Boulder

Multimodal Learning

YouTube Playlist

Maziar Raissi

Assistant Professor

Department of Applied Mathematics

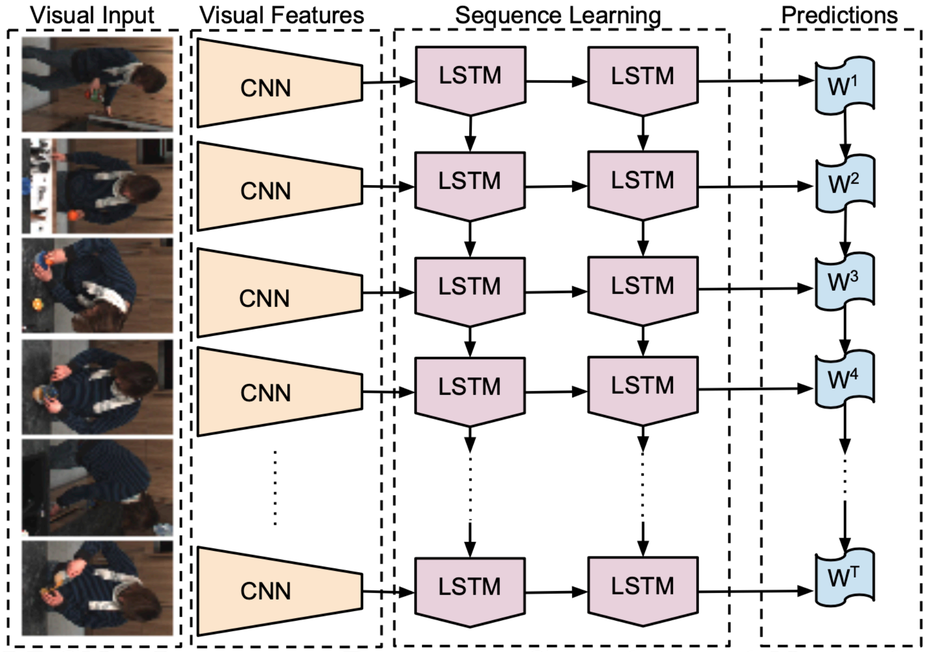
University of Colorado Boulder

maziar.raissi@colorado.edu

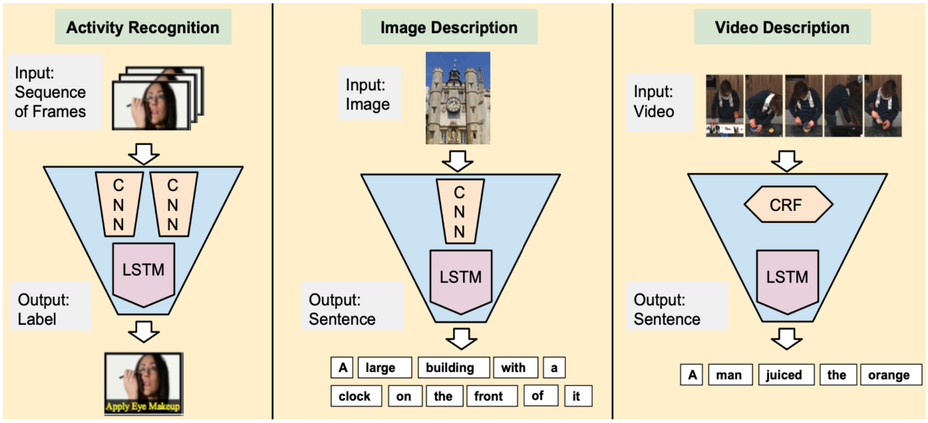
Long-term Recurrent Convolutional Networks for Visual Recognition and Description

Boulder

YouTube Playlist



LRCNs: Long-term Recurrent Convolutional Networks



– video activity recognition
– image caption generation
– video description tasks

$\langle x_1, \ldots, x_T \rangle \mapsto y$

$x \mapsto \langle y_1, \ldots, y_T \rangle$

$\langle x_1, \ldots, x_T \rangle \mapsto \langle y_1, \ldots, y_{T'} \rangle$

$v_t \rightarrow$ visual input

$\phi_V(v_t) \in \mathbb{R}^d \rightarrow$ feature transform

$\implies \langle \phi_1, \ldots, \phi_T \rangle$

$\Pr(y_t) = \text{softmax}(W_{zc} z_t + b_c)$

Recurrent Neural Networks

$h_t = g(W_{xh} x_t + W_{hh} h_{t-1} + b_h)$

$z_t = g(W_{hz} h_t + b_z)$

Long Short-Term Memory Networks (LSTMS)

