EE782 Advanced Topics in Machine Learning
06 Neural Networks for NLP

Amit Sethi

Electrical Engineering, IIT Bombay

Module objectives

- Design LSTM based networks for various NLP tasks
- Articulate the information bottleneck in encoder-decoder architectures
- Explain how attention alleviates the information bottleneck
- Understand how attention can do almost all tasks done by the other layers

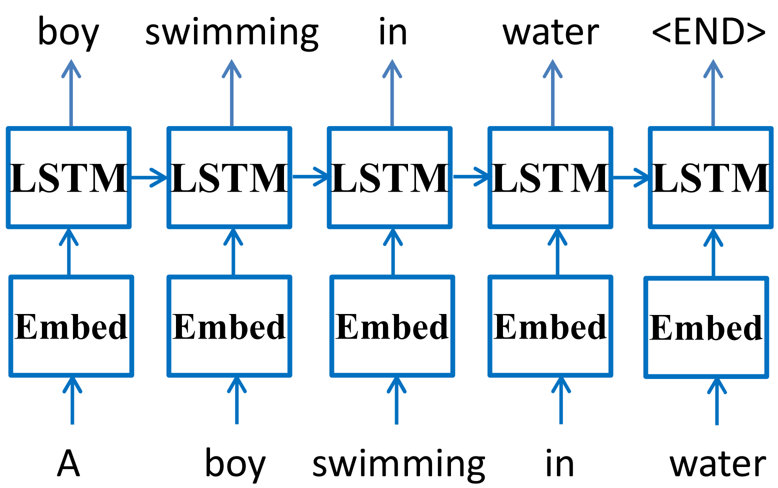
Pre-processing for NLP

- The most basic pre-processing is to convert words into an embedding using Word2Vec or GloVe

- Otherwise, a one-hot-bit input vector can be too long and sparse, and require lots on input weights

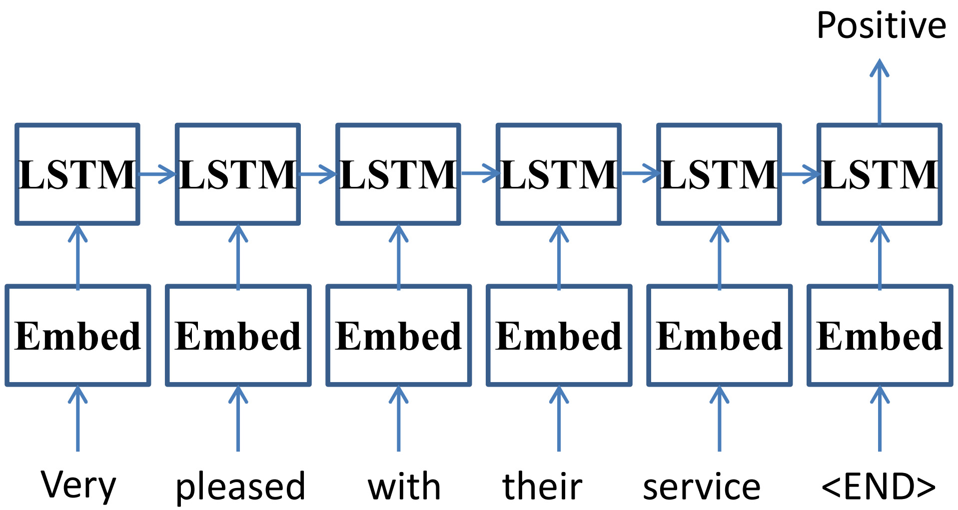
Pre-training LSTMs

- Learning to predict the next word can imprint powerful language models in LSTMs
- This captures the grammar and syntax
- Usually, LSTMs are pre-trained on corpora



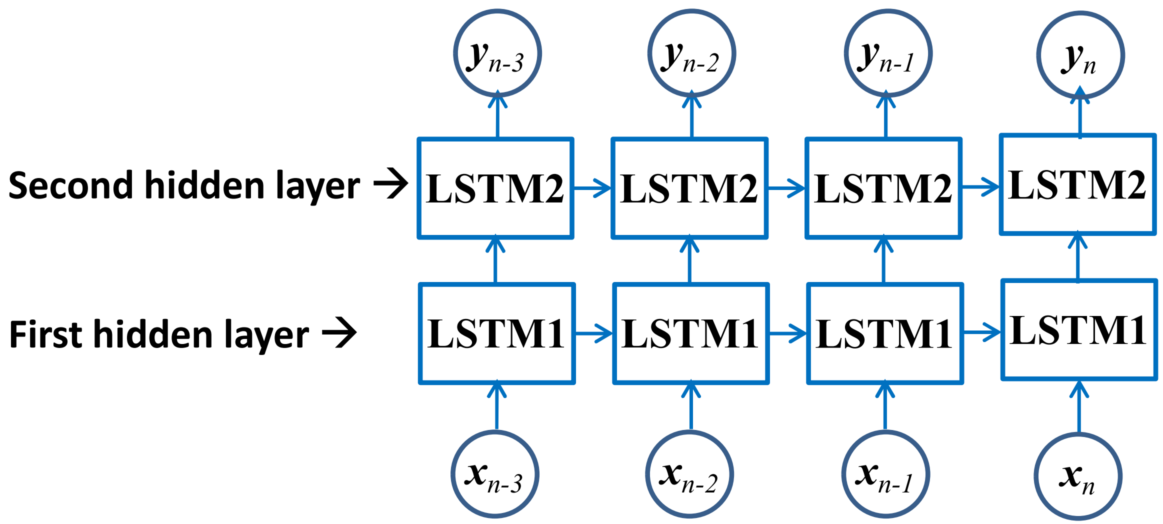
Sentiment analysis

- Very common for customer review or new article analysis
- Output before the end can be discarded (not used for backpropagation)
- This is a many-to-one task



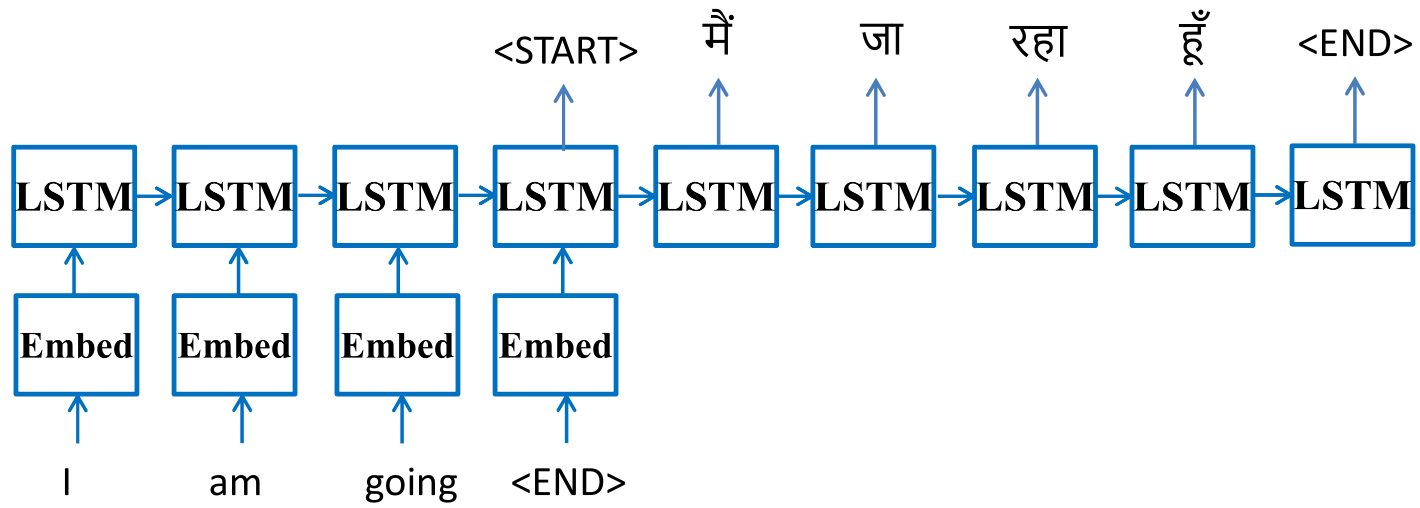
Multi-layer LSTM

- More than one hidden layer can be used



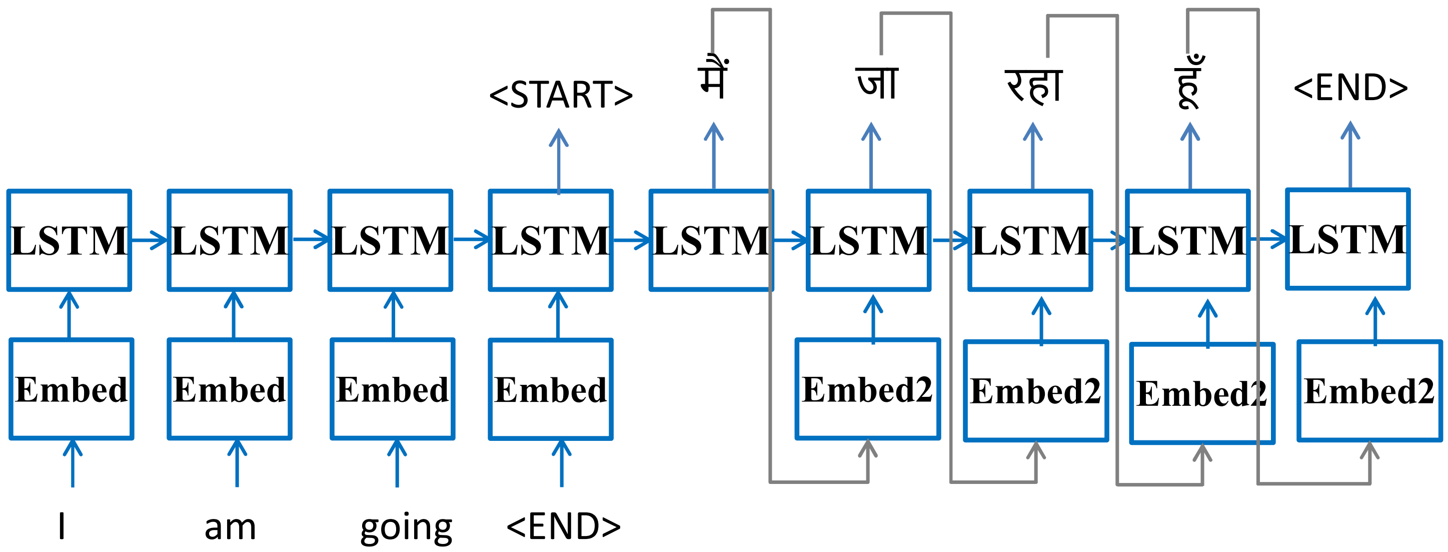
Machine translation

- A naïve model would be to use a many-to-many network and directly train it



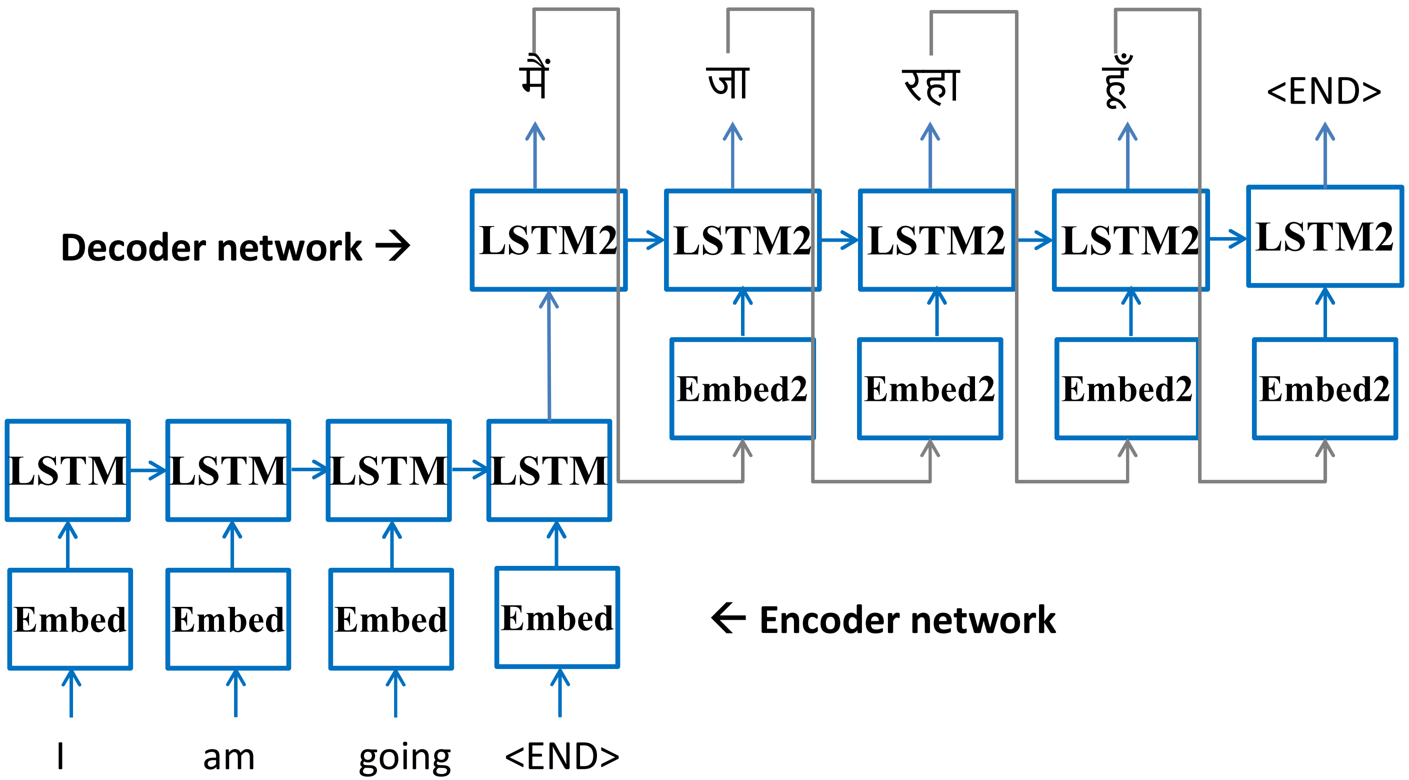
Machine translation

- One could also feed in the output to the next instance input to predict a coherent structure

