Neural Image Caption Generation with Visual Attention

Kelvin Xu, Jimmy Lei Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville,
Ruslan Salakhutdinov, Richard S. Zemel, Yoshua Bengio

Literature Review
Deep Learning - DPL302m

Today's Date, 2024

Paper Overview

- Introduction
- Related Work
- Image Caption Generation with Attention Mechanism
  - Encoder: Convolutional Features
  - Decoder: LSTM Network
- Learning Stochastic “Hard” vs Deterministic “Soft” Attention
  - Stochastic “Hard” Attention
  - Deterministic “Soft” Attention
- Training & Evaluation
- Conclusion

Content Overview

- Introduction: Why Attention?
- Challenges
- Model
    - Architecture
    - Attention Mechanism
        - Soft Attention
        - Hard Attention

Introduction

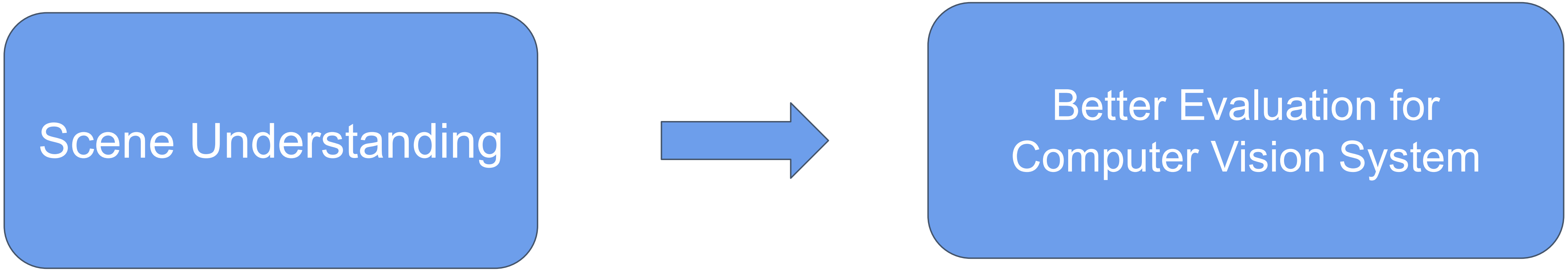
Visual System

Language System

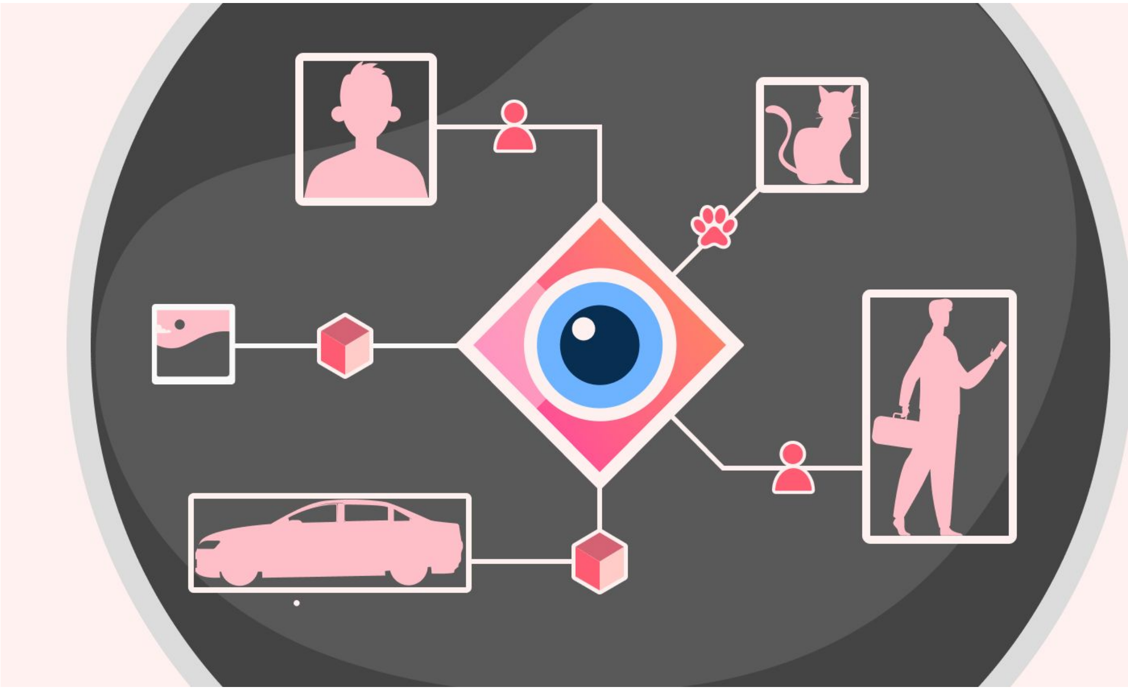
Introduction

Introduction

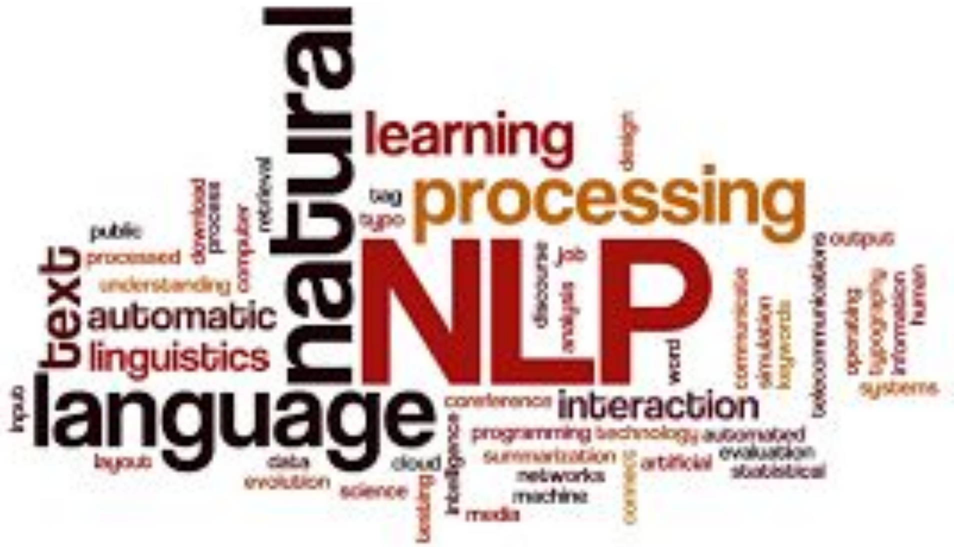
Introduction



Introduction



Computer Vision



Natural Language Processing

Introduction

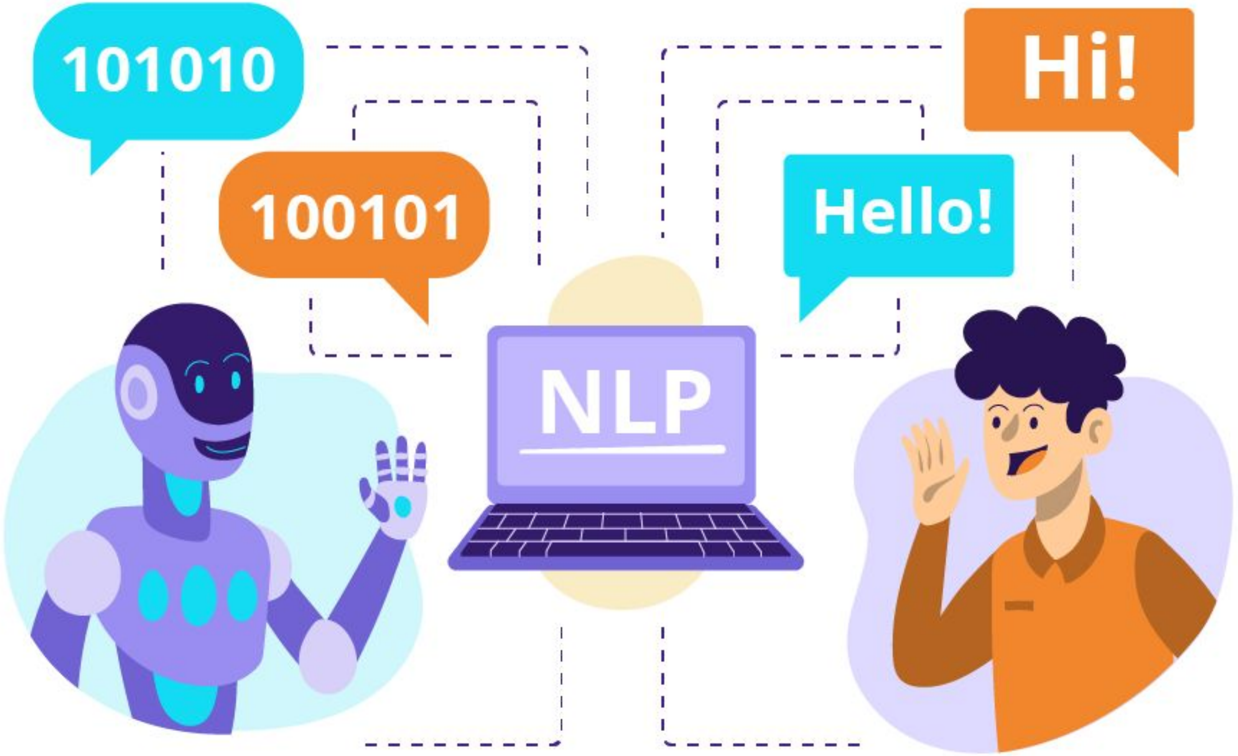
Object Recognition

The most IMPORTANT object

Relationship between objects

Introduction

Practical Meaning: AI can finally understand daily scenes and INTERACT with human!



Introduction

But Daily scenes ⇒ Visual Noises!

Challenges

- How to extract visual information from the image

- How to transform the visual information into proper natural language

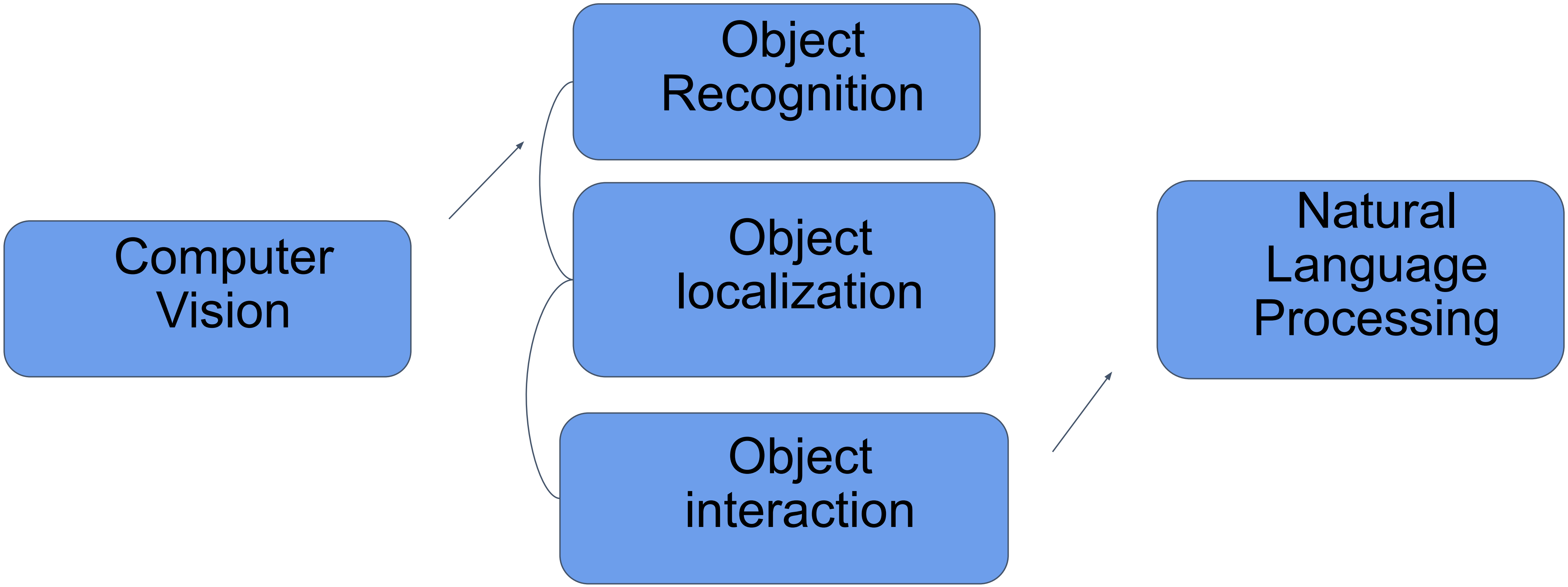
- **How to represent different parts of images and let the model focus on the IMPORTANT part**

