THE UNIVERSITY OF BRITISH COLUMBIA

Topics in AI (CPSC 532S):
Multimodal Learning with Vision, Language and Sound

Lecture 10: RNN Applications

Course **Logistics**

— **Assignment 3** due date is Wednsday 11:59pm (Thursday?)

— **Assignment 4** is out, due Friday, February 15th @ 11:59pm

— Group **Projects** form completed

— Project proposals (in class on **Feb 26th**)

Final **Project** (50% of grade total) – Reminder

- Group project (groups of 3 are encouraged, but fewer maybe possible)
- Groups are self-formed, you will not be assigned to a group
- You need to come up with a project proposal and then work on the project as a group (each person in the group gets the same grade for the project)
- Project needs to be **research** oriented (not simply implementing an existing paper); you can use code of existing paper as a starting point though

Project proposal + class presentation: 15%
Project + final presentation: 35%

Project proposal and class presentation — **15%** of grade

Presentation (~5 minutes irrespective of the group size)

1. Clear explanation of the **overall problem** you want to solve and relationship to the topics covered in class
2. What **model/algorithms** you planning to explore: at this can be somewhat abstract (e.g., CNN+RNN)
3. The **dataset(s)** you will use and how will you **evaluate** performance
4. List of **papers** you plan to read as references
5. How will you **structure the project**, who will do what and a rough timeline

Project proposal and class presentation — **15%** of grade

Presentation (~5 minutes irrespective of the group size)

1. Clear explanation of the **overall problem** you want to solve and relationship to the topics covered in class
2. What **model/algorithms** you planning to explore: this can be somewhat abstract (e.g., CNN+RNN)
3. The **dataset(s)** you will use and how will you **evaluate** performance
4. List of **papers** you plan to read as references
5. How will you **structure the project**, who will do what and a rough timeline

After presentation you will get the feedback from me

Project proposal and class presentation — **15%** of grade

Presentation (~5 minutes irrespective of the group size)

1. Clear explanation of the **overall problem** you want to solve and relationship to the topics covered in class
2. What **model/algorithms** you planning to explore: this can be somewhat abstract (e.g., CNN+RNN)
3. The **dataset(s)** you will use and how will you **evaluate** performance
4. List of **papers** you plan to read as references
5. How will you **structure the project**, who will do what and a rough timeline

After presentation you will get the feedback from me

Proposal

— Same as above but in more detail, with well defined algorithms and timeline

— Will be in the form of the **PDF** document (initial paper draft)

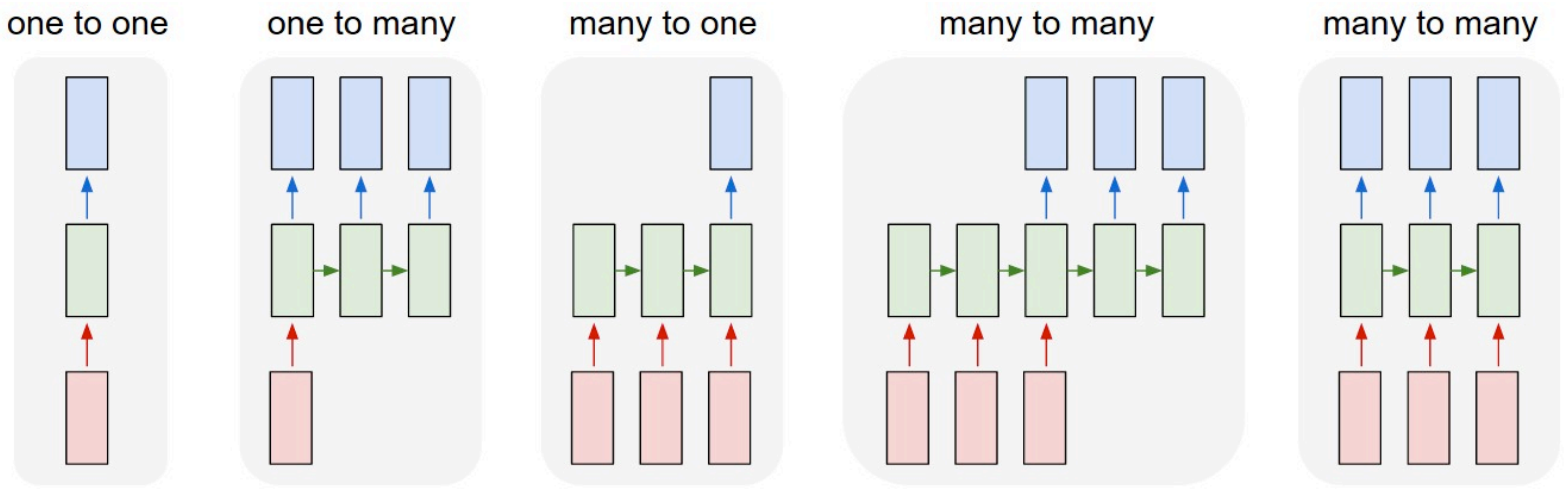
RNNs: Review

Key Enablers:

— Parameter sharing in computational graphs

— “Unrolling” in computational graphs

— Allows modeling **arbitrary length sequences**!



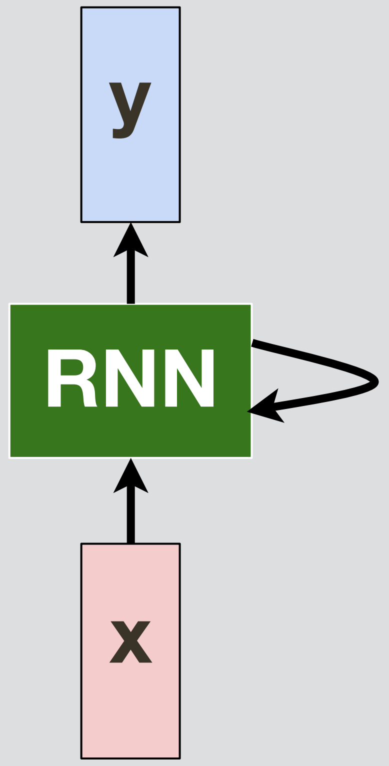
Vanilla **RNN**

$$y_t = W_{hy} h_t + b_y$$

$$h_t = f_W(h_{t-1}, x_t)$$

$$\downarrow$$

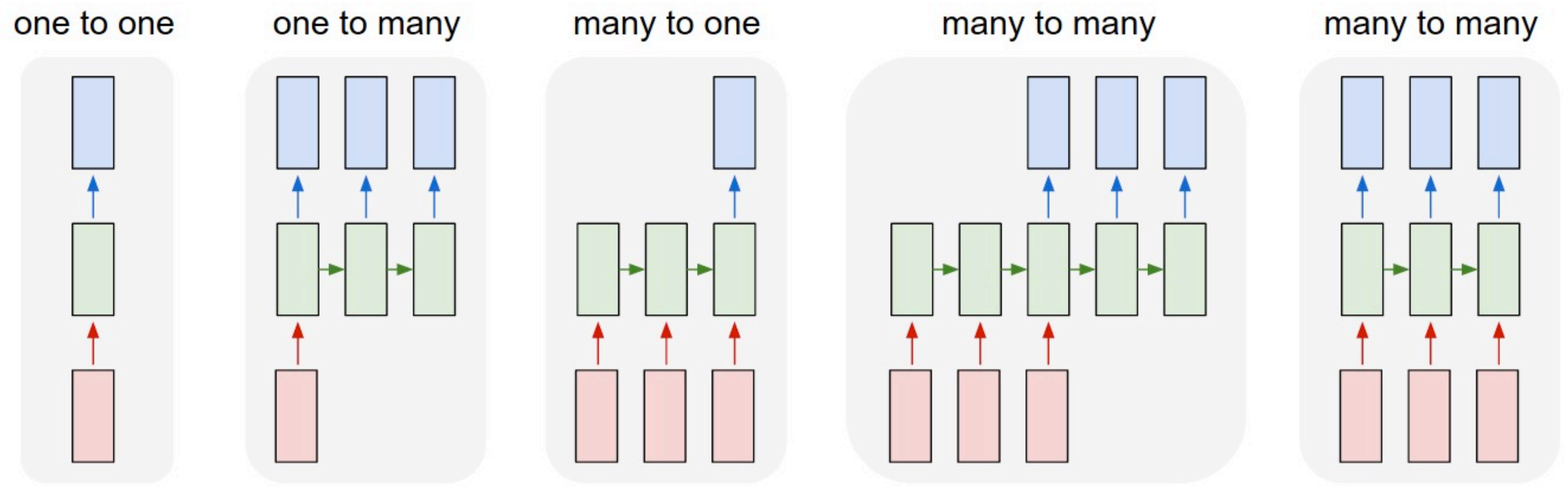
$$h_t = \tanh(W_{hh} h_{t-1} + W_{xh} x_t + b_h)$$



RNNs: Review

Key Enablers:

- Parameter sharing in computational graphs
- "Unrolling" in computational graphs
- Allows modeling **arbitrary length sequences**!



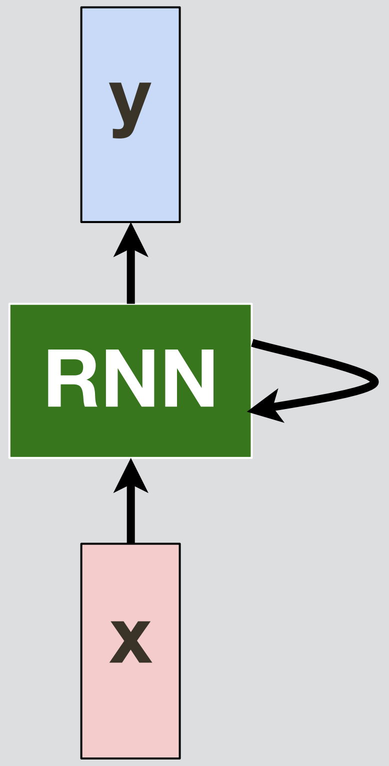
Vanilla **RNN**

$$y_t = W_{hy}h_t + b_y$$

$$h_t = f_W(h_{t-1}, x_t)$$

$$\downarrow$$

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

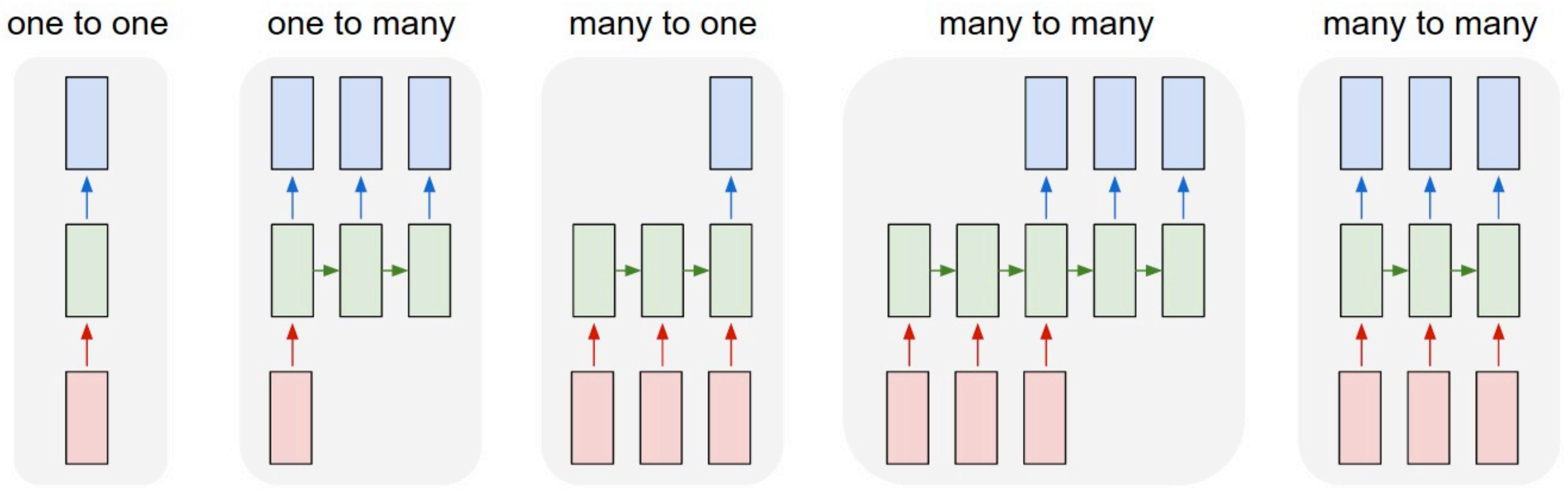


Vanishing
or
Exploding
Gradients

RNNs: Review

Key Enablers:

- Parameter sharing in computational graphs
- “Unrolling” in computational graphs
- Allows modeling **arbitrary length sequences**!



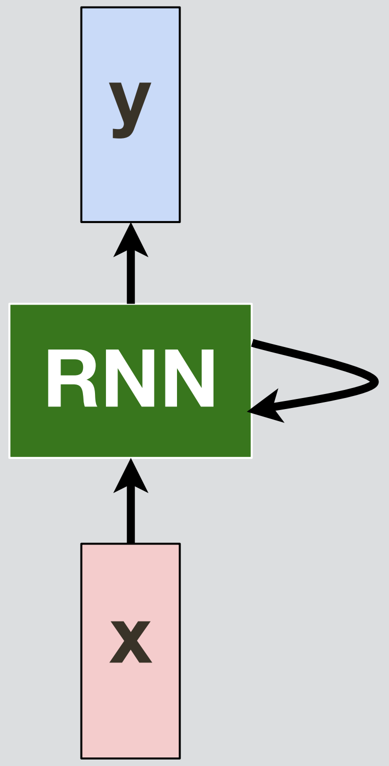
Vanilla **RNN**

$$y_t = W_{hy} h_t + b_y$$

$$h_t = f_W(h_{t-1}, x_t)$$

$$\downarrow$$

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



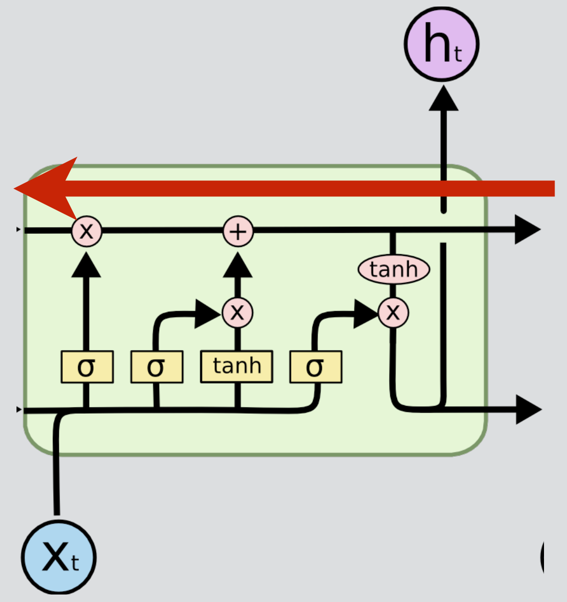
Vanishing or **Exploding** Gradients

Long-Short Term Memory (**LSTM**)

$$\begin{pmatrix} i \\ f \\ o \\ g \end{pmatrix} = \begin{pmatrix} \sigma \\ \sigma \\ \sigma \\ \tanh \end{pmatrix} W \begin{pmatrix} h_{t-1} \\ x_t \end{pmatrix}$$

$$c_t = f \odot c_{t-1} + i \odot g$$

$$h_t = o \odot \tanh(c_t)$$



Uninterrupted gradient flow!

