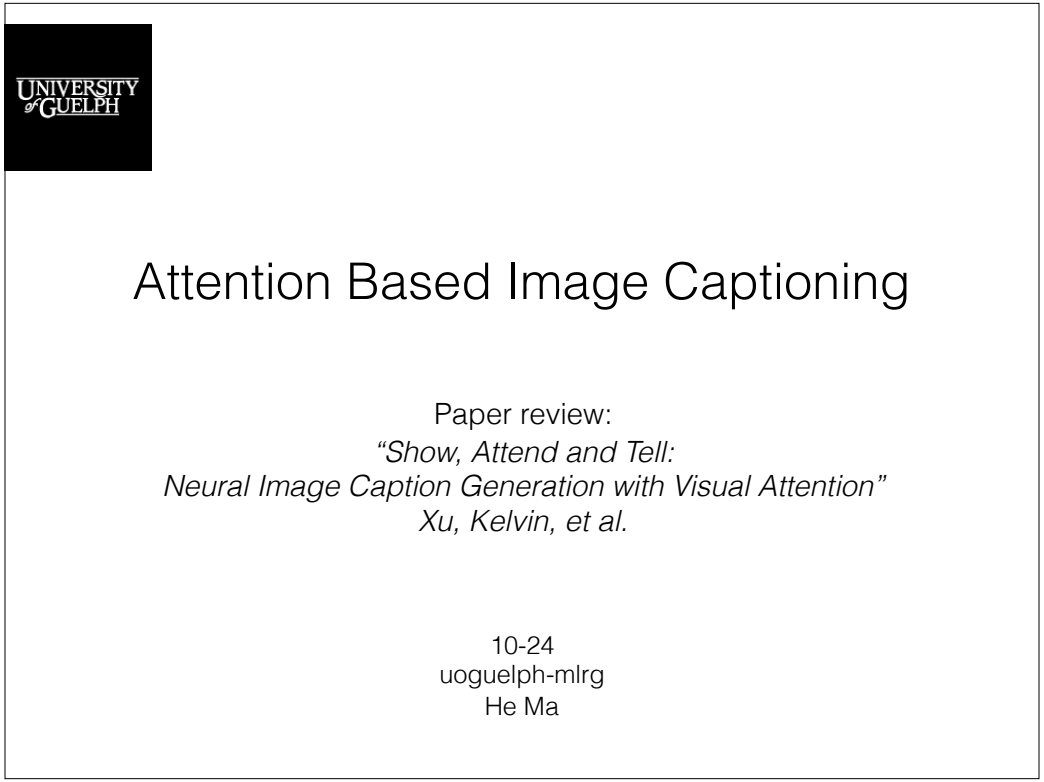
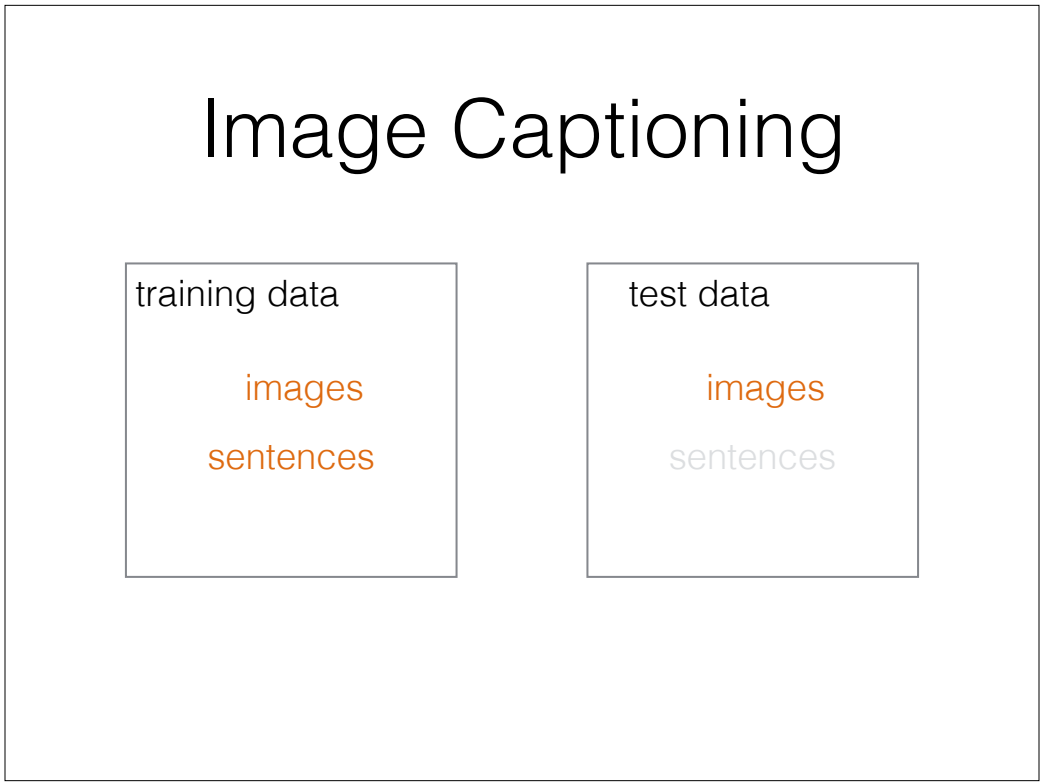


This paper actually came out last year.

