeys are riding their horses around a race track

Baseline: A monkey is playing with a dog
HGA: A monkey is pulling a dogs tail
GT: A monkey pulls on a dog's tail

Baseline: a cartoon character is talking
HGA + concepts: spongebob squarepants is talking to another character
GT: a scene from the cartoon spongebob squarepants

Figure 5. Example videos and corresponding captions from the MSVD (left) and MSRVTT (right) datasets. For each video, three random frames are shown. Baseline is GA model with five Gaussian filters. HGA is our model and GT is a sample ground truth caption.

[13] A. Karpathy and L. Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *CVPR*, pages 3128–3137, 2015.

[14] D. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[15] A. Kojima, T. Tamura, and K. Fukunaga. Natural language description of human activities from video images based on concept hierarchy of actions. *IJCV*, 50(2):171–184, 2002.

[16] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, pages 1097–1105, 2012.

[17] C.-Y. Lin. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out: Proceedings of the ACL-04 workshop*, volume 8, 2004.

[18] J. Long et al. Fully convolutional networks for semantic segmentation. In *CVPR*, pages 3431–3440, 2015.

[19] C. D. Manning et al. The stanford corenlp natural language processing toolkit. In *ACL*, pages 55–60, 2014.

[20] P. Mettes, D. C. Koelma, and C. G. Snoek. The imagenet shuffle: Reorganized pre-training for video event detection. *arXiv preprint arXiv:1602.07119*, 2016.

[21] Pan et al. Hierarchical recurrent neural encoder for video representation with application to captioning. *arXiv preprint arXiv:1511.03476*, 2015.

[22] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. ACL, 2002.

[23] J. Pennington et al. Glove: Global vectors for word representation. In *EMNLP*, volume 14, pages 1532–43, 2014.

[24] A. Piergiovanni, C. Fan, and M. S. Ryoo. Temporal attention filters for human activity recognition in videos. *arXiv preprint arXiv:1605.08140*, 2016.

[25] V. Ramanishka et al. Multimodal video description. In *Proceedings of the ACM on Multimedia Conference*, pages 1092–1096, 2016.

[26] S. Ren, K. He, R. Girshick, and J. Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *NIPS*, 2015.

[27] R. Shetty and J. Laaksonen. Frame-and segment-level features and candidate pool evaluation for video caption generation. In *ACM on Multimedia Conf.*, pages 1073–1076, 2016.

[28] N. Srivastava et al. Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(1):1929–1958, 2014.

[29] Sutskever et al. Sequence to sequence learning with neural networks. In *NIPS*, pages 3104–3112, 2014.

[30] J. Thomason et al. Integrating language and vision to generate natural language descriptions of videos in the wild. In *Coling*, volume 2, page 9, 2014.

[31] A. Torabi et al. Using descriptive video services to create a large data source for video annotation research. *arXiv preprint arXiv:1503.01070*, 2015.

[32] R. Vedantam, C. Lawrence Zitnick, and D. Parikh. Cider: Consensus-based image description evaluation. In *CVPR*, pages 4566–4575, 2015.

[33] S. Venugopalan et al. Translating videos to natural language using deep recurrent neural networks. *arXiv preprint arXiv:1412.4729*, 2014.

[34] S. Venugopalan et al. Sequence to sequence-video to text. In *ICCV*, pages 4534–4542, 2015.

[35] J. Xu et al. Msr-vtt: A large video description dataset for bridging video and language. In *CVPR*, 2016.

[36] K. Xu et al. Show, attend and tell: Neural image caption generation with visual attention. *arXiv preprint arXiv:1502.03044*, 2(3):5, 2015.

[37] R. Xu et al. Jointly modeling deep video and compositional text to bridge vision and language in a unified framework. In *AAAI*, pages 2346–2352, 2015.

[38] L. Yao et al. Describing videos by exploiting temporal structure. In *ICCV*, pages 4507–4515, 2015.

[39] Q. You et al. Image captioning with semantic attention. *arXiv preprint arXiv:1603.03925*, 2016.

[40] H. Yu, J. Wang, Z. Huang, Y. Yang, and W. Xu. Video paragraph captioning using hierarchical recurrent neural networks. *arXiv preprint arXiv:1510.07712*, 2015.

[41] Y. Yu, H. Ko, J. Choi, and G. Kim. Video captioning and retrieval models with semantic attention. *arXiv preprint arXiv:1610.02947*, 2016.

[42] C. L. Zitnick and P. Dollár. Edge boxes: Locating object proposals from edges. In *ECCV*, 2014.