Model

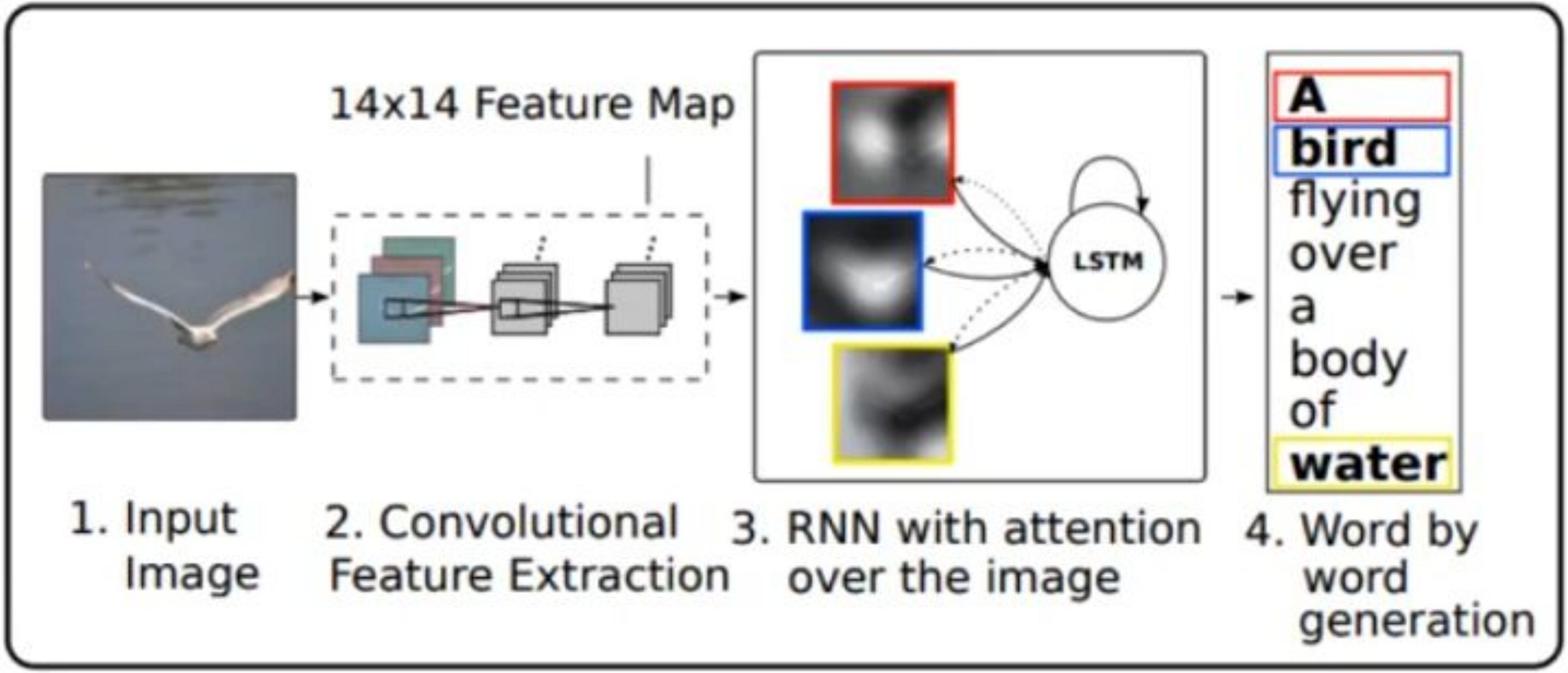
Divide the image caption generator into 2 sub-problems:

- Image understanding
- Descriptive caption generating

Introduction



Architecture



Convolutional Neural Network (CNN)

CNN consists of three types of layers:

- Convolutional layer
- Pooling layer
- Fully Connected layer

In this paper, they use a pre-trained CNN model VGG-Net 16 to do the image understanding.

NLP Part - Dependency between output

In NLP part, each output is a word in the caption. Usually the words in the one caption are highly related to each other

A <u>girl</u> holding <u>her</u> teddy bear

Long-short Term Memory Network (LSTM)

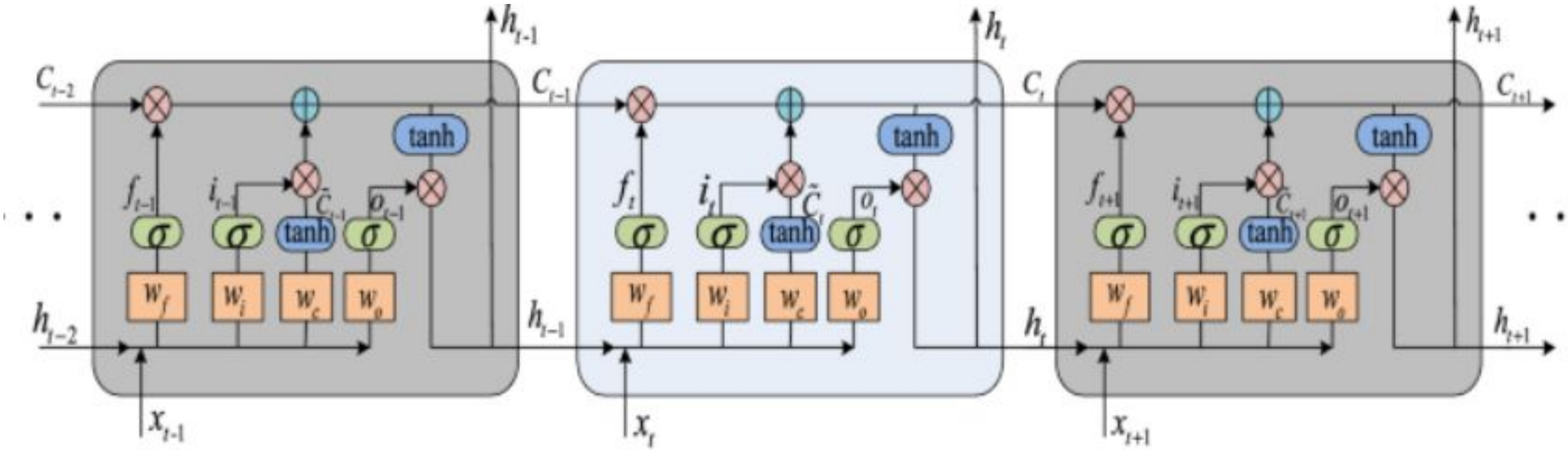
Problem: The output produced by most neural networks are independent

Solution:

Long-short Term memory network (LSTM)

A girl holding her teddy bear

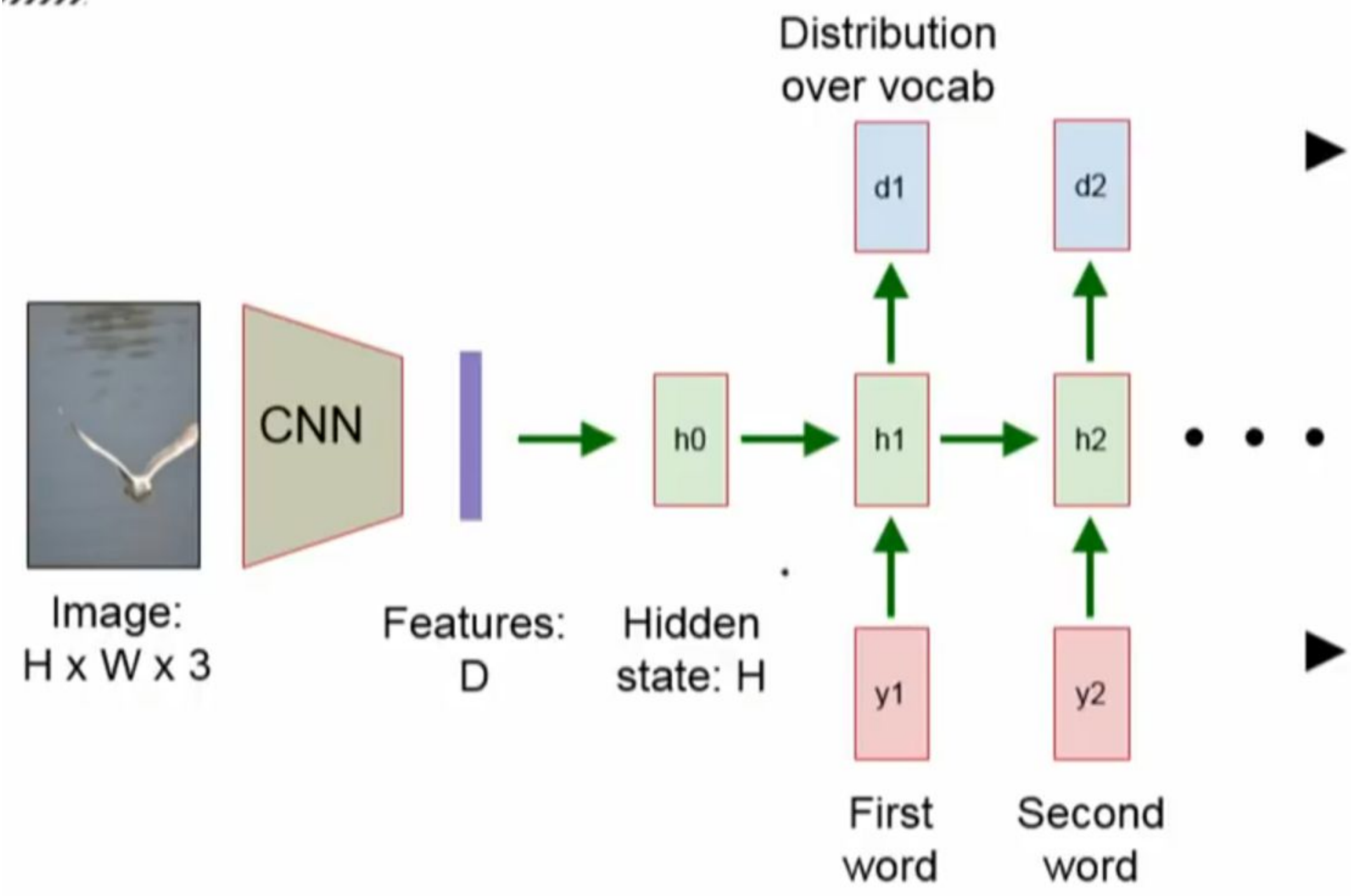
Long-short Term Memory Network (LSTM)



LSTM has a powerful memory mechanism.

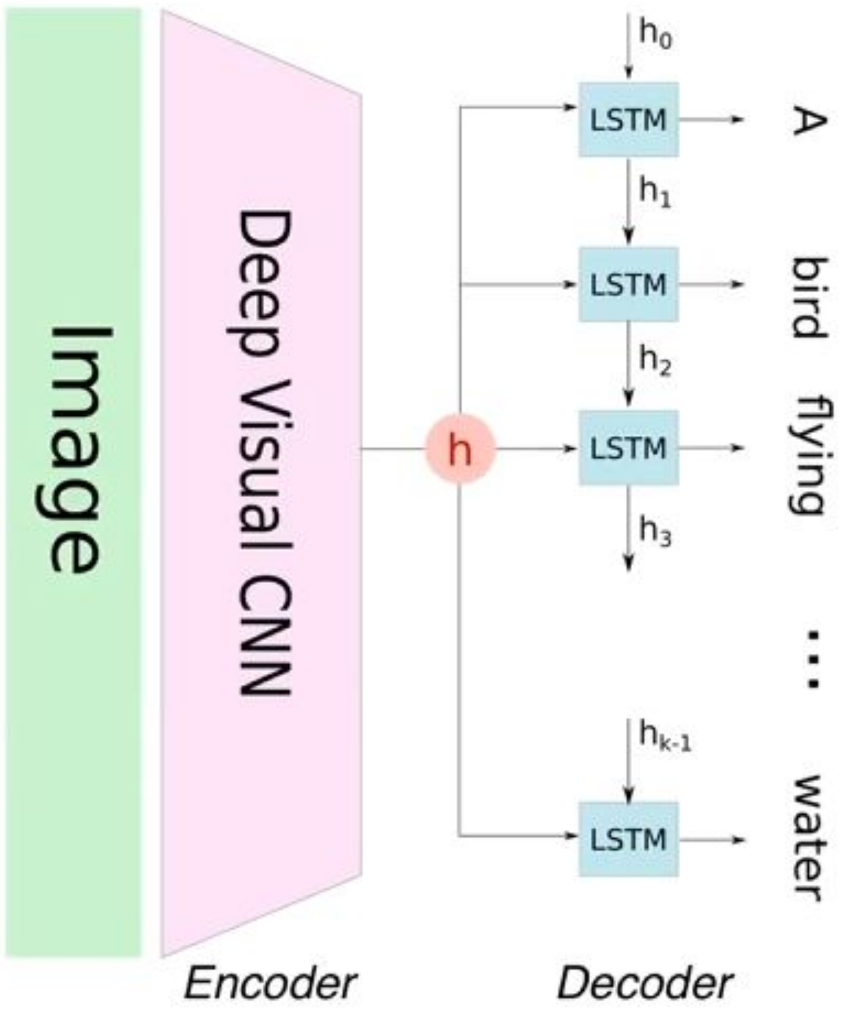
LSTM takes a new input and also inherits the hidden state and memory cell of the previous step when producing a new output.