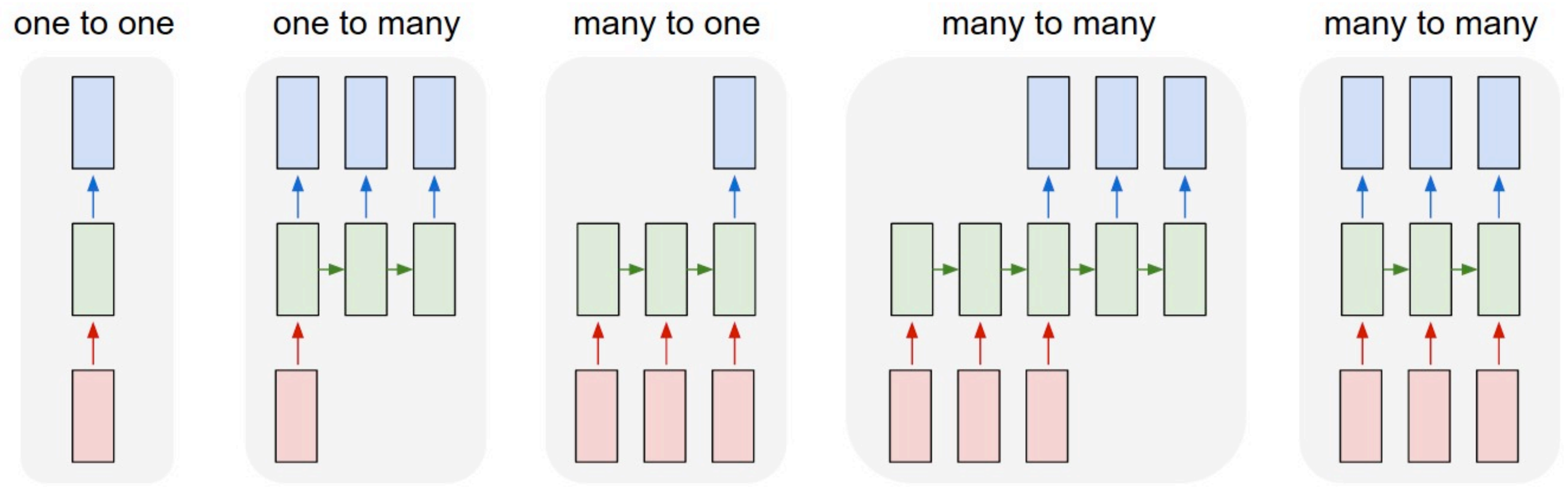
RNNs: Review

Key Enablers:

— Parameter sharing in computational graphs

— “Unrolling” in computational graphs

— Allows modeling **arbitrary length sequences**!



Loss functions: often cross-entropy (for classification); could be max-margin (like in SVM) or Squared Loss (regression)

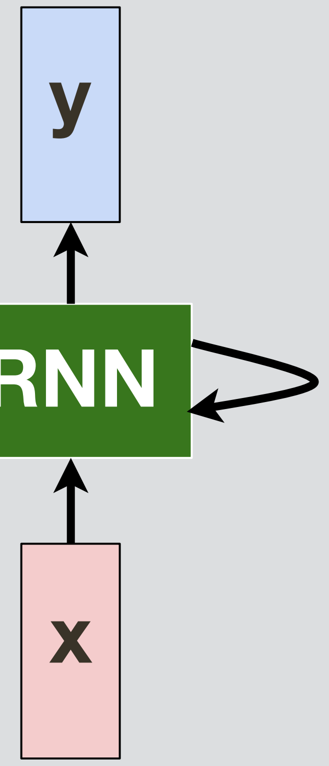
Vanilla **RNN**

$$y_t = W_{hy}h_t + b_y$$

$$h_t = f_W(h_{t-1}, x_t)$$

$$\downarrow$$

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



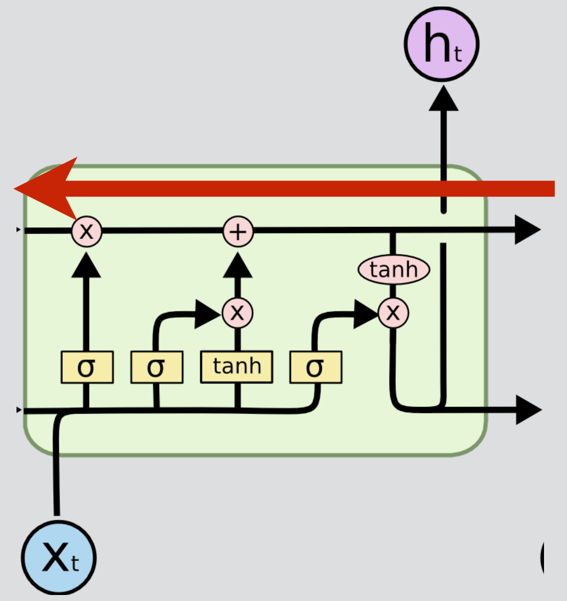
Vanishing
or
Exploding
Gradients

Long-Short Term Memory (**LSTM**)

$$\begin{pmatrix} i \\ f \\ o \\ g \end{pmatrix} = \begin{pmatrix} \sigma \\ \sigma \\ \sigma \\ \tanh \end{pmatrix} W \begin{pmatrix} h_{t-1} \\ x_t \end{pmatrix}$$

$$c_t = f \odot c_{t-1} + i \odot g$$

$$h_t = o \odot \tanh(c_t)$$



Uninterrupted gradient flow!

Sequence Level Training

During training objective is different than at test time

- **Training:** generate next word given the previous
- **Test:** generate the entire sequence given an initial state

Optimize directly evaluation metric (e.g. BLUE score for sentence generation)

Set the problem as a Reinforcement Learning:

- RNN is an Agent
- Policy defined by the learned parameters
- Action is the selection of the next word based on the policy - Reward is the evaluation metric

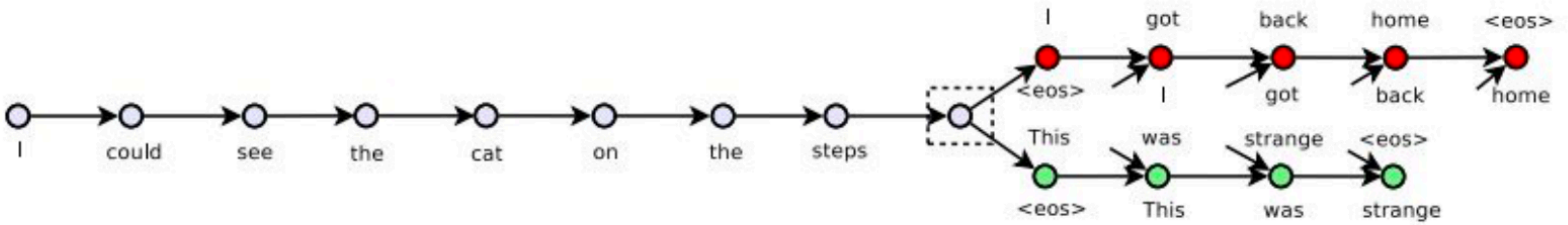
[Ranzato et al., 2016]

* slide from Marco Pedersoli and Thomas Lucas

Let us look at some actual practical uses of RNNs

Applications: Skip-thought Vectors

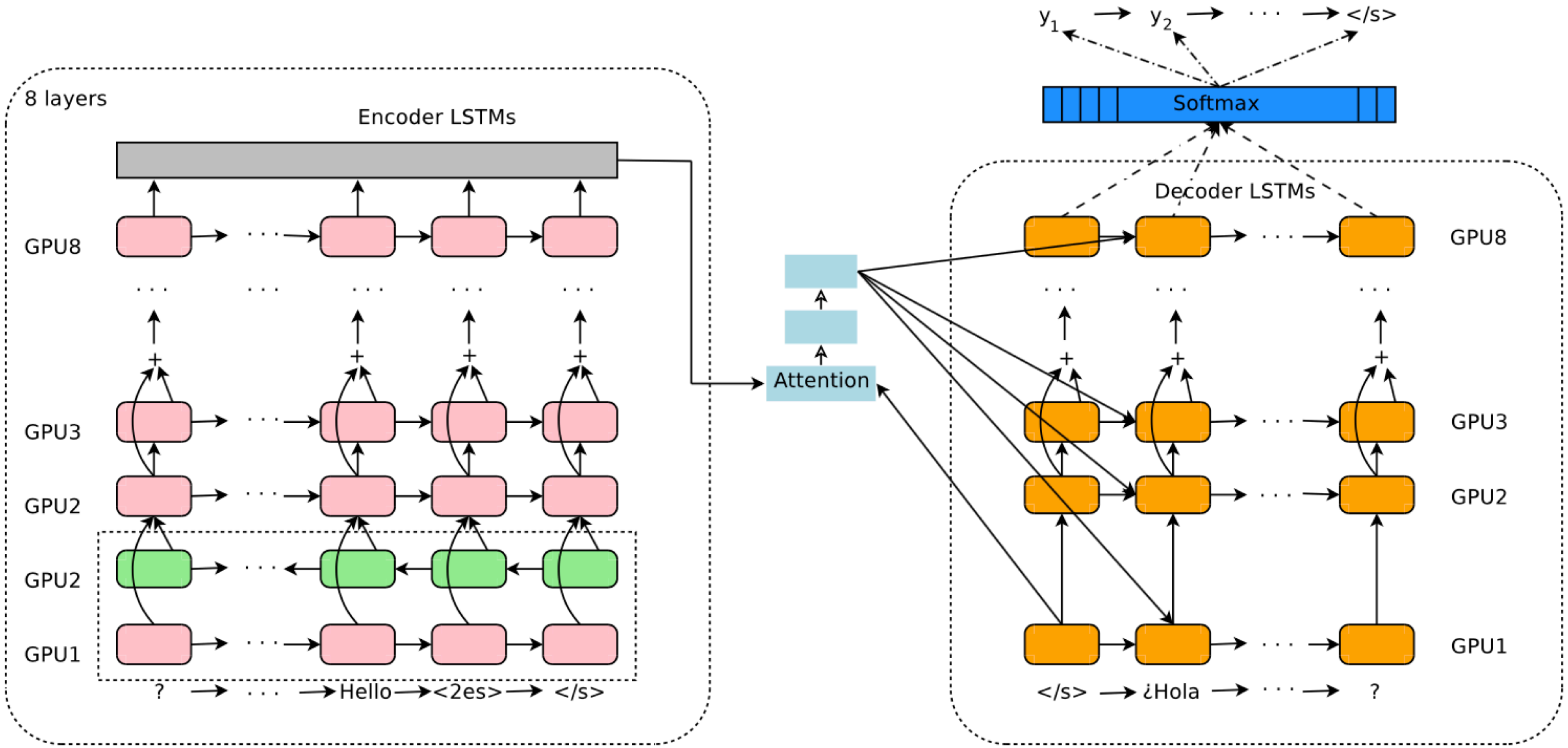
word2vec but for sentences, where each sentence is processed by an LSTM



[Kiros et al., 2015]

Applications: Google Language Translation

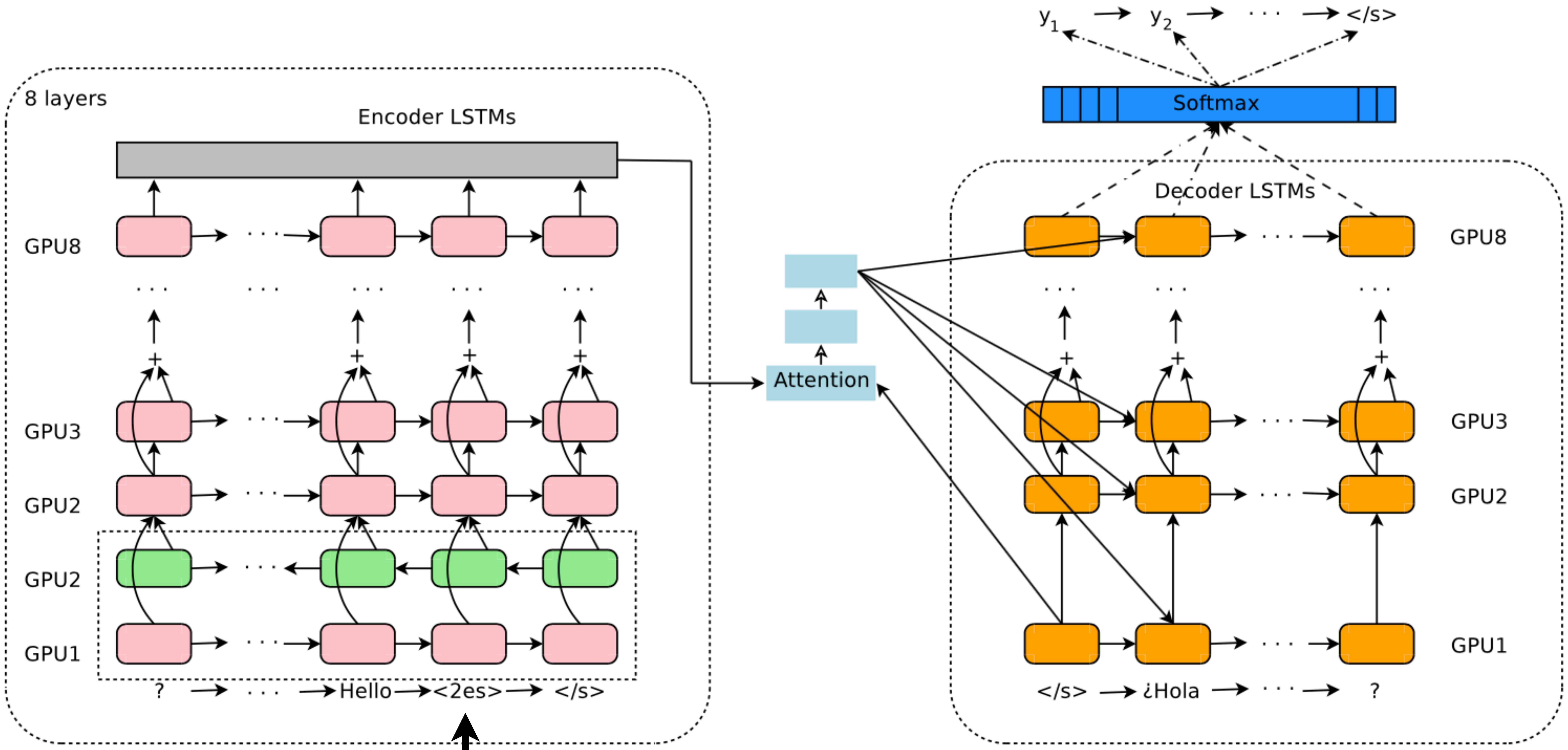
One model to translate from **any language** to any other language



[Johnson et al., 2017]

