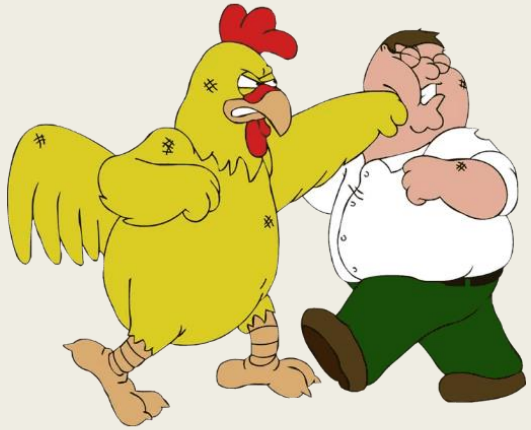
NEURAL IMAGE CAPTIONING

Davide Montagno B. (535910)



A chicken is fighting with human

[O. Vinyals, A. Toshev, S. Bengio and D. Erhan, "Show and Tell: Lessons Learned from the 2015 MSCOCO Image Captioning Challenge," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 4, pp. 652-663, 1 April 2017. - Show, Attend and Tell: Neural Image Caption Generation with Visual Attention, Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhutdinov, Richard Zemel, Yoshua Bengio]

PROBLEM & DATA

- Idea inspired by combination of Machine Translation and Object Detection [Cho et al., 2014; Bahdanau et al., 2014;Sutskever et al., 2014 , .Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate,"arXiv:1409.0473, 2014.]
- Attention **based model** **[extension Bahdanau et al. (2014); Mnih et al. (2014); Baet al. (2014)]**
  - *Soft Attention*
  - *Hard Attention*
  - *Doubling Soft Attention*
- **Goal**: Learn to attend

Flickr8k – Images with captions

Flickr30k – Images with captions

MS COCO (Microsoft) – Images with captions

Alignment datasets

- References
- Tokenization

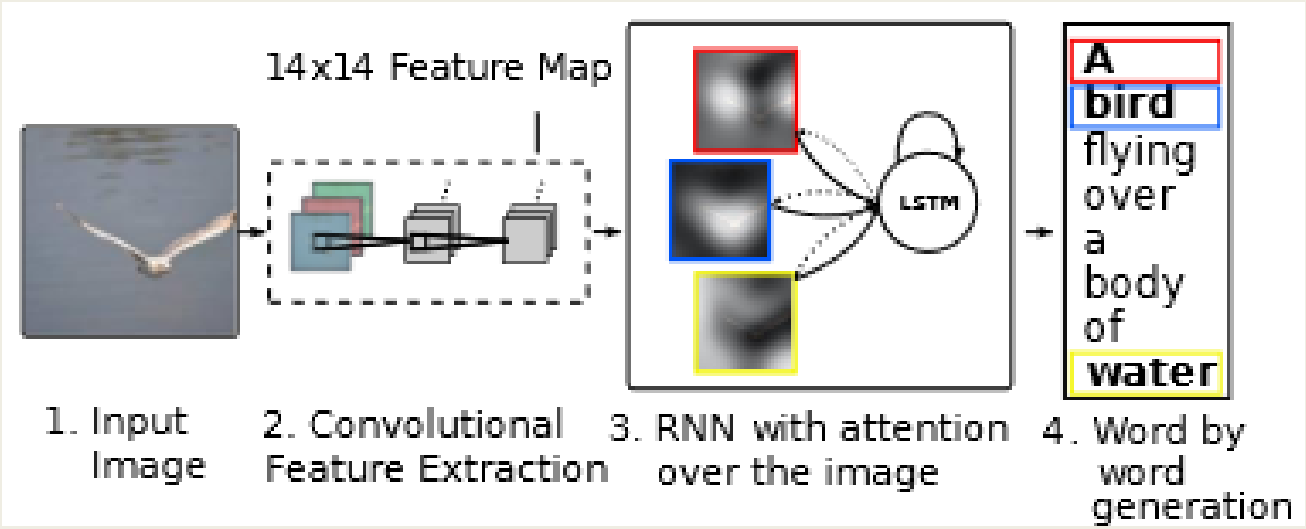
ATTENTION BASED MODEL - ABM

- Historically
  - *Encoder-Decoder*
  - *CNN+RNN*
  - *R-CNN+RNN*
- New model
  - *Dynamic information based on attention mechanism*
  - *No information loss*

CONSTRUCTION STEP BY STEP

ABM – GLOBAL VIEW

- Images fed as input
- Encode each image by using a pool of filters
- RNN + Attention