This reason I selected this paper is that image caption is a joint area in deep learning combining both image and language models. It seems that the attention method used in the paper is well recognized as it is cited several hundreds times already. So in this presentation, I will show what is the contribution of the work in this paper.



First, what is image captioning.

The image captioning task is when

Image-Sentence Databases

- Image:
  - Microsoft COCO (82k images)
  - Flickr8k (8k images)
  - Flickr30k (30k images)
  - SBU Im2text (1M images)
- 5 captions each image

a man riding a bike on a dirt path through a forest.
bicyclist raises his fist as he rides on desert dirt trail.
this dirt bike rider is smiling and raising his fist in triumph.
a man riding a bicycle while pumping his fist in the air.
a mountain biker pumps his fist in celebration.

credit: Karpathy, et al. Automated Image Captioning with ConvNets and Recurrent Net

For example

Professional describers give those five reference sentences as the correct description of this image. The model tries to learn the ability of generating an similar and sensible description on image.

Image Captioning Examples

a bathroom with a sink and a mirror

a close up of a remote control on a table

a bench sitting in the middle of a park

credit: Karpathy, et al. Automated Image Captioning with ConvNets and Recurrent Net

Here are some examples of generated descriptions: we can see the model tries to describe what is happening in the image and resulting description is relevant but sometimes may not be accurate. How do we evaluate how accurate is the description? We evaluate based on comparing with reference sentences.

Fixed vocabulary size=10000

Image Captioning Metrics

- BLEU score:
  based on n-grams. BLEU-N is length-N score. The larger N, the harder candidate matches reference, the harder to get100% score.
  considers adequacy and fluency.

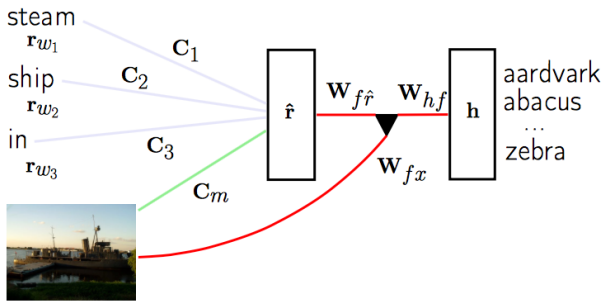
- METEOR score:
  consider exactness, stem, synonym, paraphrase

Fixed vocabulary size=10000

Similar to the metrics used in machine translation since the output is the same as a normal language model. It is just during training the input is an image instead of a sentence.

Image Captioning Related Works

- Kiros, et al. "Multimodal Neural Language Models". (2014)



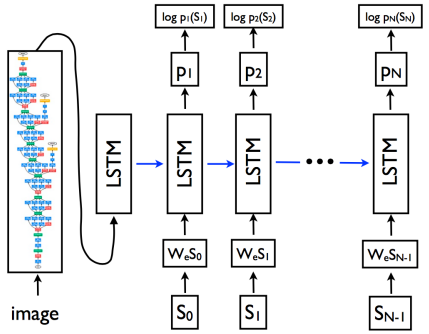
(b) Factored 3-way Log-Bilinear Model

Fixed vocabulary size=10000

In the first one by Kiros, a log-bilinear model is proposed which consists of a two-layer MLP basically. The models tries to predict the next word based on the previous n-1 words conditioned on the outputs of CNN. The outputs of the convolution network is used as a bias and for gating.

Image Captioning Related Works

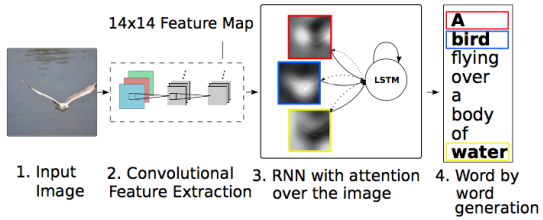
- Kiros, et al. "Multimodal Neural Language Models". (2014)
- Vinyals, et al. "Show and Tell: A Neural Image Caption Generator." (2015)



The second one is a paper also came out last year, they used the last hidden layer of a CNN for the initial hidden states of LSTM, and tries to predict the next words based on the previously n-1 words and the hidden states.