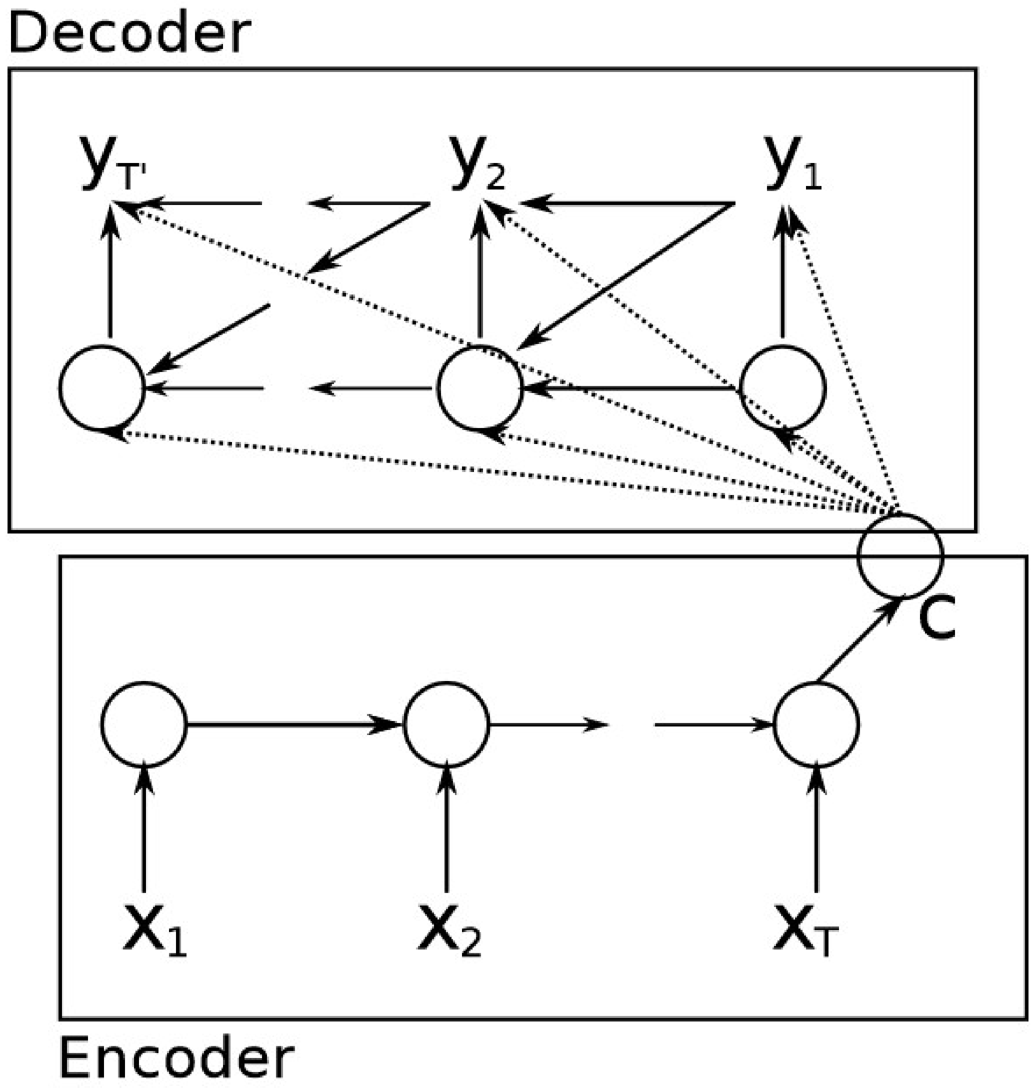


Machine translation

- In actuality, one would use separate LSTMs pre-trained on two different languages



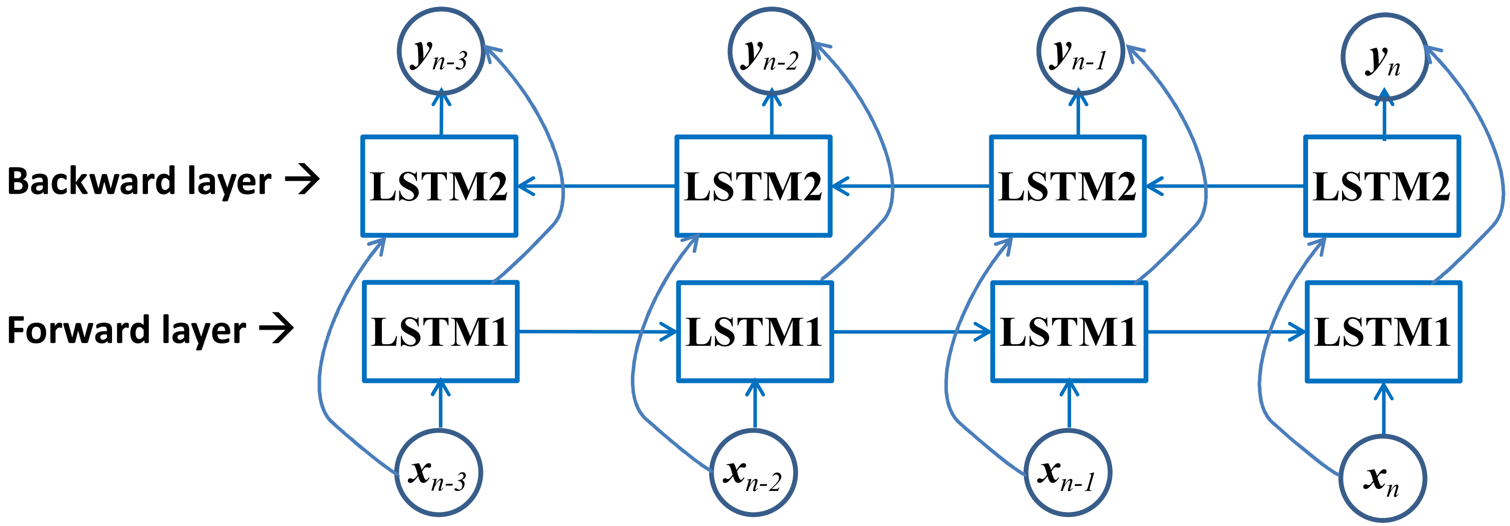
Machine translation using encoder-decoder



Source: “Learning Phrase Representations using RNN Encoder–Decoderfor Statistical Machine Translation”, Cho et al., ArXiv 2014

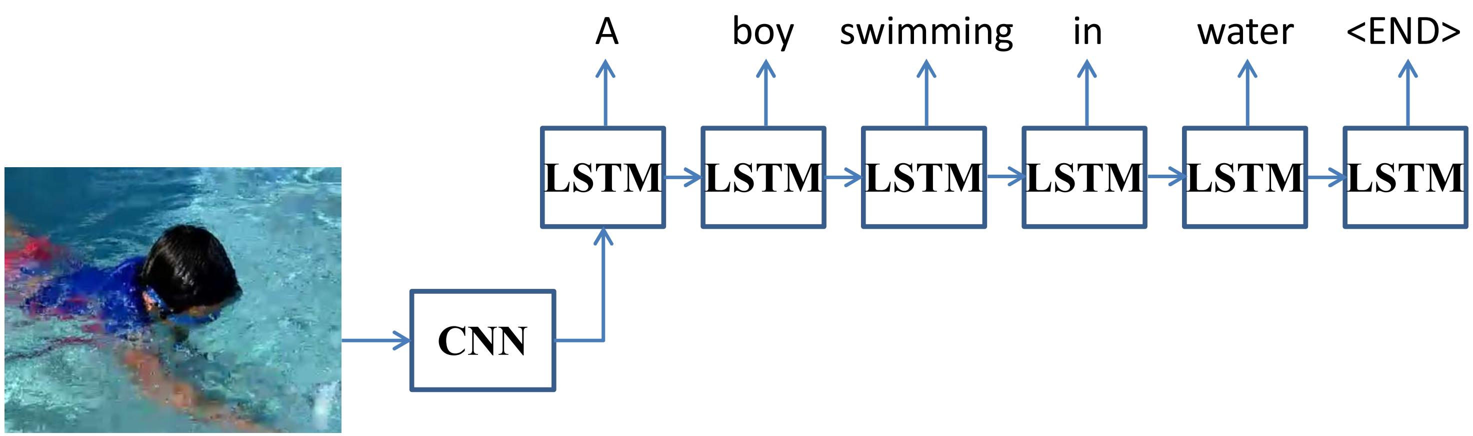
Bi-directional LSTM

- Many problems require a reverse flow of information as well
- For example, POS tagging may require context from future words



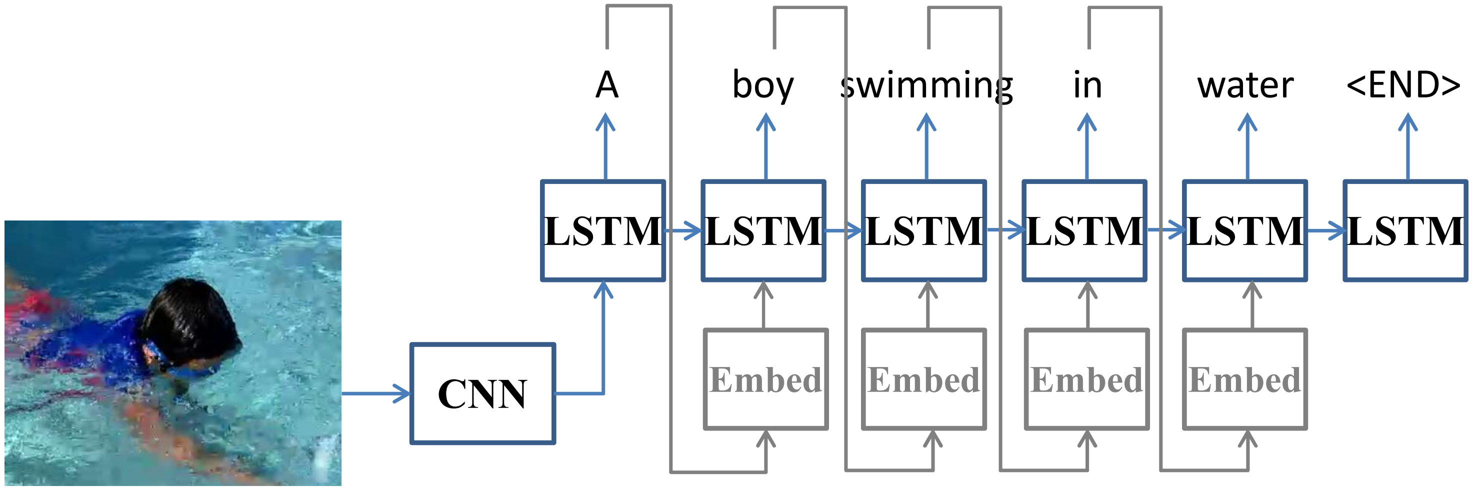
Sentence generation

- Very common for image captioning
- Input is given only in the beginning
- This is a one-to-many task

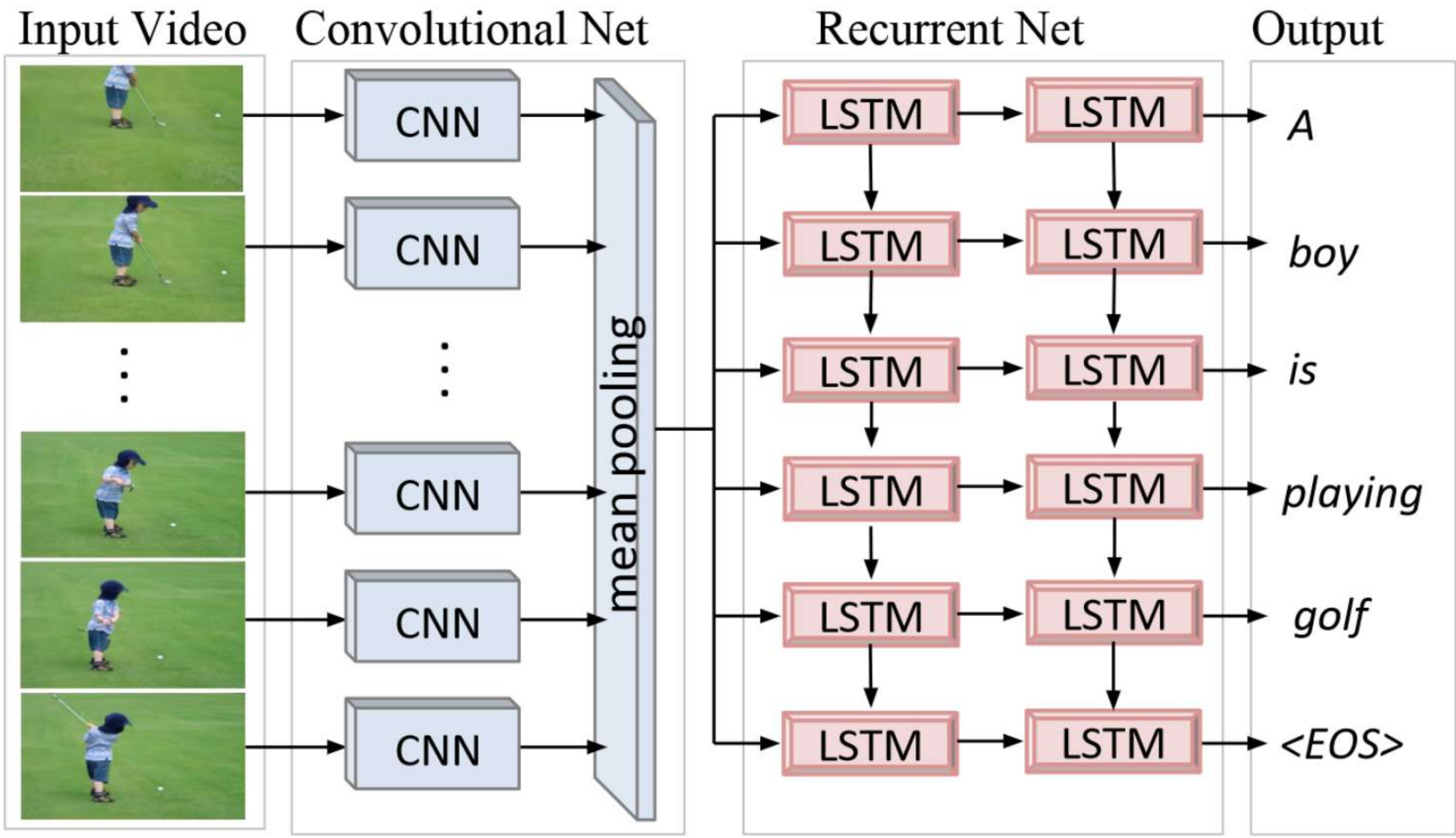


Sentence generation

- Very common for image captioning
- Input is given only in the beginning
- This is a one-to-many task