Attention Mechanism

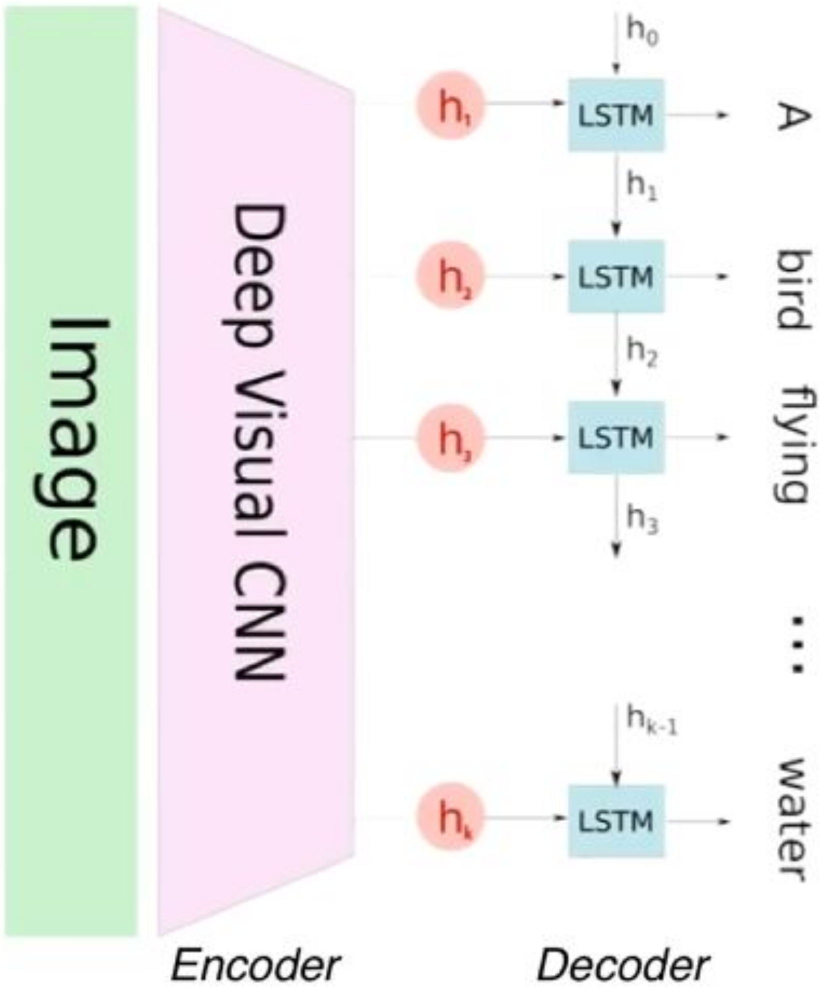


- RNN looks at the whole image only once
- What if RNN looks at different parts of the image at each time step?

Model without Attention



Model with Attention

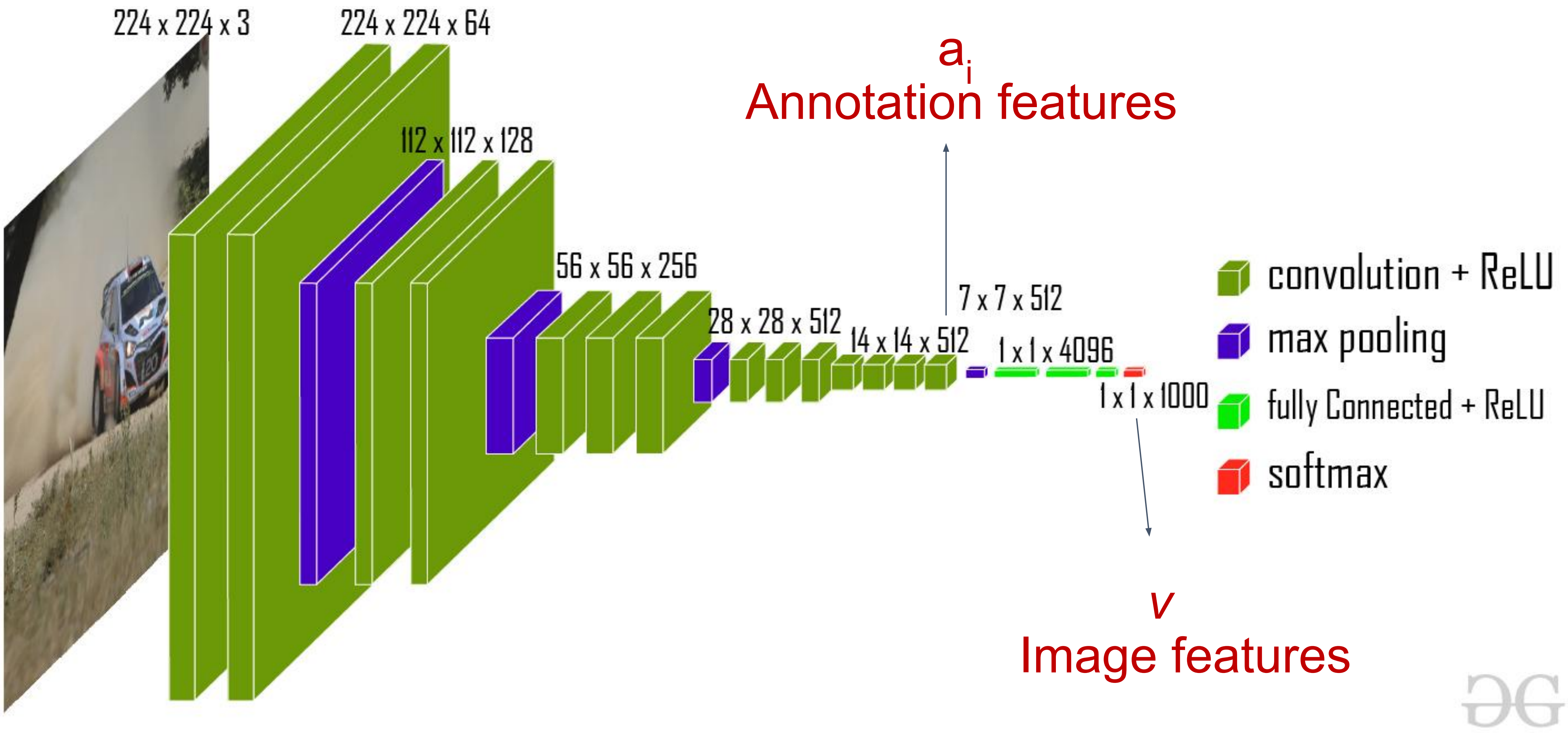


3.1.1. Encoder: Convolutional Features

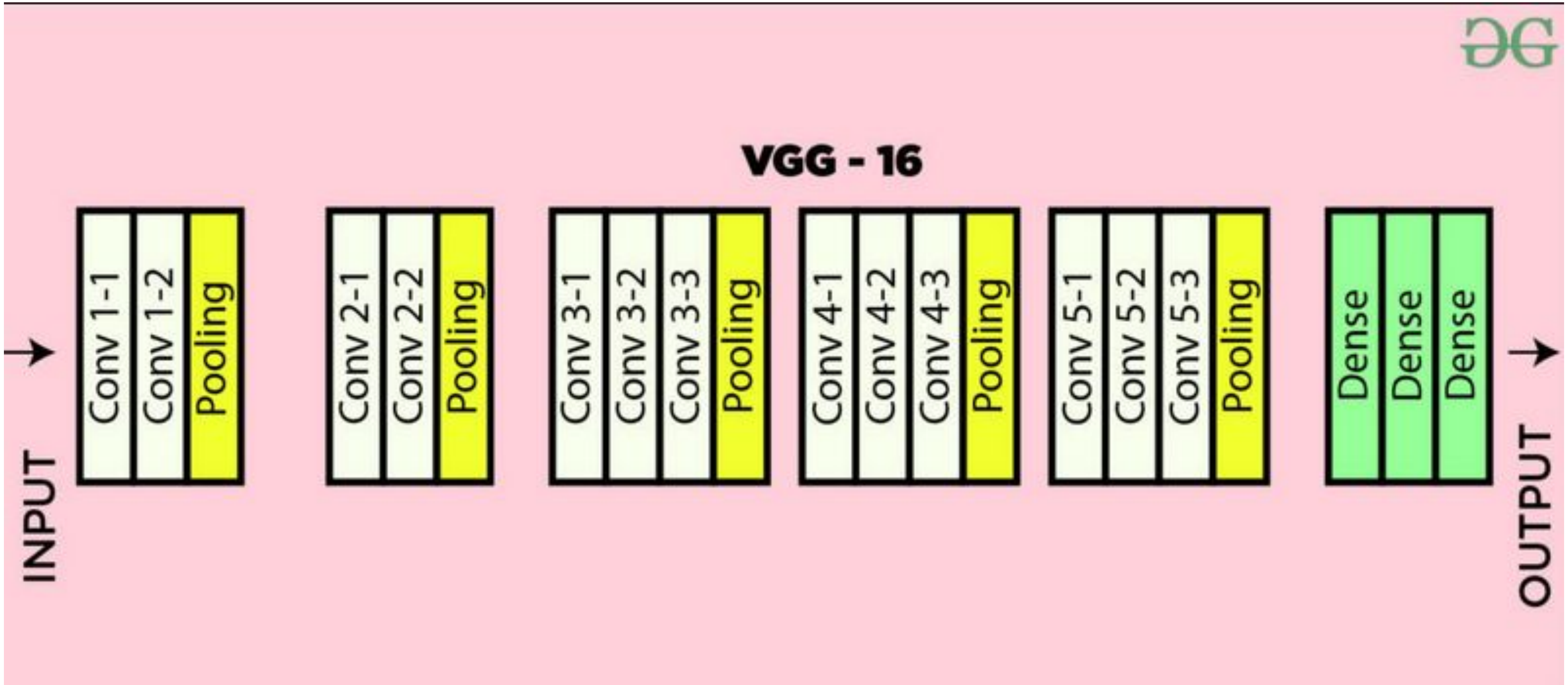
We use a convolutional neural network in order to extract a set of feature vectors which we refer to as annotation vectors. The extractor produces L vectors, each of which is a D-dimensional representation corresponding to a part of the image.