Image Captioning Related Works

- Kiros, et al. "Multimodal Neural Language Models". (2014)
- Vinyals, et al. "Show and Tell: A Neural Image Caption Generator." (2014)
- Kelvin, et al. "Show, Attend and Tell: Neural Image Caption Generation with Visual Attention". (2015)



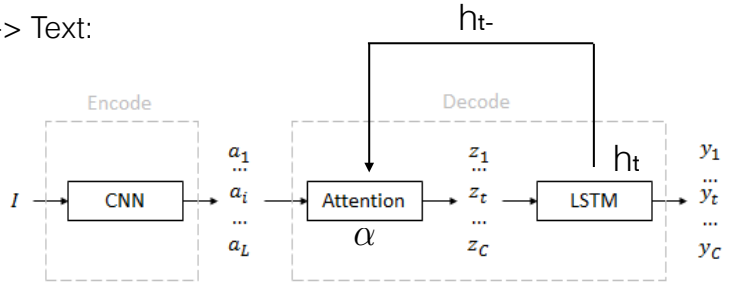
We can see that in both papers, the role of the image is like a static condition, either like a bias or initial hidden state.

But this paper I will talk about today convert the input image to an attentional region sequence and this sequence to generate words sequence.

Similar to the second one, it uses only the early feature map of a CNN model as the input to LSTM.

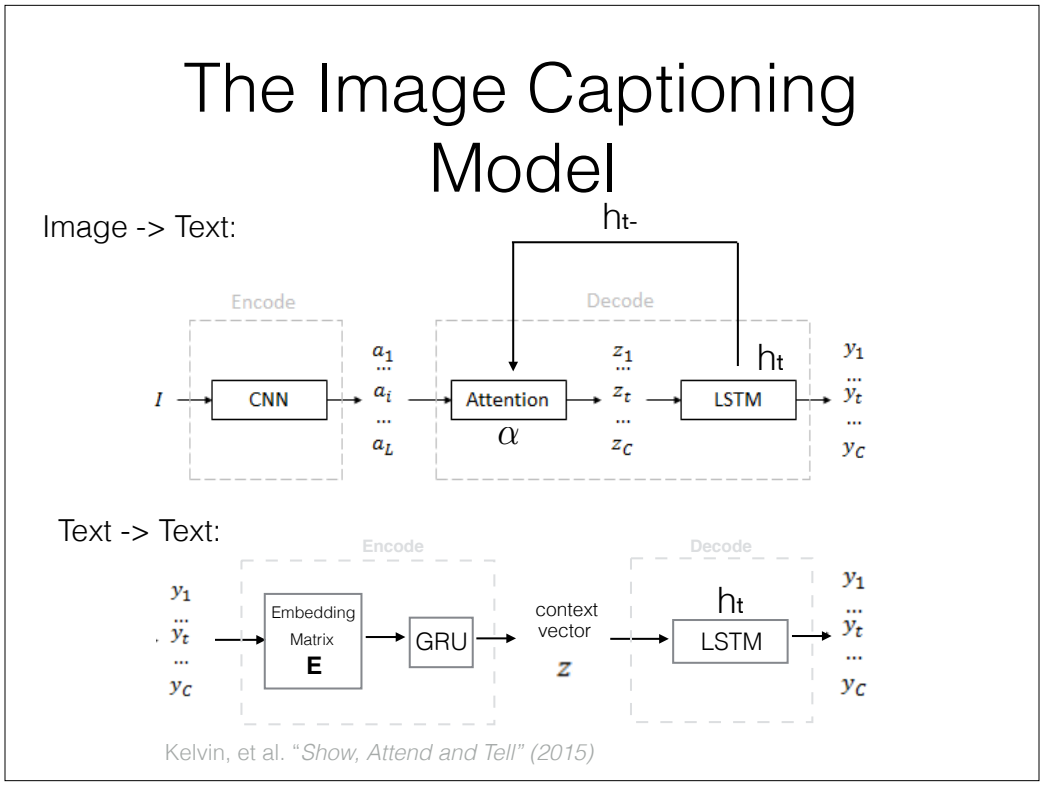
The Image Captioning Model

Image -> Text:



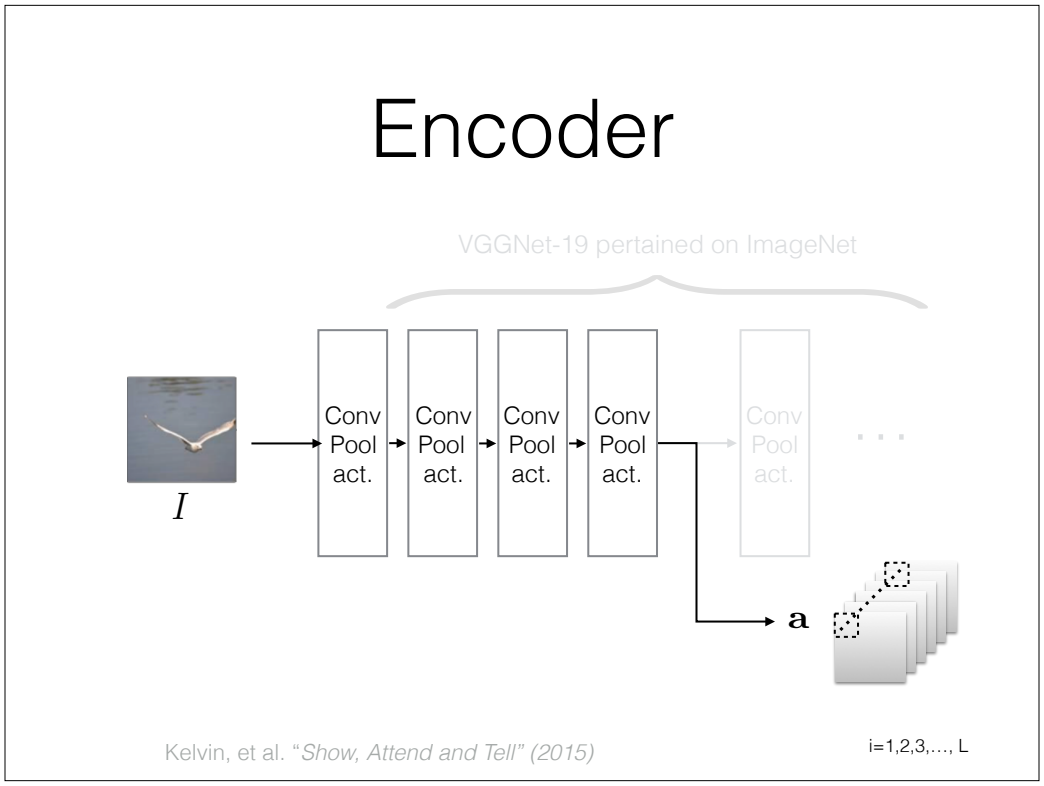
Kelvin, et al. "*Show, Attend and Tell" (2015)*

Same as the figure on previous slide, the overall structure of the model consists of an encoder and a decoder, and in the decoder there is an attentional interface which generates a image context for each word at time step t.



This is similar to a text to text language model, where the input sequence is embedded as a context embedding at each time step.

For the text-text part: see(credit: http://licstar.net/archives/328)

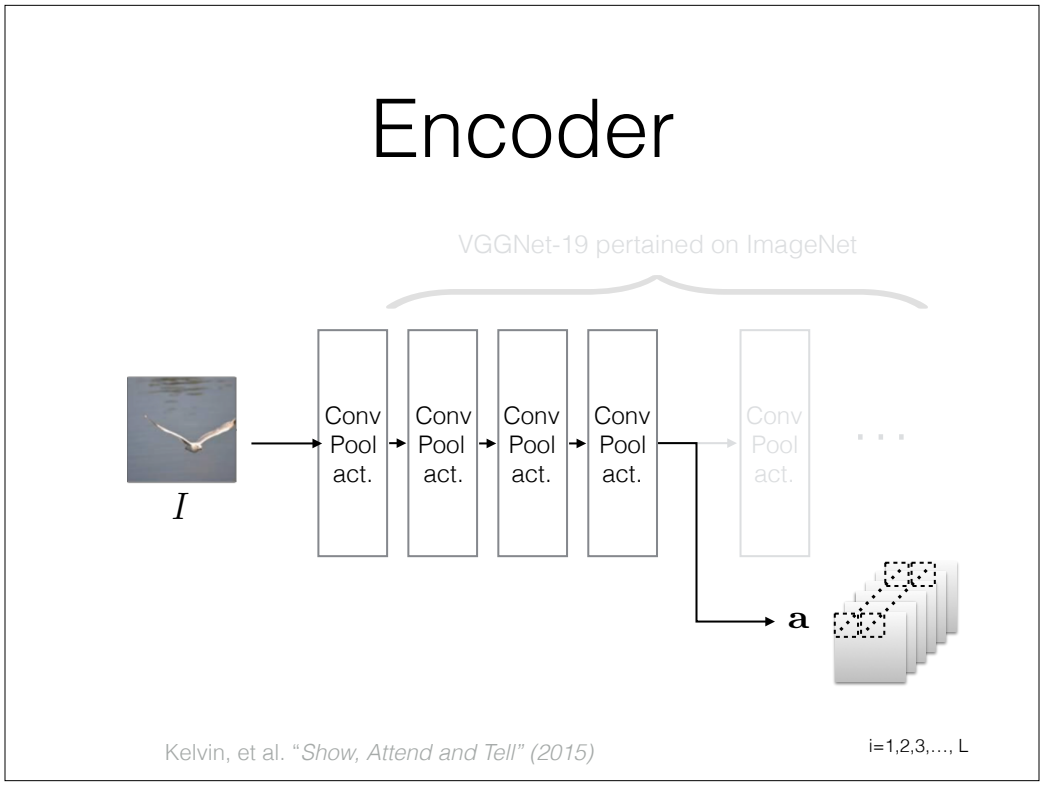


To elaborate on details:

The encoder is just a CNN, for example, VGG. They take only the lower Conv layers and output the features.

Note that on each feature, a single pixel corresponds to a region in the original image if you deconv it.

i is for indexing the feature in a feature map, a_i is the i-th feature

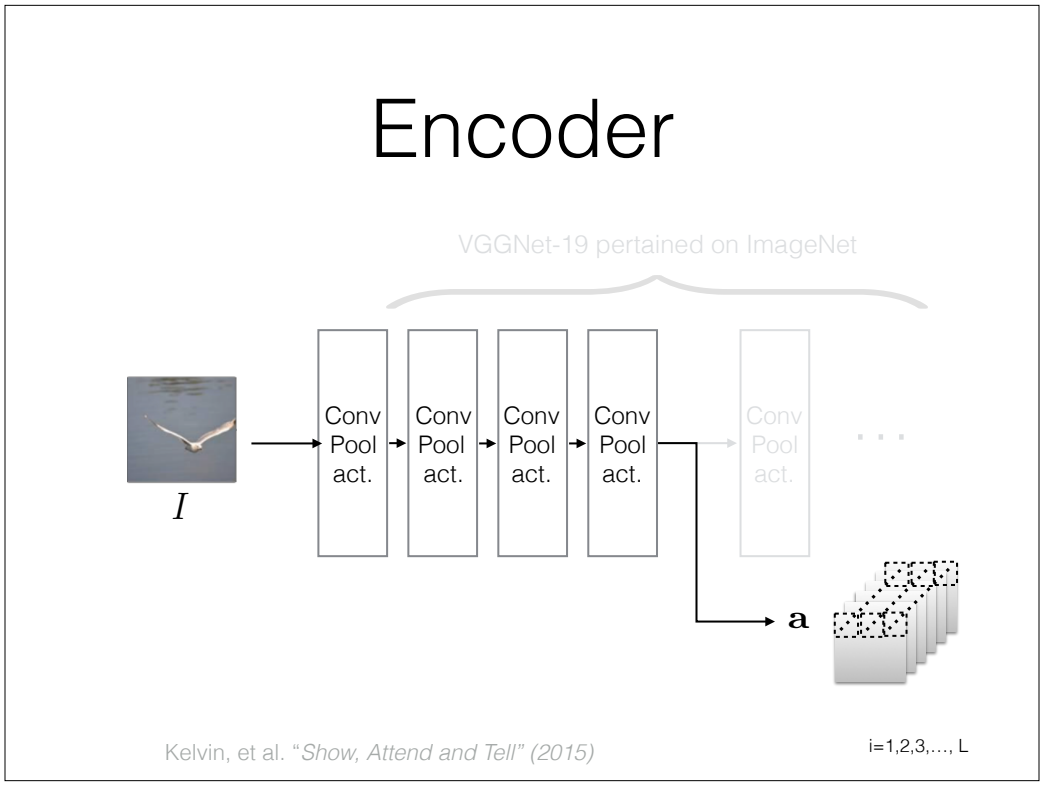


To elaborate on details:

The encoder is just a CNN, for example, VGG. They take only the lower Conv layers and output the features.

Note that on each feature, a single pixel corresponds to a region in the original image if you deconv it.

i is for indexing the feature in a feature map, a_i is the i-th feature



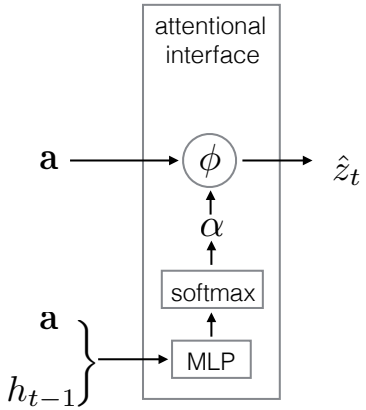
To elaborate on details:

The encoder is just a CNN, for example, VGG. They take only the lower Conv layers and output the features.

Note that on each feature, a single pixel corresponds to a region in the original image if you deconv it.