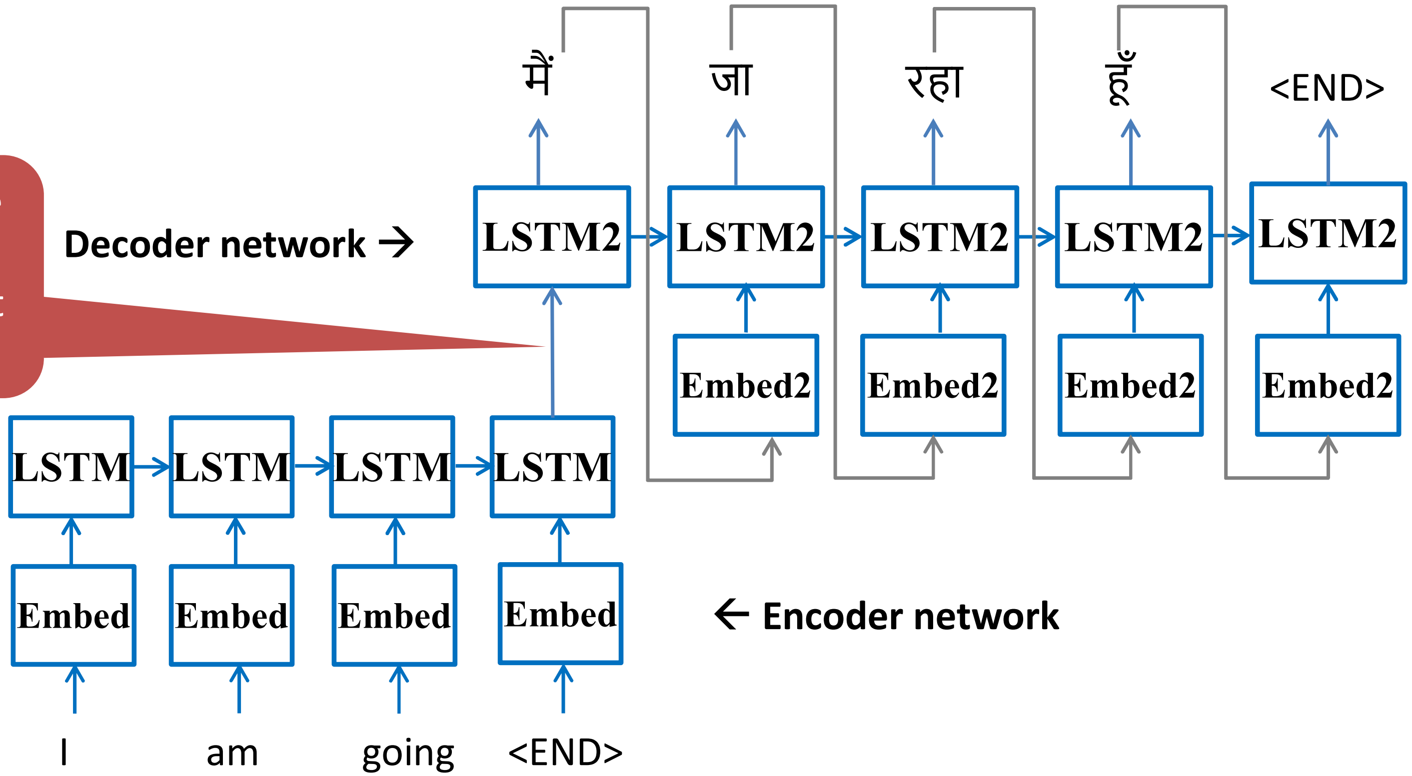
Video Caption Generation



Source: "Translating Videos to Natural Language Using Deep Recurrent Neural Networks", Venugopal et al., ArXiv 2014

Information bottlenech in the context vector

The entire info about the previous sentence has to flow through this context vector ***c***



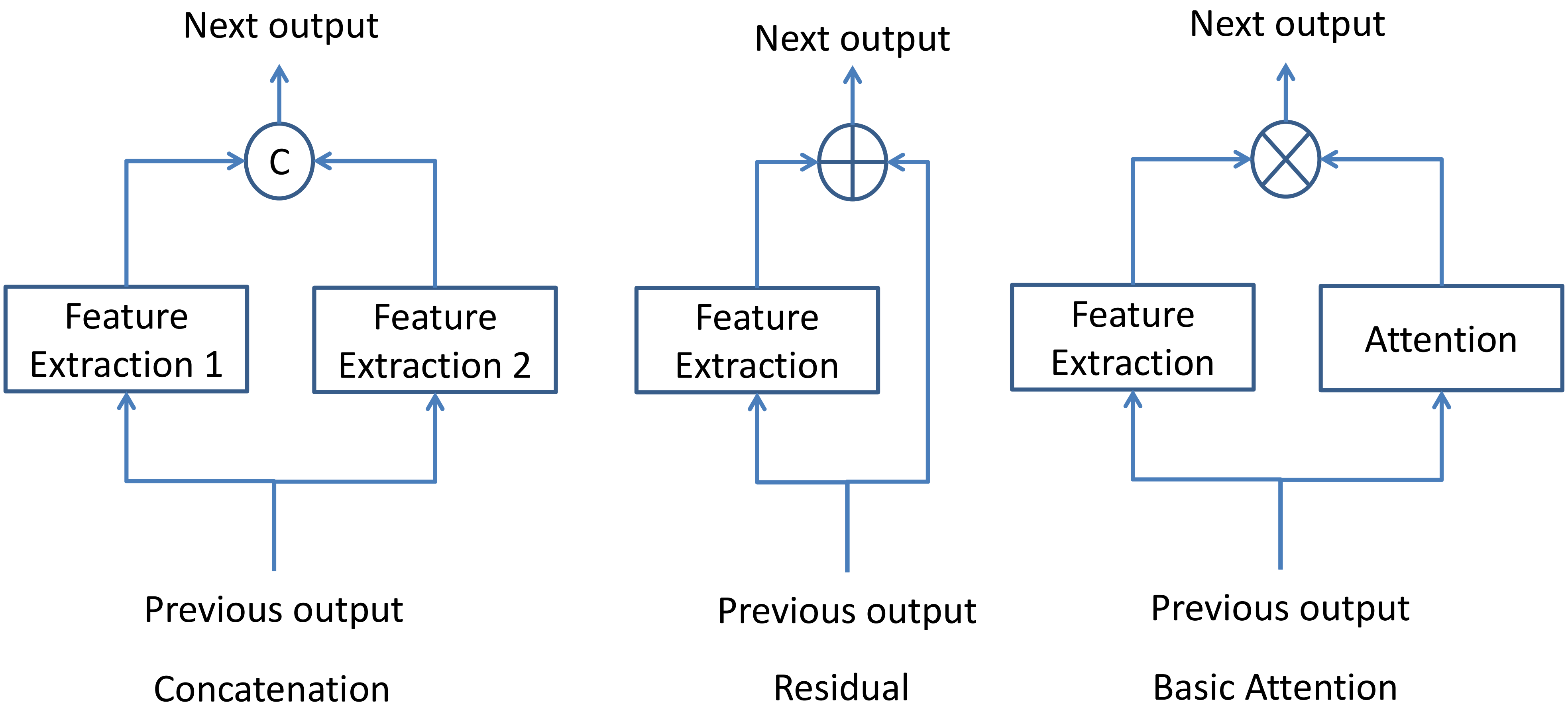
Pooling summarizes feature vectors over several locations

- Average pooling: $1/N \sum_i x_i$
- Max pooling: $\max_i (x_i)$
- Attention: $\sum_i a_i x_i$
  - Often, $a_i = \mathrm{softmax}(e_i)$
  - And $e_i = f(q, k_i)$

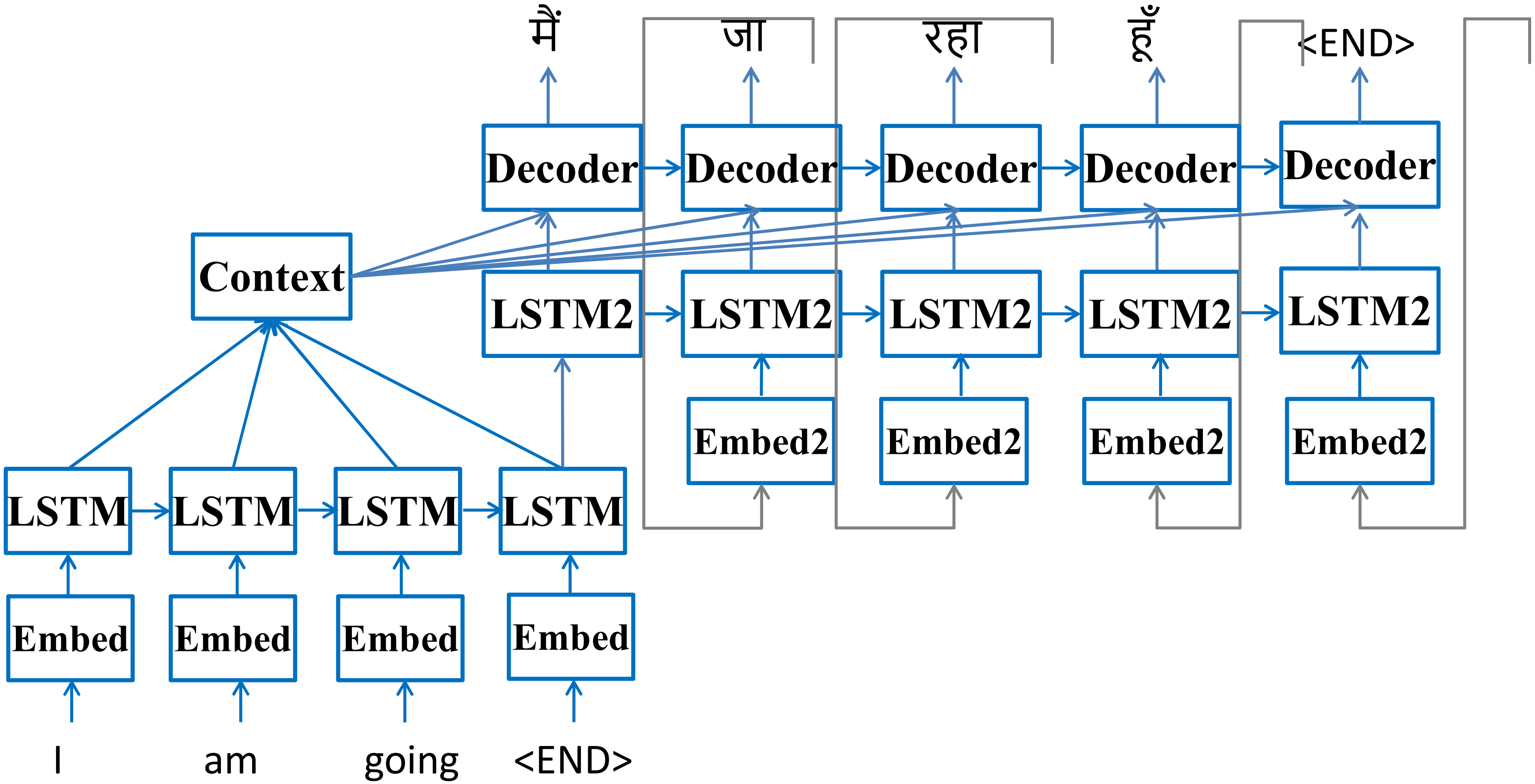
Attention is like intelligent pooling

- E.g. kernel methods: $y = \sum_i k(x,x_i)\, y_i$
  - Query is x
  - Key is x_i
  - Value is y_i
  - Permutation-invariant in x_i
- In softmax attention, the sum of the attentions is *1*

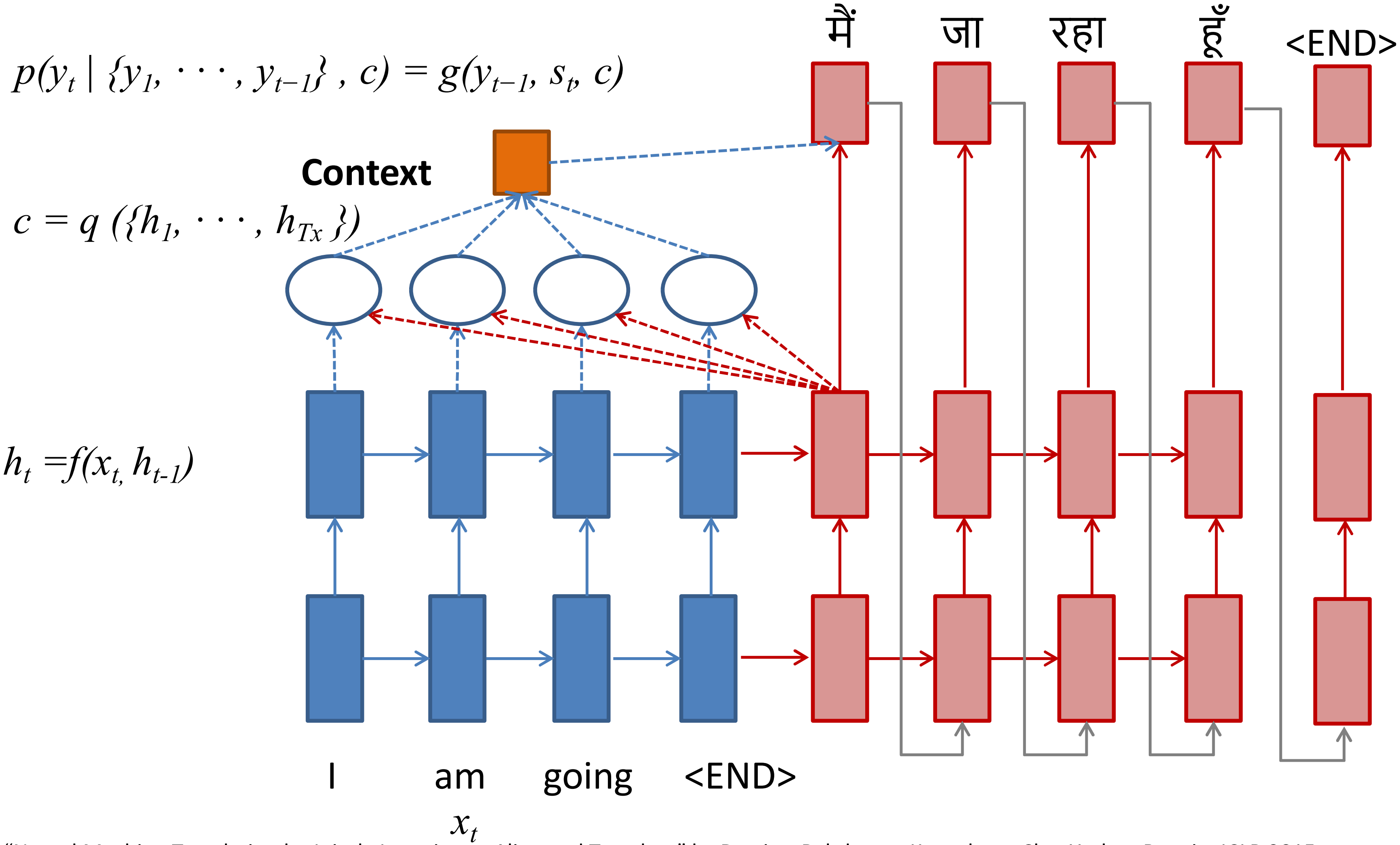
Side-branches in neural networks



LSTM with Attention Mechanism



LSTM with Attention Mechanism



“Neural Machine Translation by Jointly Learning to Align and Translate” by Dzmitry Bahdanau, Kyunghyun Cho, Yoshua Bengio, ICLR 2015