Applications: Google Language Translation

One model to translate from **any language** to any other language

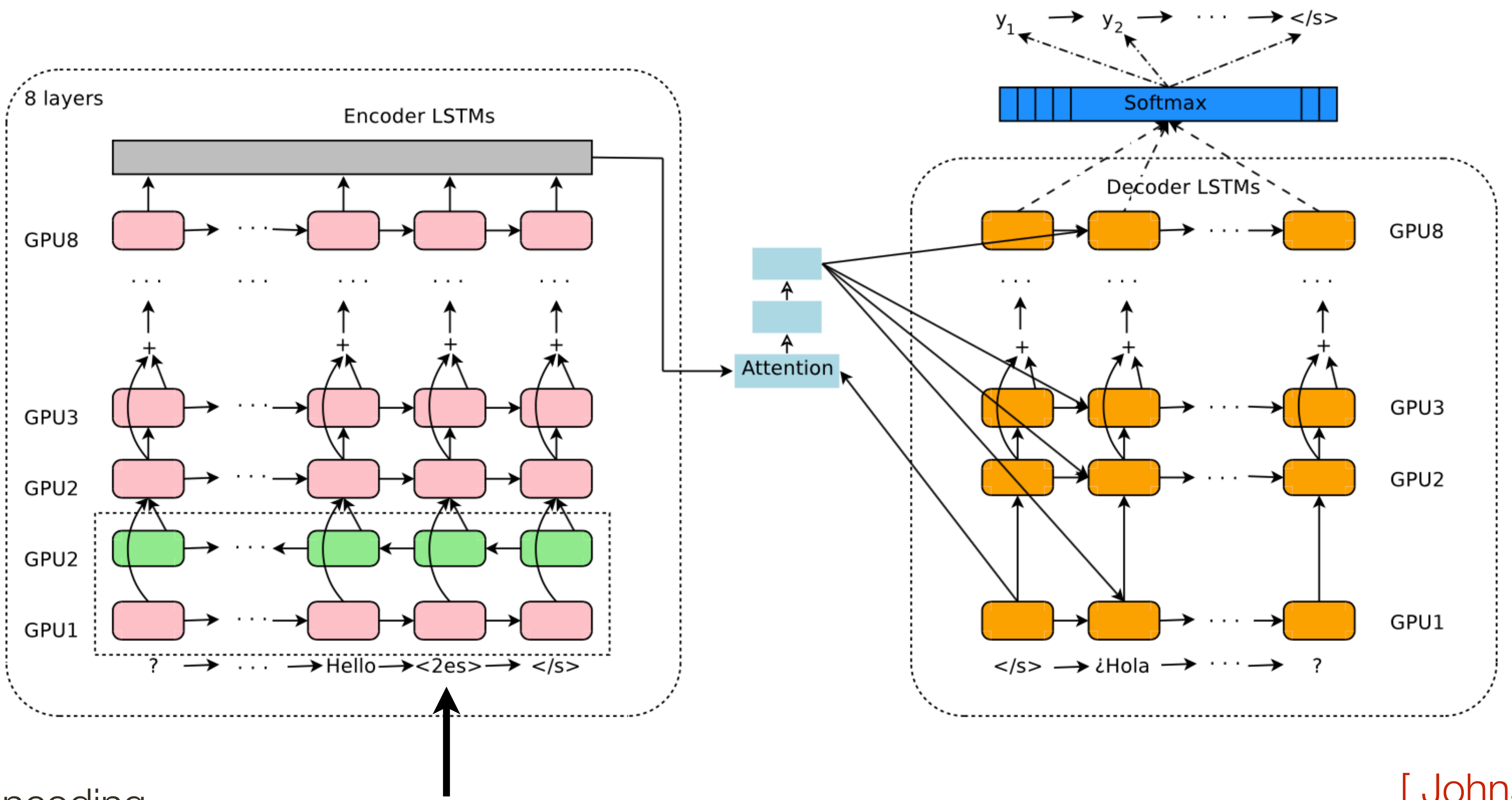


Token designating
target language

[Johnson et al., 2017]

Applications: Google Language Translation

One model to translate from **any language** to any other language



Flipped order encoding

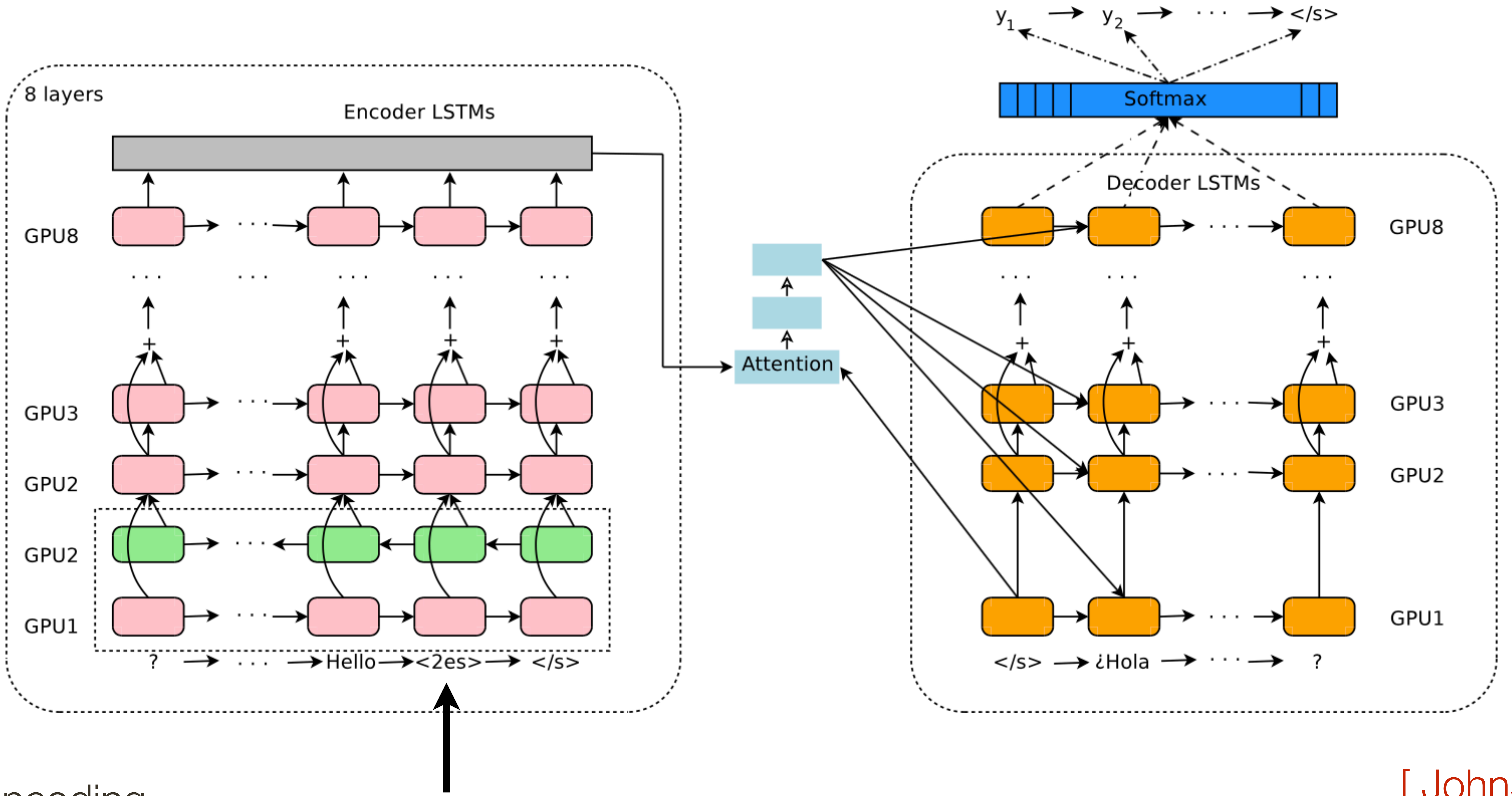
Why?

Token designating **target** language

[Johnson et al., 2017]

Applications: Google Language Translation

One model to translate from **any language** to any other language



Flipped order encoding

Token designating **target** language

[Johnson et al., 2017]

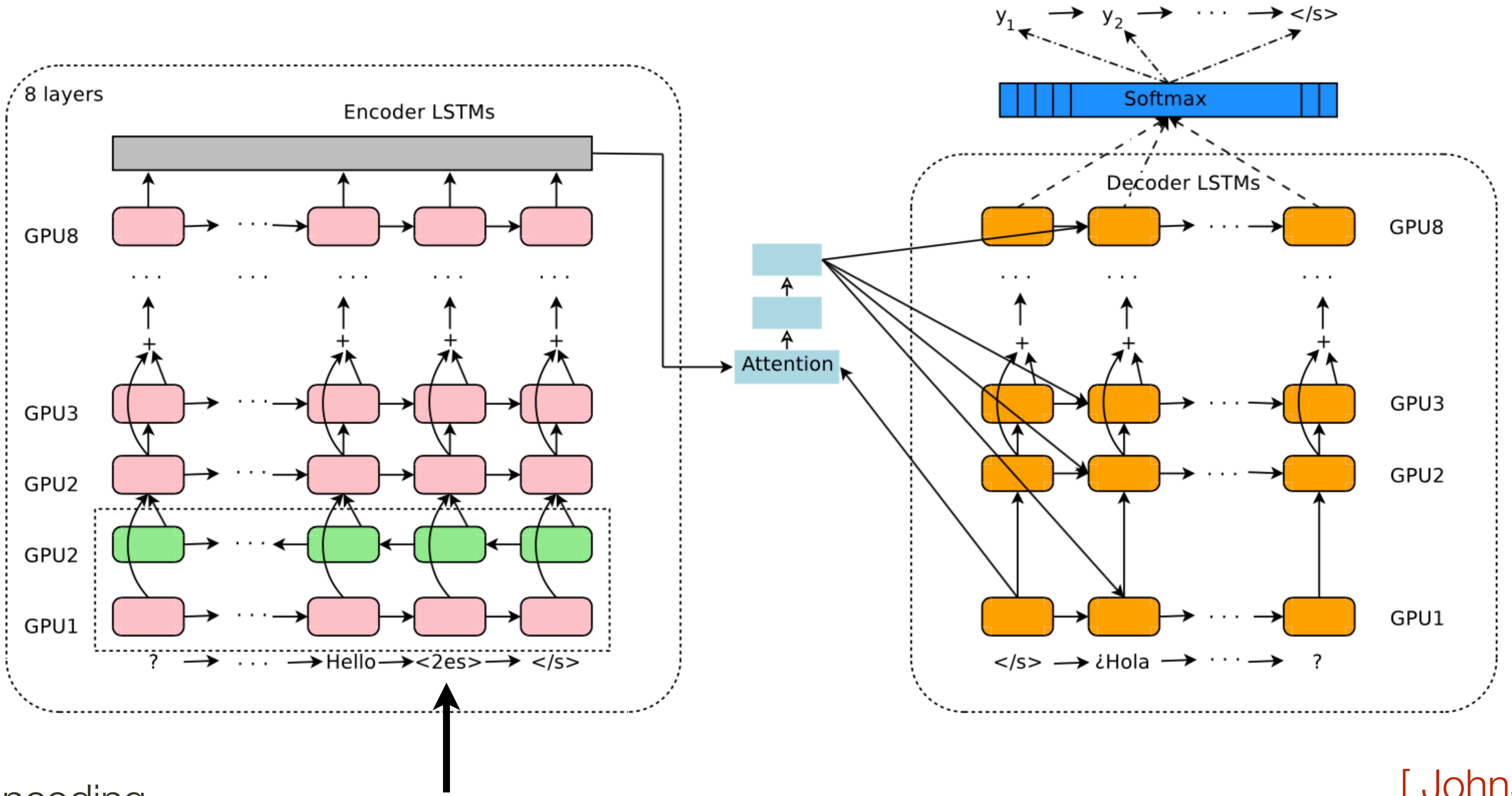
8! layer LSTM decoder and encoder

Applications: Google Language Translation

One model to translate from **any language** to any other language

Residual at other layers (ResNet style)

Bi-directional at lower layers



Flipped order encoding

Token designating **target** language

[Johnson et al., 2017]

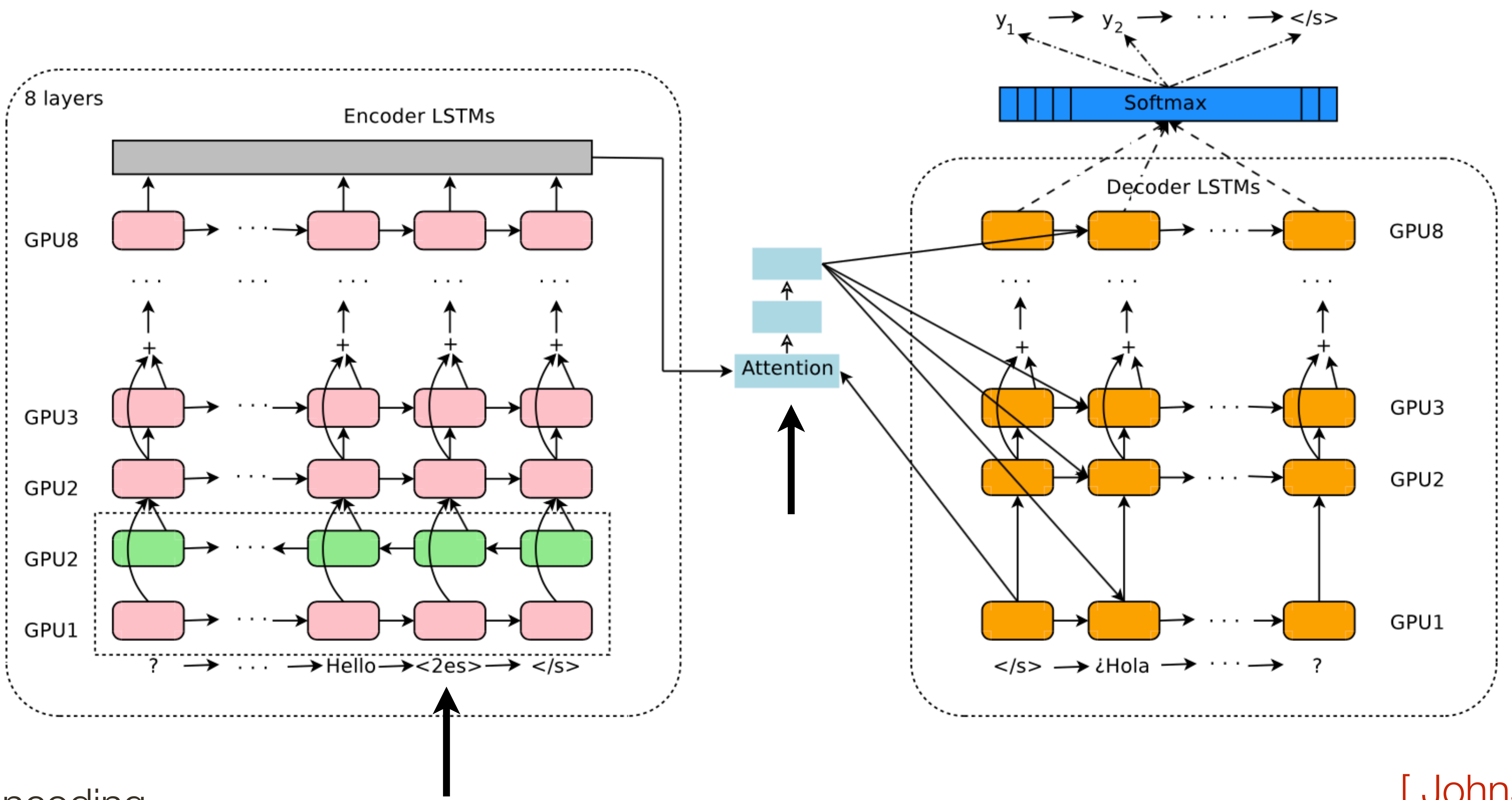
8! layer LSTM decoder and encoder

Applications: Google Language Translation

One model to translate from **any language** to any other language

Residual at other layers (ResNet style)