$i_t = \sigma(W_{xi} x_t + W_{hi} h_{t-1} + b_i)$

$f_t = \sigma(W_{xf} x_t + W_{hf} h_{t-1} + b_f)$

$o_t = \sigma(W_{xo} x_t + W_{ho} h_{t-1} + b_o)$

$g_t = \phi(W_{xc} x_t + W_{hc} h_{t-1} + b_c)$

$c_t = f_t \odot c_{t-1} + i_t \odot g_t$

$h_t = o_t \odot \phi(c_t)$

UCF-101 (Activity Recognition)	Single Input Type		Weighted Average	
Model	RGB	Flow	1/2, 1/2	1/3, 2/3
Single frame (split-1)	69.00	72.20	75.71	79.04
LRCN-fc$_6$ (split-1)	71.12	76.95	81.97	82.92
LRCN-fc$_7$ (split-1)	70.68	69.36	79.01	80.51
Single frame (all splits)	67.70	72.19	75.87	78.84
LRCN-fc$_6$ (all splits)	**68.19**	**77.46**	80.62	**82.66**

Retrieval	R@1	R@5	R@10	Med*r*
Caption to Image (Flickr30k)				
DeViSE [8]	6.7	21.9	32.7	25
SDT-RNN [36]	8.9	29.8	41.1	16
DeFrag [15]	10.3	31.4	44.5	13
m-RNN [25]	12.6	31.2	41.5	16
ConvNet [18]	11.8	34.0	46.3	13
LRCN$_{2f}$ (ours)	**17.5**	**40.3**	**50.8**	**9**
Image to Caption (Flickr30k)				
DeViSE [8]	4.5	18.1	29.2	26
SDT-RNN [36]	9.6	29.8	41.1	16
DeFrag [15]	16.4	40.2	54.7	8
m-RNN [25]	18.4	40.2	50.9	10
ConvNet [18]	14.8	39.2	50.9	10
LRCN$_{2f}$ (ours)	**23.6**	**46.6**	**58.3**	**7**
Caption to Image (COCO)				
LRCN$_{2f}$ (ours)	29.0	61.6	74.8	3
Image to Caption (COCO)				
LRCN$_{2f}$ (ours)	39.1	69.0	80.9	2

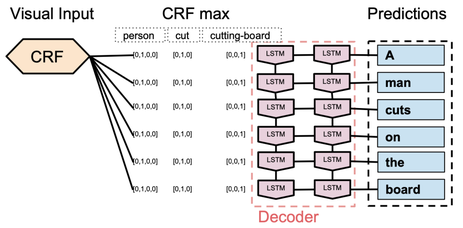
Image description	Flickr30k [28]			
BLEU Scores	B-1	B-2	B-3	B-4
m-RNN [25]	54.79	23.92	19.52	-
1NN fc$_8$ base (ours)	37.34	18.66	9.39	4.88
1NN fc$_7$ base (ours)	38.81	20.16	10.37	5.54
LRCN (ours)	**58.72**	**39.06**	**25.12**	**16.46**
	COCO 2014 [24]			
	B-1	B-2	B-3	B-4
1NN fc$_7$ base (c5)	46.23	26.39	15.07	08.73
LRCN (ours)	**66.86**	**48.92**	**34.89**	**24.92**

Median rank, Medr, of the first retrieved ground truth image or caption and Recall@K, the number of images or captions for which a correct caption or image is retrieved within the top K results.

Limited availability of video description datasets!

TACoS multilevel (Video Description)

Architecture	Input	BLEU
SMT [30]	CRF max	24.9
SMT [29]	CRF prob	26.9
(a) LSTM Encoder-Decoder (ours)	CRF max	25.3
(b) LSTM Decoder (ours)	CRF max	27.4
(c) LSTM Decoder (ours)	CRF prob	**28.8**



Donahue, Jeffrey, et al. "Long-term recurrent convolutional networks for visual recognition and description." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015.

Show and Tell: A Neural Image Caption Generator

YouTube Video

Helping visually impaired people better understand the content of images on the web.

– Pascal dataset
– Flickr30k dataset
– SBU dataset
– COCO dataset

$I \rightarrow$ input image
$S = \{S_1, S_2, \ldots\} \rightarrow$ target sequence of words
$p(S|I) \rightarrow$ likelihood
S_t comes from a given dictionary
NIC: Neural Image Caption