ABM – Model details pt. 1

- Use of any encoding function
  - *Use of Oxford VGGnet pretrained on ImageNet [Simonyan & Zisserman, 2014]*
  - ***Extract features from 4° max-pool layer** (14x14x512)*
  - *Encoded as matrix (196x512)*
  - ***Fixed Markovian Space***
- Time proportional to longest sentence in dataset
  - *Creation of dictionary*
  - *Sampling length*
  - *Mini-batch of 64 on sampled length*

ABM – Model details pt. 2

- **Visualize the attention**
  - *Image 224*224, but code is 14*14*512*
    - Upsample to 2^4=16 -> 224*224
- **Evaluation Metrics**
  - ***BLEU*** *[K. Papineni, S. Roukos, T. Ward, and W. J. Zhu, "BLEU: A methodfor automatic evaluation of machine translation," inProc. 40thAnnu. Meeting Assoc. Comput. Linguistics, 2002, pp. 311–318]*
  - ***CIDRE*** *[R. Vedantam, C. L. Zitnick, and D. Parikh, "CIDEr: Consensus-based image description evaluation," inarXiv:1411.5726, 2015]*
  - ***METEOR*** *(with AlexNet – harmonic mean btw precision and recall (weighted majorly on recall)) [Denkowski & Lavie, 2014]*
- **Regularization**
  - *Dropout*
  - *Early Stopping on BLEU*
- **Whetlab** used for Flickr8K dataset

ABM – FROM ANNOTATION TO CONTEXT

ENCODER

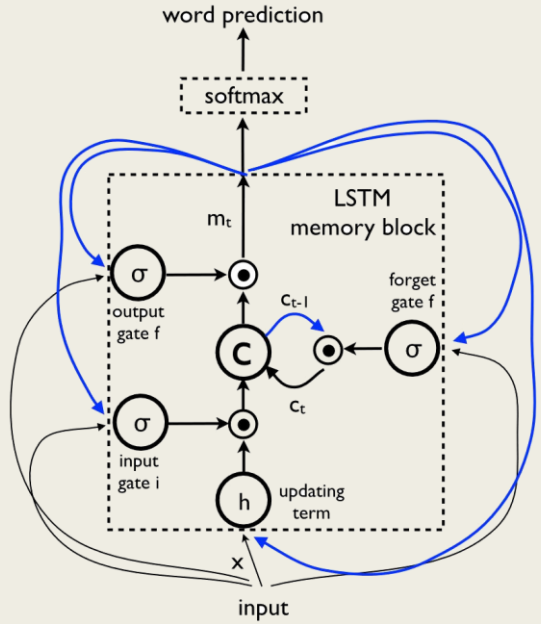
- Image to vector of words taken (Eq 1)
- Extract set of feature from encoder (**annotation vector «a»**) – (Eq 2)

DECODER

- Long-Short Time Memory (LSTM) – (Fig 1 – Eq 3) [S. Hochreiter and J. Schmidhuber, "Long short-term memory,"Neural Comput., vol. 9, no. 8, pp. 1735–1780, 1997.]
  - *Conditioned on context vector (dynamic representation of the actual information)*

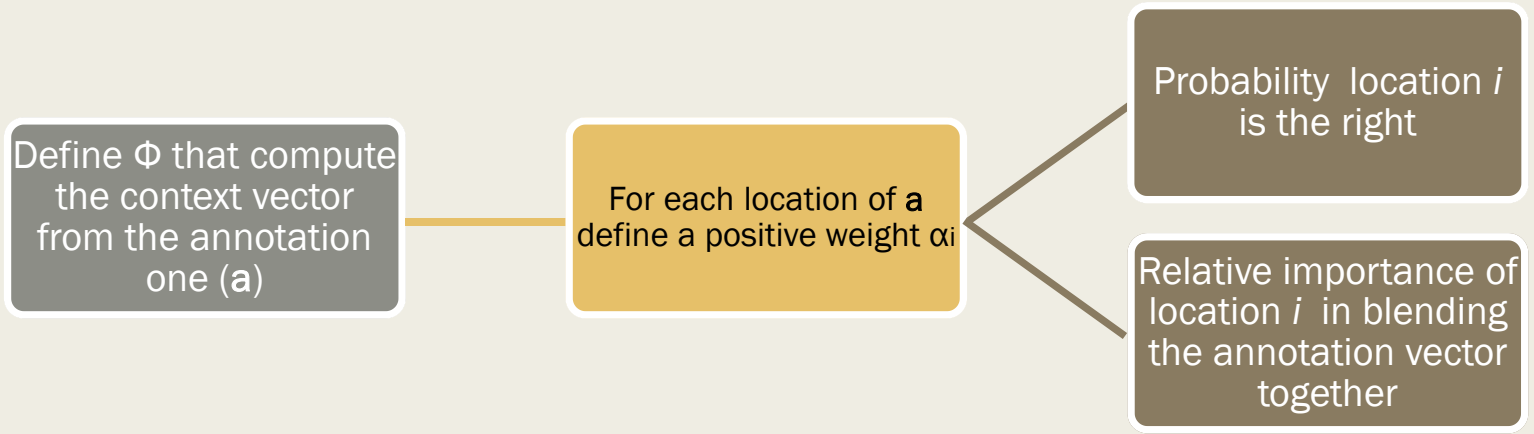
$$y = \{y_1, \ldots, y_c\}, y_i \in \mathbb{R}^k$$