LSTM with Attention Mechanism

$$p(y_i \mid y_1, \ldots, y_{i-1}, \boldsymbol{x}) = g(y_{i-1}, s_i, c_i)$$

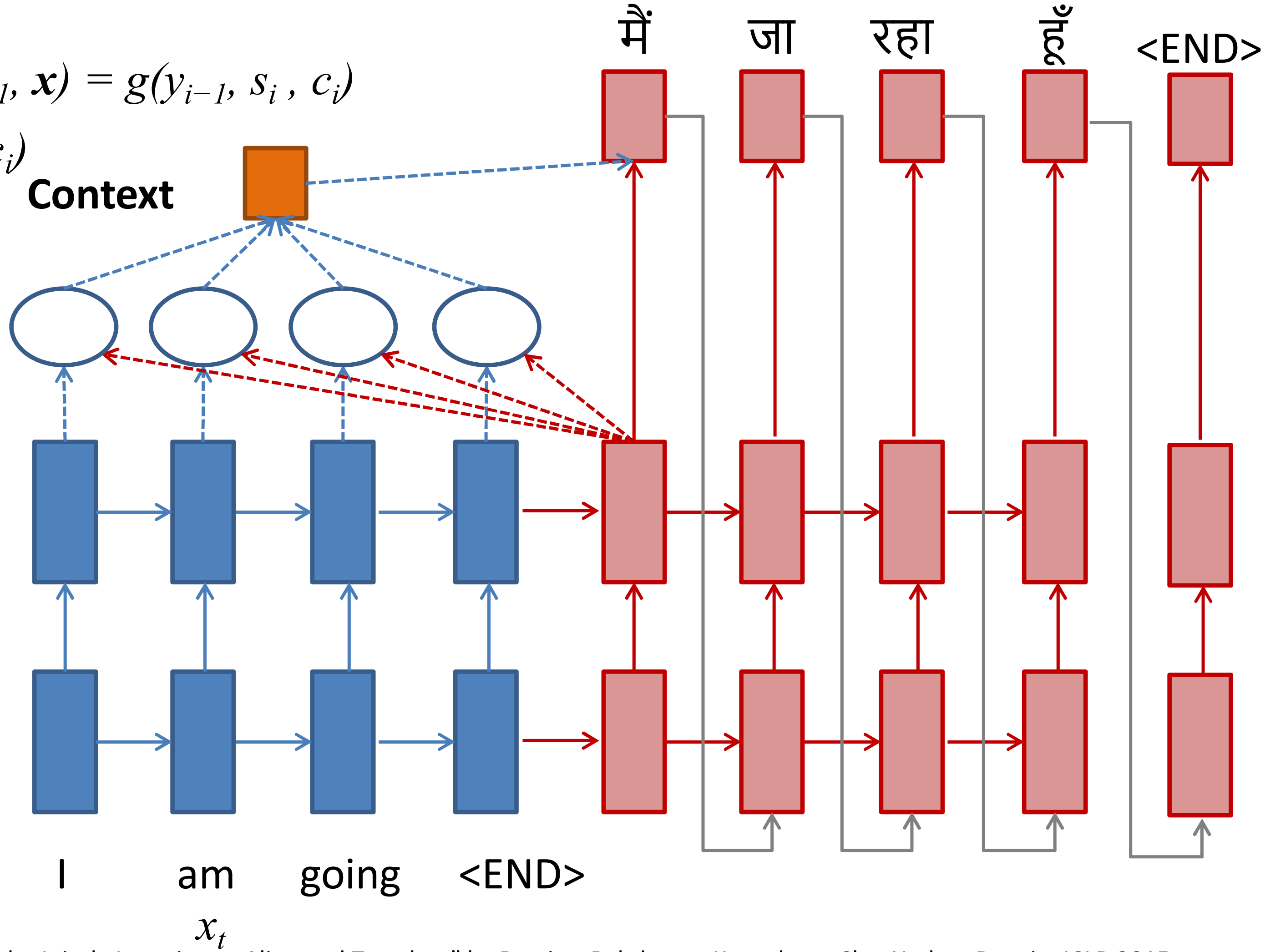
$$s_i = f(s_{i-1}, y_{i-1}, c_i)$$

$$c_i = \sum_j \alpha_{ij} h_j$$

$$\alpha_{ij} = softmax(e_{ij})$$

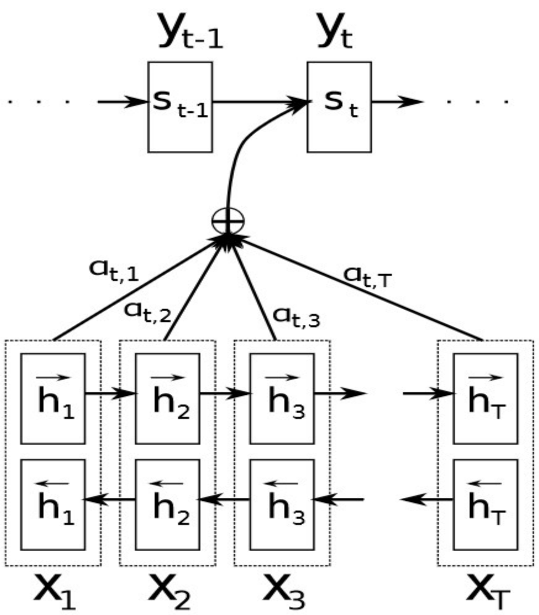
$$e_{ij} = a(s_{i-1}, h_j)$$

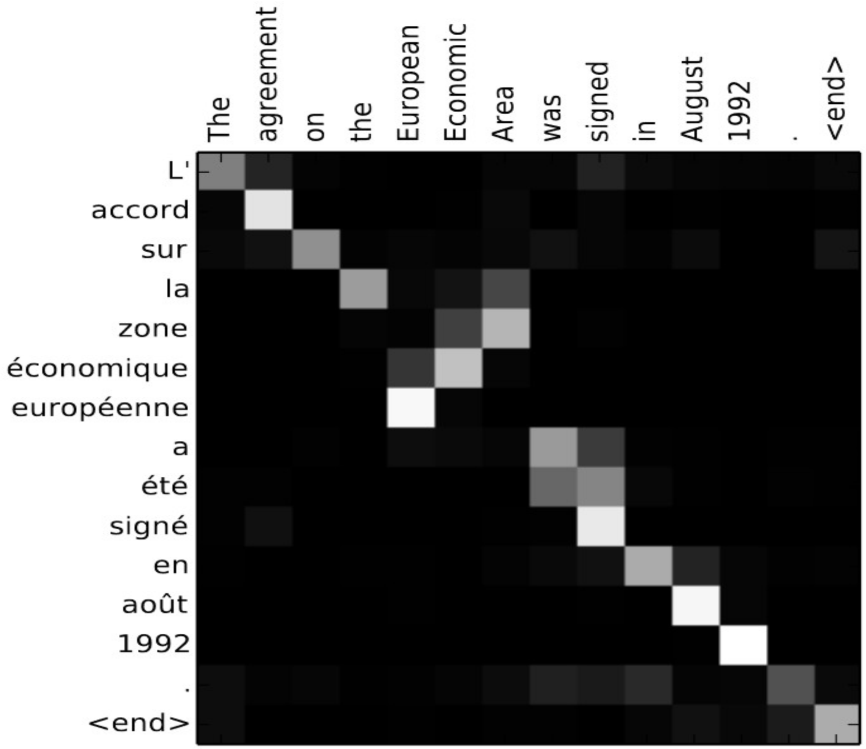
$$hj = f(xj, h_{j-1})$$



“Neural Machine Translation by Jointly Learning to Align and Translate” by Dzmitry Bahdanau, Kyunghyun Cho, Yoshua Bengio, ICLR 2015

Attention between encoder and decoder





Source: "Neural Machine Translation by Jointly Learning to Align and Translate," by Bahdanau, Cho, Bengio, 2014

How attention changes information flow

- Previously

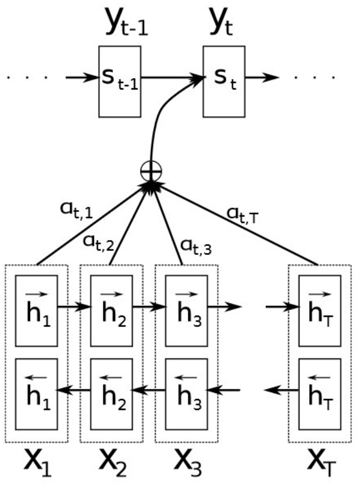
$$h_t = f\left(x_t, h_{t-1}\right)$$

$$c = q\left(\{h_1, \cdots, h_{T_x}\}\right)$$

$$q\left(\{h_1, \cdots, h_T\}\right) = h_T$$

$$p(\mathbf{y}) = \prod_{t=1}^{T} p(y_t \mid \{y_1, \cdots, y_{t-1}\}, c)$$

$$p(y_t \mid \{y_1, \cdots, y_{t-1}\}, c) = g(y_{t-1}, s_t, c)$$



- With attention

$$p(y_i | y_1, \ldots, y_{i-1}, \mathbf{x}) = g(y_{i-1}, s_i, c_i)$$

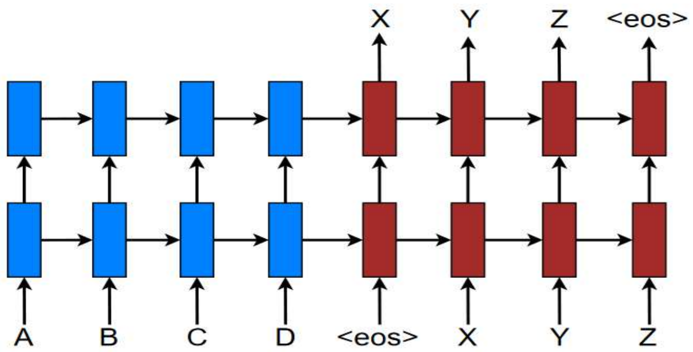
$$s_i = f(s_{i-1}, y_{i-1}, c_i)$$

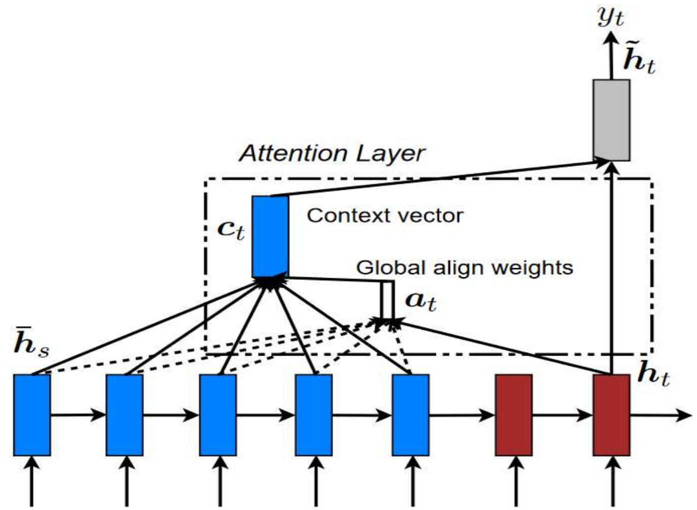
$$c_i = \sum_{j=1}^{T_x} \alpha_{ij} h_j \qquad \alpha_{ij} = \frac{\exp\left(e_{ij}\right)}{\sum_{k=1}^{T_x} \exp\left(e_{ik}\right)}$$

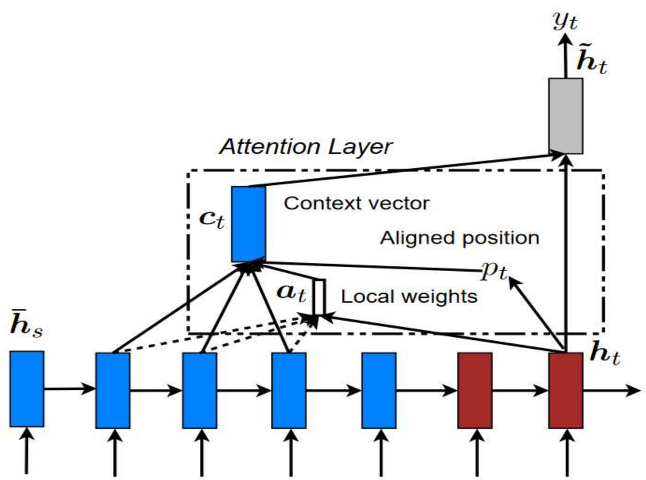
$$e_{ij} = a(s_{i-1}, h_j)$$

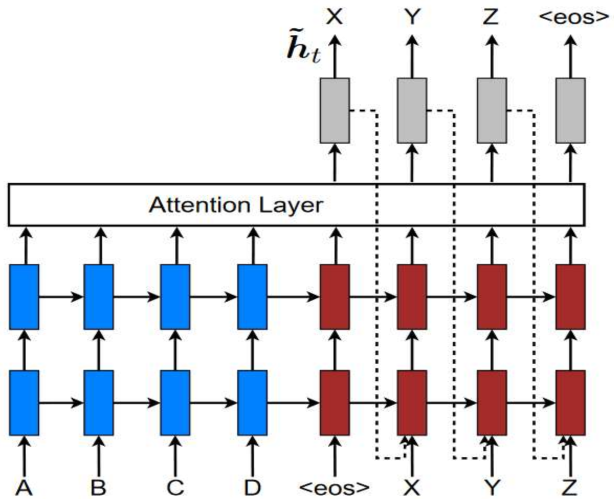
Source: "Neural Machine Translation by Jointly Learning to Align and Translate," by Bahdanau, Cho, Bengio, 2014

No vs. global vs. local attention



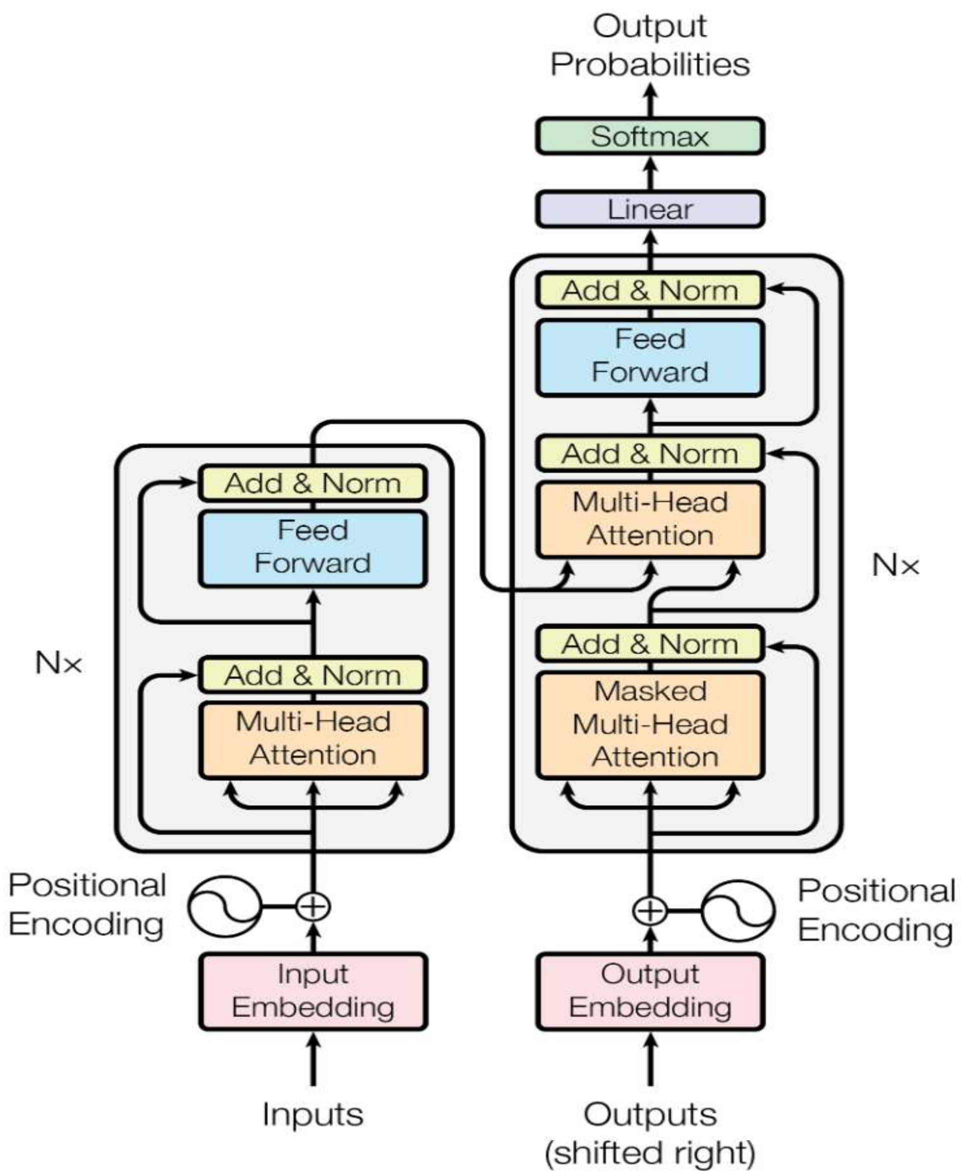






Source: "Effective Approaches to Attention-based Neural Machine Translation," by Luong, Pham, Manning, 2015