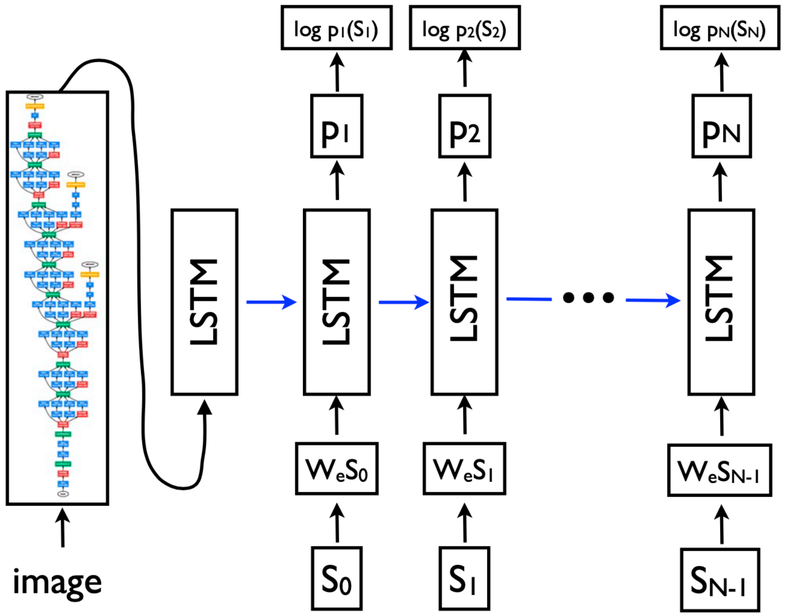
$$\theta^\star = \arg\max_\theta \sum_{(I,S)} \log p(S|I;\theta)$$

$$\log p(S|I) = \sum_{t=0}^{N} \log \underbrace{p(S_t|I, S_0, \ldots, S_{t-1})}_{\text{model with an RNN}}$$

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

CNN input (image & words)

$$\begin{aligned}
i_t &= \sigma(W_{ix}x_t + W_{im}m_{t-1}) \\
f_t &= \sigma(W_{fx}x_t + W_{fm}m_{t-1}) \\
o_t &= \sigma(W_{ox}x_t + W_{om}m_{t-1}) \\
c_t &= f_t \odot c_{t-1} + i_t \odot h(W_{cx}x_t + W_{cm}m_{t-1}) \\
m_t &= o_t \odot c_t \\
p_{t+1} &= \text{Softmax}(m_t)
\end{aligned}$$



$$\begin{aligned}
x_{-1} &= \text{CNN}(I) \\
x_t &= W_e S_t, \quad t \in \{0 \ldots N-1\} \\
p_{t+1} &= \text{LSTM}(x_t), \quad t \in \{0 \ldots N-1\}
\end{aligned}$$

$S_0 \rightarrow$ start symbol
$S_N \rightarrow$ end symbol
$S_t \rightarrow$ one-hot-vector
$W_e \rightarrow$ word embeddings

$$L(I,S) = -\sum_{t=1}^{N} \log p_t(S_t)$$

BeamSearch: Iteratively consider the set of the k best sentences up to time t as candidates to generate sentences of size $t+1$, and keep only the resulting best k of them. (beam size: k = 20)

Dataset name	size		
	train	valid.	test
Pascal VOC 2008 [6]	-	-	1000
Flickr8k [26]	6000	1000	1000
Flickr30k [33]	28000	1000	1000
MSCOCO [20]	82783	40504	40775
SBU [24]	1M	-	-

Analysis of Embeddings

Nearest neighbors of a few example words

Word	Neighbors
car	van, cab, suv, vehicule, jeep
boy	toddler, gentleman, daughter, son
street	road, streets, highway, freeway
horse	pony, donkey, pig, goat, mule
computer	computers, pc, crt, chip, compute

Approach	PASCAL (xfer)	Flickr 30k	Flickr 8k	SBU
NIC	**59**	**66**	**63**	**28**
Human	69	68	70	

novel sentence not present in the training set

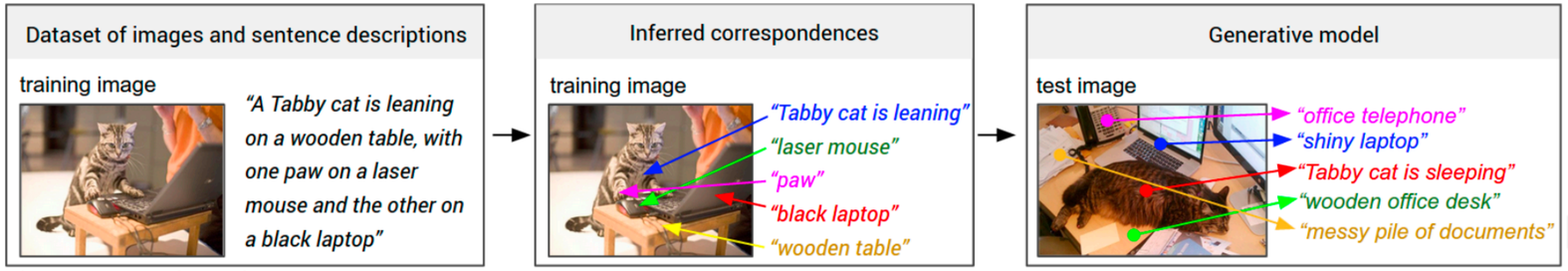
A man throwing a frisbee in a park.
A man holding a frisbee in his hand.
A man standing in the grass with a frisbee.

Vinyals, Oriol, et al. "Show and tell: A neural image caption generator." Proceedings of the IEEE conference on computer vision and pattern recognition. 2015.