Bi-directional at lower layers



Flipped order encoding

Token designating **target** language

[Johnson et al., 2017]

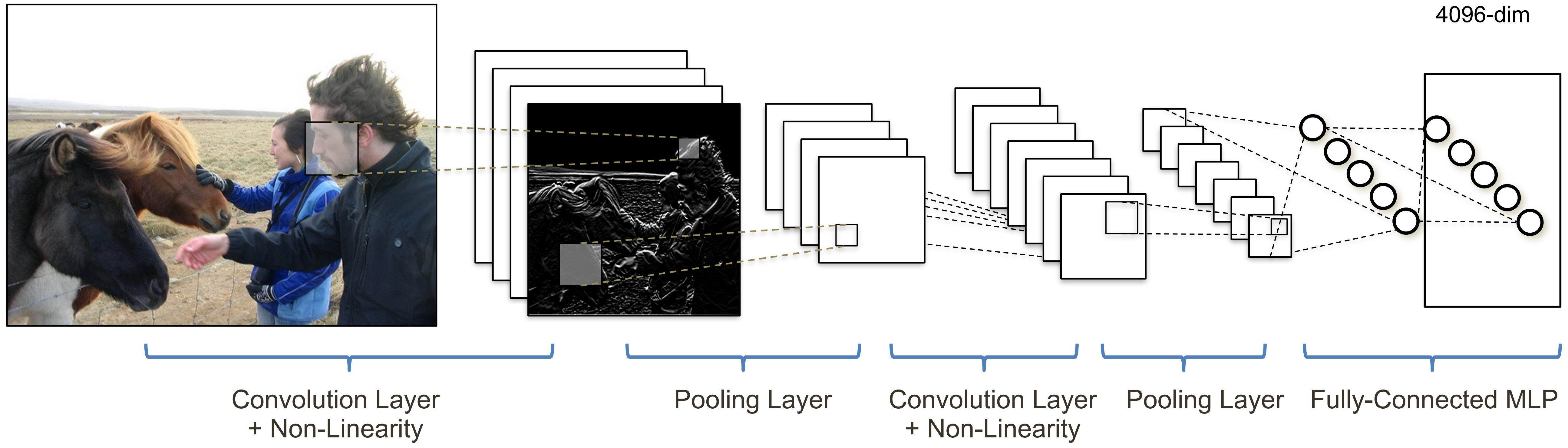
8! layer LSTM decoder and encoder

Applications: Neural Image Captioning

* slide from Dhruv Batra

Applications: Neural Image Captioning

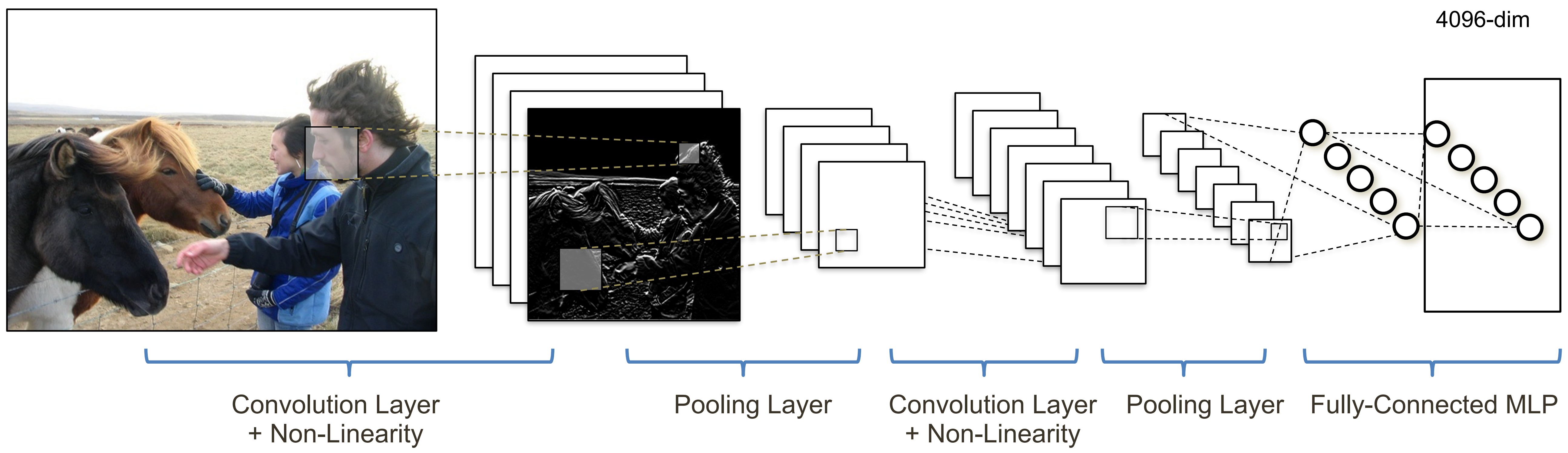
Image Embedding (VGGNet)



* slide from Dhruv Batra

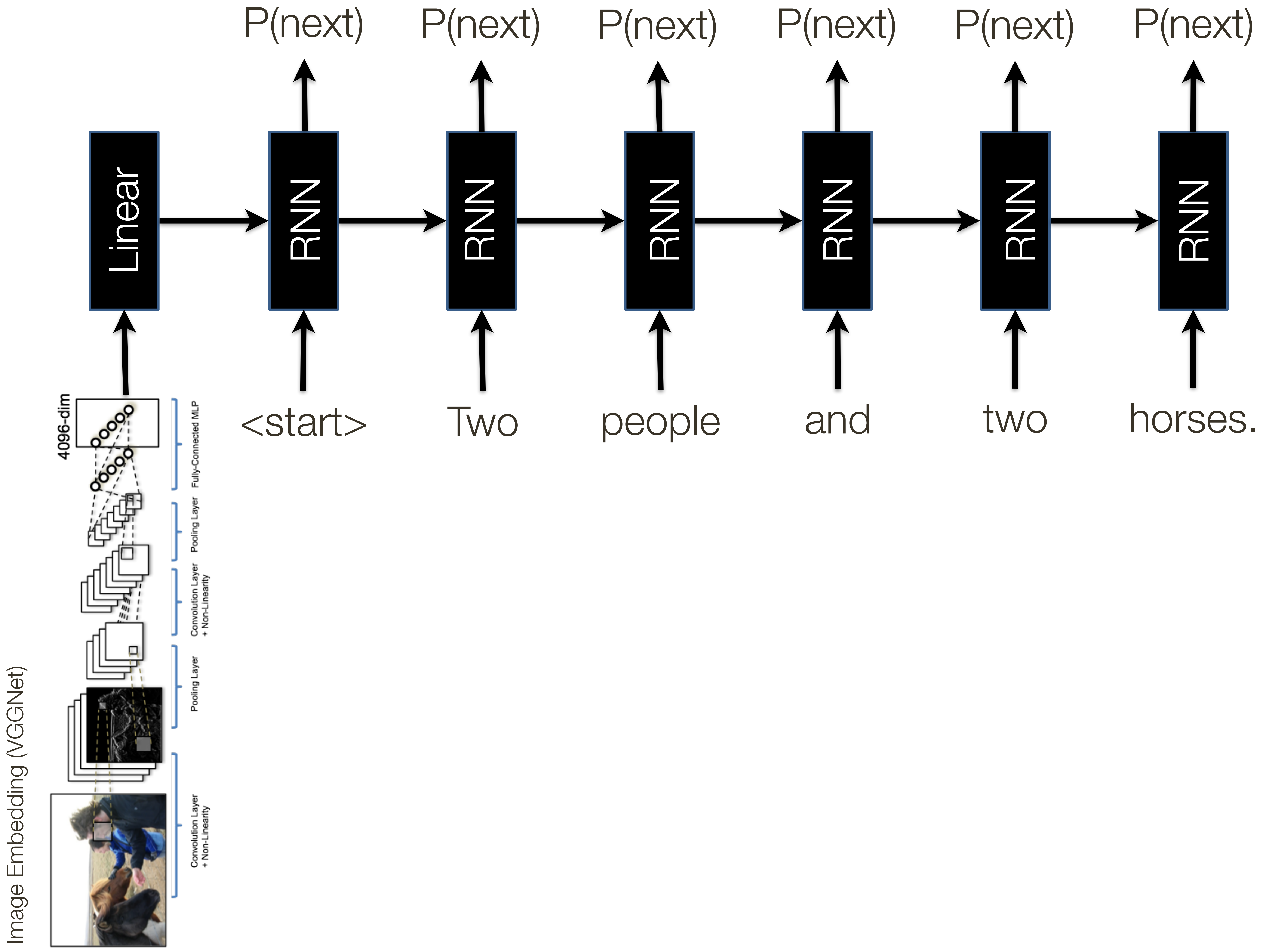
Applications: Neural Image Captioning

Image Embedding (VGGNet)



* slide from Dhruv Batra

Applications: Neural Image Captioning



* slide from Dhruv Batra

Applications: Neural Image Captioning

Good results

A cat sitting on a suitcase on the floor

A cat is sitting on a tree branch

A dog is running in the grass with a frisbee

A white teddy bear sitting in the grass

Two people walking on the beach with surfboards

A tennis player in action on the court

Two giraffes standing in a grassy field

A man riding a dirt bike on a dirt track

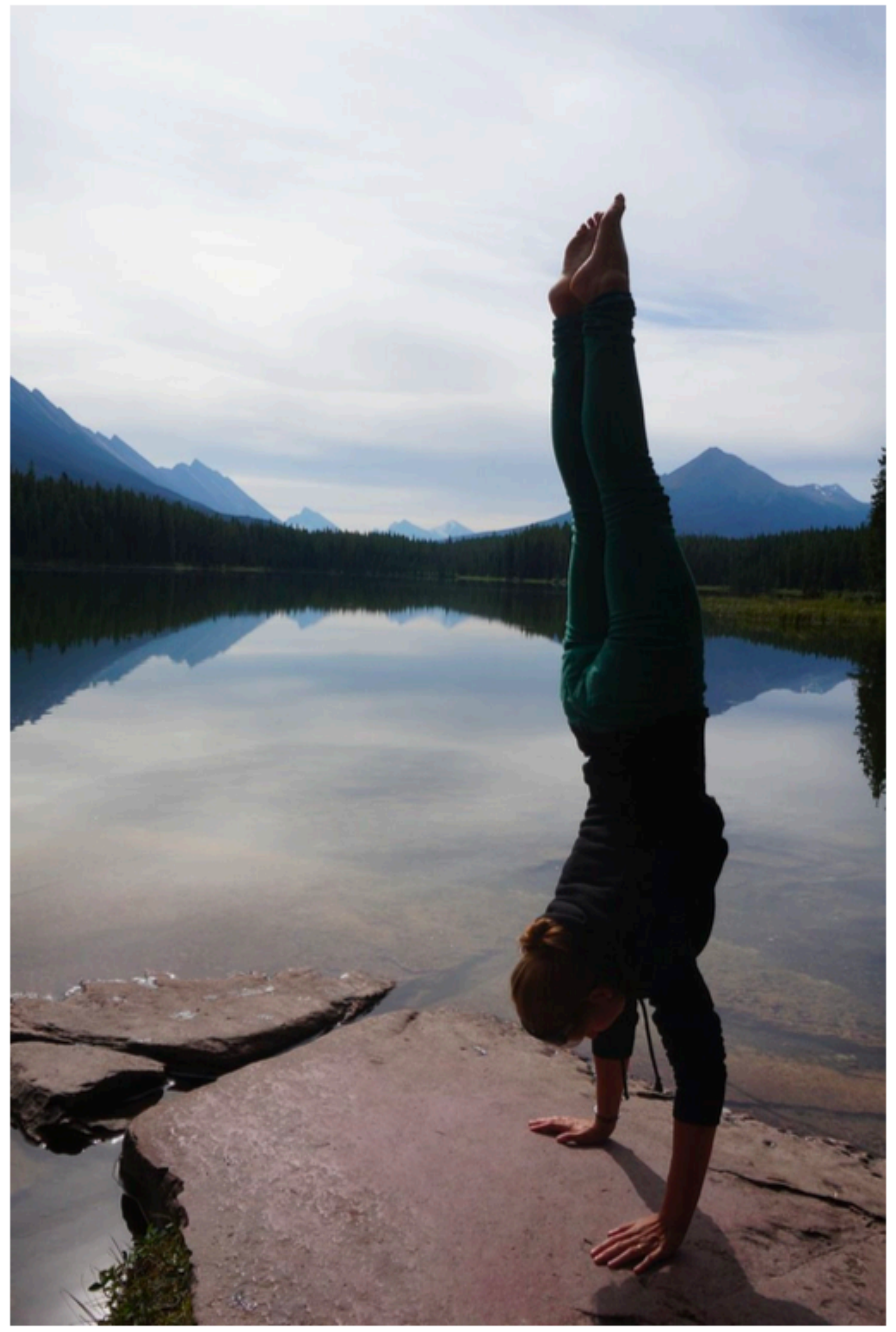
* slide from Fei-Dei Li, Justin Johnson, Serena Yeung, **cs231n Stanford**