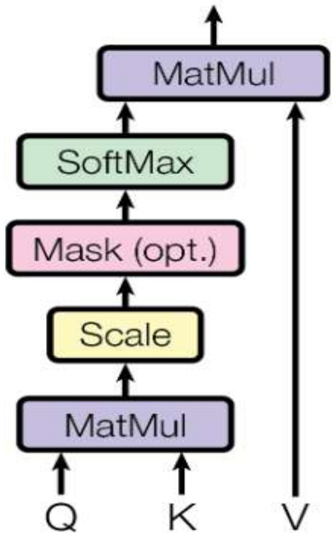
Transformer networks



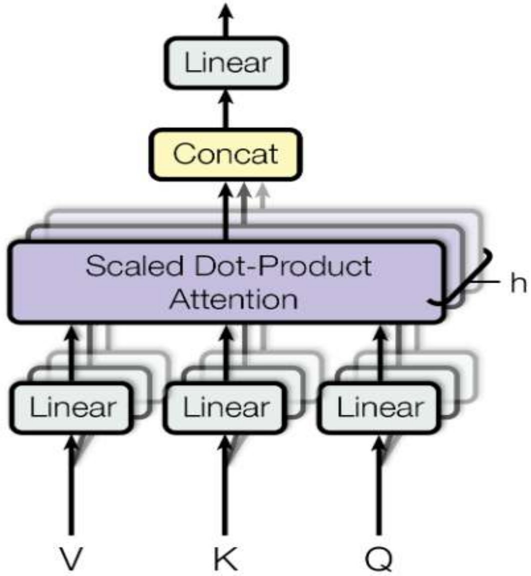
Source: "Attention Is All You Need," by Vaswani et al., 2017

Attention in Transformer networks

Scaled Dot-Product Attention



Multi-Head Attention



$$\mathrm{Attention}(Q, K, V) = \mathrm{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

$$\mathrm{MultiHead}(Q, K, V) = \mathrm{Concat}(\mathrm{head_1}, \ldots, \mathrm{head_h})W^O$$
$$\text{where } \mathrm{head_i} = \mathrm{Attention}(QW_i^Q, KW_i^K, VW_i^V)$$

Source: "Attention Is All You Need," by Vaswani et al., 2017

Details of transfomer by Vaswani et al. (2017)

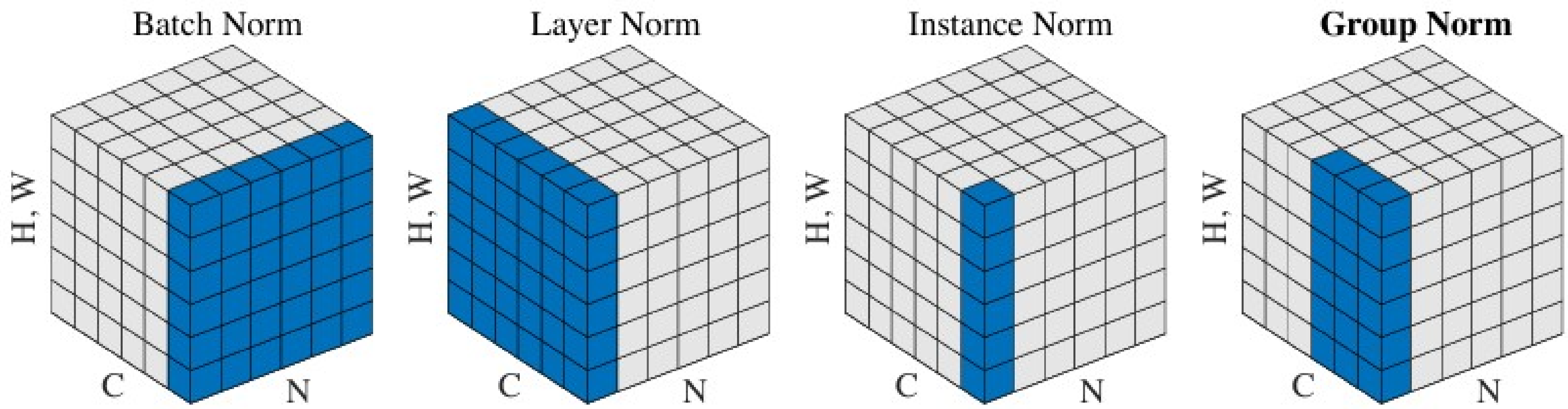
$$\mathrm{Attention}(Q, K, V) = \mathrm{softmax}(\frac{QK^T}{\sqrt{d_k}})V$$

$$\mathrm{MultiHead}(Q, K, V) = \mathrm{Concat}(\mathrm{head_1}, ..., \mathrm{head_h})W^O$$
$$\text{where } \mathrm{head_i} = \mathrm{Attention}(QW_i^Q, KW_i^K, VW_i^V)$$

$$PE_{(pos,2i)} = sin(pos/10000^{2i/d_{\mathrm{model}}})$$
$$PE_{(pos,2i+1)} = cos(pos/10000^{2i/d_{\mathrm{model}}})$$

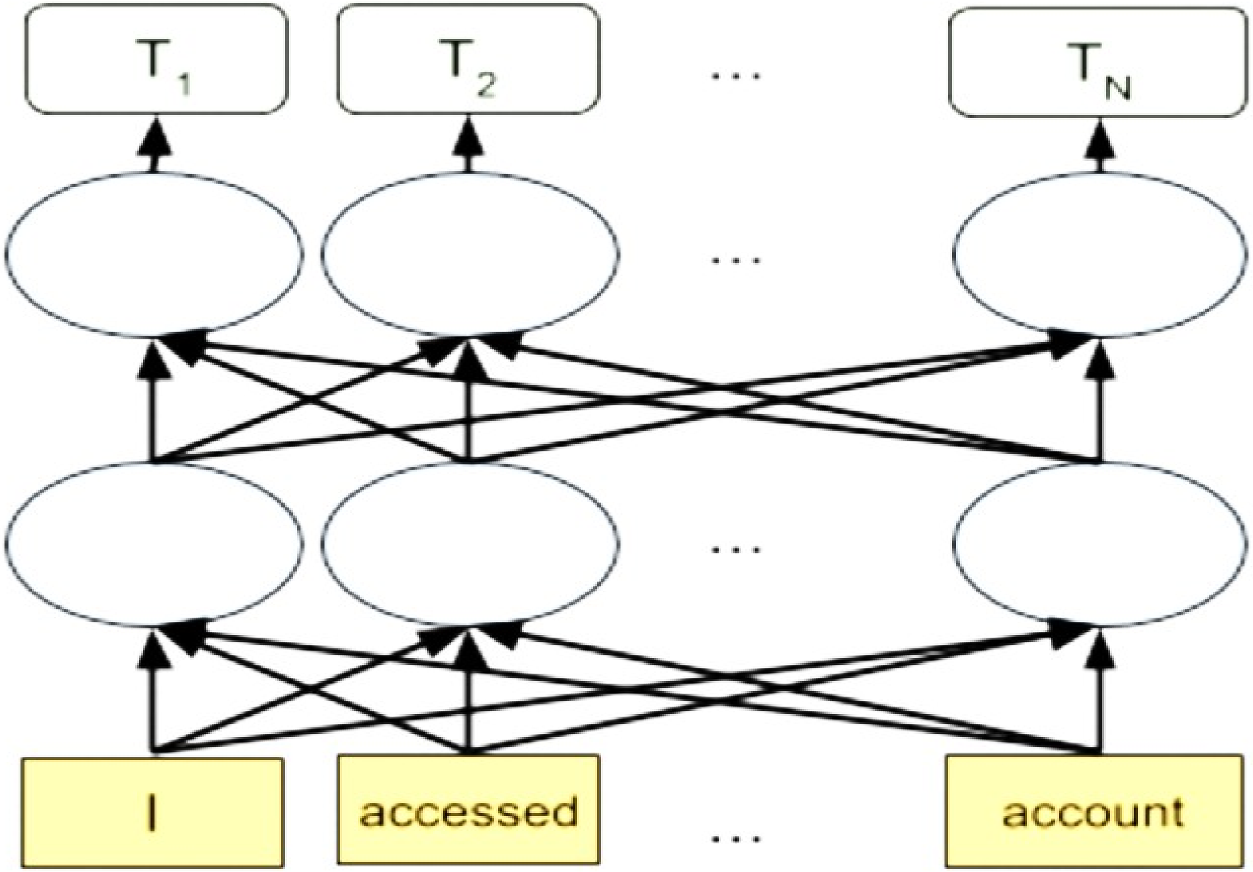
Source: "Attention Is All You Need," by Vaswani et al., 2017

Layer norm visualized



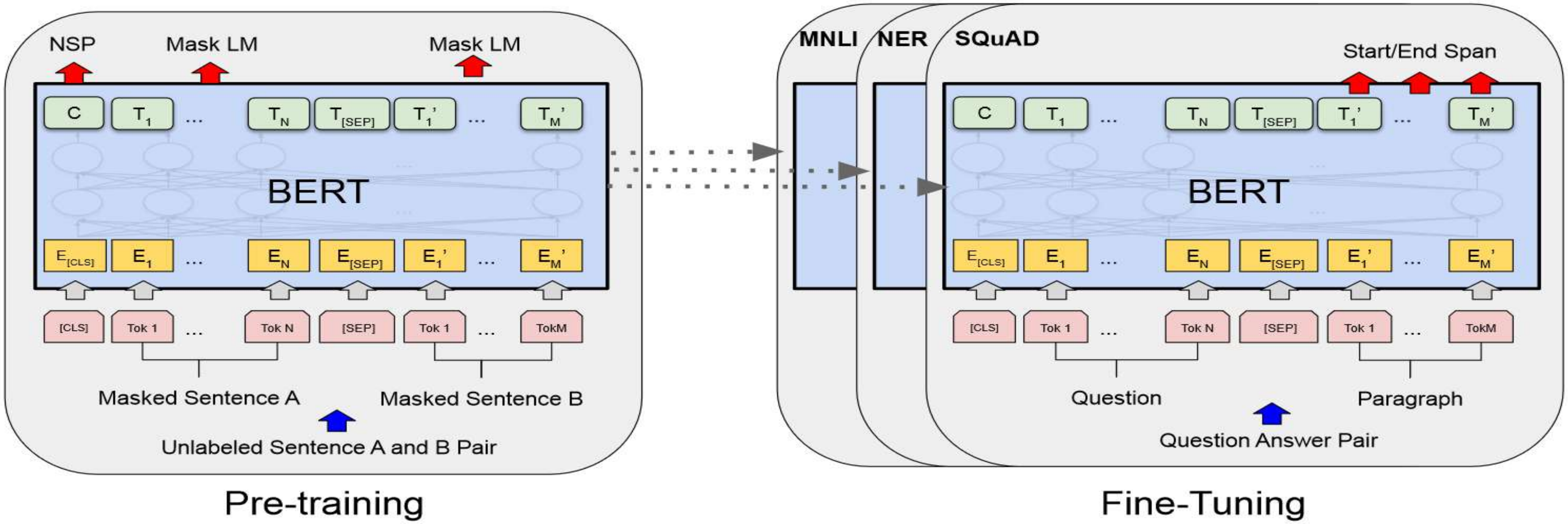
"Batch Normalization" by Ioffe, Szegedy, 2015
"Layer Normalization" by Ba, Kiros, Hinton, 2016
"Group Normalization" by Wu, He, 2017

BERT



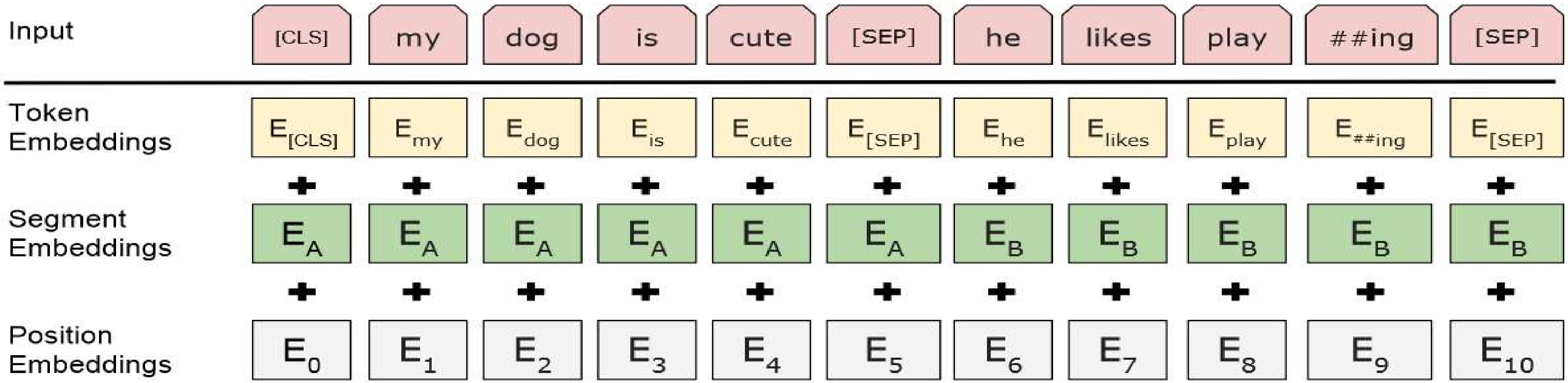
Source: "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," by Devli et al., 2018

BERT



Source: "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," by Devli et al., 2018

BERT



Source: "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," by Devli et al., 2018