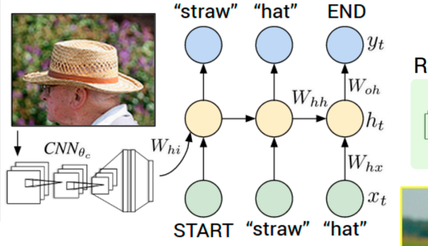
Deep Visual-Semantic Alignments for Generating Image Descriptions

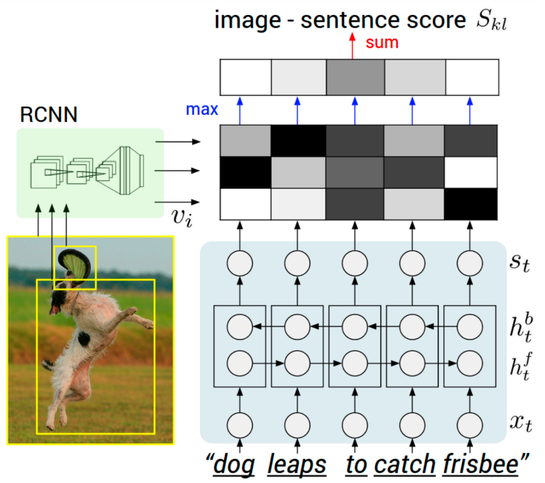
Boulder

YouTube Playlist



Multimodal RNN





Representing images

R-CNN $\rightarrow$ top 19 detected locations + whole image

$I_b \rightarrow$ pixels inside each bounding box

$v = W_m \text{CNN}_{\theta_c}(I_b) + b_m$

$\text{CNN}_{\theta_c}(I_b) \rightarrow$ 4096-dimentional activations of the fully-connected layer immediately before the classifier

$W_m \in \mathbb{R}^{h \times 4096}$

$1000 \leq h \leq 1600 \rightarrow$ dimension of the multimodal space

$\theta_c \rightarrow$ approximately 60 million parameters

$\{v_i \in \mathbb{R}^h : i = 1, 2, \ldots, 20\} \rightarrow$ representation of image

Representing sentences

Bidirectional RNN

Sequence of N words

$x_t = W_w \mathbb{I}_t$

$e_t = f(W_e x_t + b_e)$

$h_t^f = f(e_t + W_f h_{t-1}^f + b_f)$

$h_t^b = f(e_t + W_b h_{t+1}^b + b_b)$

$s_t = f(W_d(h_t^f + h_t^b) + b_d).$

Training data: inferred correspondences

$$b_v = W_{hi}[CNN_{\theta_c}(I)]$$
$$h_t = f(W_{hx}x_t + W_{hh}h_{t-1} + b_h + \mathbb{1}(t=1) \odot b_v)$$
$$y_t = softmax(W_{oh}h_t + b_o).$$

Alignment objective

$v_i^T s_t \rightarrow$ a measure of similarity btw i-th region & t-th word

$S_{kl} = \sum_{t \in g_l} max_{i \in g_k} v_i^T s_t$

score btw image k & sentence l

$g_k \rightarrow$ set of image fragments

$g_l \rightarrow$ set of sentence fragments

every word s_t aligns to the single best image region

Loss

$$\mathcal{C}(\theta) = \sum_k \Big[\underbrace{\sum_l max(0, S_{kl} - S_{kk} + 1)}_{\text{rank images}} + \underbrace{\sum_l max(0, S_{lk} - S_{kk} + 1)}_{\text{rank sentences}} \Big]$$

$k = l$ denotes a corresponding image and sentence pair

encourages aligned image-sentences pairs to have a higher score than misaligned pairs, by a margin!

Decoding text segment alignments to images

Sentence with N words

Image with M bounding boxes

$a_t \in \{1, \ldots, M\}, t = 1, \ldots, N$

latent alignment variables

Markov Random Field (MRF)

$$E(a) = \sum_{t=1}^{N} \psi_t^U(a_t) + \sum_{t=1}^{N-1} \psi_t^B(a_t, a_{t+1})$$

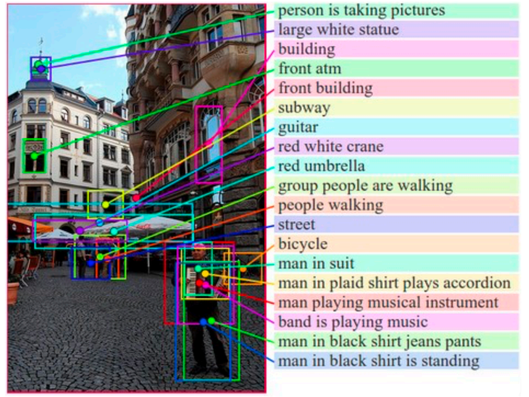
$\psi_t^U(a_t = i) = v_i^T s_t$

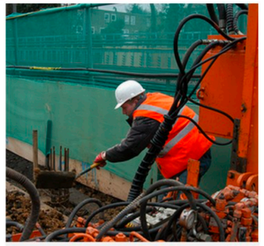
$\psi_t^B(a_t, a_{t+1}) = \beta \mathbb{1}[a_t = a_{t+1}]$

$\min_a E(a) \rightarrow$ using dynamic programing

$\beta = 0 \rightarrow$ single word alignment

$\beta \gg 0 \rightarrow$ aligning the entire sentence to a single region





construction worker in orange safety vest is working on road.