Applications: Neural Image Captioning

Failure cases

A woman is holding a cat in her hand

A person holding a computer mouse on a desk



A woman standing on a beach holding a surfboard

A bird is perched on a tree branch

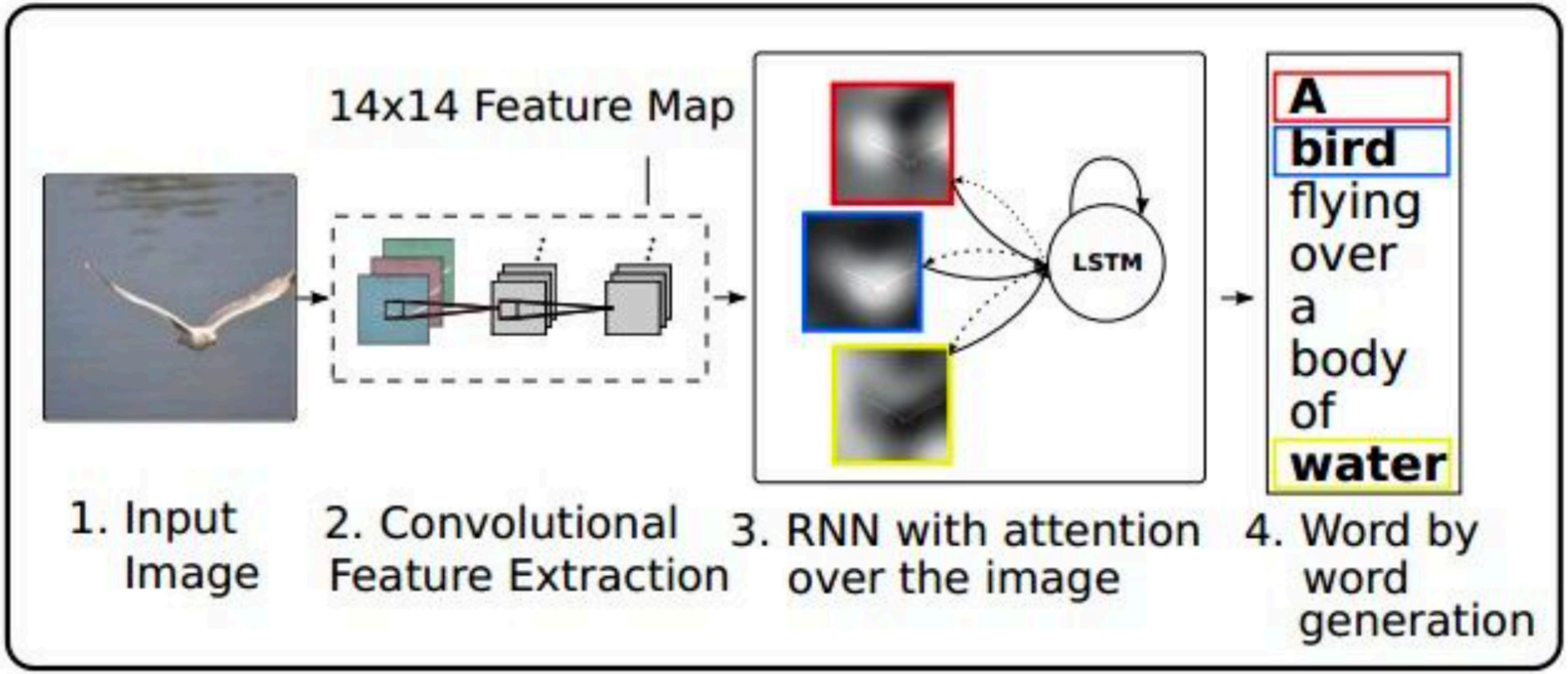
A man in a baseball uniform throwing a ball

* slide from Fei-Dei Li, Justin Johnson, Serena Yeung, **cs231n Stanford**

Applications: Image Captioning with Attention

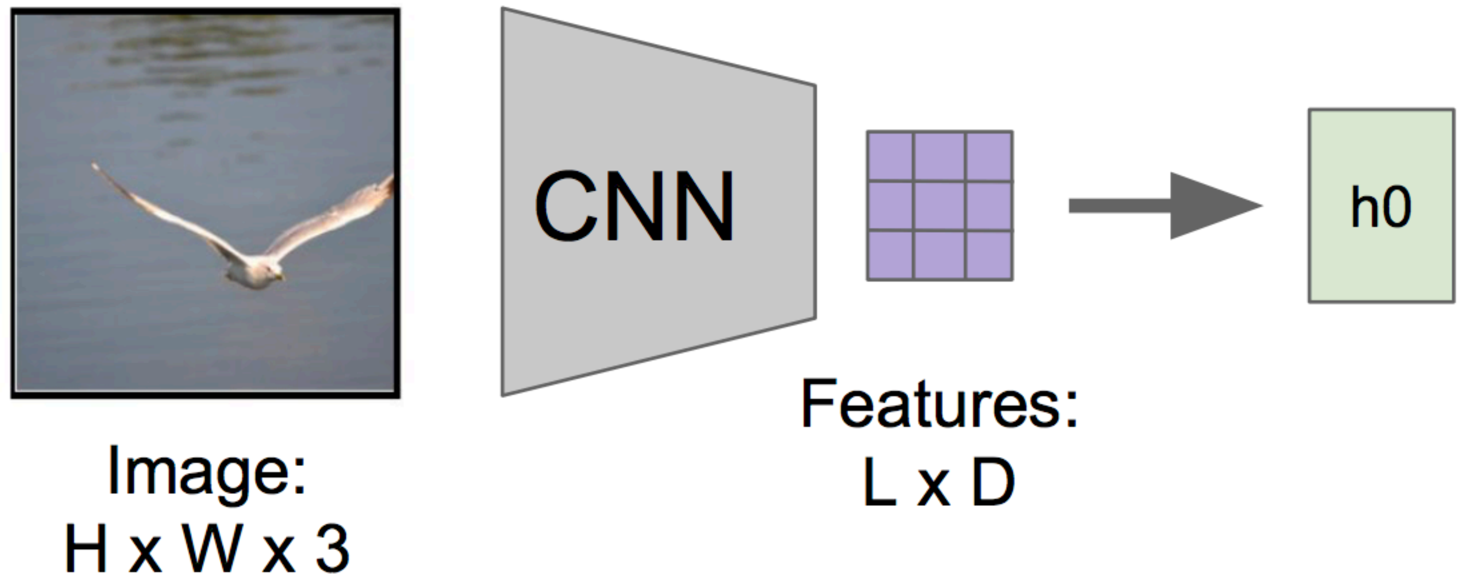
[Xu et al., ICML 2015]

RNN focuses its attention at a different spatial location when generating each word



Applications: Image Captioning with Attention

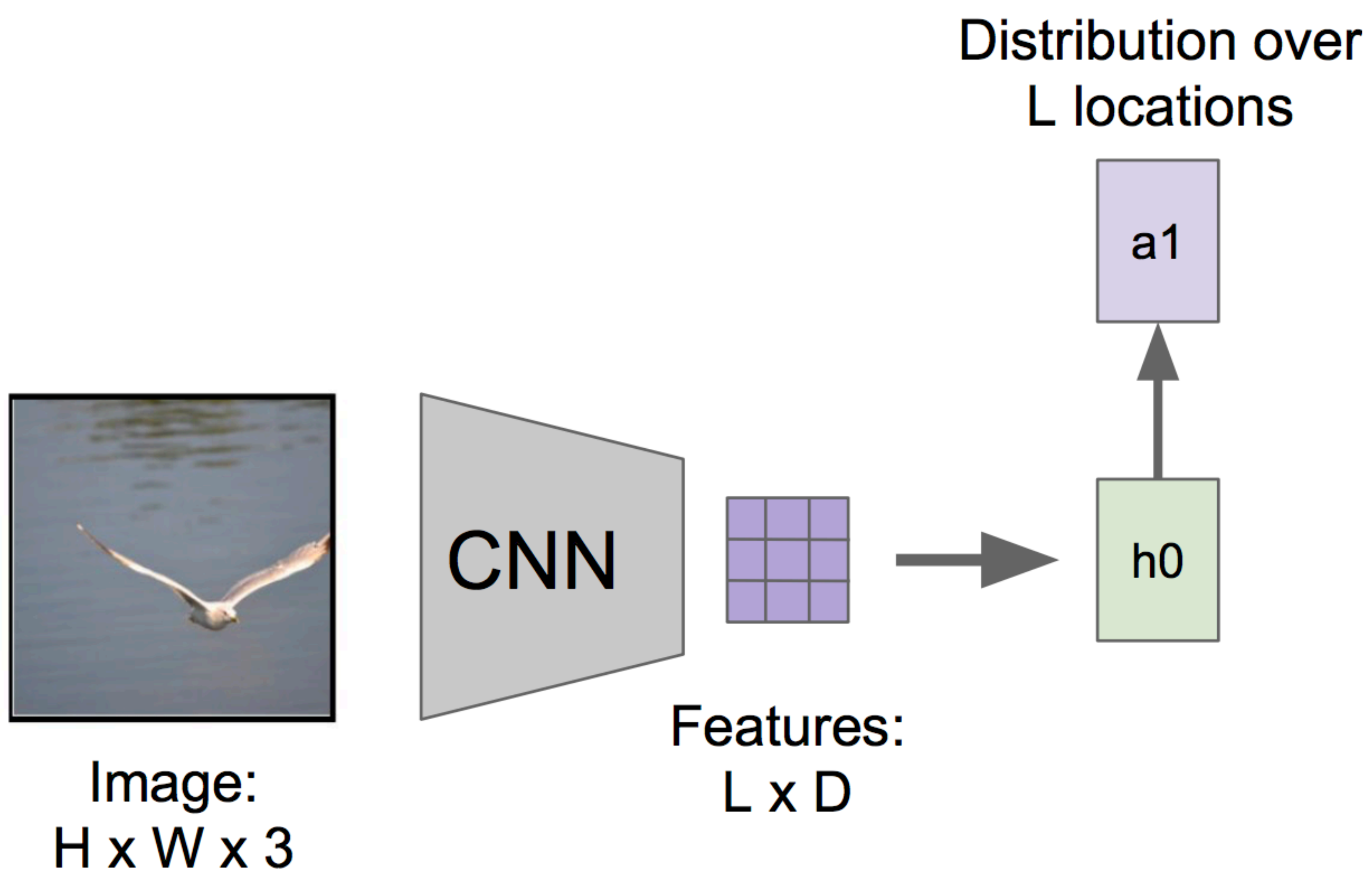
[Xu et al., ICML 2015]



* slide from Fei-Dei Li, Justin Johnson, Serena Yeung, **cs231n Stanford**

Applications: Image Captioning with Attention

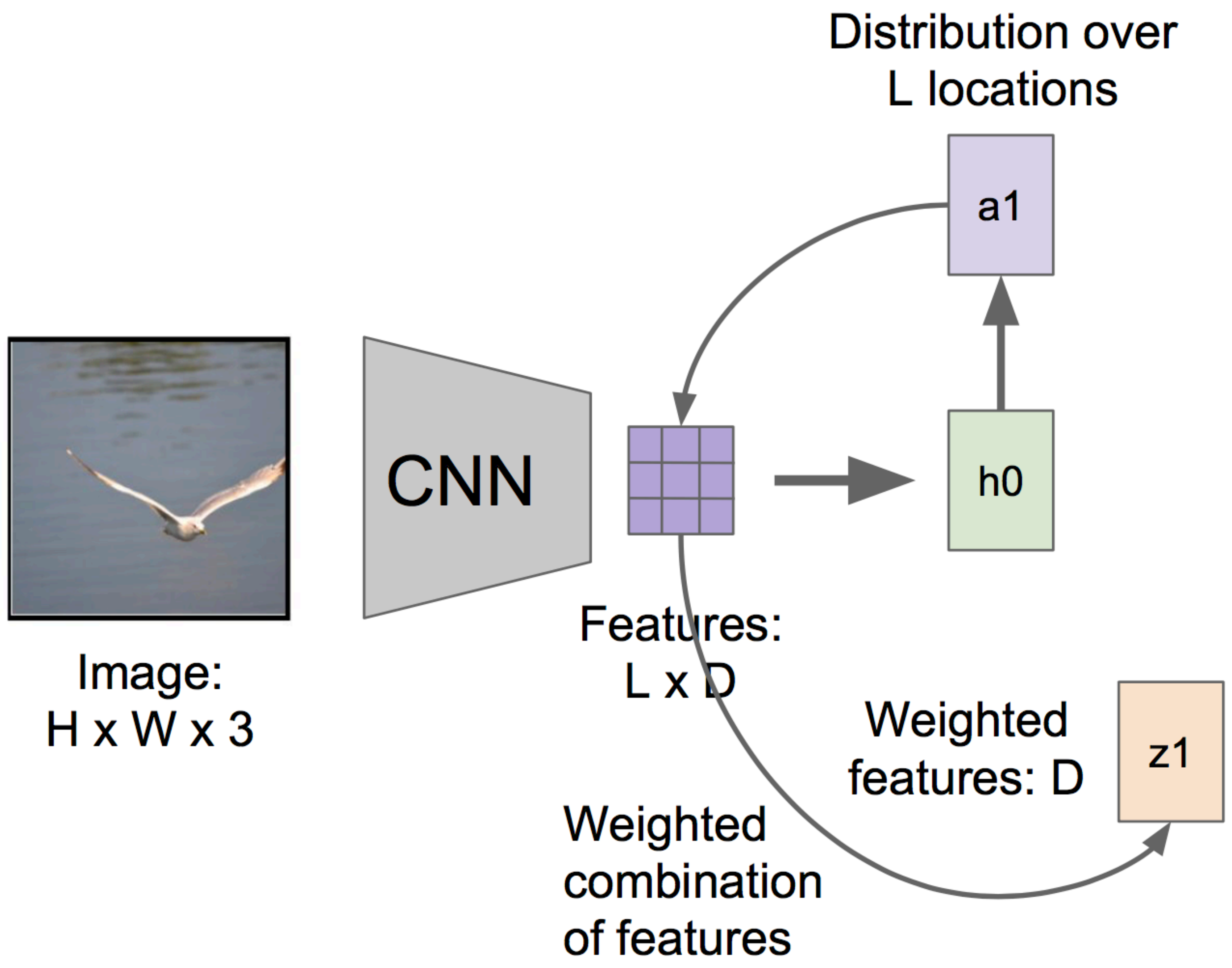
[Xu et al., ICML 2015]