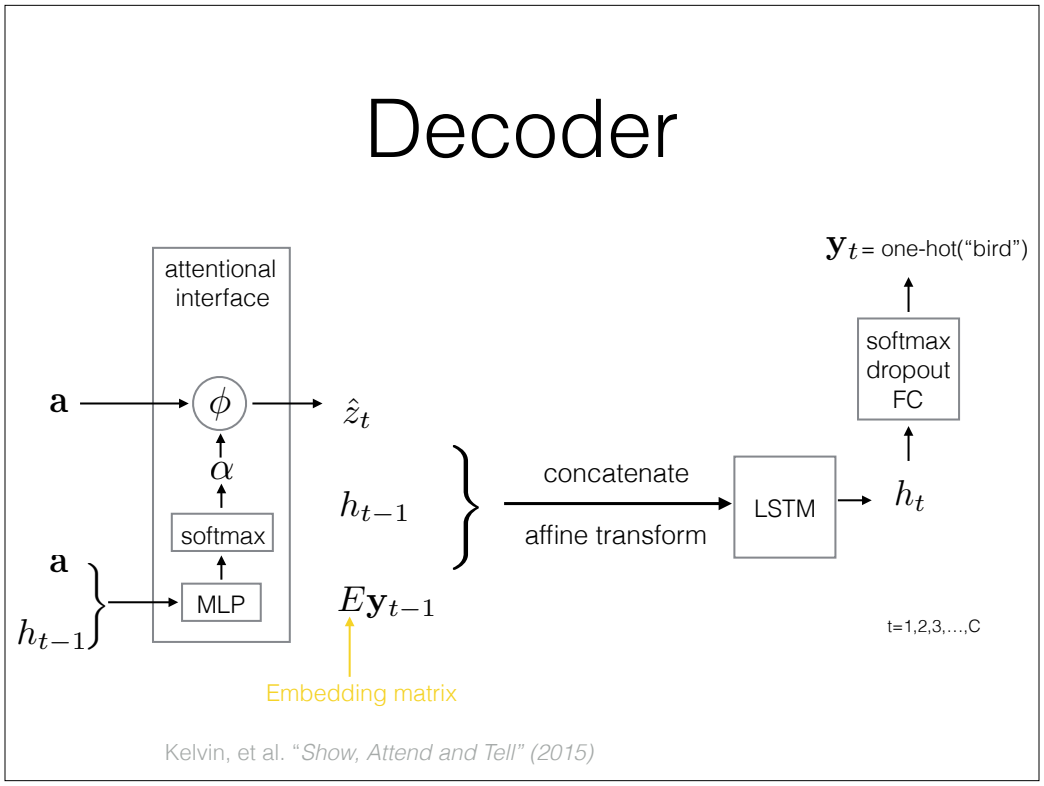
i is for indexing the vector

Decoder



Kelvin, et al. "*Show, Attend and Tell" (2015)*

The decoder put an attention function on those annotation vectors to select which a vector or some vectors to focus on, and generate a image context z at time t as compared to word context.



The image context is then fed into the LSTM along with the previous hidden state and previously generated word for generating the next words. This is part is the same as a regular RNN language model.

The interesting part is the attentional interface where z is calculated from the current image features and the previous hidden state. So what it implies is that the what to see in the current image not only depends on the image itself but also depends on what previously seen.

t is for indexing the words in a sentence. y_t is the t-th word

Attentional Interface

$$\hat{z}_t = \begin{cases} \sum_i s_{t,i}\mathbf{a}_i & \text{if hard attention} \\ \sum_i \alpha_{t,i}\mathbf{a}_i & \text{if soft attention} \end{cases}$$

$\mathbf{a}$ Image annotation vectors (Regions)

s_t Region selecting one-hot indicator

α_t Region selecting probability (Attention)

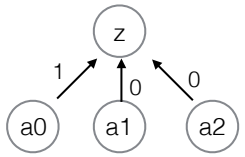
z_t Context vector (Selected Regions)

Kelvin, et al. "*Show, Attend and Tell" (2015)*

image: Region selection index (attention locations) ->s
region selecting probablity - >alpha
image annotations(Regions) -> a
resulting selected regions - > z

Hard Attention ϕ

$$\hat{z}_t = \sum_i s_{t,i}\mathbf{a}_i$$



- Not differentiable

Kelvin, et al. "*Show, Attend and Tell" (2015)*

in making a hard choice at every point, this hard attention function returns a sampled annotation among annotations at each time point.

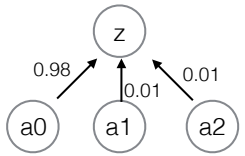
// ———————————————————————-
We need to maximize the probability of generating the correct sentence given the corresponding annotations.

To deal with the original intractable log Expectation, we instead maximize its lower bound which is the Expectation of the log.