$$a = \{\mathbf{a}_1, \ldots, \mathbf{a}_L\}, \ \mathbf{a}_i \in \mathbb{R}^D$$

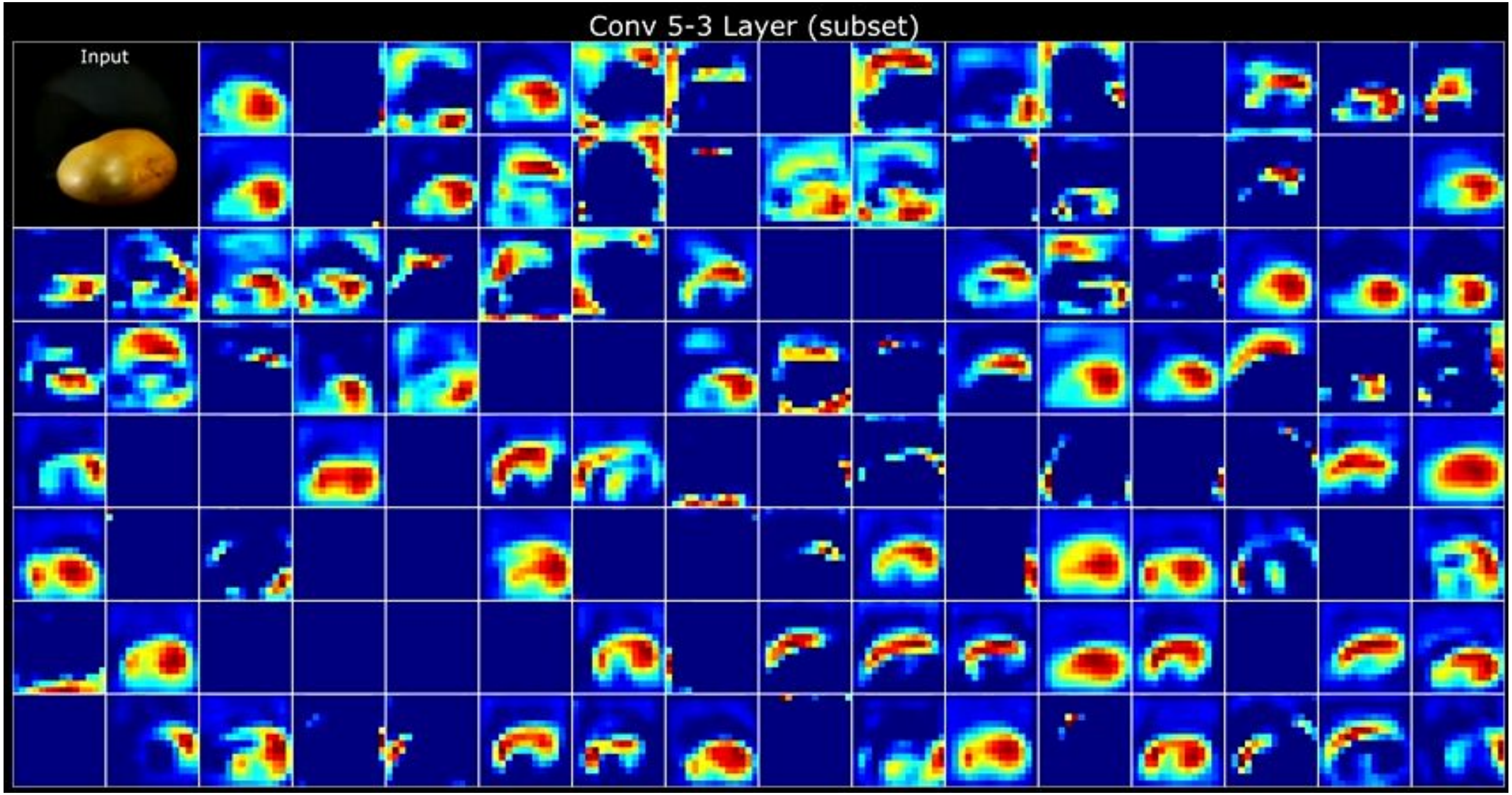
Annotation features



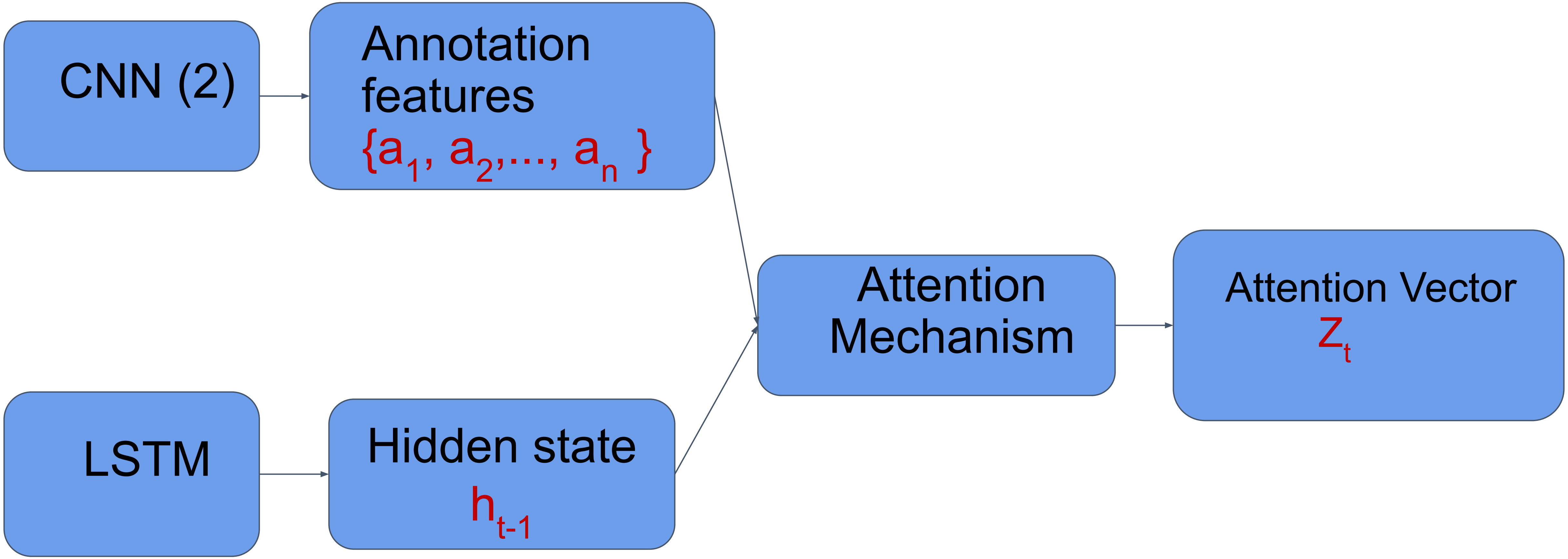
Annotation features



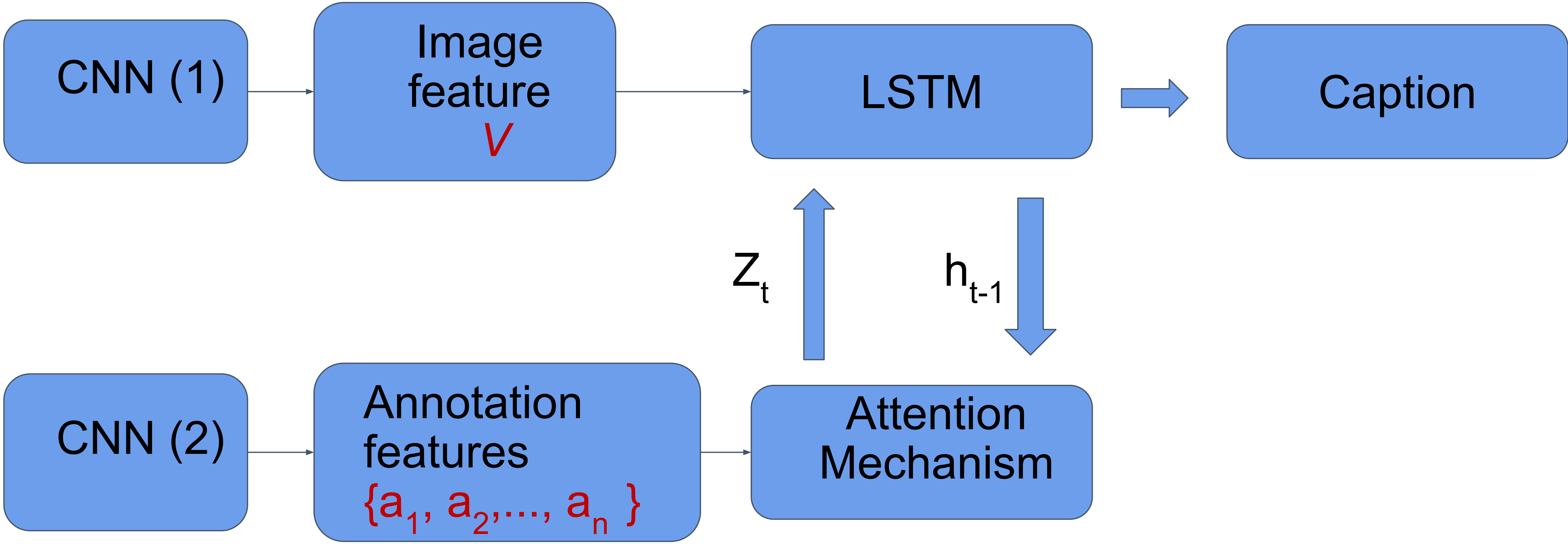
Annotation features



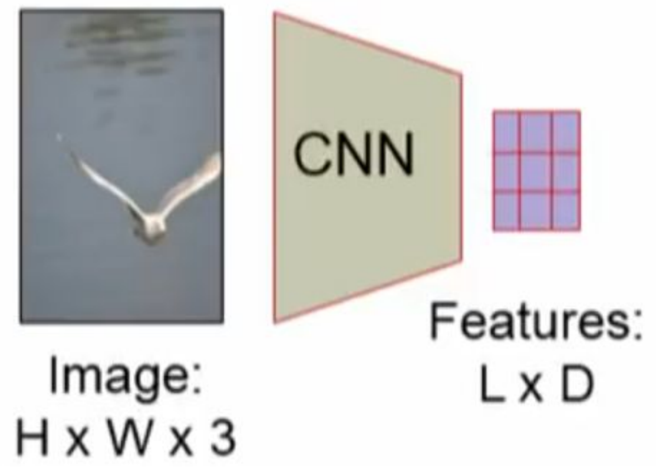
Attention Part



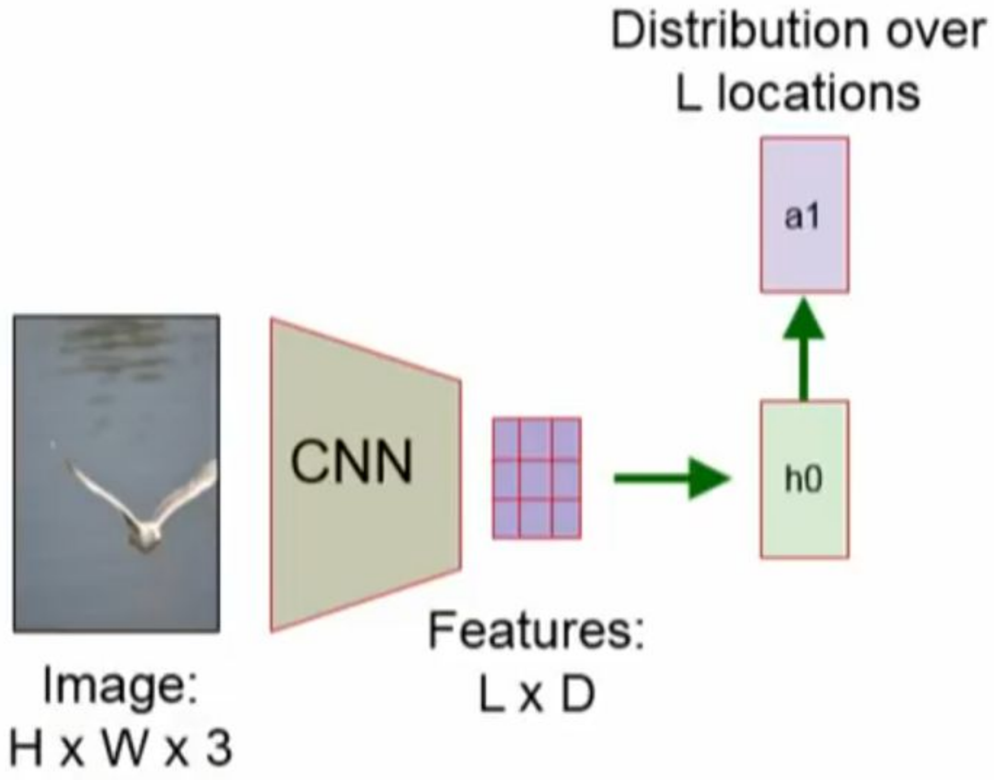
Model with attention



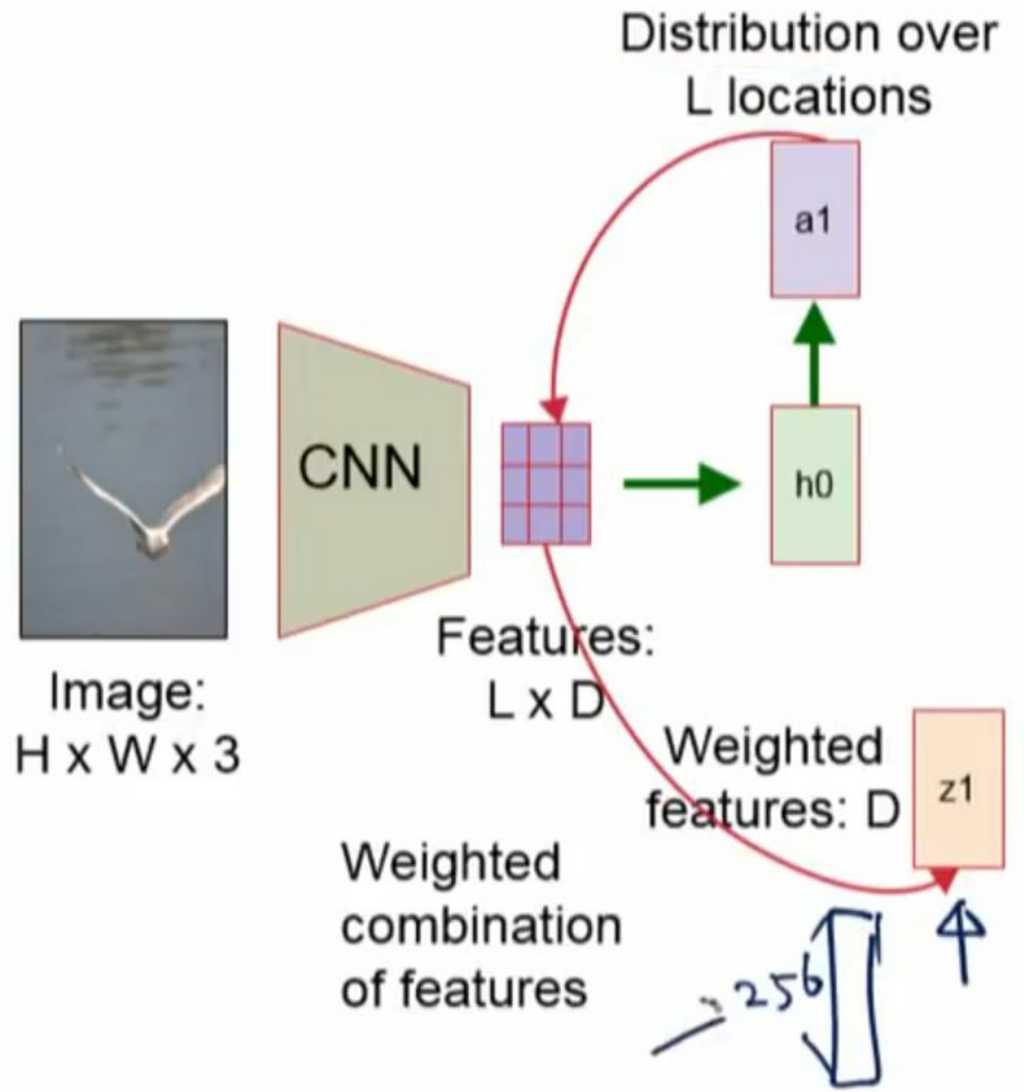
Attention Mechanism