Vision Transformer

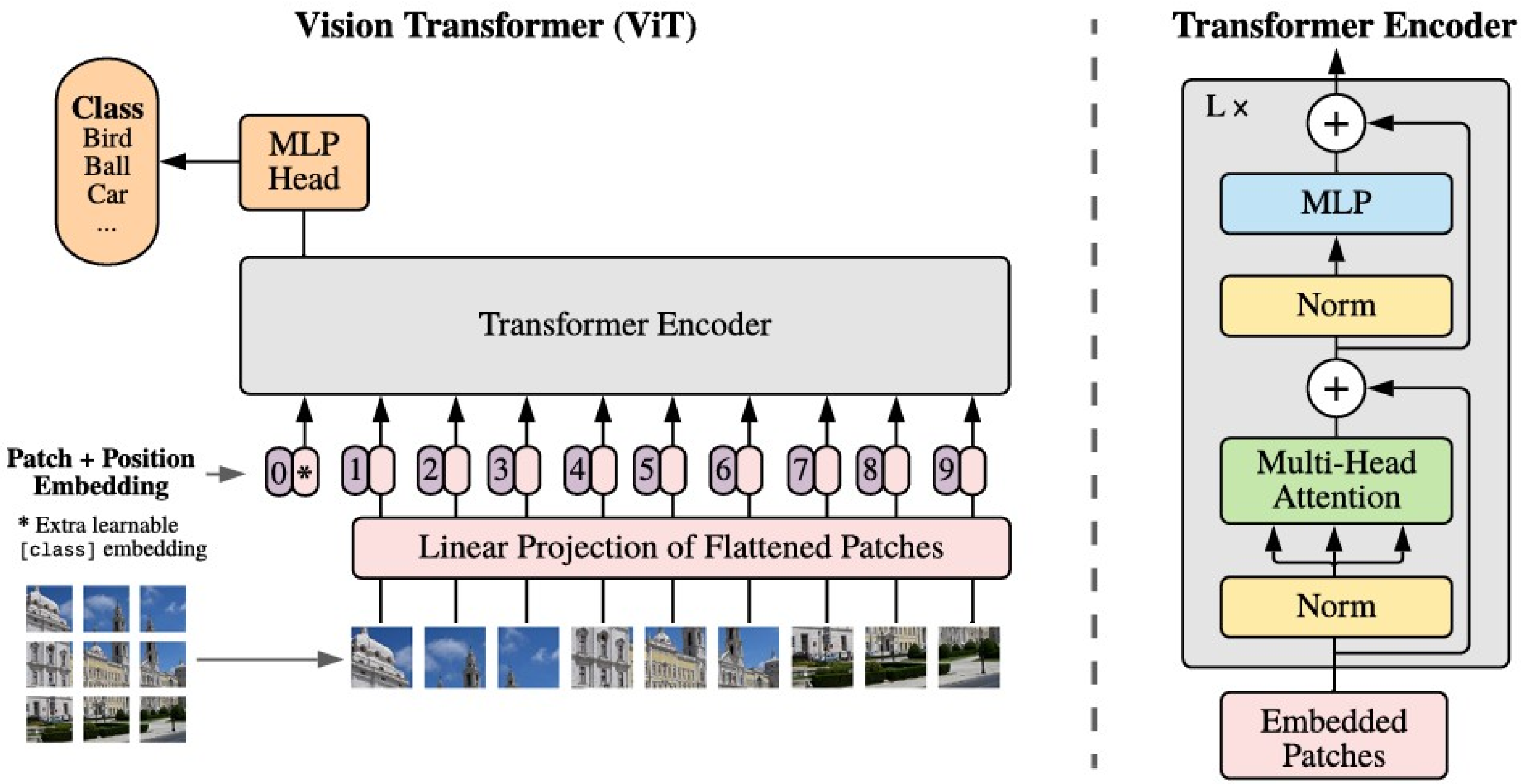


Figure 1: Model overview. We split an image into fixed-size patches, linearly embed each of them, add position embeddings, and feed the resulting sequence of vectors to a standard Transformer encoder. In order to perform classification, we use the standard approach of adding an extra learnable "classification token" to the sequence. The illustration of the Transformer encoder was inspired by Vaswani et al. (2017).

"An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale," Dosovitskiy et al. ICLR 2021 https://arxiv.org/pdf/2010.11929.pdf

Vision Transformer

$$\mathbf{z}_0 = [\mathbf{x}_{\text{class}};\, \mathbf{x}_p^1\mathbf{E};\, \mathbf{x}_p^2\mathbf{E}; \cdots;\, \mathbf{x}_p^N\mathbf{E}] + \mathbf{E}_{pos}, \qquad \mathbf{E} \in \mathbb{R}^{(P^2 \cdot C)\times D},\, \mathbf{E}_{pos} \in \mathbb{R}^{(N+1)\times D}$$

$$\mathbf{z'}_\ell = \mathrm{MSA}(\mathrm{LN}(\mathbf{z}_{\ell-1})) + \mathbf{z}_{\ell-1}, \qquad \ell = 1 \ldots L$$

$$\mathbf{z}_\ell = \mathrm{MLP}(\mathrm{LN}(\mathbf{z'}_\ell)) + \mathbf{z'}_\ell, \qquad \ell = 1 \ldots L$$

$$\mathbf{y} = \mathrm{LN}(\mathbf{z}_L^0)$$

$$[\mathbf{q}, \mathbf{k}, \mathbf{v}] = \mathbf{z}\mathbf{U}_{qkv} \qquad \mathbf{U}_{qkv} \in \mathbb{R}^{D\times 3D_h},$$

$$A = \mathrm{softmax}\left(\mathbf{q}\mathbf{k}^\top / \sqrt{D_h}\right) \qquad A \in \mathbb{R}^{N\times N},$$

$$\mathrm{SA}(\mathbf{z}) = A\mathbf{v}\,.$$

$$\mathrm{MSA}(\mathbf{z}) = [\mathrm{SA}_1(z); \mathrm{SA}_2(z); \cdots; \mathrm{SA}_k(z)]\,\mathbf{U}_{msa} \qquad \mathbf{U}_{msa} \in \mathbb{R}^{k\cdot D_h \times D}$$

“An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale,” Dosovitskiy et al. ICLR 2021 https://arxiv.org/pdf/2010.11929.pdf

Association of words to image regions

A woman is throwing a frisbee in a park.

A dog is standing on a hardwood floor.

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

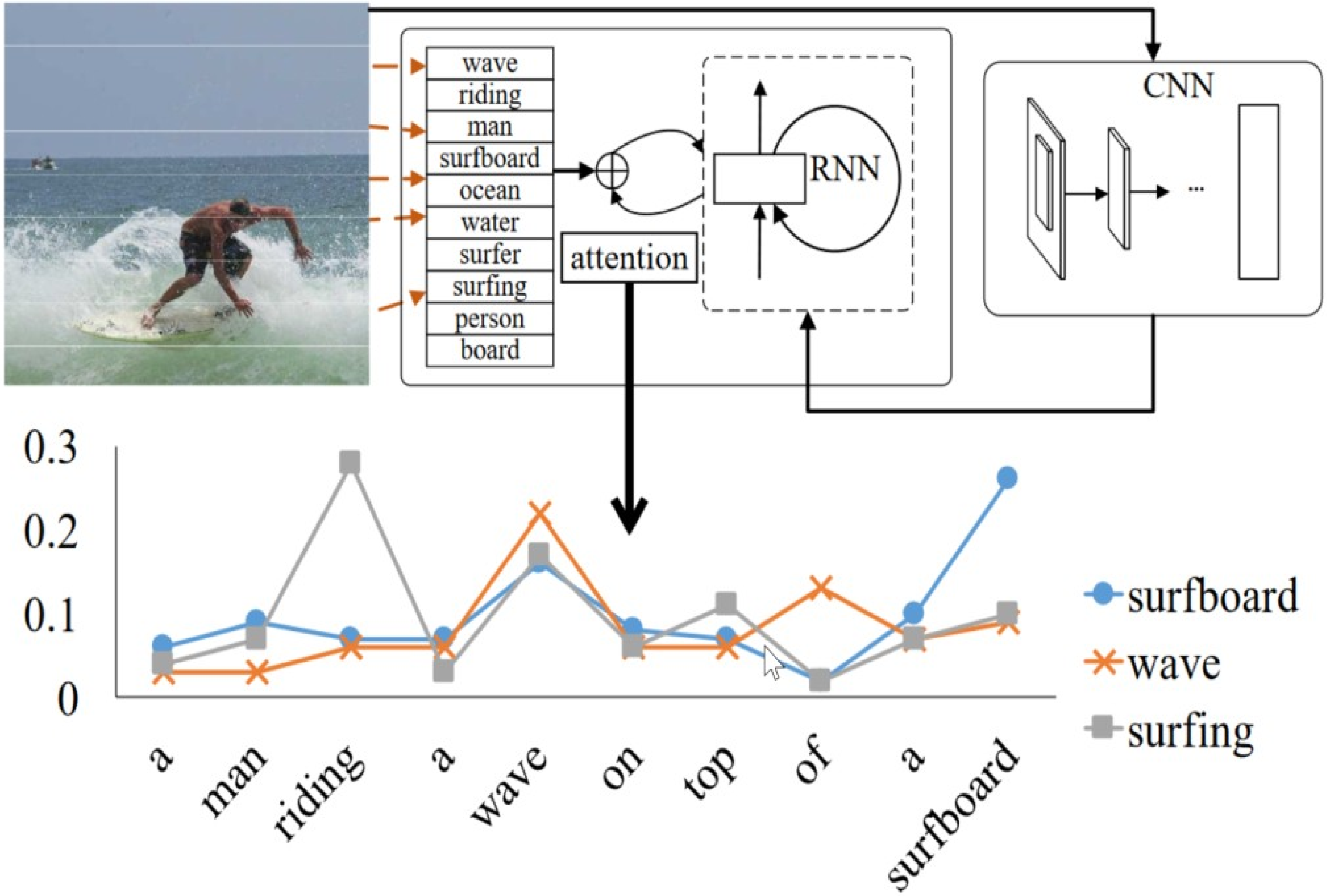
A little girl sitting on a bed with a teddy bear.

A group of people sitting on a boat in the water.

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

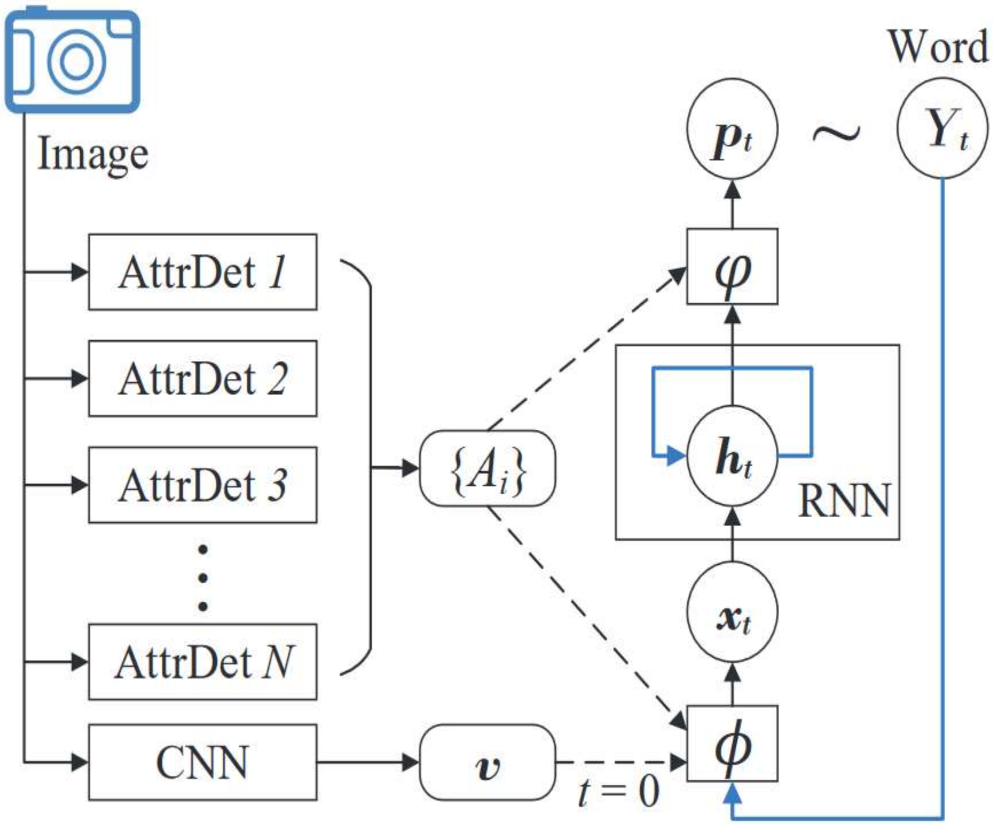
Source: Xu, Kelvin, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. "Show, attend and tell: Neural image caption generation with visual attention." In *International conference on machine learning*, pp. 2048-2057. 2015.

Injecting attribute extraction into the attention process



Source: You Q, Jin H, Wang Z, Fang C, Luo J. Image captioning with semantic attention. InProceedings of the IEEE conference on computer vision and pattern recognition 2016 (pp. 4651-4659).

Injecting attribute extraction into the attention process

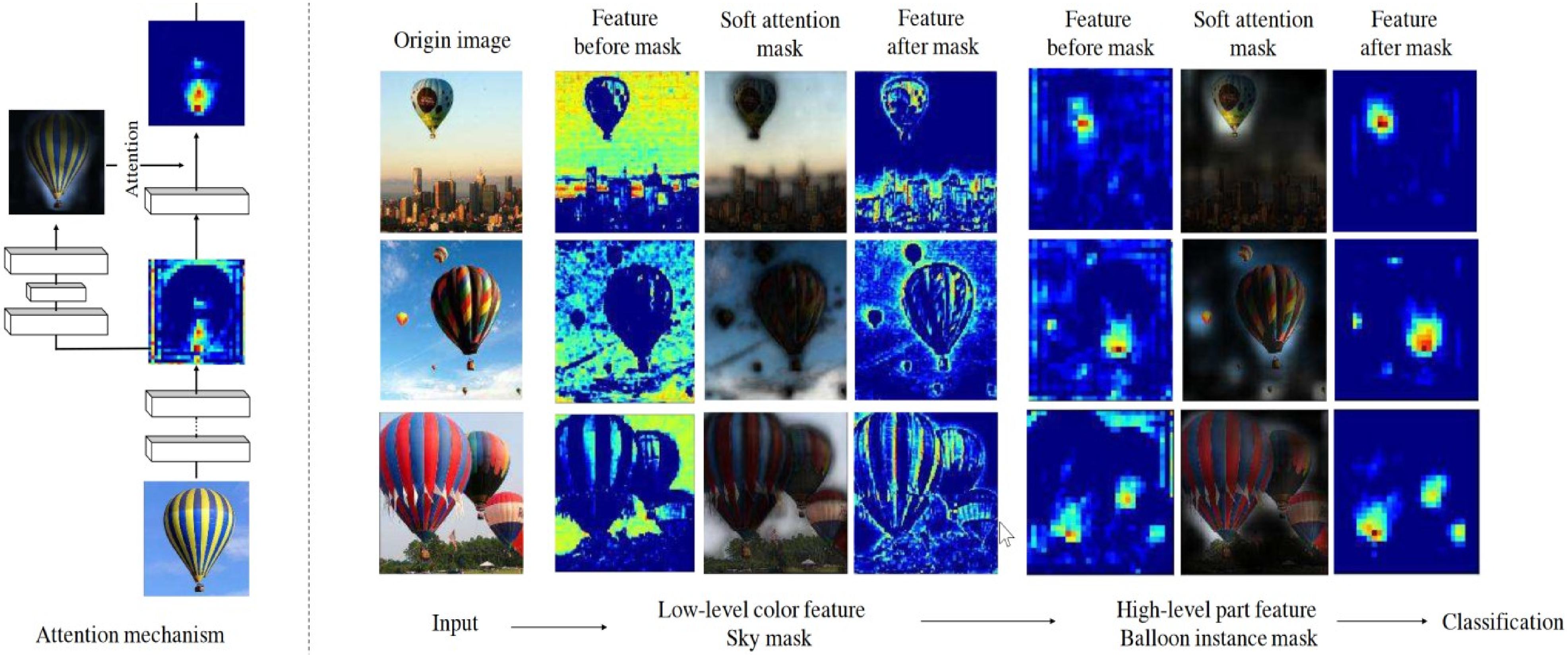


Source: You Q, Jin H, Wang Z, Fang C, Luo J. Image captioning with semantic attention. InProceedings of the IEEE conference on computer vision and pattern recognition 2016 (pp. 4651-4659).