Then lower bound can be computed through sampling the neural network.

Soft Attention ϕ

$$\hat{z}_t = \sum_i \alpha_{t,i} \mathbf{a}_i$$



- Differentiable

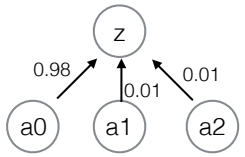
Kelvin, et al. "*Show, Attend and Tell" (2015)*

this soft attention function returns a weighted sum of annotations at each time point.

can be optimized by minimizing the negative log-likelihood

Soft Attention ϕ

$$\hat{z}_t = \sum_i \alpha_{t,i} \mathbf{a}_i$$



- Differentiable,
- Optimized through end-to-end Backprop on:

$$L_d = -log(p(\mathbf{y}|\mathbf{x}))$$

Kelvin, et al. "*Show, Attend and Tell" (2015)*

this soft attention function returns a weighted sum of annotations at each time point.

can be optimized by minimizing the negative log-likelihood

Soft Attention ϕ

$$t = 0 \quad \hat{z}_0 = \sum_i \alpha_{0,i} \mathbf{a}_i$$

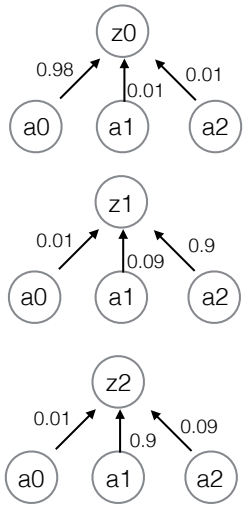
$$\mathbf{y_0} = \text{“Cat”}$$

$$t = 1 \quad \hat{z}_1 = \sum_i \alpha_{1,i} \mathbf{a}_i$$

$$\mathbf{y_1} = \text{“eats”}$$

$$t = 2 \quad \hat{z}_2 = \sum_i \alpha_{2,i} \mathbf{a}_i$$

$$\mathbf{y_2} = \text{“fish”}$$



Kelvin, et al. “*Show, Attend and Tell” (2015)*

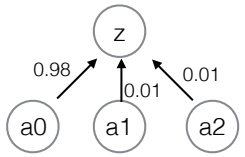
To improve more on the soft attention:

encourage the model to pay equal attention to every annotation of the image.

or force the attention move away from the previous attention

Soft Attention ϕ

$$\hat{z}_t = \sum_i \alpha_{t,i} \mathbf{a}_i$$



- Differentiable,
- Optimized through end-to-end Backprop on:

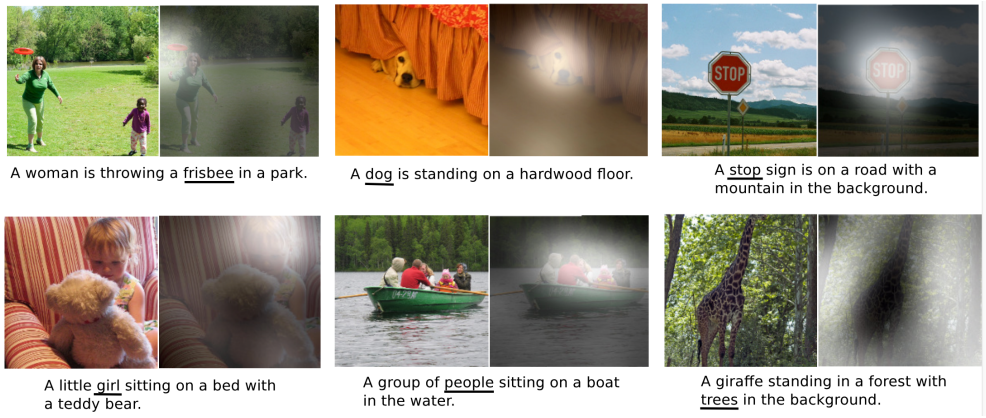
$$L_d = -log(p(\mathbf{y}|\mathbf{x})) + \lambda \sum_i^L (1 - \sum_t^C \alpha_{ti})^2$$

Kelvin, et al. "*Show, Attend and Tell" (2015)*

So they add another regularization term to do that

Now lets look at some results to get an idea of how this model perform. Here is the visualization of the learned attentional coefficient by deconvolution.

Results



Kelvin, et al. "*Show, Attend and Tell" (2015)*

Here is more result. The model is able to relate the same object in the pixel space to that in the text space.