Karpathy, Andrej, and Li Fei-Fei. "Deep visual-semantic alignments for generating image descriptions." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015.

Show, Attend and Tell: Neural Image Caption Generation with Visual Attention

Boulder

YouTube Playlist

Encoder

$y = \{y_1, y_2, \ldots, y_C\} \rightarrow$ caption

$y_i \in \mathbb{R}^K \rightarrow$ 1-of-K encoded word

$K \rightarrow$ size of the vocabulary

$C \rightarrow$ length of the caption

$a = \{a_1, a_2, \ldots, a_L\} \rightarrow$ set of feature vectors (annotation vectors)

$a_i \in \mathbb{R}^D$ extracted by a CNN

Example: $14 \times 14 \times 512$ feature map

$L = 196$ & $D = 512$

Stochastic (Hard) Attention

$$\hat{z} = \sum_{i=1}^{L} s_{t,i} a_i$$

$s_t \rightarrow$ attention location (random variable)

$s_{t,i} = 1$ iff the i-th location (out of L) is used to extract visual features

$p(s_{t,i} = 1 | s_1, \ldots, s_{t-1}, a) = \alpha_{t,i}$

$\alpha_t = \text{softmax}(e_t)$

$e_t = f_{\text{att}}(a, h_{t-1}) \rightarrow$ MLP

$h_t \rightarrow$ hidden state of a decoder (LSTM)

$$\log p(y|a) = \log \sum_s p(s|a) p(y|s,a)$$

$$\geq \sum_s p(s|a) \log p(y|s,a) = L_s$$

$$\frac{\partial L_s}{\partial W} = \sum_s p(s|a) \left[\frac{\partial \log p(y|s,a)}{\partial W} + \log p(y|s,a) \frac{\partial \log p(s|a)}{\partial W} \right]$$

Monte Carlo

$$\frac{\partial L_s}{\partial W} \approx \frac{1}{N} \sum_{n=1}^{N} \left[\frac{\partial \log p(y|\tilde{s}^n, a)}{\partial W} + \log p(y|\tilde{s}^n, a) \frac{\partial \log p(\tilde{s}^n|a)}{\partial W} \right]$$

$\tilde{s}^n = (\tilde{s}_1^n, \tilde{s}_2^n, \ldots)$ & $\tilde{s}_t^n \sim \text{Multinoulli}_L(\{\alpha_i^n\})$

Variance Reduction

$b_k = 0.9 b_{k-1} + 0.1 \log p(y|\tilde{s}_k, a) \rightarrow$ moving average baseline (after seeing k-th mini-batch)

$H[s] \rightarrow$ entropy of the multinoulli distribution

$$\frac{\partial L_s}{\partial W} \approx \frac{1}{N} \sum_{n=1}^{N} \left[\frac{\partial \log p(y|\tilde{s}^n, a)}{\partial W} + \lambda_r (\log p(y|\tilde{s}^n, a) - b) \frac{\partial \log p(\tilde{s}^n|a)}{\partial W} + \lambda_e \frac{\partial H[\tilde{s}^n]}{\partial W} \right]$$

This formulation is equivalent to the REINFORCE learning rule!

Deterministic (Soft) Attention

$$\mathbb{E}_{p(s_t|a)}[\hat{z}_t] = \sum_{i=1}^{L} \alpha_{t,i} a_i \qquad \hat{z}_t = \sum_{i=1}^{N} \alpha_{t,i} a_i \qquad \hat{z}_t = \beta_t \sum_{i=1}^{N} \alpha_{t,i} a_i$$

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

Decoder

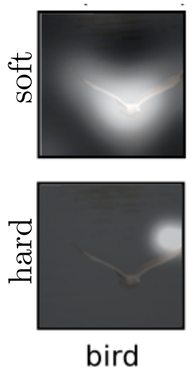
$i_t = \sigma(W_i E y_{t-1} + U_i h_{t-1} + Z_i \hat{z}_t + b_i) \rightarrow$ input gate

$f_t = \sigma(W_f E y_{t-1} + U_f h_{t-1} + Z_f \hat{z}_t + b_f) \rightarrow$ forget gate

$o_t = \sigma(W_o E y_{t-1} + U_o h_{t-1} + Z_o \hat{z}_t + b_o) \rightarrow$ output gate

$c_t = f_t c_{t-1} + i_t \tanh(W_c E y_{t-1} + U_c h_{t-1} + Z_c \hat{z}_t + b_c) \rightarrow$ memory

$h_t = o_t \tanh(c_t) \rightarrow$ hidden state



A stop sign is on a road with a mountain in the background.