* slide from Fei-Dei Li, Justin Johnson, Serena Yeung, **cs231n Stanford**

Applications: Image Captioning with Attention

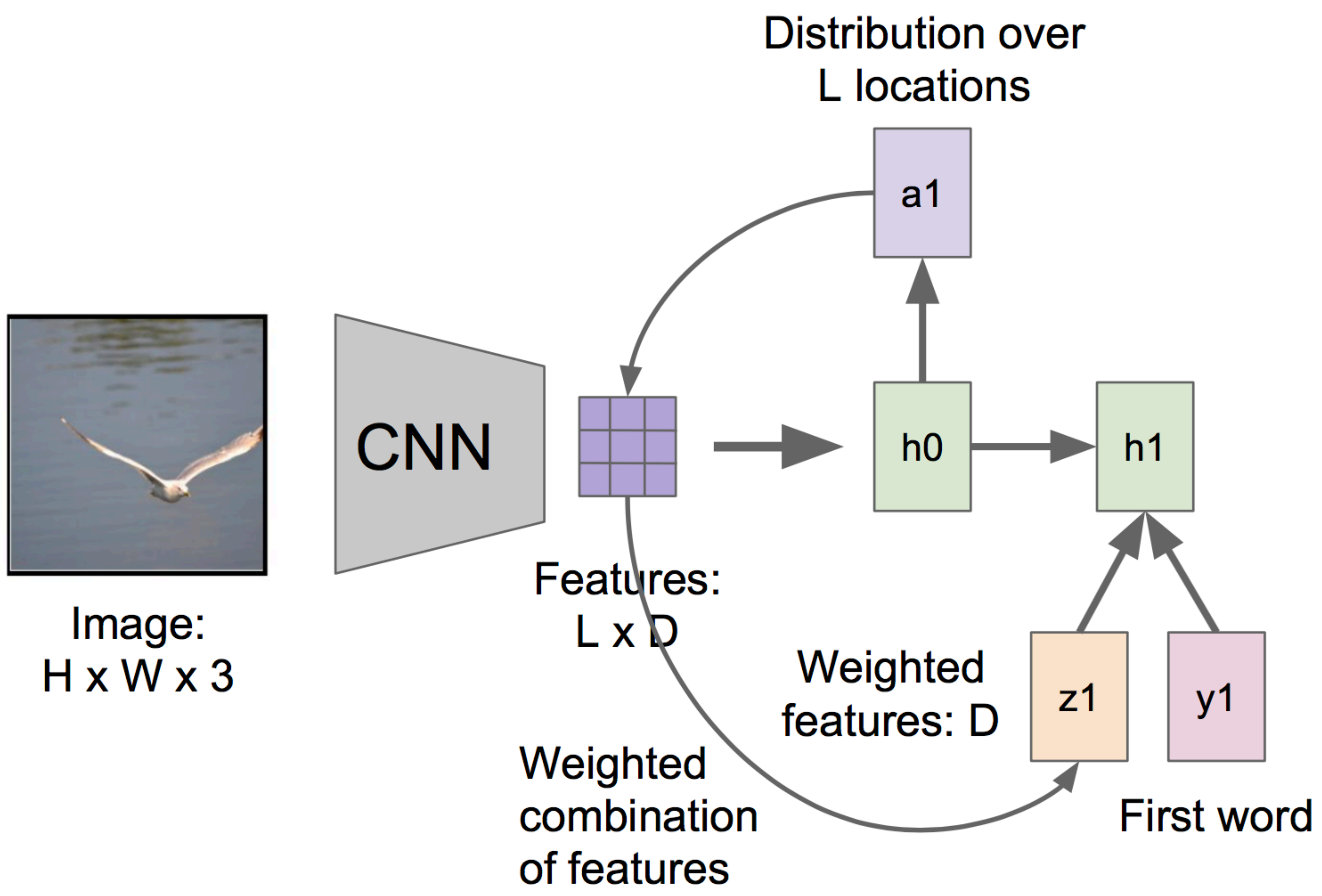
[Xu et al., ICML 2015]



$$z = \sum_{i=1}^{L} p_i v_i$$

Applications: Image Captioning with Attention

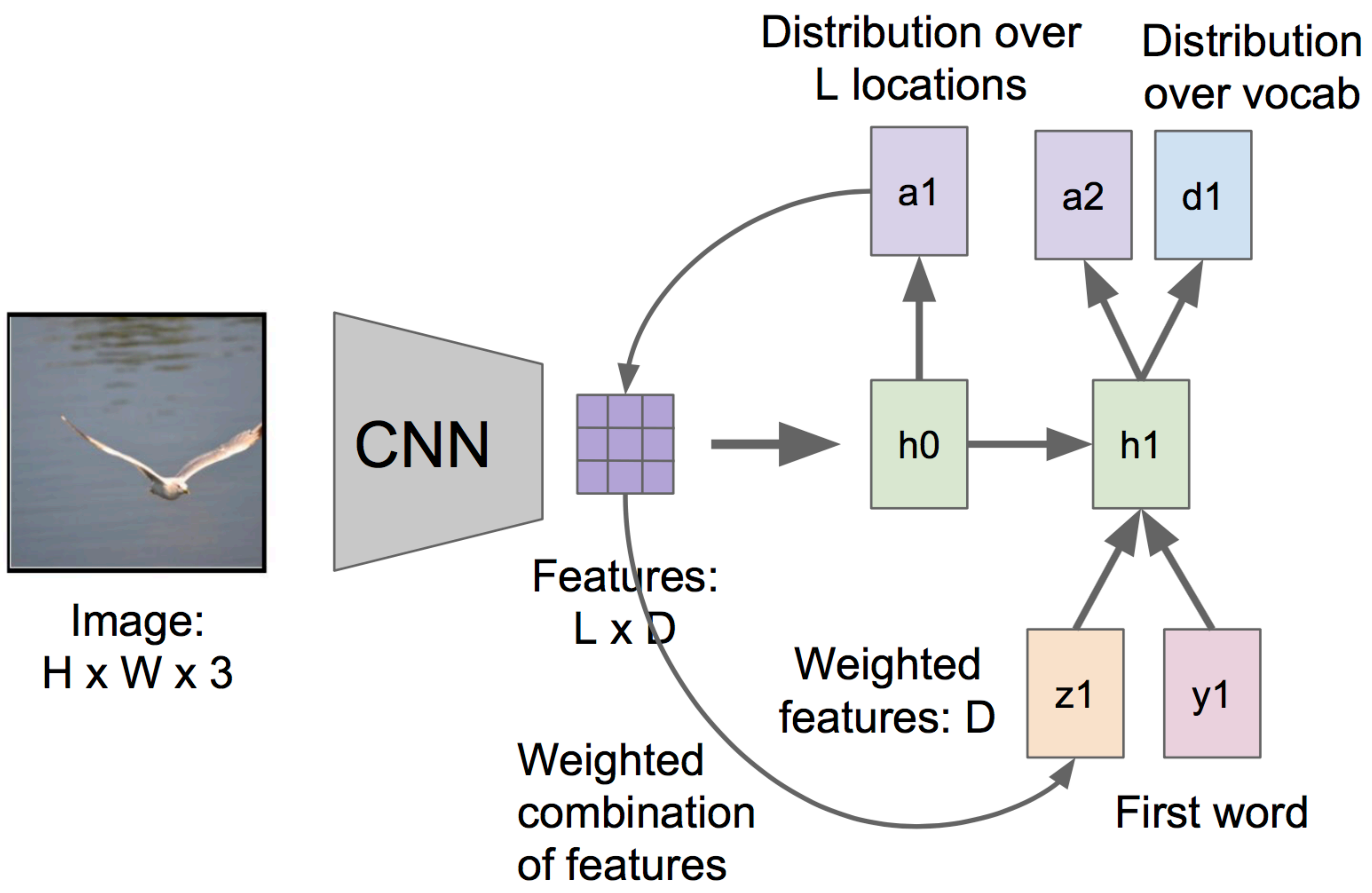
[Xu et al., ICML 2015]



* slide from Fei-Dei Li, Justin Johnson, Serena Yeung, **cs231n Stanford**

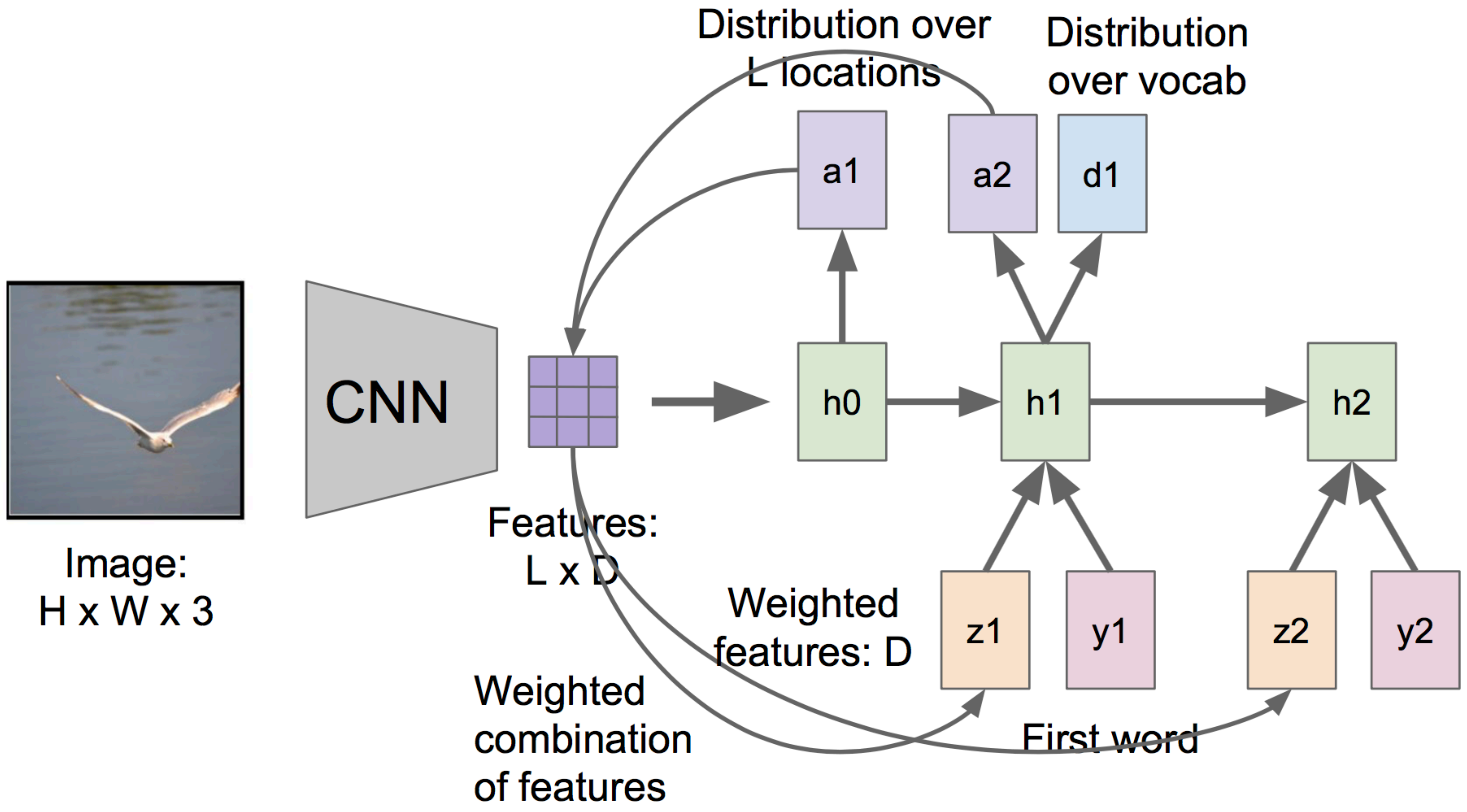
Applications: Image Captioning with Attention

[Xu et al., ICML 2015]



Applications: Image Captioning with Attention

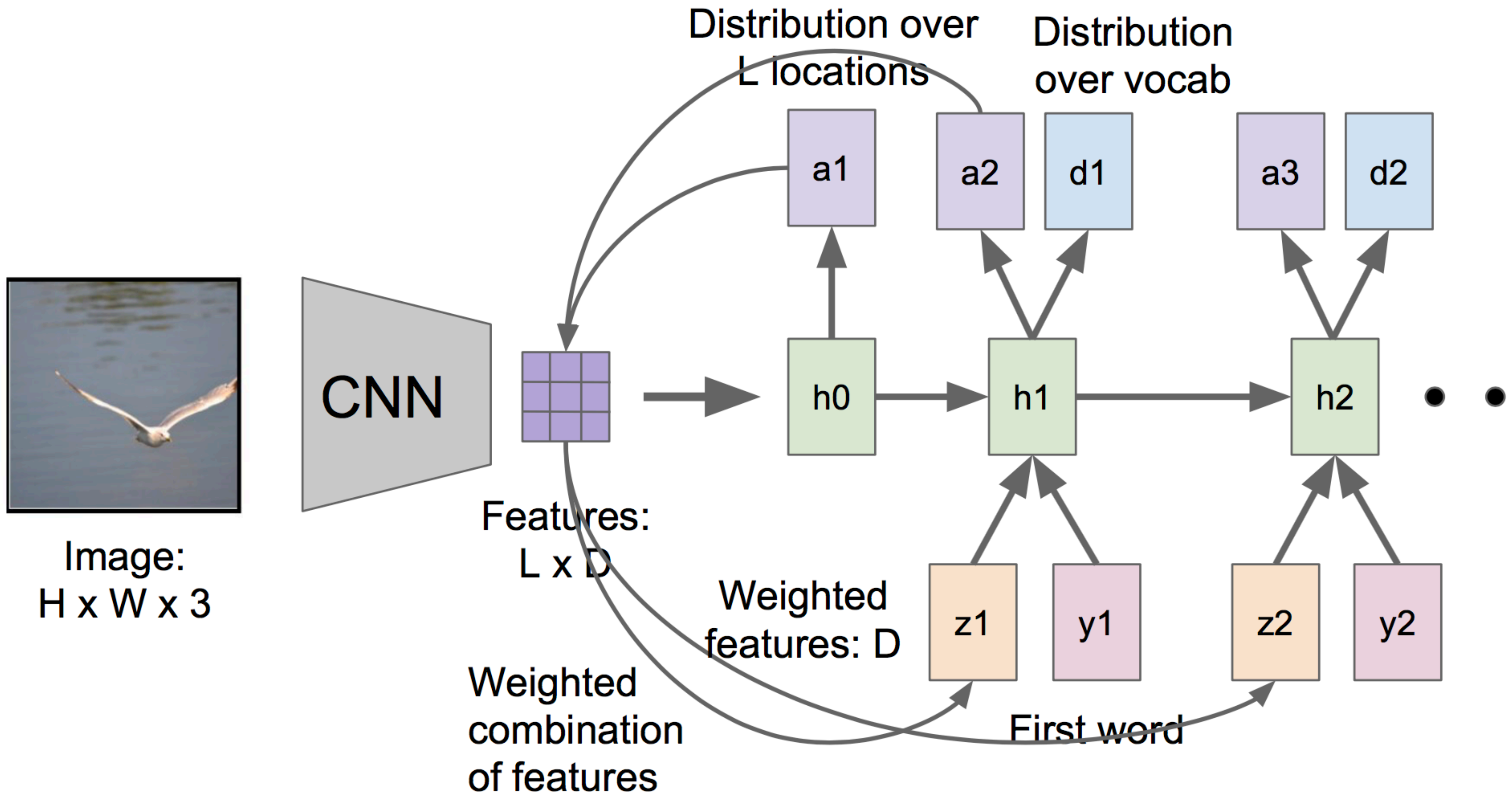
[Xu et al., ICML 2015]



* slide from Fei-Dei Li, Justin Johnson, Serena Yeung, **cs231n Stanford**

Applications: Image Captioning with Attention

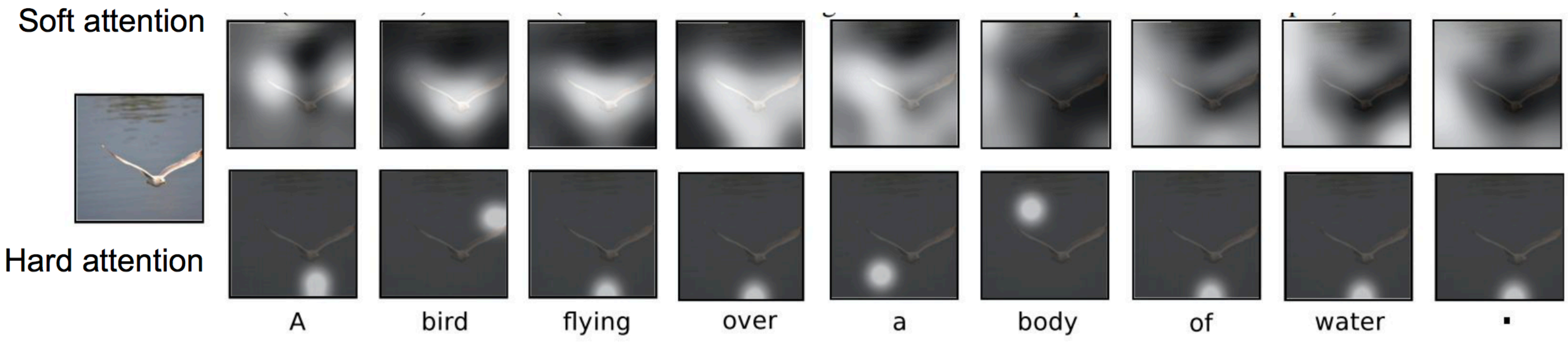
[Xu et al., ICML 2015]