$$a = \{a_1, \ldots, a_c\}, a_i \in \mathbb{R}^D$$



$$i_t = \sigma\left(W_{ix}x_t + W_{im}m_{t-1}\right)$$
$$f_t = \sigma\left(W_{fx}x_t + W_{fm}m_{t-1}\right)$$
$$o_t = \sigma\left(W_{ox}x_t + W_{om}m_{t-1}\right)$$
$$c_t = f_t \odot c_{t-1} + i_t \odot h\left(W_{cx}x_t + W_{cm}m_{t-1}\right)$$
$$m_t = o_t \odot c_t$$
$$p_{t+1} = \mathrm{Softmax}\left(m_t\right),$$

ABM
-
FROM ANNOTATION TO CONTEXT
-
Decoder pt.2



ABM – FROM ANNOTATION TO CONTEXT

- **Mathematical Background**
  - Next word depends on words already generated (Eq. 1)
    - *Then, also the context vector is computed (Eq. 2)*
  - Encoder Initialization (Eq. 3-4)
  - ***DEEP OUTPUT LAYER***. Compute output by combining

$$p\left(\mathbf{y}_t \mid \mathbf{a}, \mathbf{y}_1^{t-1}\right) \propto \exp\left(\mathbf{L}_o\left(\mathbf{E}\mathbf{y}_{t-1} + \mathbf{L}_h\mathbf{h}_t + \mathbf{L}_z\hat{\mathbf{z}}_t\right)\right)$$

Where $\mathbf{L}_o \in \mathbb{R}^{K\times m}, \mathbf{L}_h \in \mathbb{R}^{m\times n}, \mathbf{L}_z \in \mathbb{R}^{m\times D}$, and $\mathbf{E}$ are learned parameters initialized randomly.

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

$$\alpha_{ti} = \frac{exp(e_{ti})}{\sum_{k=1}^{L} exp(e_{tk})}$$

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

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

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

ABM – DETERMINISTIC «SOFT» ATTENTION

- Take the expectation of the context vector (Eq 1)
- Formulation of deterministic attention by computing a **soft attention** weighted annotation vector (Φ) (Eq 2) - [Bahdanauet al. (2014)]
- Previous step: Fed the system with soft α weighted context
- Model smooth and differentiable under the deterministic attention => **backpropagation**.

$$\mathbb{E}_{p(s_t|a)}\left[\hat{\mathbf{z}}_\ell\right] = \sum_{i=1}^{L} \alpha_{l,i}\mathbf{a}_i$$

$$\phi\left(\{\mathbf{a}_i\}, \{\alpha_i\}\right) = \sum_{i}^{L} \alpha_i \mathbf{a}_i$$

ABM – STOCHASTIC «HARD» ATTENTION

- **«St» location variable:** model focus attention for t-th word.
  - *Defining one hot variable «St,i».*
  - *Attention locations treated as latent variables*
  - *Assign multinoulli distribution parameterized by {αi}*
- Context locations treated as random variables (Eq 1.1)
- **New Objective Function Ls (pt. 1)**
  - *Variational Lower Bound on the initial distribution (marginal – Eq 2.1)*
  - *Use of **Monte Carlo based sampling approximation** of the gradient, by sampling «St» (Eq. 2.2) from a multinoulli distribution*

$$p\left(s_{t,i} = 1 \mid s_{j<t}, \mathbf{a}\right) = \alpha_{t,i}$$
$$\hat{\mathbf{z}}_t = \sum_i s_{t,i}\mathbf{a}_i$$

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

ABM – STOCHASTIC «HARD» ATTENTION

New Objective Function Ls (pt. 2)

- Moving average baseline for reduce the variance in Monte Carlo estimation
  - Or better (Eq1) [Mnih et al. (2014) and Ba et al. (2014)]
  - *Adding entropy term to further reduce the estimation (Eq 2 – lambdas are parameters set by cross-validation)* *[equivalent to REINFORCE learning rule - (Williams, 1992)]*
- **RESULT: the context vector is a function that returns a sampled a_i at every point in time based on a multinouilli distribution parameterized by α**

$$b_k = 0.9 \times b_{k-1} + 0.1 \times \log p\left(\mathbf{y} \mid \tilde{s}_k, \mathbf{a}\right)$$