A giraffe standing in a forest with trees in the background.

$$\sum_{i=1}^{L} \alpha_{t,i} = 1, \sum_{t=1}^{C} \alpha_{t,i} \approx \tau \geq L/D$$

$$L_d = -\log p(y|a) + \lambda \sum_i (\tau - \sum_{t=1}^{C} \alpha_{t,i})^2$$

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

$E \rightarrow$ embedding matrix

MLP

$$c_0 = f_{\text{init},c}\left(\frac{1}{L} \sum_{i=1}^{L} a_i\right)$$

$$h_0 = f_{\text{init},h}\left(\frac{1}{L} \sum_{i=1}^{L} a_i\right)$$

Xu, Kelvin, et al. "Show, attend and tell: Neural image caption generation with visual attention." *International conference on machine learning*. 2015.

Layer Normalization

YouTube Video

Batch Normalization

$a^\ell \rightarrow$ vector representation of the summed inputs to the neurons in layer ℓ

$a_i^\ell = (w_i^\ell)^T h_\ell$

$W^\ell \rightarrow$ weight matrix

$w_i^\ell \rightarrow$ incoming weights to the i-th hidden unit

$h^\ell \rightarrow$ bottom-up inputs

$h_i^{\ell+1} = f(a_i^\ell + b_i^\ell)$ (b_i^ℓ: bias parameter)

$\mu_i^\ell = \mathbb{E}_{x \sim p(x)}[a_i^\ell]$

$\sigma_i^\ell = \sqrt{\mathbb{E}_{x \sim p(x)}[(a_i^\ell - \mu_i^\ell)^2]}$

$\overline{a}_i^\ell = \frac{g_i^\ell}{\sigma_i^\ell}(a_i^\ell - \mu_i^\ell) \rightarrow$ normalized summed inputs

$g_i^\ell \rightarrow$ gain parameter

Layer Normalization

$$\mu^\ell = \frac{1}{H}\sum_{i=1}^{H} a_i^\ell \qquad \sigma^\ell = \sqrt{\frac{1}{H}\sum_{i=1}^{H}(a_i^\ell - \mu_i^\ell)^2}$$

$H \rightarrow$ number of hidden units in a layer

different training cases have different normalization terms

no constraints on the size of a mini-batch

batch size could be one

Layer Normalized RNNs

different sequence lengths for different training cases

$x^t \rightarrow$ current input

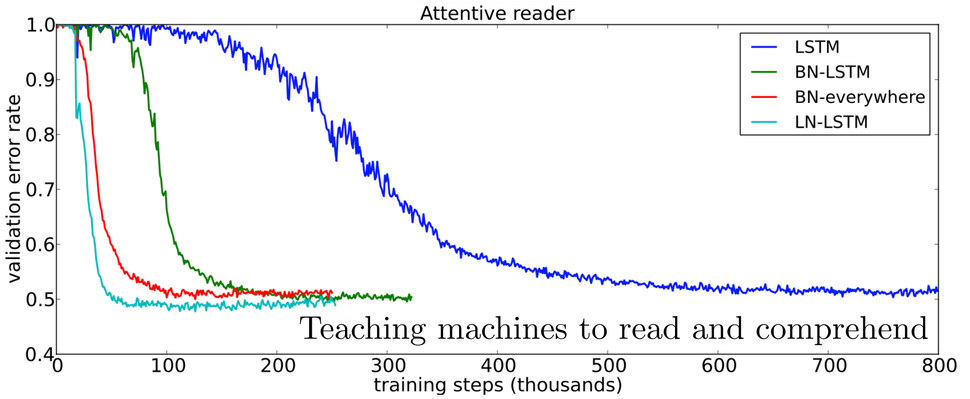
$h^{t-1} \rightarrow$ previous vector of hidden states

$a^t = W_{hh} h^{t-1} + W_{xh} x^t$

(a^t: summed inputs in the recurrent layer)