* slide from Fei-Dei Li, Justin Johnson, Serena Yeung, **cs231n Stanford**

Applications: Image Captioning with Attention

[Xu et al., ICML 2015]



* slide from Fei-Dei Li, Justin Johnson, Serena Yeung, **cs231n Stanford**

Applications: Image Captioning with Attention

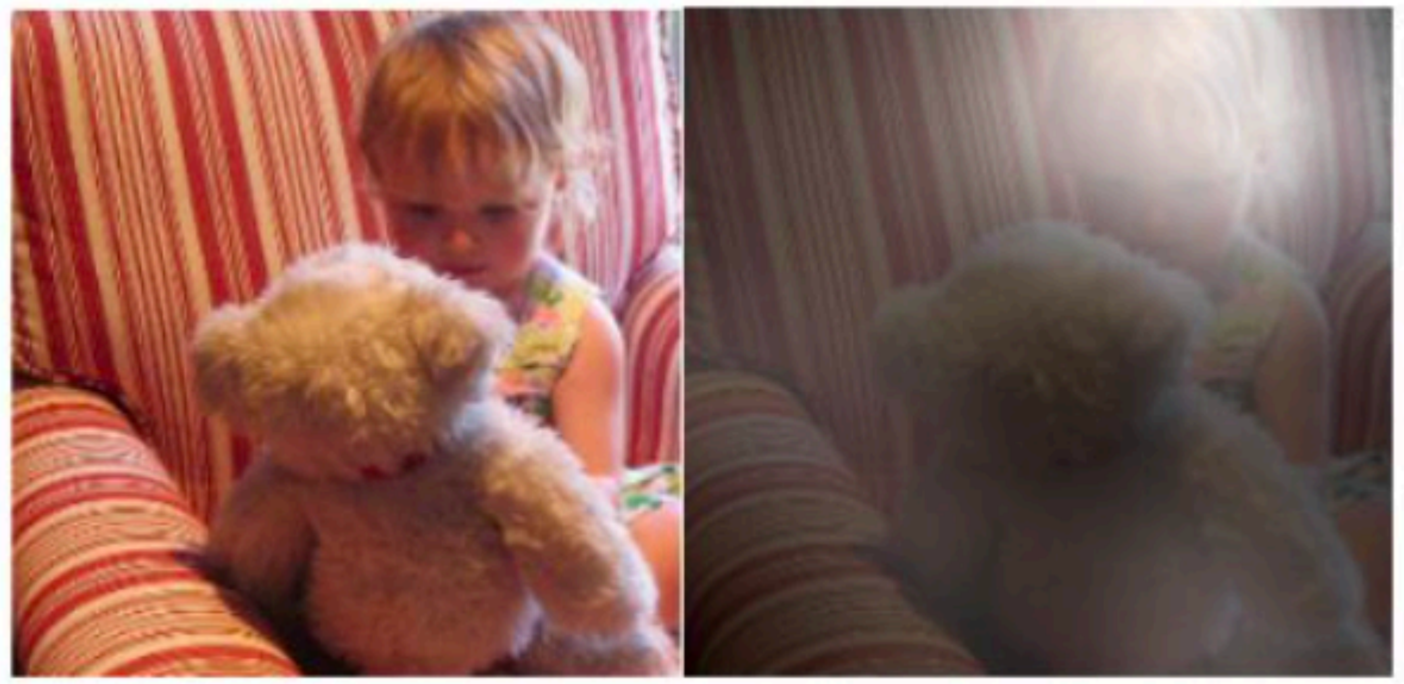
[Xu et al., ICML 2015]

Good results

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.

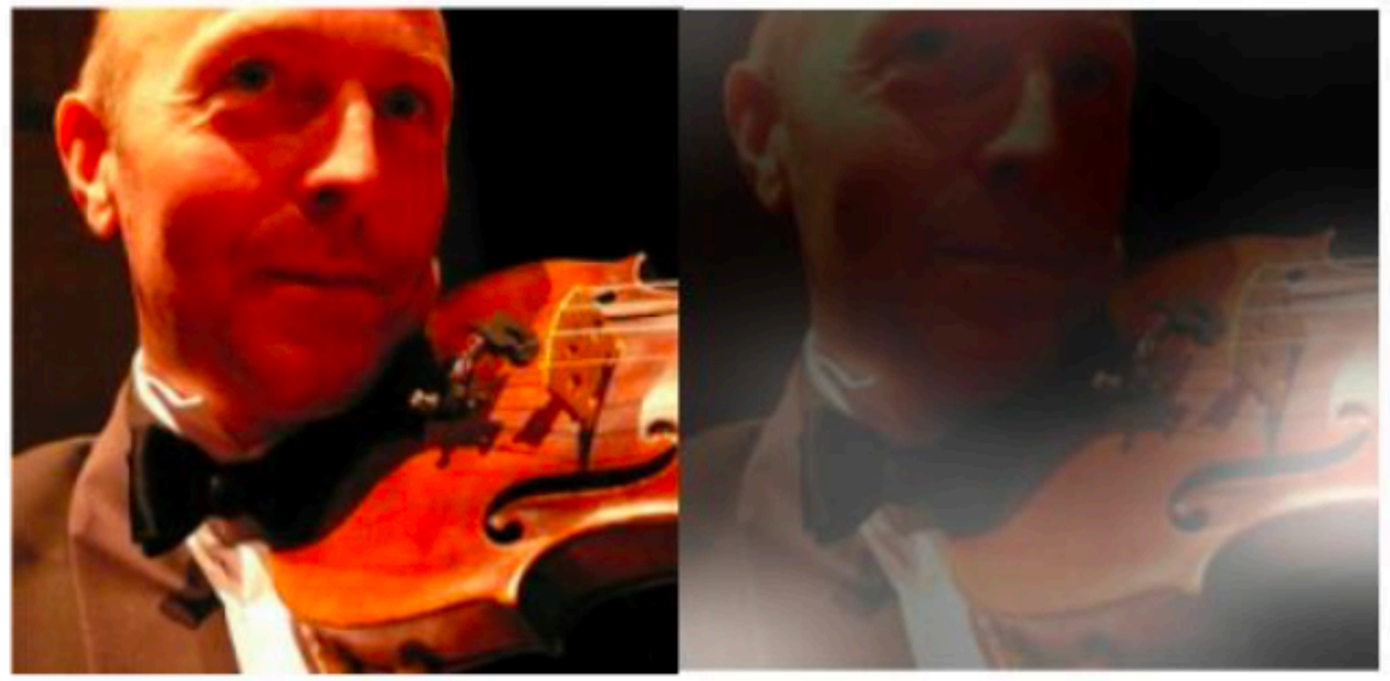
Applications: Image Captioning with Attention

[Xu et al., ICML 2015]

Failure results

A large white bird standing in a forest.

A woman holding a clock in her hand.



A man wearing a hat and
a hat on a skateboard.

A person is standing on a beach
with a surfboard.

A woman is sitting at a table
with a large pizza.

A man is talking on his cell phone
while another man watches.

* slide from Fei-Dei Li, Justin Johnson, Serena Yeung, **cs231n Stanford**

Applications: Typical Visual Question Answering (VQA)

Image

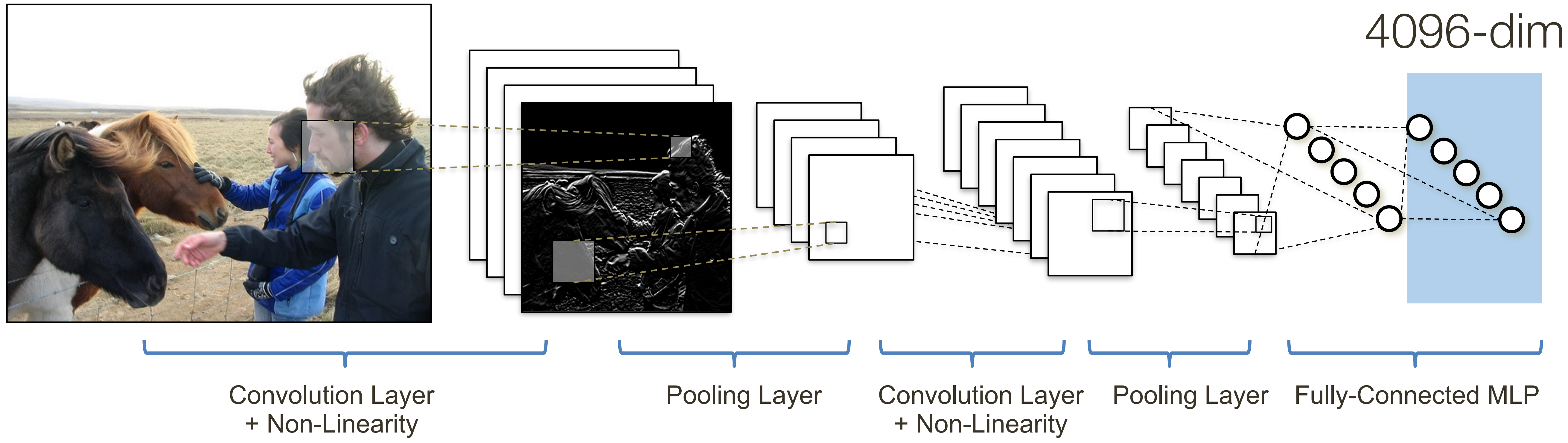
Question

"How many horses are in this image?"

* slide from Dhruv Batra

Applications: Typical Visual Question Answering (VQA)

Image Embedding (VGGNet)



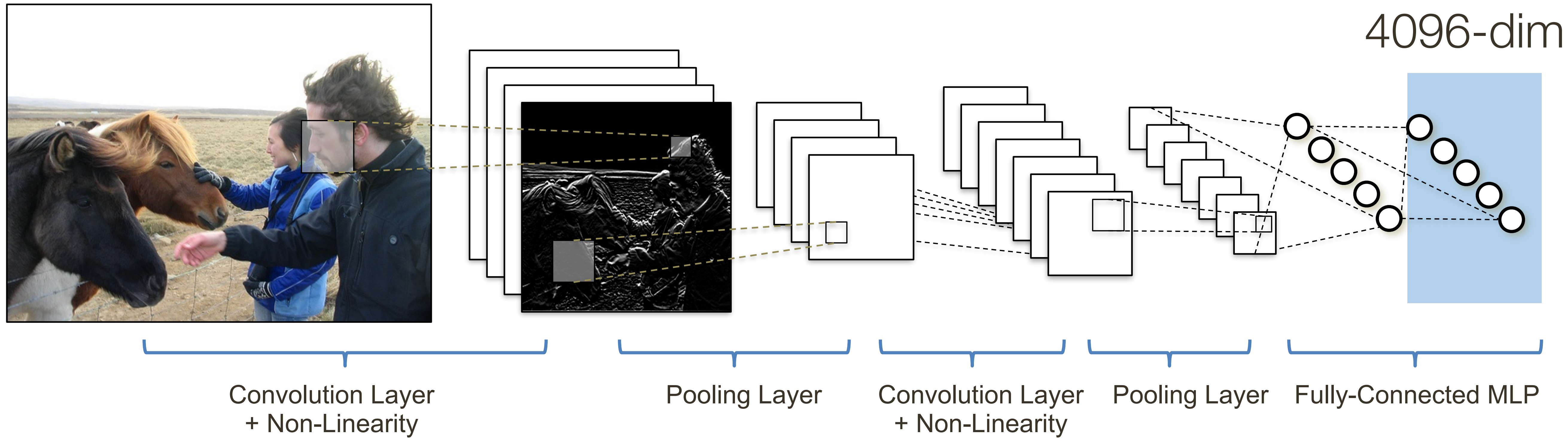
Question

"How many horses are in this image?"

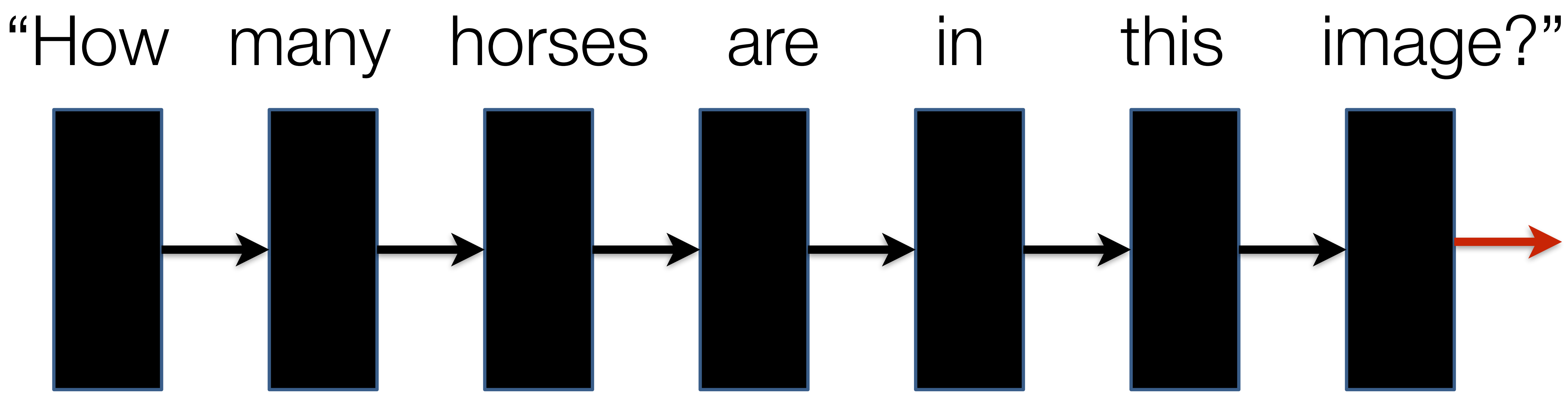
* slide from Dhruv Batra

Applications: Typical Visual Question Answering (VQA)

Image Embedding (VGGNet)