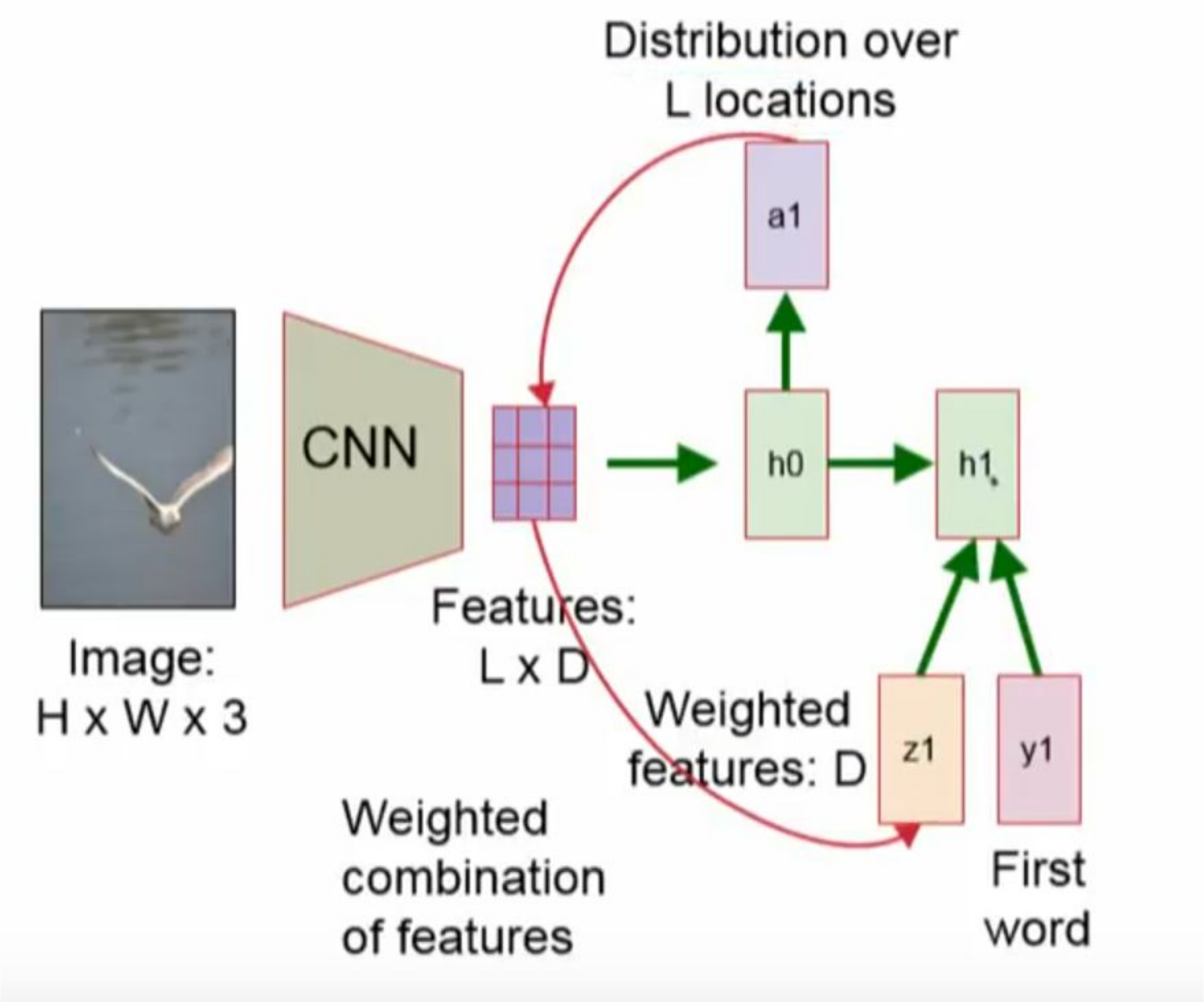
Attention Mechanism



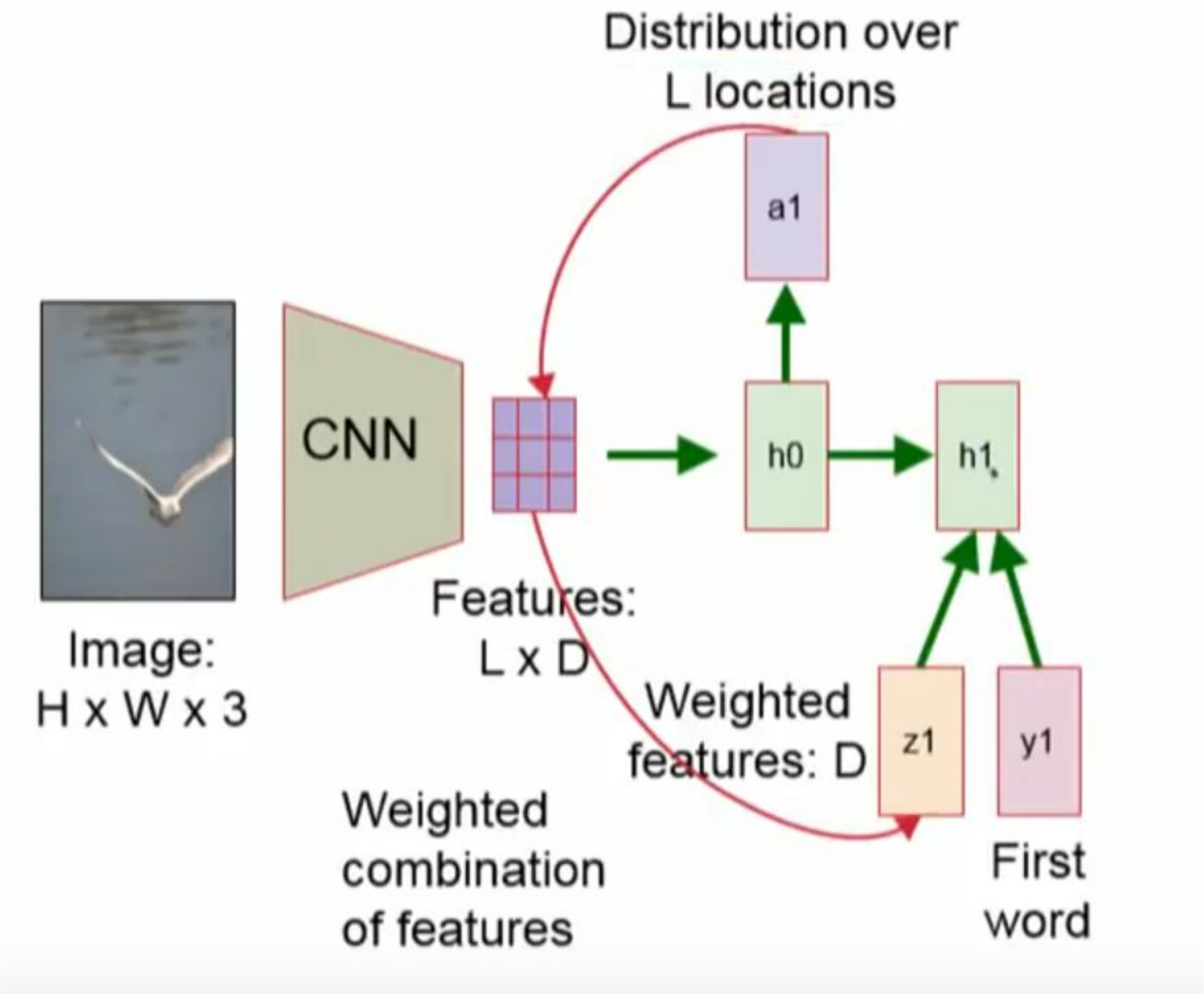
Attention Mechanism



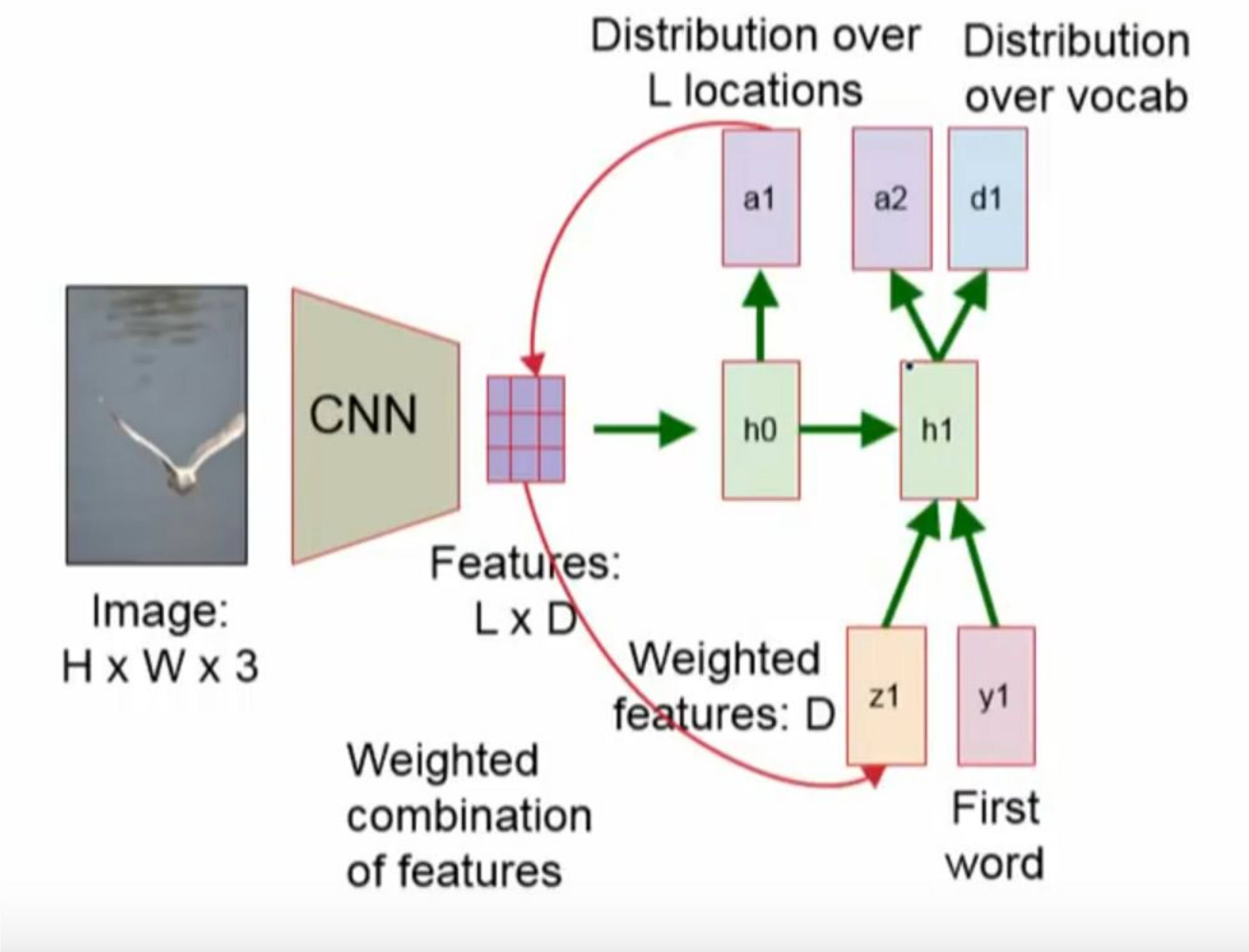
Attention Mechanism



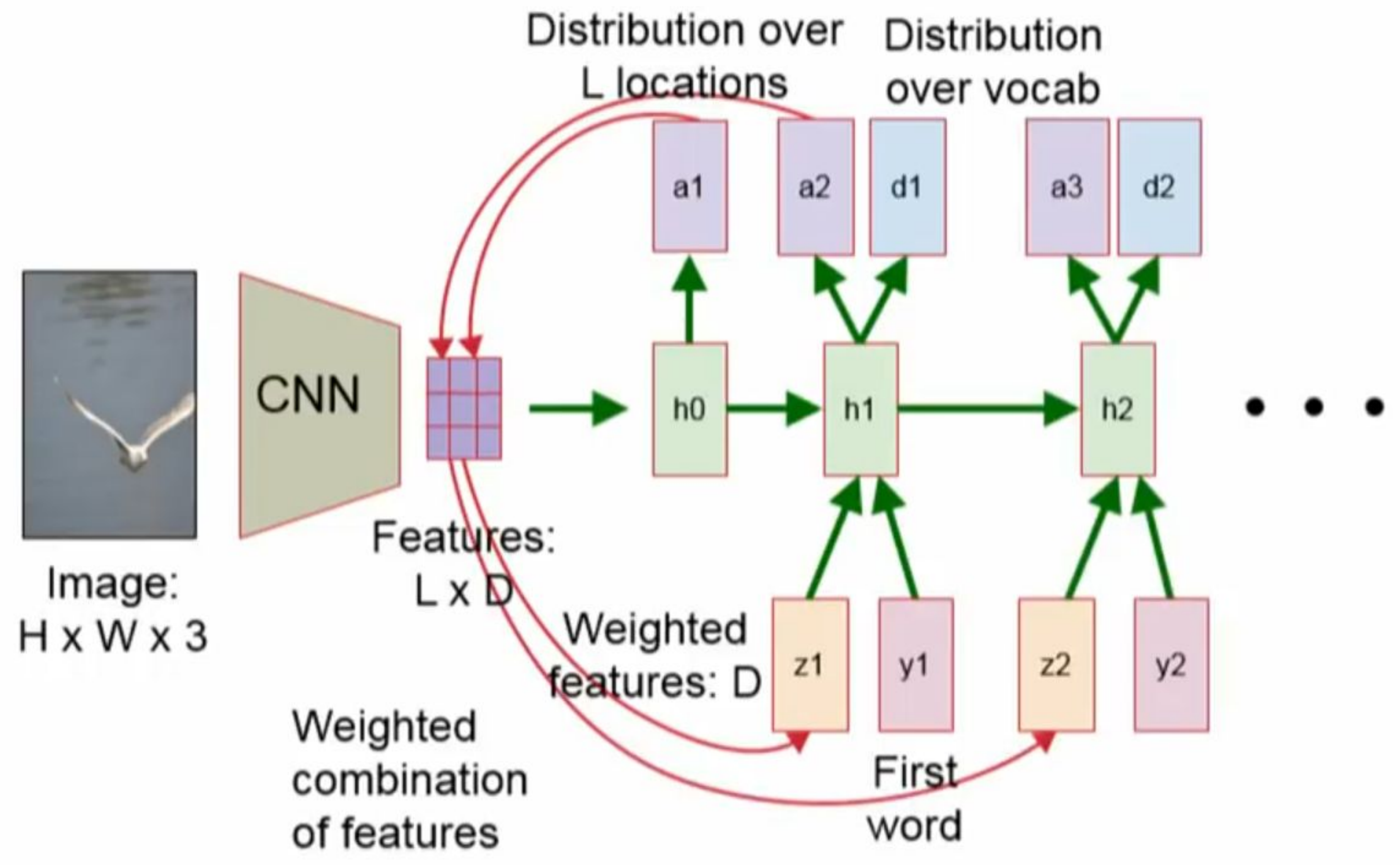
Attention Mechanism



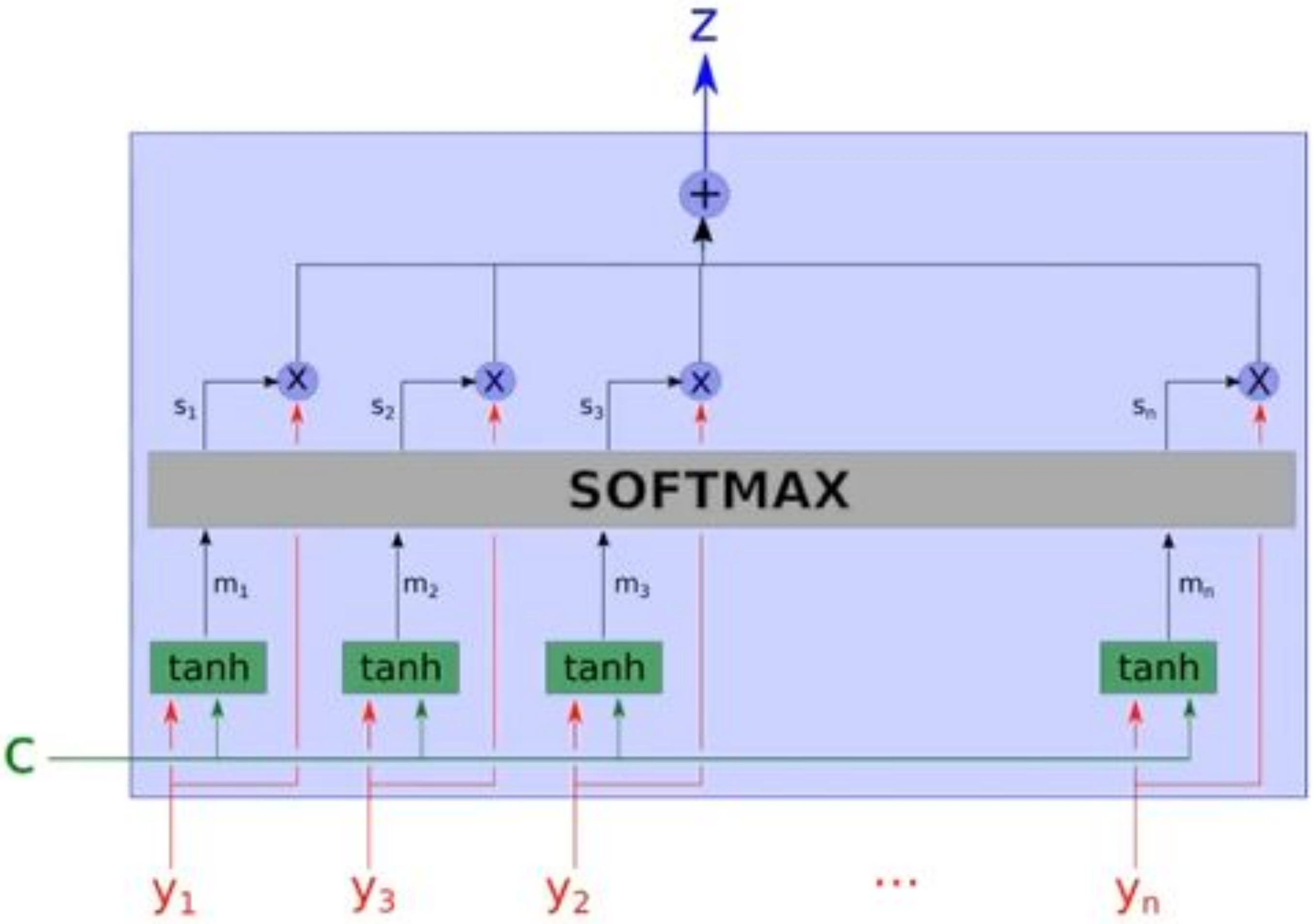
Attention Mechanism



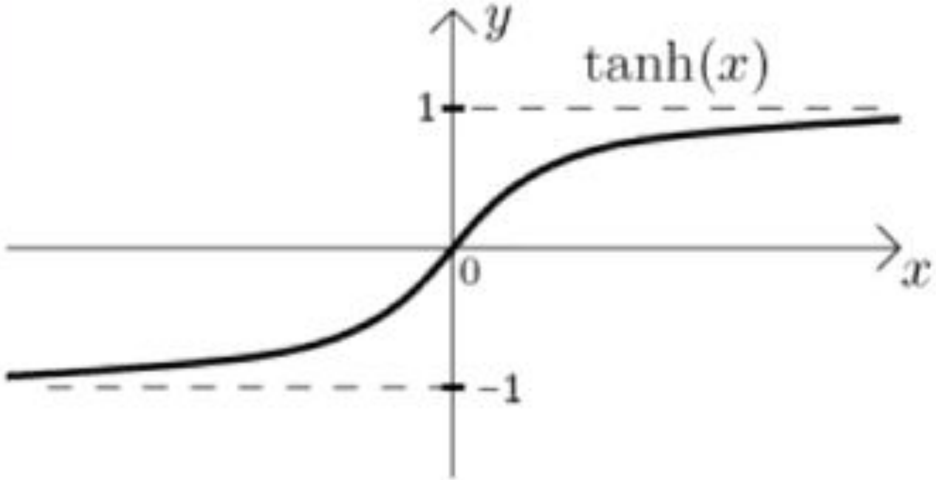
Attention Mechanism



Attention Unit

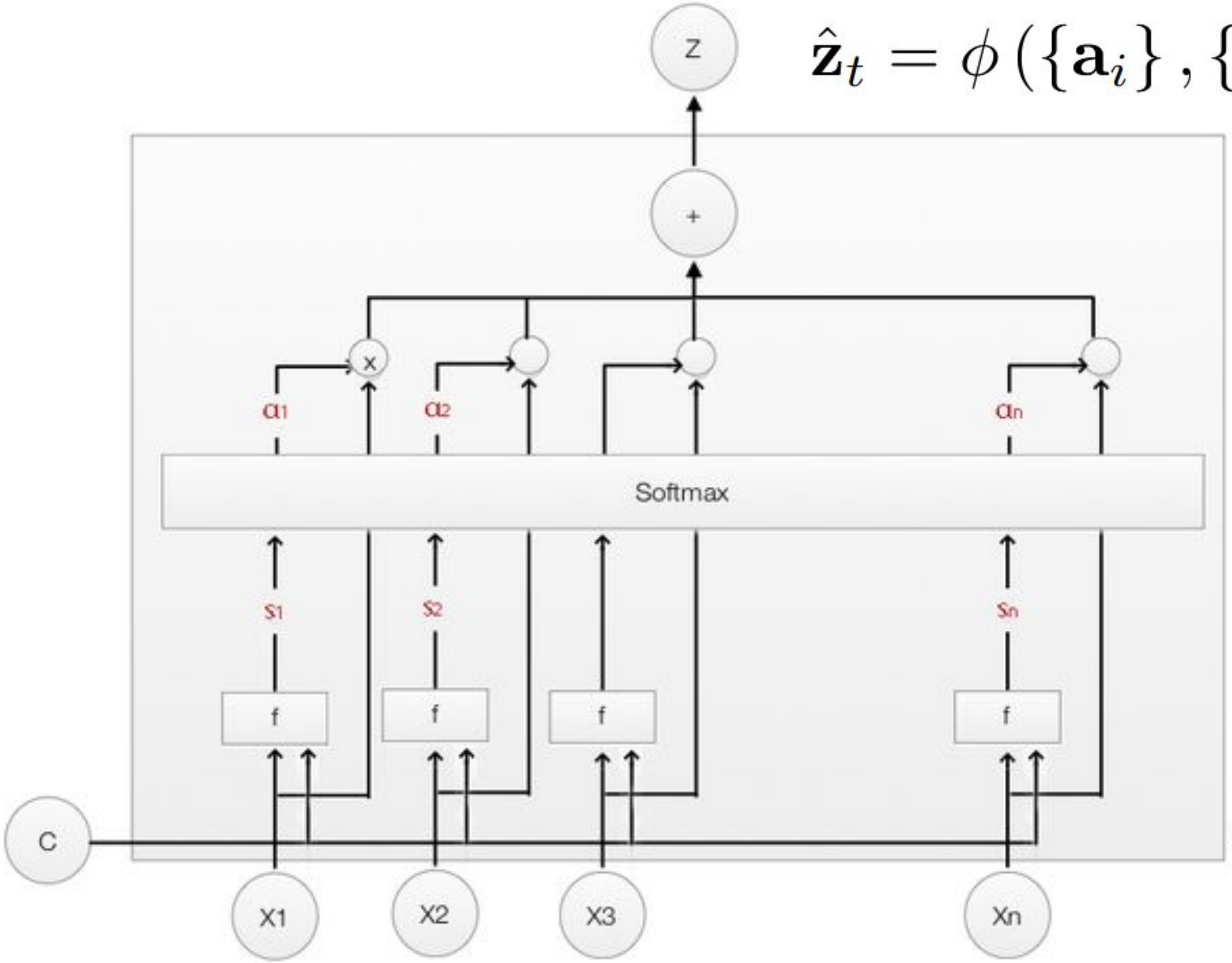


$$m_i = \tanh\left(y_i W_{y_i} + C W_c\right)$$