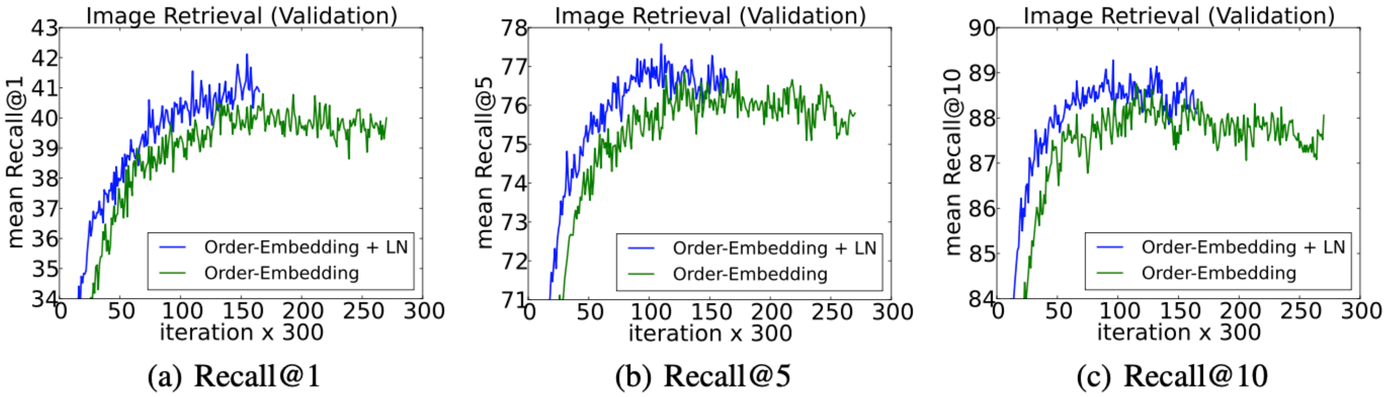
$h_i^t = f(\frac{g_i}{\sigma^t}(a_i^t - \mu^t) + b_i)$

$$\mu^t = \frac{1}{H}\sum_{i=1}^{H} a_i^t \qquad \sigma^t = \sqrt{\frac{1}{H}\sum_{i=1}^{H}(a_i^t - \mu^t)^2}$$



	Weight matrix re-scaling	Weight matrix re-centering	Weight vector re-scaling	Dataset re-scaling	Dataset re-centering	Single training case re-scaling
Batch norm	Invariant	No	Invariant	Invariant	Invariant	No
Weight norm	Invariant	No	Invariant	No	No	No
Layer norm	Invariant	Invariant	No	Invariant	No	Invariant

Experiments with layer normalization on 6 tasks, with a focus on recurrent neural networks: image-sentence ranking, question-answering, contextual language modeling, generative modeling, handwriting sequence generation and MNIST classification.



(a) Recall@1 (b) Recall@5 (c) Recall@10

Images and sentences from the Microsoft COCO dataset are embedded into a common vector space.

Ba, Jimmy Lei, Jamie Ryan Kiros, and Geoffrey E. Hinton. "Layer normalization." *arXiv preprint arXiv:1607.06450* (2016).

Bottom-Up and Top-Down Attention for Image Captioning and Visual Question Answering

YouTube Video

$I \rightarrow$ image

$V = \{v_1, \ldots, v_k\} \rightarrow$ a possibly variably-sized set of k image features

$v_i \in \mathbb{R}^D \rightarrow$ encodes a salient region of the image

Bottom-Up Attention Model

Faster R-CNN: stage 1 $\rightarrow$ region proposal network (RPN)
stage 2 $\rightarrow$ region of interest (RoI) pooling

$v_i \rightarrow$ mean-pooled convolutional feature for region i

$D = 2048$

– train on Visual Genome data

– predict attribute classes in addition to object classes

Captioning Model

"soft" top-down attention

$h_t = \text{LSTM}(x_t, h_{t-1})$

Top-Down Attention LSTM

$x_t^1 = [h_{t-1}^2, \overline{v}, W_e \Pi_t] \rightarrow$ input vector to the attention LSTM

$h_{t-1}^2 \rightarrow$ previous output of the language LSTM

$\overline{v} = \frac{1}{k} \sum_i v_i \rightarrow$ mean-pooled image features

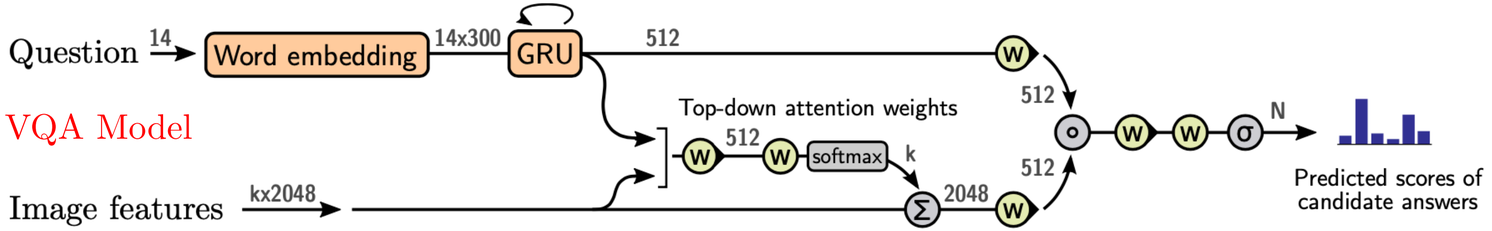
$W_e \Pi_t \rightarrow$ embedding of the previous generated word