Outline

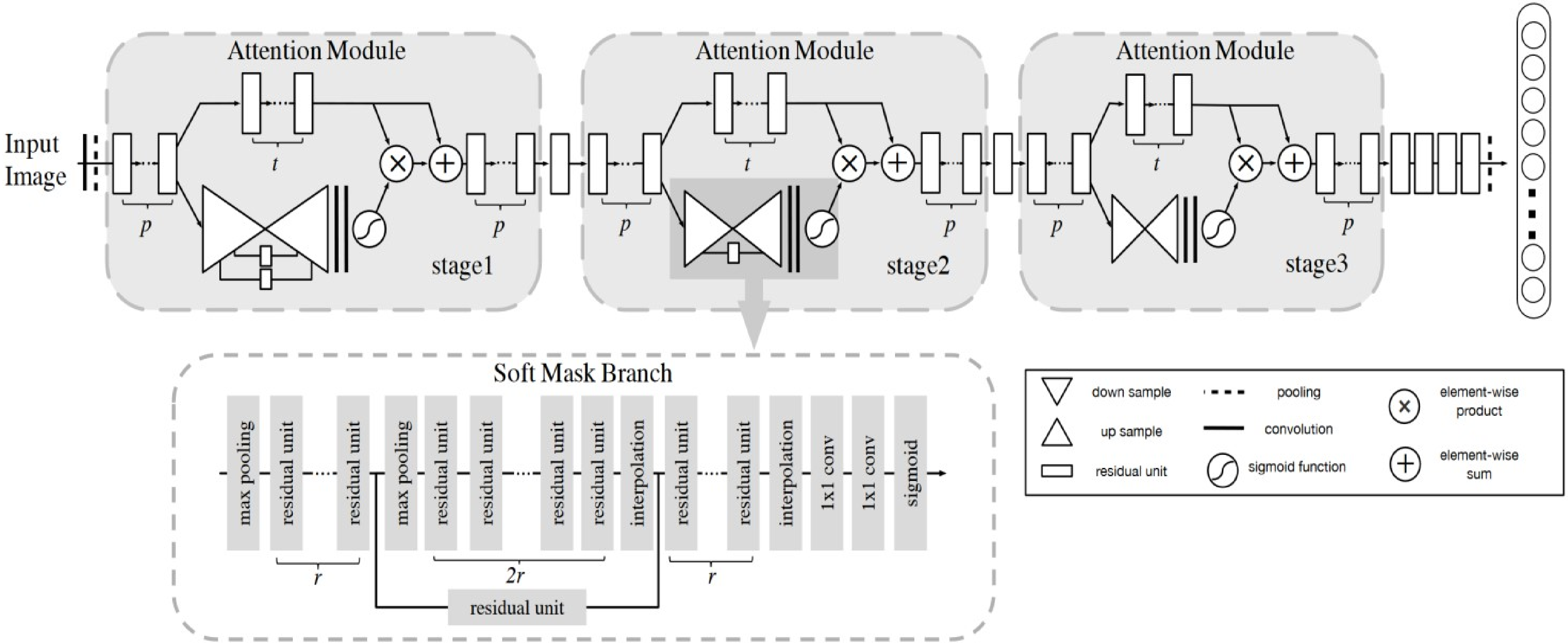
- Attention in NLP
- **Attention in vision**
  - Attention for image captioning
  - **Attention for image recognition**
  - Attention for segmentation and detection
  - Attention in other vision applications

Residual attention network



Source: Wang F, Jiang M, Qian C, Yang S, Li C, Zhang H, Wang X, Tang X. Residual attention network for image classification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition 2017 (pp. 3156-3164).

Two branches of residual attention network



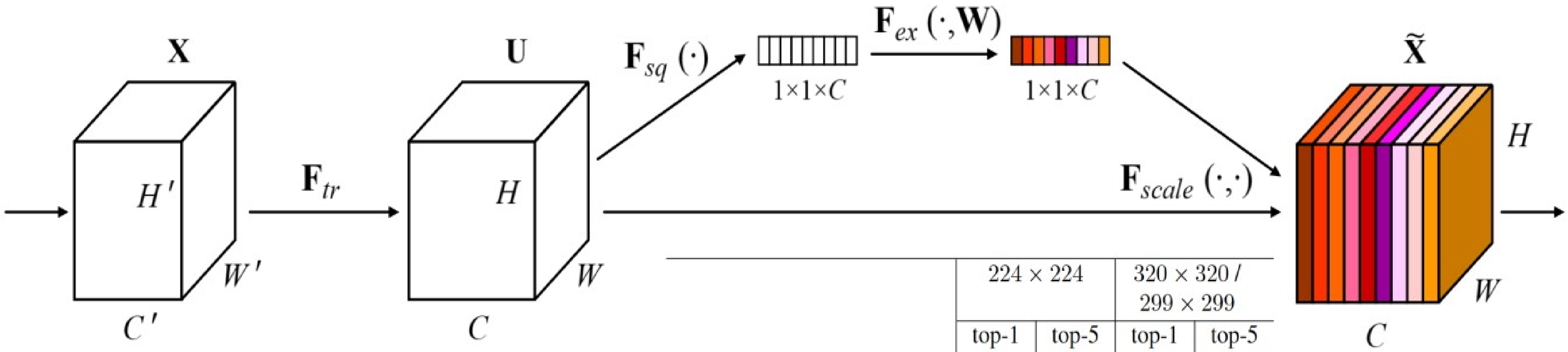
Code snippet (GitHub: koichiro11 / residual-attention-network):

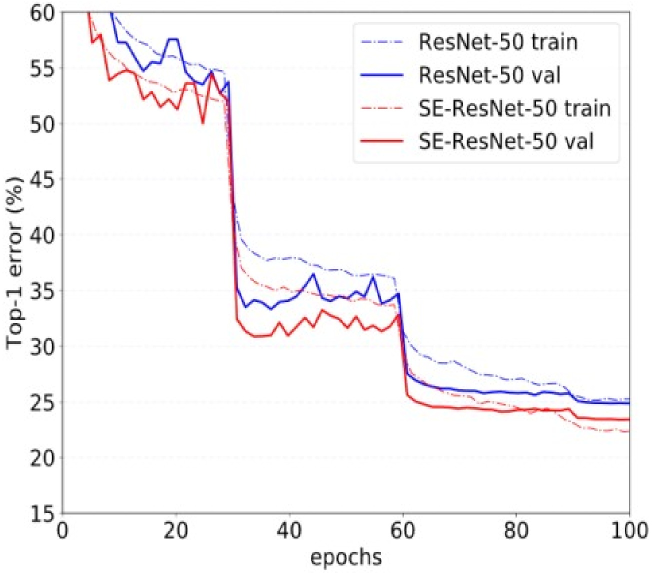
```
with tf.variable_scope("attention"):
 output = (1 + output_soft_mask) * output_trunk
```

```
with tf.variable_scope("output"):
 output_soft_mask = tf.layers.conv2d…
 output_soft_mask = tf.nn.sigmoid(output_soft_mask)
```

Paper: Wang F, Jiang M, Qian C, Yang S, Li C, Zhang H, Wang X, Tang X. Residual attention network for image classification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition 2017 (pp. 3156-3164).

Squeeze and excitation networks

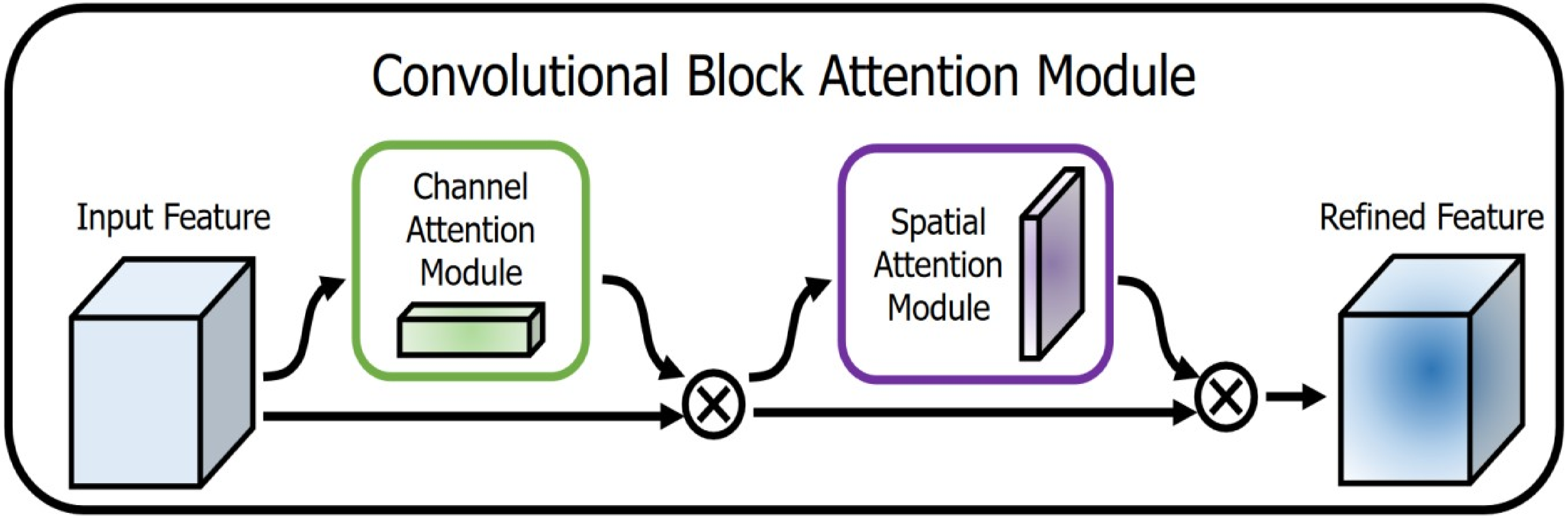




	224 × 224		320 × 320 / 299 × 299	
	top-1 err.	top-5 err.	top-1 err.	top-5 err.
ResNet-152 [10]	23.0	6.7	21.3	5.5
ResNet-200 [11]	21.7	5.8	20.1	4.8
Inception-v3 [44]	-	-	21.2	5.6
Inception-v4 [42]	-	-	20.0	5.0
Inception-ResNet-v2 [42]	-	-	19.9	4.9
ResNeXt-101 (64 × 4d) [47]	20.4	5.3	19.1	4.4
DenseNet-264 [14]	22.15	6.12	-	-
Attention-92 [46]	-	-	19.5	4.8
Very Deep PolyNet [51] †	-	-	18.71	4.25
PyramidNet-200 [8]	20.1	5.4	19.2	4.7
DPN-131 [5]	19.93	5.12	18.55	4.16
SENet-154	**18.68**	**4.47**	**17.28**	**3.79**
NASNet-A (6@4032) [55] †	-	-	17.3‡	3.8‡
SENet-154 (post-challenge)	-	-	**16.88**‡	**3.58**‡

Source: Hu J, Shen L, Sun G. Squeeze-and-excitation networks. InProceedings of the IEEE conference on computer vision and pattern recognition 2018 (pp. 7132-7141).

Convolutional block attention module

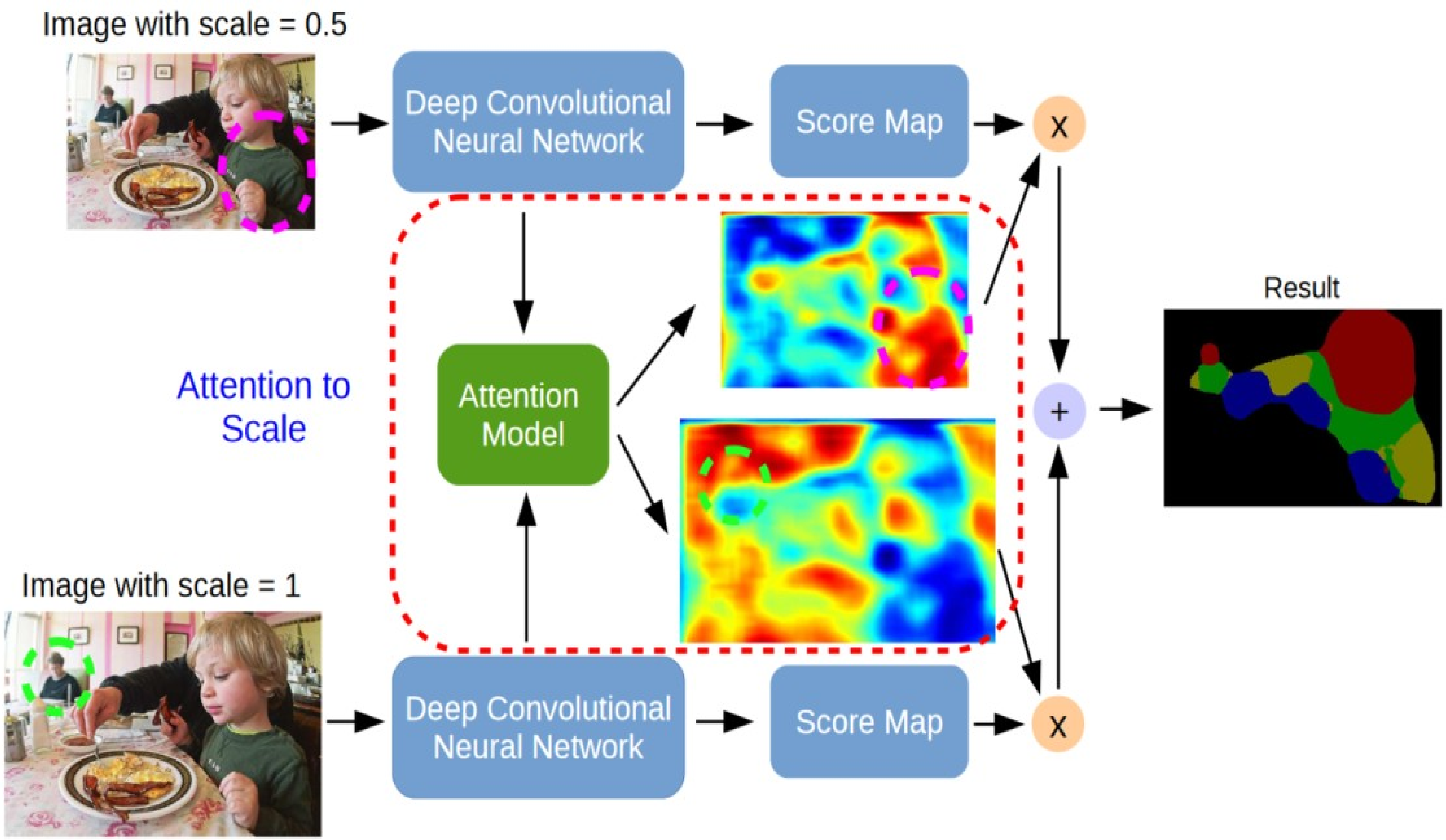


Source: Woo S, Park J, Lee JY, So Kweon I. Cbam: Convolutional block attention module. InProceedings of the European Conference on Computer Vision (ECCV) 2018 (pp. 3-19).