Attention Unit

Attention Unit



$$\hat{\mathbf{z}}_t = \phi\left(\{\mathbf{a}_i\}, \{\alpha_i\}\right) = \sum_i \alpha_i \, \mathbf{a}_i$$

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

$$e_{ti} = f_{\text{att}}(\mathbf{a}_i, \mathbf{h}_{t-1})$$

$$= \tanh(W_c C + W_x X_i) = \tanh(W_c h_{t-1} + W_x x_i)$$

Type of Attention

- **Soft Attention**: different parts, different subregions

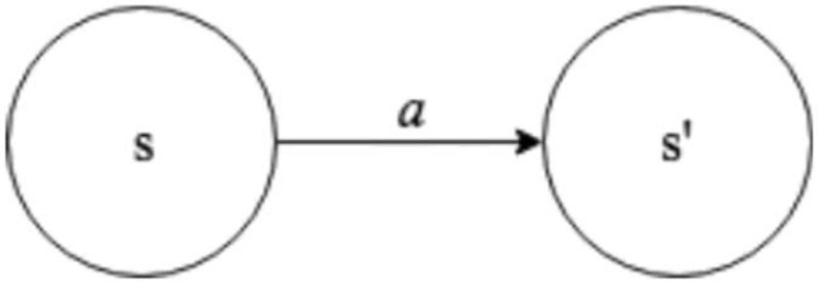
- **Hard Attention**: only ONE subregion