Results

Dataset	Model	BLEU				METEOR
		BLEU-1	BLEU-2	BLEU-3	BLEU-4	
Flickr8k	Google NIC(Vinyals et al., 2014)†Σ	63	41	27	—	—
	Log Bilinear (Kiros et al., 2014a)°	65.6	42.4	27.7	17.7	17.31
	Soft-Attention	**67**	44.8	29.9	19.5	18.93
	Hard-Attention	**67**	**45.7**	**31.4**	**21.3**	**20.30**
Flickr30k	Google NIC†∘Σ	66.3	42.3	27.7	18.3	—
	Log Bilinear	60.0	38	25.4	17.1	16.88
	Soft-Attention	66.7	43.4	28.8	19.1	**18.49**
	Hard-Attention	**66.9**	**43.9**	**29.6**	**19.9**	18.46
COCO	CMU/MS Research (Chen & Zitnick, 2014)a	—	—	—	—	20.41
	MS Research (Fang et al., 2014)†a	—	—	—	—	20.71
	BRNN (Karpathy & Li, 2014)°	64.2	45.1	30.4	20.3	—
	Google NIC†∘Σ	66.6	46.1	32.9	24.6	—
	Log Bilinear°	70.8	48.9	34.4	24.3	20.03
	Soft-Attention	70.7	49.2	34.4	24.3	**23.90**
	Hard-Attention	**71.8**	**50.4**	**35.7**	**25.0**	23.04

Kelvin, et al. "*Show, Attend and Tell" (2015)*

Here is some quantitative result

We can see that the proposed attention model outperforms other models among all metrics. And the hard attention model seems giving higher score in most results.

Attention Related works

- Ren, et al. "*End-to-End Instance Segmentation and Counting with Recurrent **Attention***" (2016) Link
- Chan, et al. "*Listen, **Attend** and Spell" (2016) (Voice recognition)*
- *Mei, et al. "Listen, **Attend**, and Walk: Neural Mapping of Navigational Instructions to Action Sequences" (2016)*
- *Xu, et al. "Ask, **Attend** and Answer: Exploring Question-Guided Spatial Attention for Visual Question Answering" (2015)*
- *Mnih et al."Recurrent models of visual **attention**" (2014)*
- *Ba, et al. "Multiple object recognition with visual **attention**" (2014)*

Following Thor and Terrence's presentation on Chris Olah's blog.

I searched a little bit and found some other attention based works in recent years.

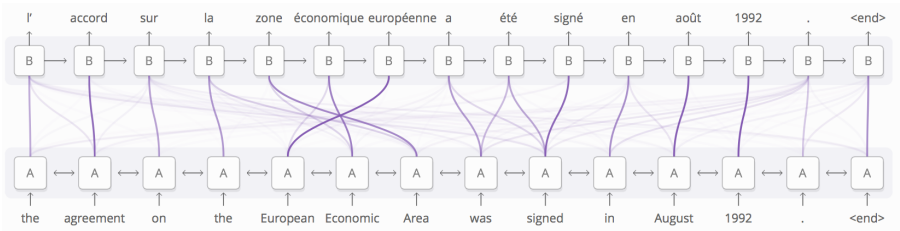
I came across this attention idea when attending the OpenPower Workshop where Professor Richard Zemel gives a brief review of his students' recent work on instance segmentation and counting. I think this attention idea is pretty cool and as it is applied in more advanced image tasks.

See paper

Attention Related works

- Image captioning (Xu, et al., 2015)
- Voice recognition (Chan, *et al.* 2015)
- Translation (Bahdanau, *et al.* 2014)
- Conversational modeling (Vinyals & Le, 2015)

Recent development on applying the attentional interfaces includes the following as summarized by Chris Olah's blog on distill.



- Chris Olah's blog summarized:

  "attentional interfaces can be used whenever one wants to interface with a neural network that has a repeating structure in its output."

He took a further step and summarized that: …

Repeating structure is obvious when talking about RNNs and language models, the output of RNNs is structured. This is quite intuitive when understanding a sentence.

So attentional interface and Recurrent model come hand in hand.

//———————————-

The structured output can then be generated by focusing on part of a subset of the information they're given each time.

This is quite intuitive when understanding a sentence since the useful information is not evenly distributed on every parts of the sentence and for generating a certain output word, only a part of the sentence is mostly useful. This is same as captioning an image.

Discussion

- Interpretation of the visual attention
  - hierarchical(Conv) and iterative(LSTM)
- contribution of the paper
  - extract attentional region sequence from an image
  - visualize what the model sees

I think this idea of applying recurrent model in understating an image also makes sense. As human understands an image not only in hierarchical way (Convolution), but also in an iterative way by looking part by part at an image and tries to relate between parts (LSTM). When asking about a certain attribute of an image, like where is the dog in the image, human needs to look around and focus on some local parts of the image (Attention).

———————————————-

provides a way of visualizing where the model is looking at.

The main contribution of the paper is the way they extract attentional region sequence from a static image. which essentially convert the problem to a sequence to sequence problem.

END

//———————————————————-

- Thank you