Outline

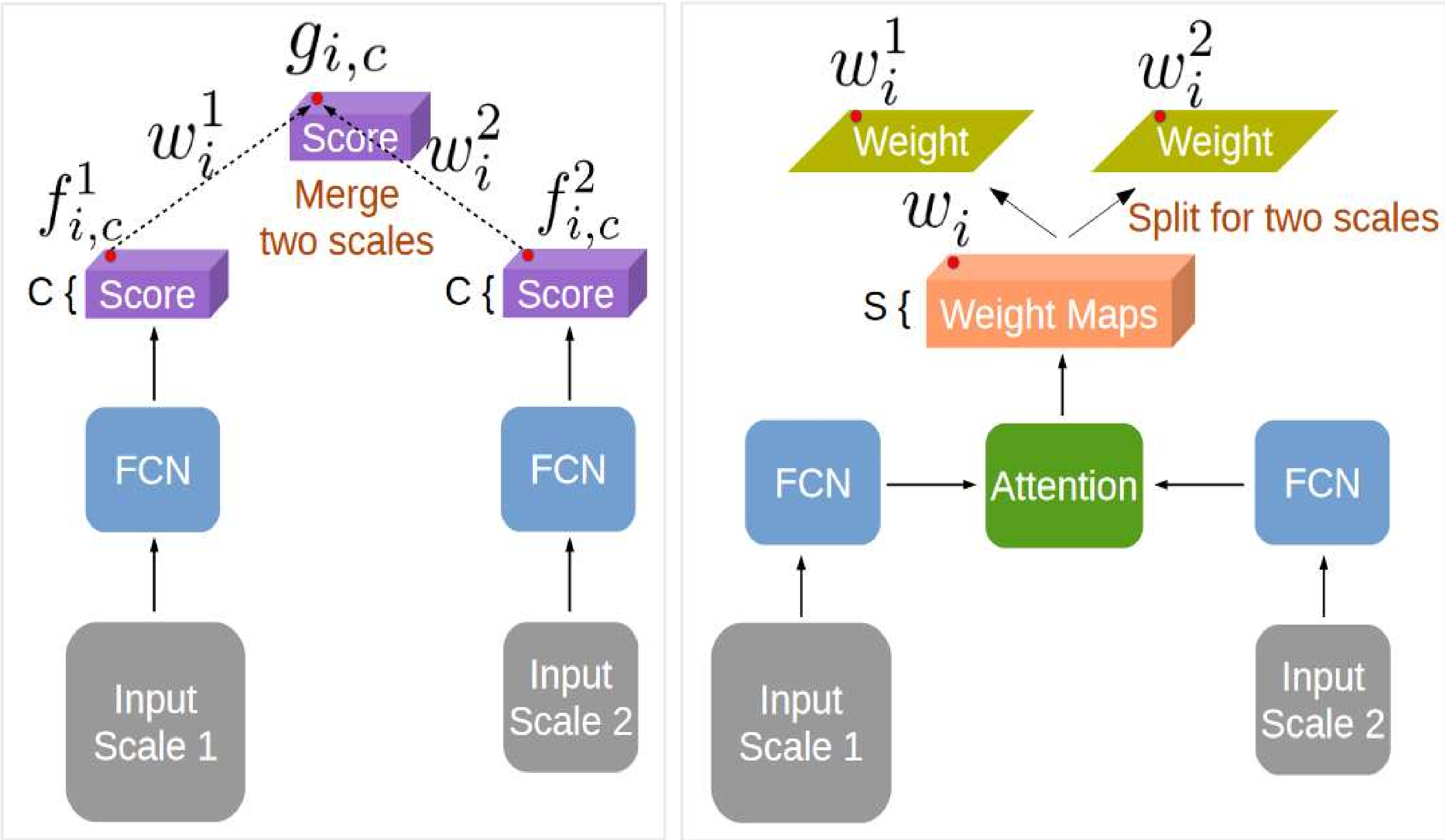
- Attention in NLP
- **Attention in vision**
  - Attention for image captioning
  - Attention for image recognition
  - **Attention for segmentation and detection**
  - Attention in other vision applications

Multi-scale attention for segmentation



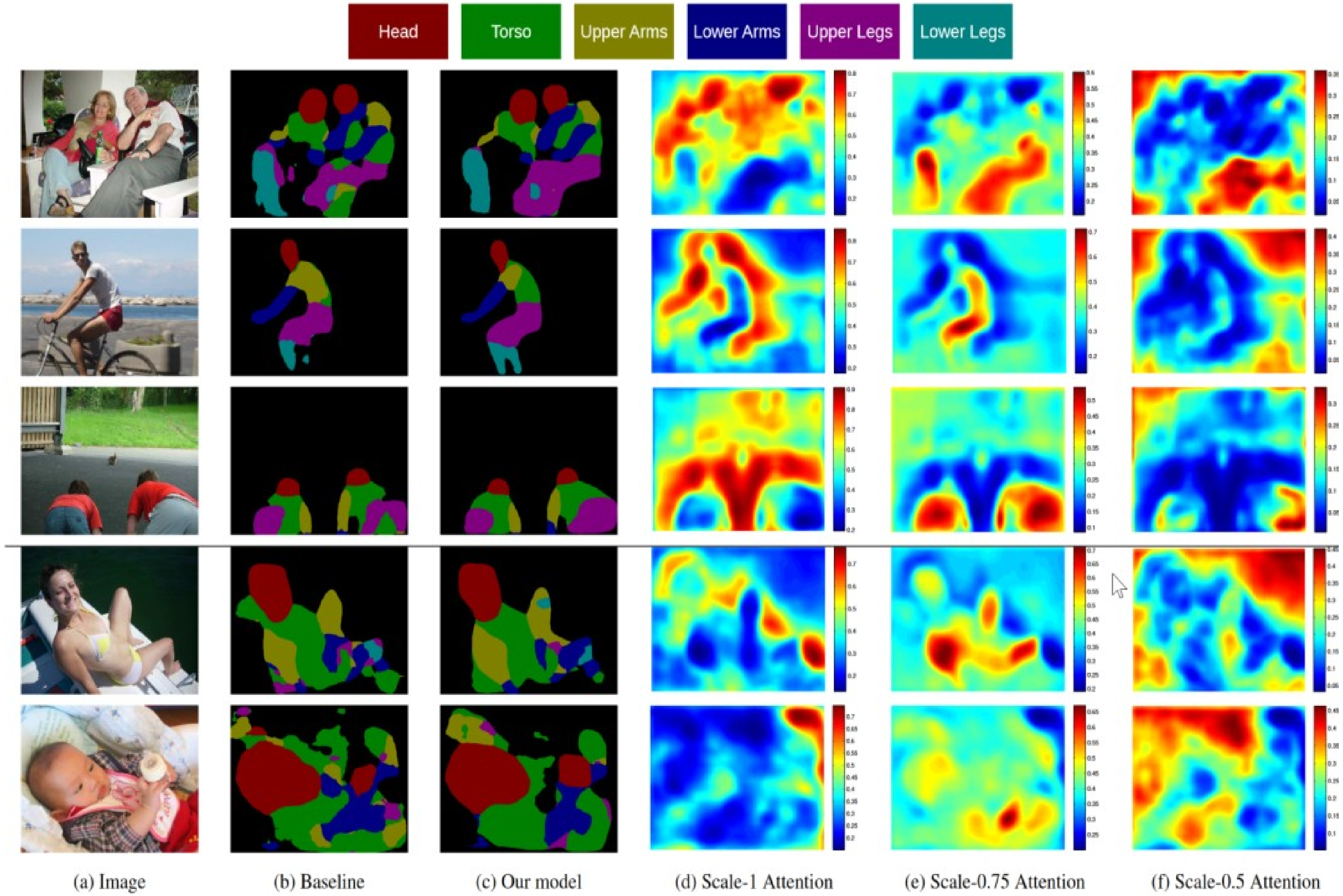
Source: Chen LC, Yang Y, Wang J, Xu W, Yuille AL. Attention to scale: Scale-aware semantic image segmentation. InProceedings of the IEEE conference on computer vision and pattern recognition 2016 (pp. 3640-3649).

Multi-scale attention for segmentation



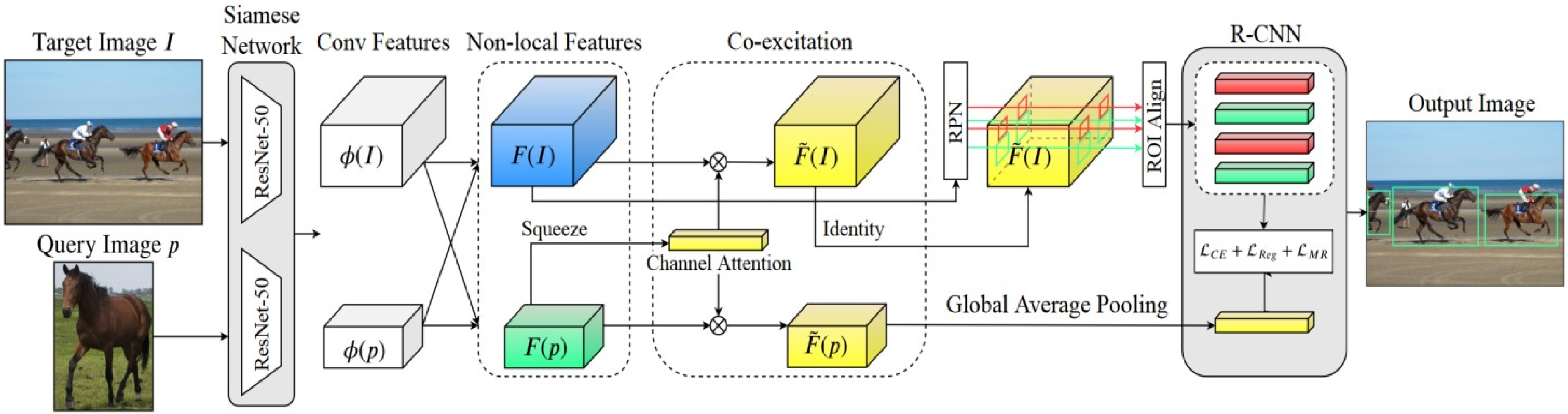
Source: Chen LC, Yang Y, Wang J, Xu W, Yuille AL. Attention to scale: Scale-aware semantic image segmentation. InProceedings of the IEEE conference on computer vision and pattern recognition 2016 (pp. 3640-3649).

Multi-scale attention for segmentation



Source: Chen LC, Yang Y, Wang J, Xu W, Yuille AL. Attention to scale: Scale-aware semantic image segmentation. InProceedings of the IEEE conference on computer vision and pattern recognition 2016 (pp. 3640-3649).

One-shot object detection

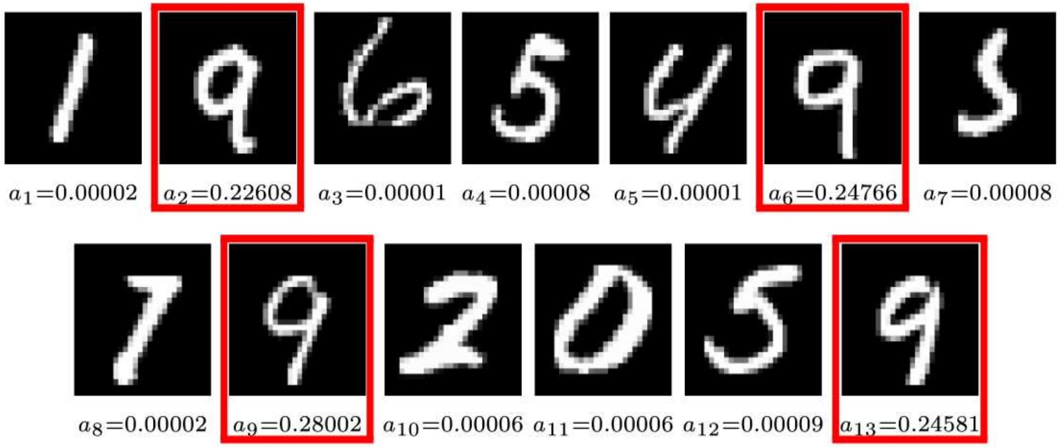


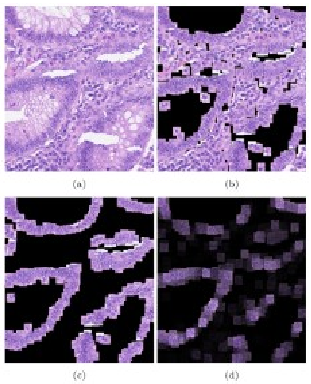
Source: Hsieh TI, Lo YC, Chen HT, Liu TL. One-Shot Object Detection with Co-Attention and Co-Excitation. InAdvances in Neural Information Processing Systems 2019 (pp. 2721-2730).

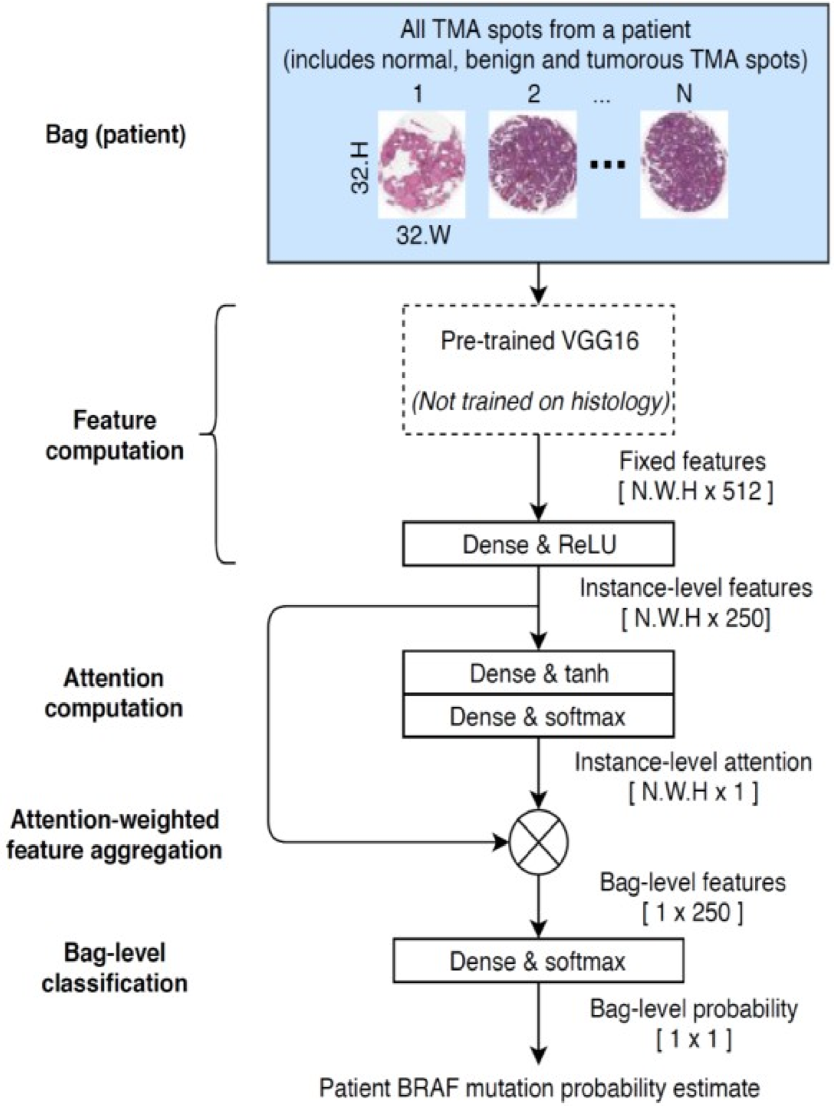
Outline

- Attention in NLP
- **Attention in vision**
  - Attention for image captioning
  - Attention for image recognition
  - Attention for segmentation and detection
  - **Attention in other vision applications**

Attention-based Multiple Instance Learning







Source: Ilse M, Tomczak JM, Welling M. Attention-based deep multiple instance learning. In 35th International Conference on Machine Learning, ICML 2018 2018 Jan 1 (pp. 3376-3391).
And, adapted in unpublished work by Deepak Anand, Kumar Yashashwi, Neeraj Kumar, Swapnil Rane, Peter Gann, and Amit Sethi