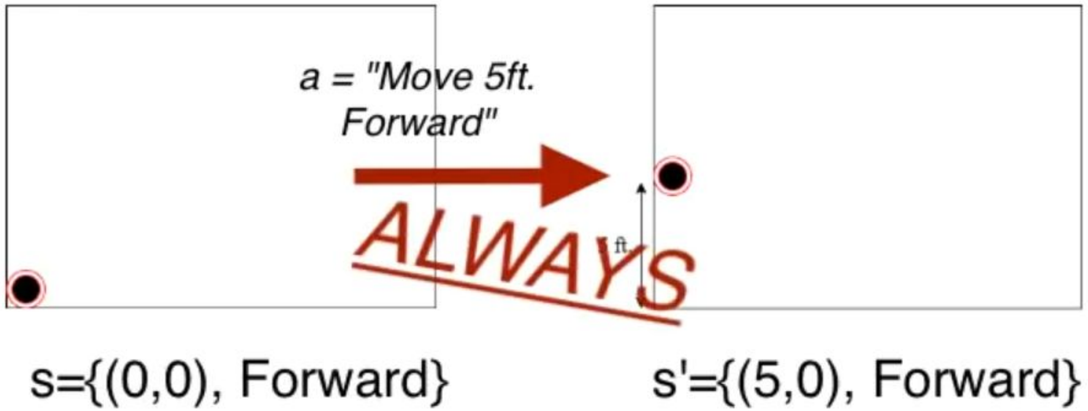
Type of Attention

- **Soft Attention**: different parts, different subregions

$$z = \sum_n s_n y_n$$

- Soft Attention is <u>Deterministic</u>

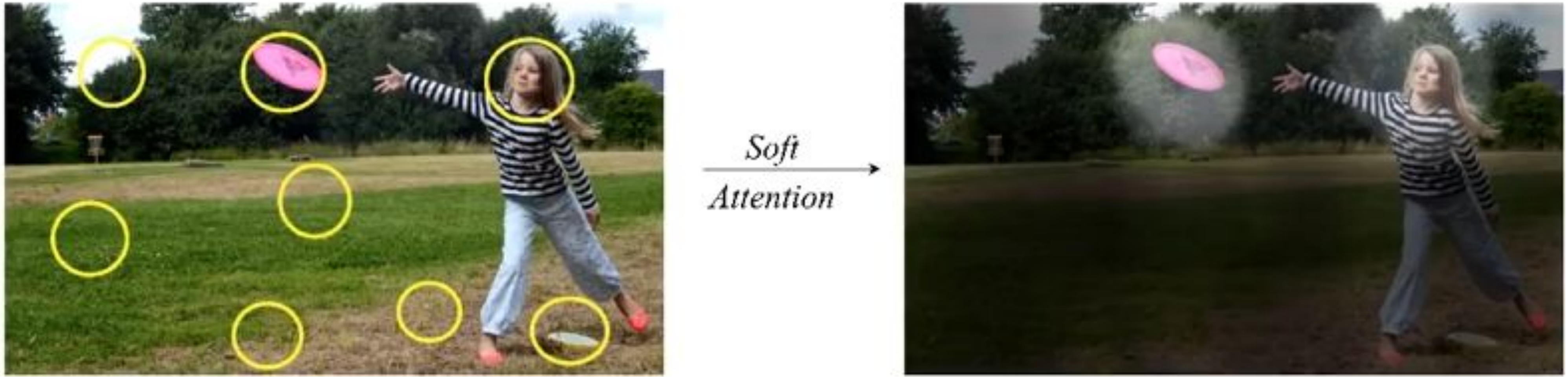




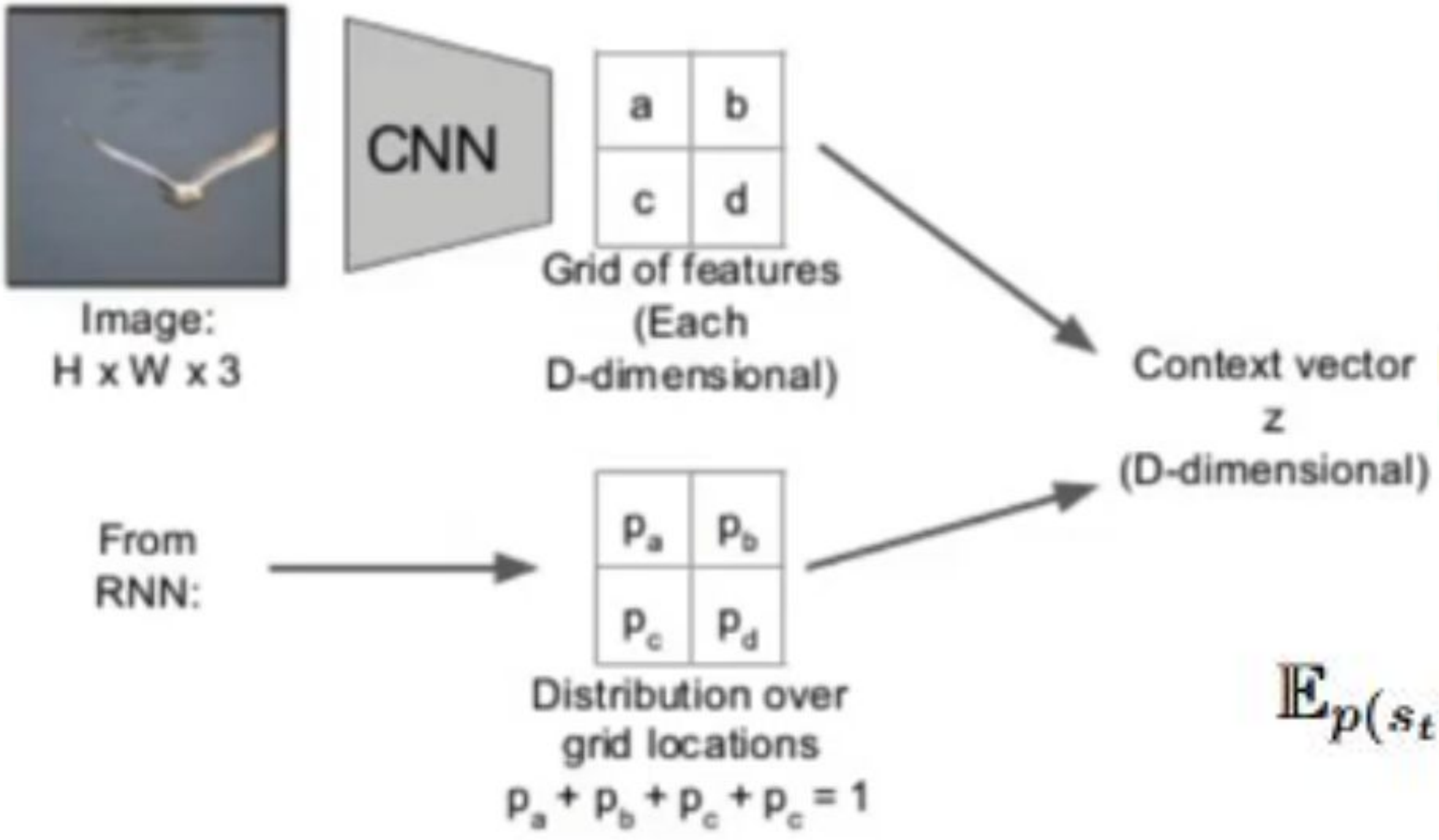
- **Soft Attention**: different parts, different subregions

$$z = \sum_n s_n y_n$$

- Soft Attention is <u>Deterministic</u>



Implementing Soft Attention

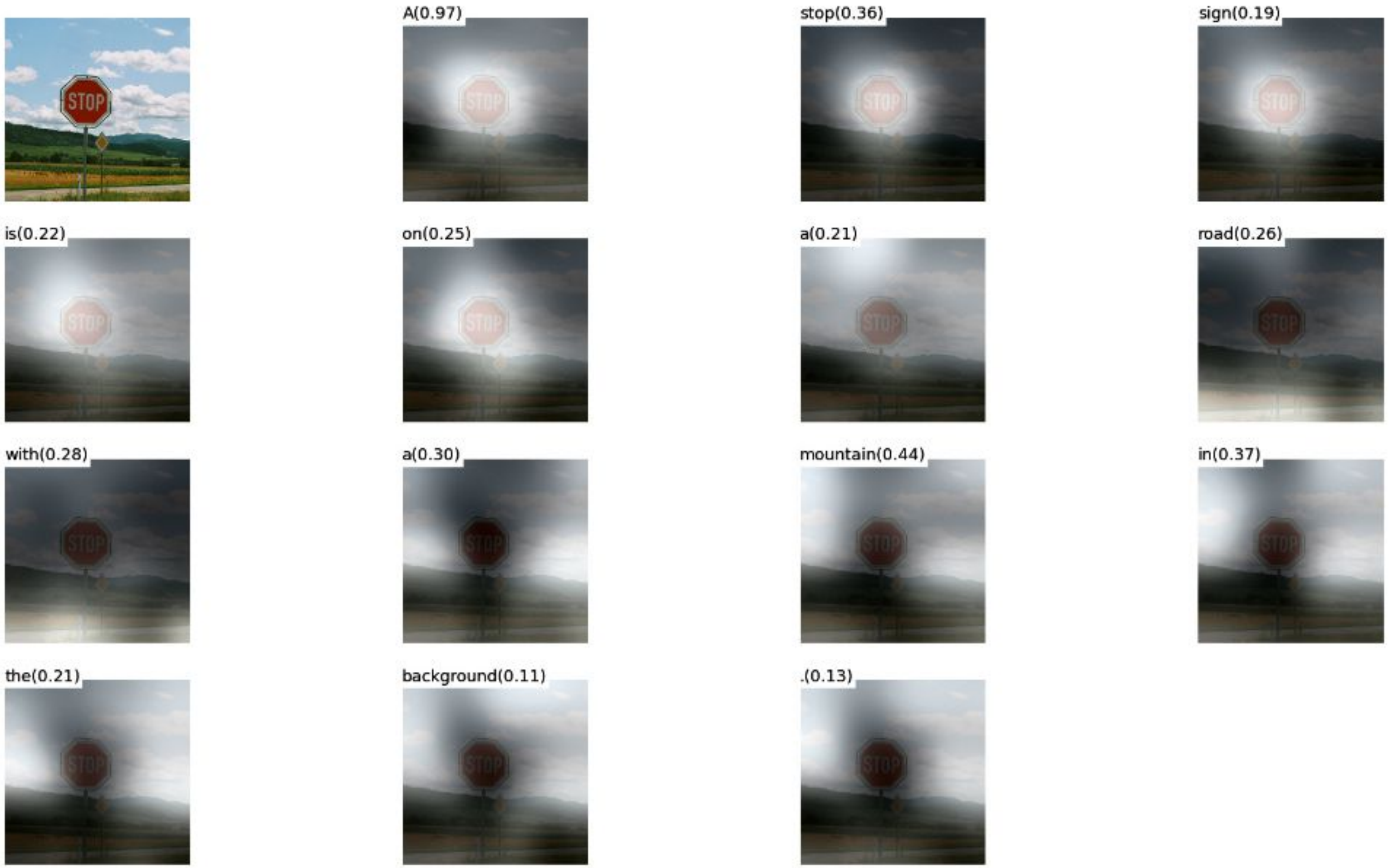


$$e_{ti} = f_{\text{att}}(\mathbf{a}_i, \mathbf{h}_{t-1})$$

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

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

Positive Example



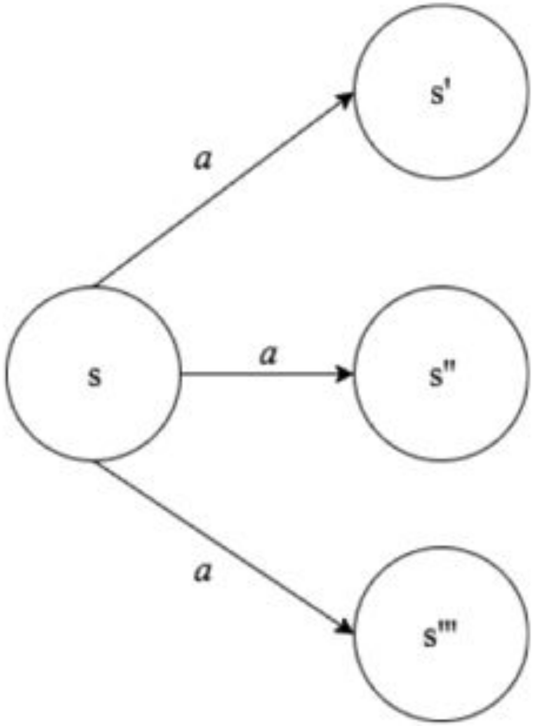
(b) A stop sign is on a road with a mountain in the background.