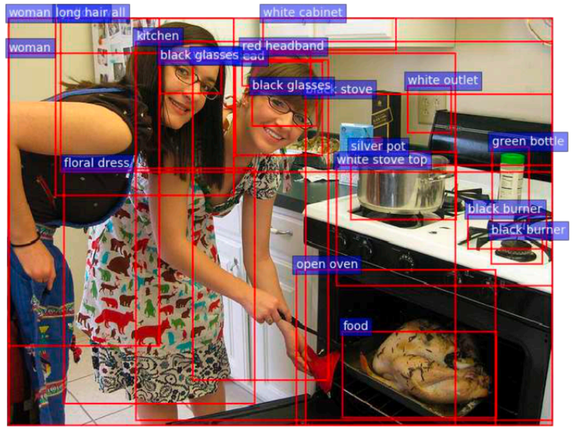
$W_e \in \mathbb{R}^{E \times |\Sigma|} \rightarrow$ word embedding matrix

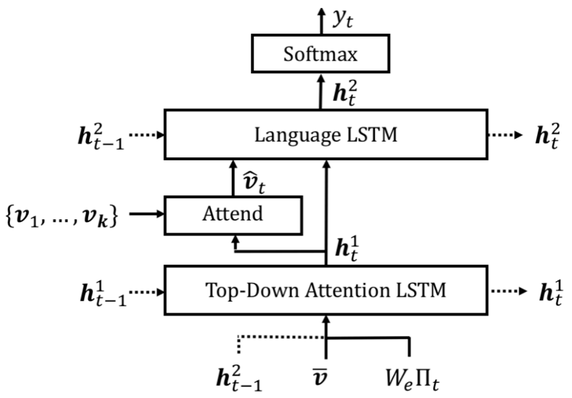
$\Sigma \rightarrow$ vocabulary

$\Pi_t \rightarrow$ one-hot-encoding of the input word at step t

$h_t^1 = \text{LSTM}(x_t^1, h_{t-1}^1)$







Concatenation | Weighted sum over image locations | Element-wise product

$$\hat{v}_t = \sum_{i=1}^{k} \alpha_{i,t} v_i$$ ($\alpha_{i,t}$: normalized attention weights)

$$\alpha_t = \text{softmax}(a_t)$$

$$a_{i,t} = w_a^T \tanh(W_{va} v_i + W_{ha} h_t^1)$$

Language LSTM

$$x_t^2 = [\hat{v}_t, h_t^1], h_t^2 = \text{LSTM}(x_t^2, h_{t-1}^2)$$

$$p(y_t | y_1, \ldots, y_{t-1}) = \text{softmax}(W_p h_t^2 + b_p)$$

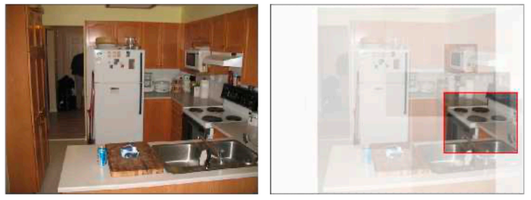
$$p(y_1, \ldots, y_T) = \prod_{t-1}^{T} p(y_t | y_1, \ldots, y_{t-1})$$

Objective

$(y_1^*, \ldots, y_T^*) \rightarrow$ target ground truth sequence

$$L_{XE}(\theta) = -\sum_{t=1}^{T} \log p_\theta(y_t^* | y_1^*, \ldots, y_{t-1}^*)$$ cross-entropy loss

Two men playing frisbee in a dark field



Question: What room are they in? Answer: kitchen

Anderson, Peter, et al. "Bottom-up and top-down attention for image captioning and visual question answering." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018.

Zero-Shot Text-to-Image Generation

YouTube Video

DALL-E: A 12-billion parameter autoregressive transformer trained on 250 million image-text pairs from the internet

Stage 1: dVAE: discrete variational auto-encoder

$q_\phi : x \in \mathbb{R}^{256\times256\times3} \mapsto z \in \{1,\dots,8192\}^{32\times32}$

$p_\theta : z \in \{1,\dots,8192\}^{32\times32} \mapsto x \in \mathbb{R}^{256\times256\times3}$

Gumbel-softmax relaxation

This reduces the context size of the transformer by a factor of 192 without a large degradation in visual quality!

(a) a tapir made of accordion. a tapir with the texture of an accordion.

(b) an illustration of a baby hedgehog in a christmas sweater walking a dog

(c) a neon sign that reads "backprop". a neon sign that reads "backprop". backprop neon sign

(d) the exact same cat on the top as a sketch on the bottom

Stage 2: Train an autoregressive transformer $p_\psi(y, z)$ to model the joint distribution over the text and image tokens.

Ramesh, Aditya, et al. "Zero-Shot Text-to-Image Generation." *arXiv preprint arXiv:2102.12092* (2021).

Boulder

Questions?

YouTube Playlist