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

$$\tilde{s}_t \sim \text{Multinoulli}_L\left(\{\alpha_i\}\right)$$

ABM – DOUBLY STOCHASTIC ATTENTION

- **RECALL.** The sum of weights for the annotation vector **must be 1** (softmax)
- Introducing sort of regularization (Eq 1)
  - *Force the model to pay equal attention to every part of the image* *[Show, Attend and Tell: Neural Image Caption Generation with Visual Attention, Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhutdinov, Richard Zemel, Yoshua Bengio]*
- The soft attention model predicts also β from the previous hidden state at each time step by Eq 2
  - *The attention weights put more emphasis on the objects in the images by including β.*

$$\sum_t \alpha_{ti} \approx 1$$

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

$$where$$

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

TRAINING – MINIMIZE PENALIZED LIKELIHOOD

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

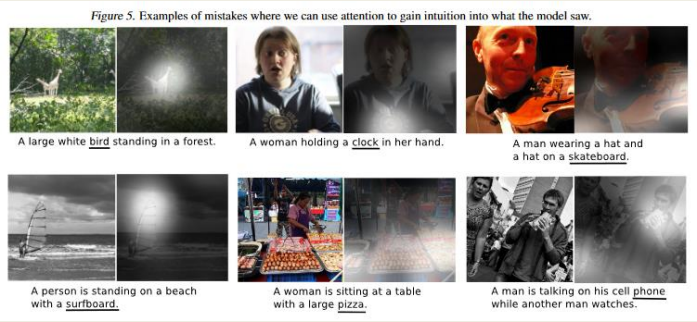


Figure 5. Examples of mistakes where we can use attention to gain intuition into what the model saw.

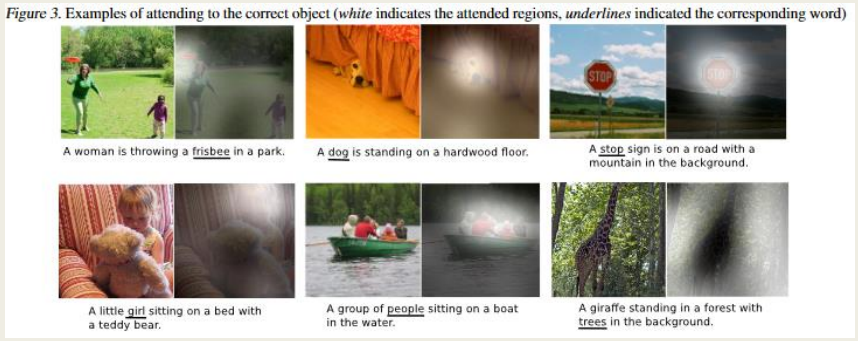


Figure 3. Examples of attending to the correct object (*white* indicates the attended regions, *underlines* indicated the corresponding word)

Table 1. BLEU-1,2,3,4/METEOR metrics compared to other methods, † indicates a different split, (—) indicates an unknown metric, ○ indicates the authors kindly provided missing metrics by personal communication, Σ indicates an ensemble, *a* indicates using AlexNet

Dataset	Model	BLEU				
		BLEU-1	BLEU-2	BLEU-3	BLEU-4	METEOR
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

ABM - RESULTS

- Visualize attention component
  - *Another interpretability level*
- Historically
  - Object detection for ideal candidate
  - Pre-initialized static objects
- Model effectively pay attention in different “non-object” regions
  - *As the model generates each word, its attention changes to reflect different relevant parts of the image*

TABLE 1
Scores on the MSCOCO Development Set for Two Models: NIC, Which Was the Model Which We Developed in [46], and NICv2, Which Was the Model After We Tuned and Refined Our System for the MSCOCO Competition

Metric	BLEU-4	METEOR	CIDER
NIC	27.7	23.7	85.5
NICv2	**32.1**	**25.7**	**99.8**
Random	4.6	9.0	5.1
Nearest Neighbor	9.9	15.7	36.5
Human	21.7	25.2	85.4

PERSONAL CONSIDERATION

- Powerful method for predict captions
  - *LSTM require that the sequential data be processed in order*
    - As in the encoding part (pre-trained), we could use transformer (no fixed Markovian Space => more parallelism)
- Try the proposed model
  - ***Original:*** *https://github.com/kelvinxu/arctic-captions*
    - Better version: https://github.com/Lorne0/arctic-captions

THANK YOU FOR LISTENING!