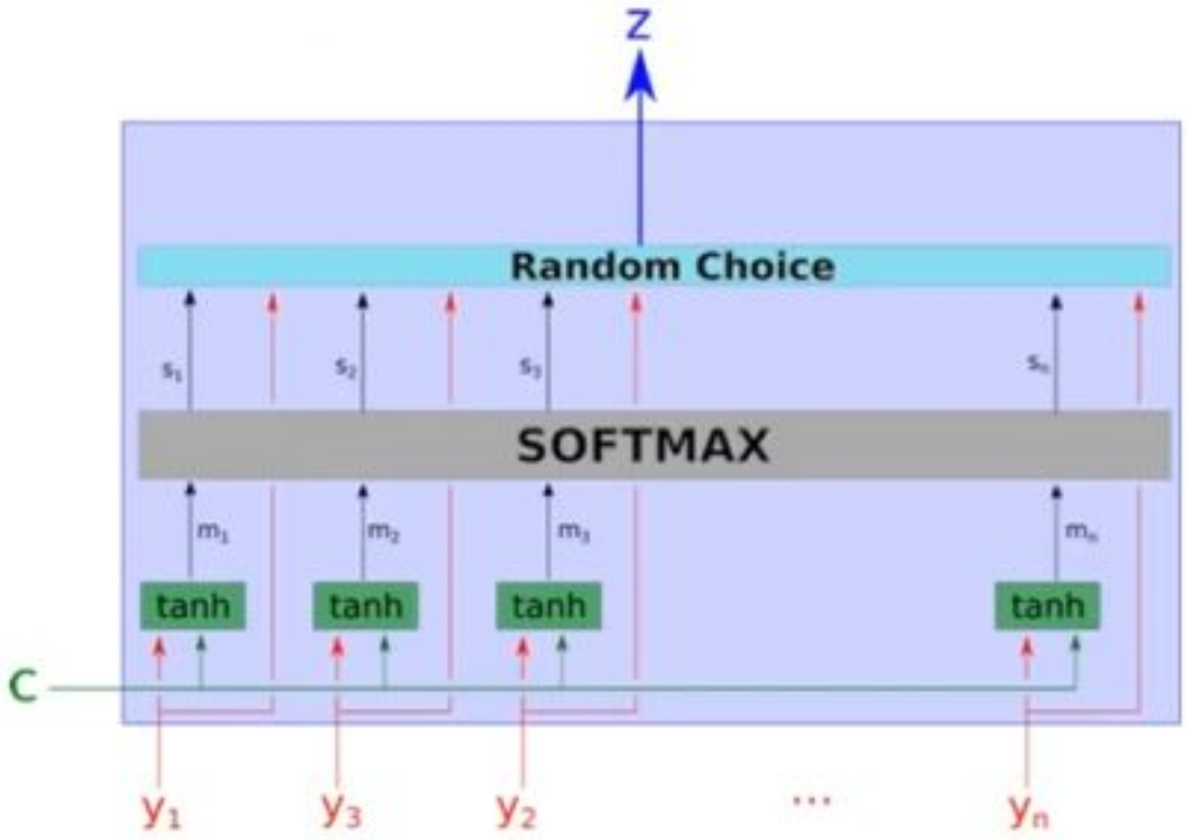
Negative Example

(b) A woman holding a clock in her hand.

2. Hard Attention: only ONE subregion

Hard Attention is *Stochastic*





Implementing Hard Attention

$$p(s_{t,i} = 1 \mid s_{j<t}, \mathbf{a}) = \alpha_{t,i}$$

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

We represent the location variable s_t as where the model decides to focus attention when generating the t^{th} word. $s_{t,i}$ is an indicator one-hot variable which is set to 1 if the i-th location (out of L) is the one used to extract visual features. By treating the attention locations as intermediate latent variables, we can assign a multinoulli distribution parametrized by $\{\alpha_i\}$, and view $\hat{z}_t$ as a random variable:

We define a new objective function L_s that is a variational lower bound on the marginal log-likelihood $\log p(\mathbf{y} \mid \mathbf{a})$ of observing the sequence of words $\mathbf{y}$ given image features $\mathbf{a}$. The learning algorithm for the parameters W of the models can be derived by directly optimizing L_s:

$$\begin{aligned} L_s &= \sum_s p(s \mid \mathbf{a}) \log p(\mathbf{y} \mid s, \mathbf{a}) \\ &\leq \log \sum_s p(s \mid \mathbf{a}) p(\mathbf{y} \mid s, \mathbf{a}) \\ &= \log p(\mathbf{y} \mid \mathbf{a}) \end{aligned}$$

Implementing Hard Attention

$$\frac{\partial L_s}{\partial W} = \sum_s p(s \mid \mathbf{a}) \left[\frac{\partial \log p(\mathbf{y} \mid s, \mathbf{a})}{\partial W} + \log p(\mathbf{y} \mid s, \mathbf{a}) \frac{\partial \log p(s \mid \mathbf{a})}{\partial W} \right]. \quad (11)$$

$$= \sum_s p(s \mid \mathbf{a}) \frac{\partial \log p(\mathbf{y} \mid s, \mathbf{a})}{\partial W} + p(s \mid \mathbf{a}) \log p(\mathbf{y} \mid s, \mathbf{a}) \frac{\partial \log p(s \mid \mathbf{a})}{\partial W}$$

Implementing Hard Attention

- This means that Monte Carlo Sampling can be performed!

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

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

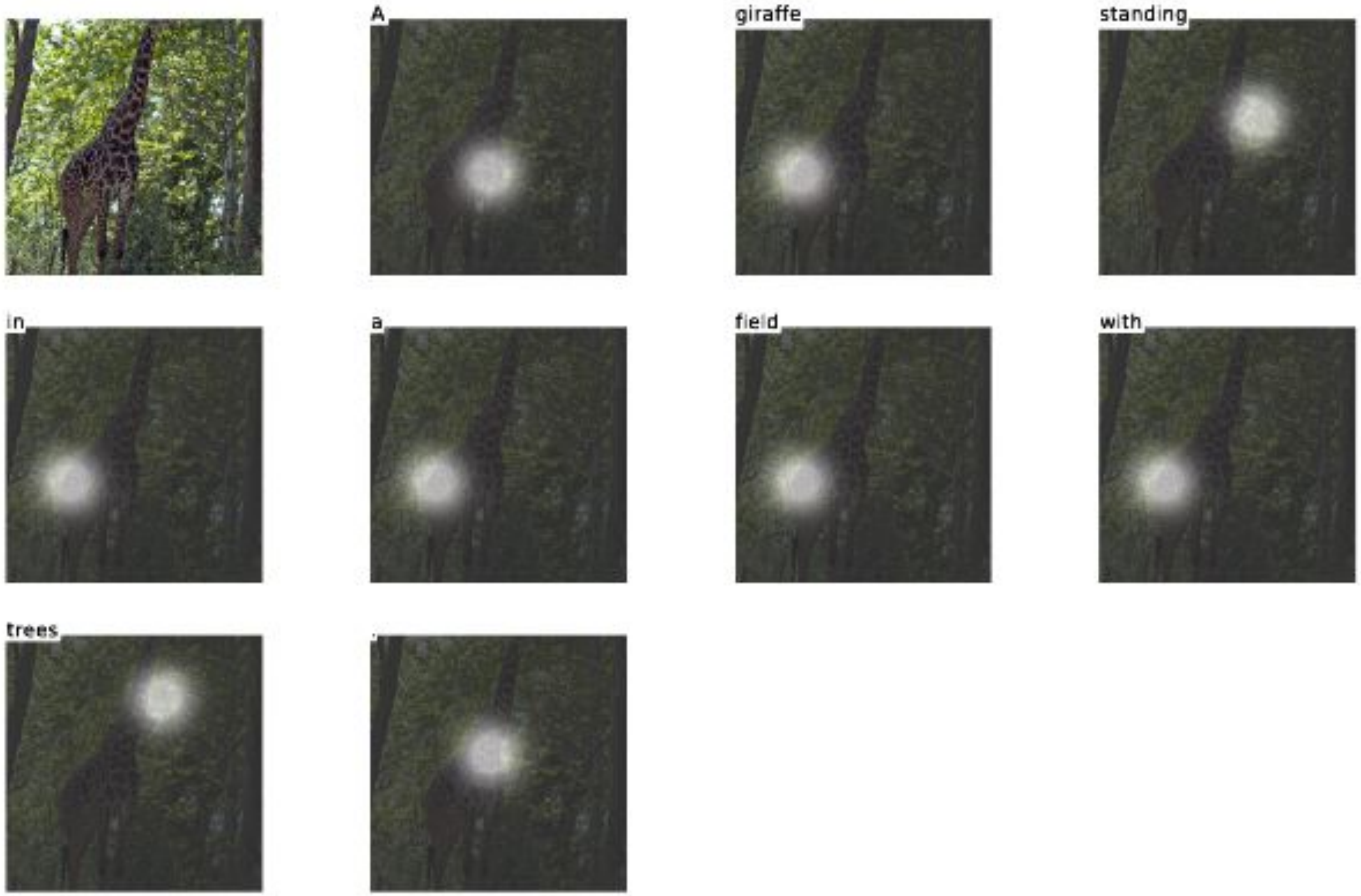
Implementing Hard Attention

- The issue is that the variance in this estimate is too high

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

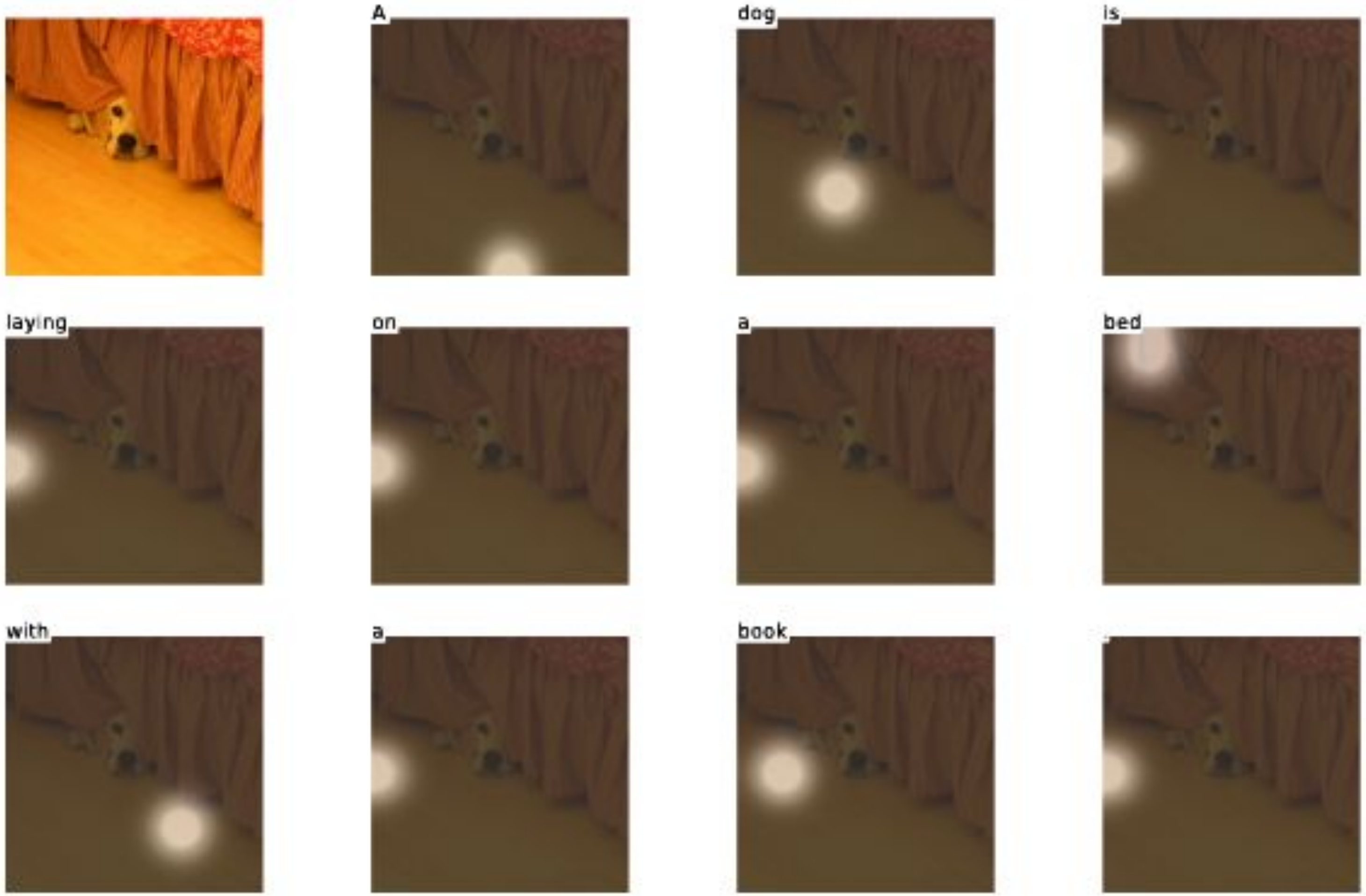
$$\frac{\partial L_s}{\partial W} \approx \frac{1}{N} \sum_{n=1}^{N} \Big[\frac{\partial \log p(\mathbf{y} \mid \tilde{s}^n, \mathbf{a})}{\partial W} + \lambda_r (\log p(\mathbf{y} \mid \tilde{s}^n, \mathbf{a}) - b) \frac{\partial \log p(\tilde{s}^n \mid \mathbf{a})}{\partial W} + \lambda_e \frac{\partial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

Positive Example



(a) A giraffe standing in a field with trees.

Negative Example



(a) A dog is laying on a bed with a book.

Doubly Stochastic Attention

- To encourage the model to look at various parts of the image

$$L_d = -\log(P(\mathbf{y}|\mathbf{x})) + \lambda \sum_{i}^{L} (1 - \sum_{t}^{C} \alpha_{ti})^2$$

Result

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

		BLEU				
Dataset	**Model**	**BLEU-1**	**BLEU-2**	**BLEU-3**	**BLEU-4**	**METEOR**
Flickr8k	Google NIC(Vinyals et al., 2014)$^{\dagger\Sigma}$	63	41	27	—	—
	Log Bilinear (Kiros et al., 2014a)$^{\circ}$	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$^{\dagger\circ\Sigma}$	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)$^{\dagger a}$	—	—	—	—	20.71
	BRNN (Karpathy & Li, 2014)$^{\circ}$	64.2	45.1	30.4	20.3	—
	Google NIC$^{\dagger\circ\Sigma}$	66.6	46.1	32.9	24.6	—
	Log Bilinear$^{\circ}$	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