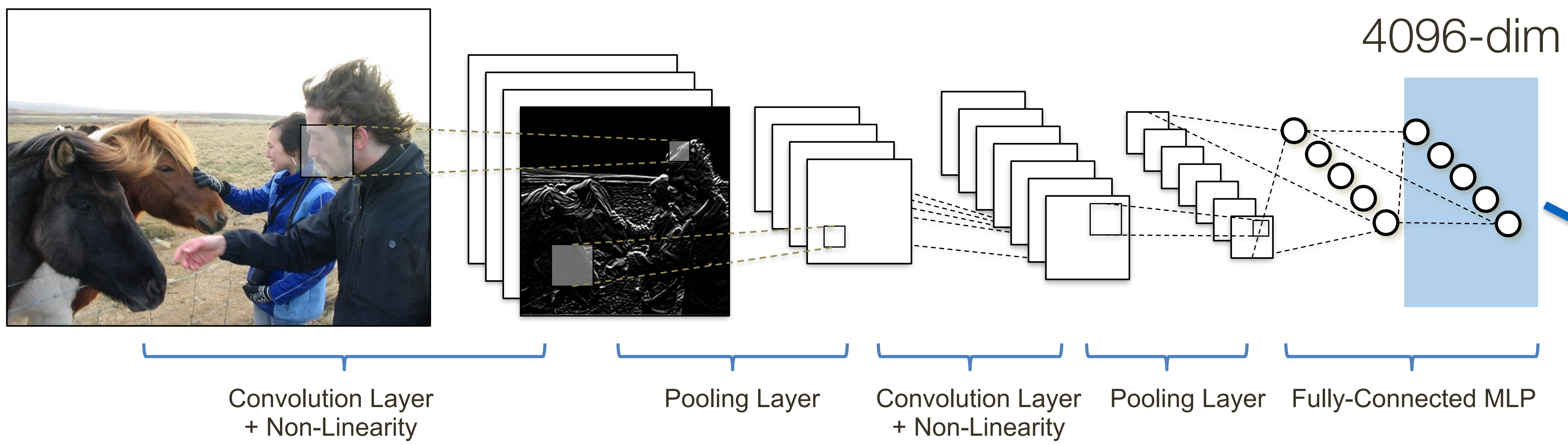
Question Embedding (LSTM)



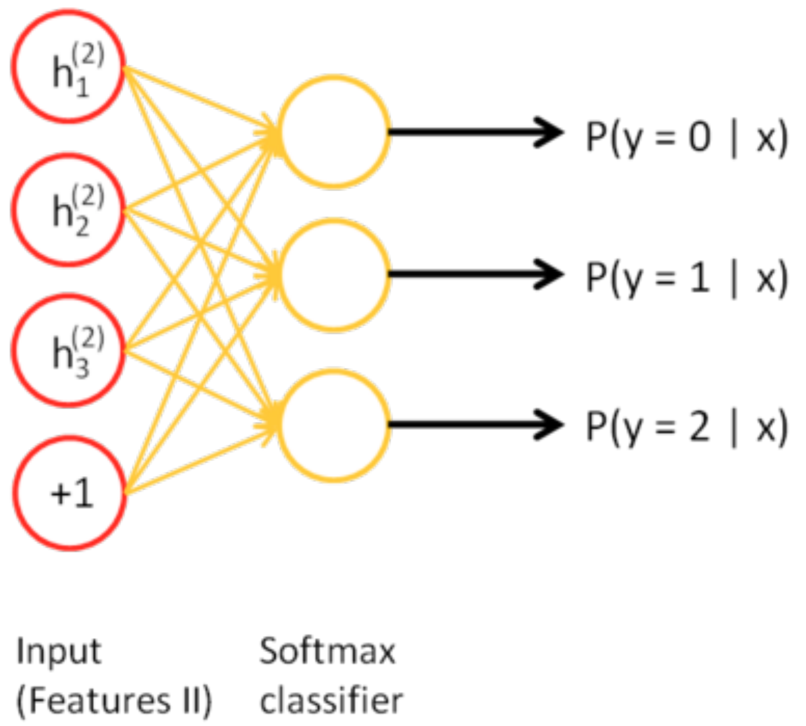
* slide from Dhruv Batra

Applications: Typical Visual Question Answering (VQA)

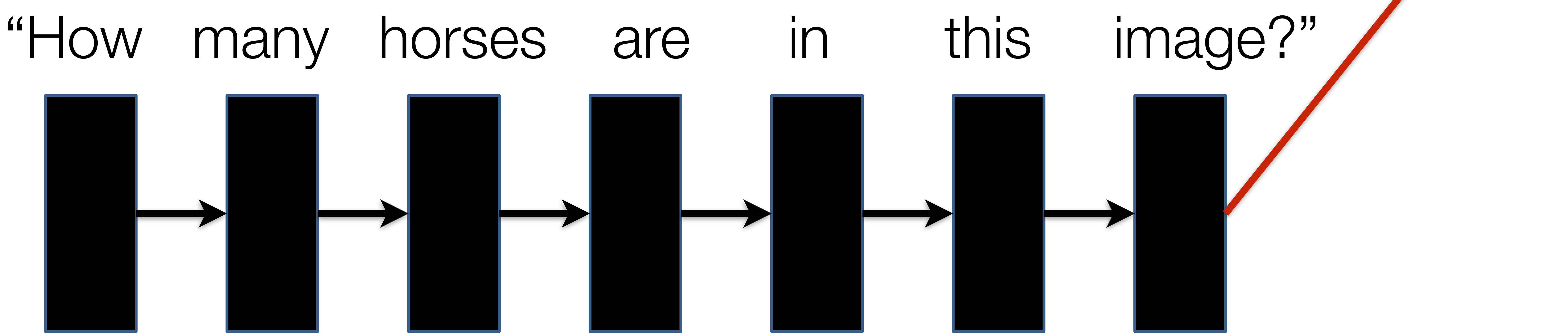
Image Embedding (VGGNet)



Neural Network
Softmax
over **top K answers**



Question Embedding (LSTM)



* slide from Dhruv Batra

Applications: Activity Detection

Activity: A collection of human/object movements with a particular semantic meaning

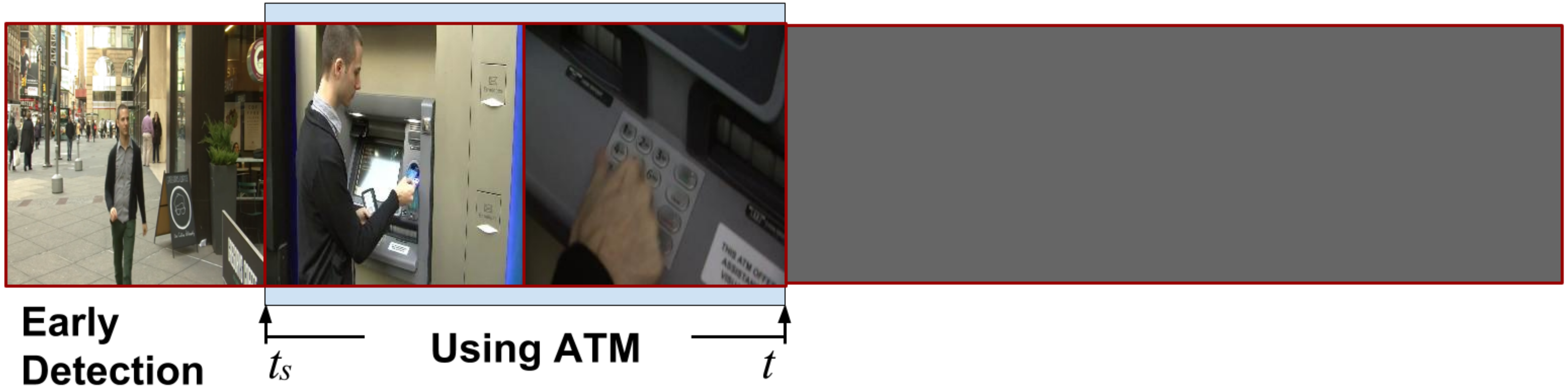
Action Recognition: Finding if a video segment contains such a movement

Action Detection: Finding a segment (beginning and start) and recognize the action in it

[Ma et al., 2014]

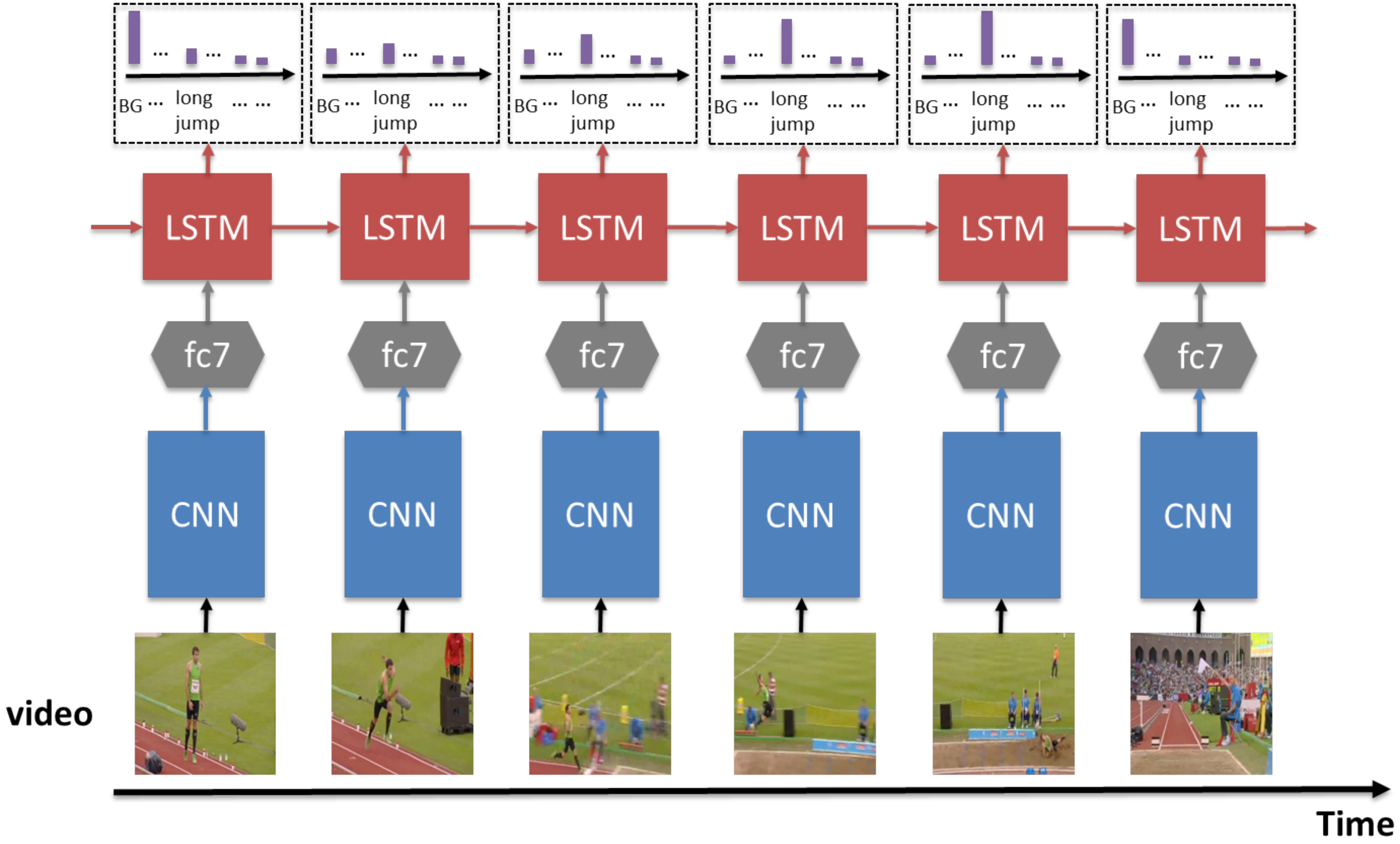
Applications: Activity Detection

Early Detection: Recognize when an action starts and try to predict which action is performed as quickly as possible.



[Ma et al., 2014]

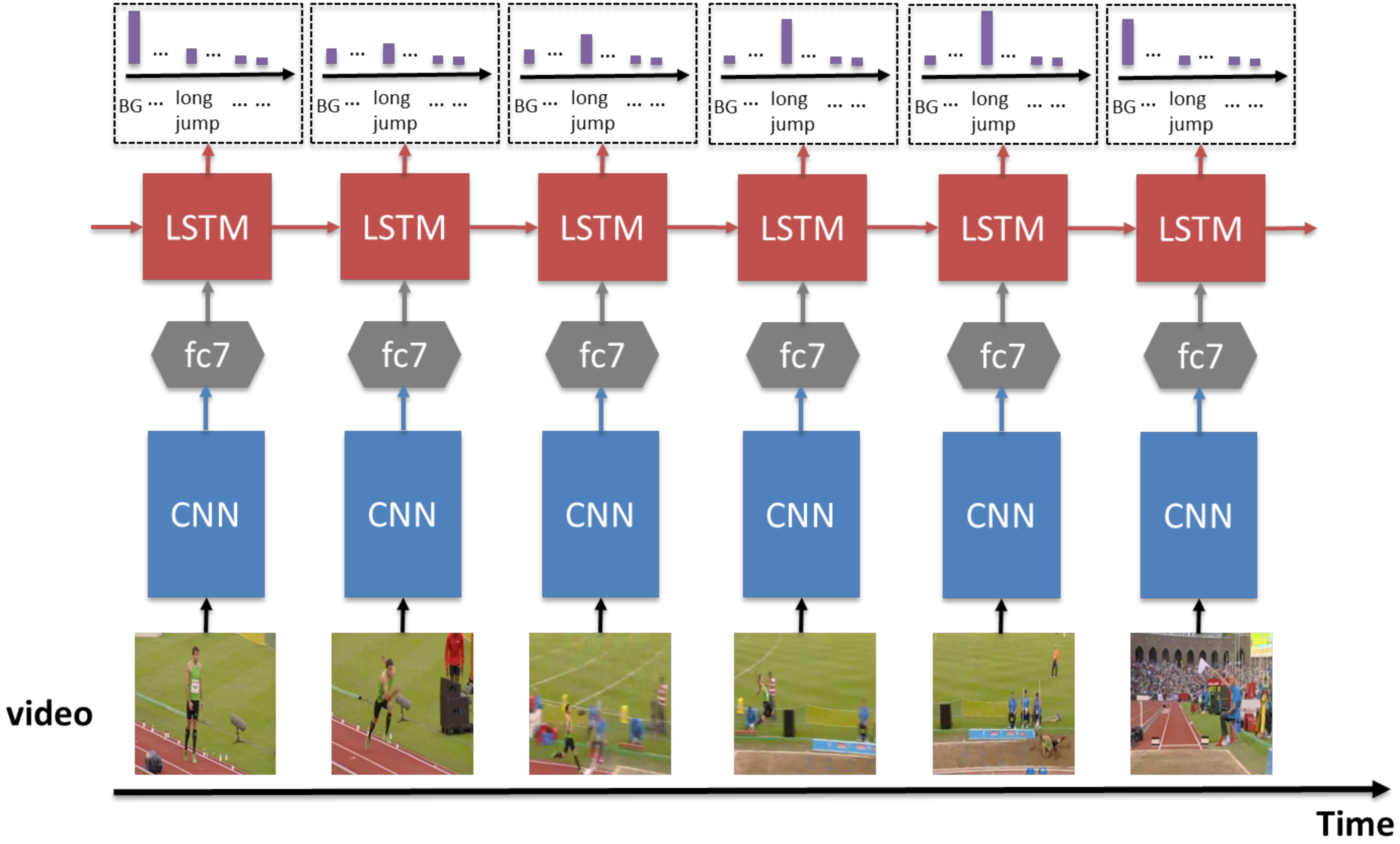
Applications: Activity Detection



[Ma et al., 2014]

Applications: Activity Detection

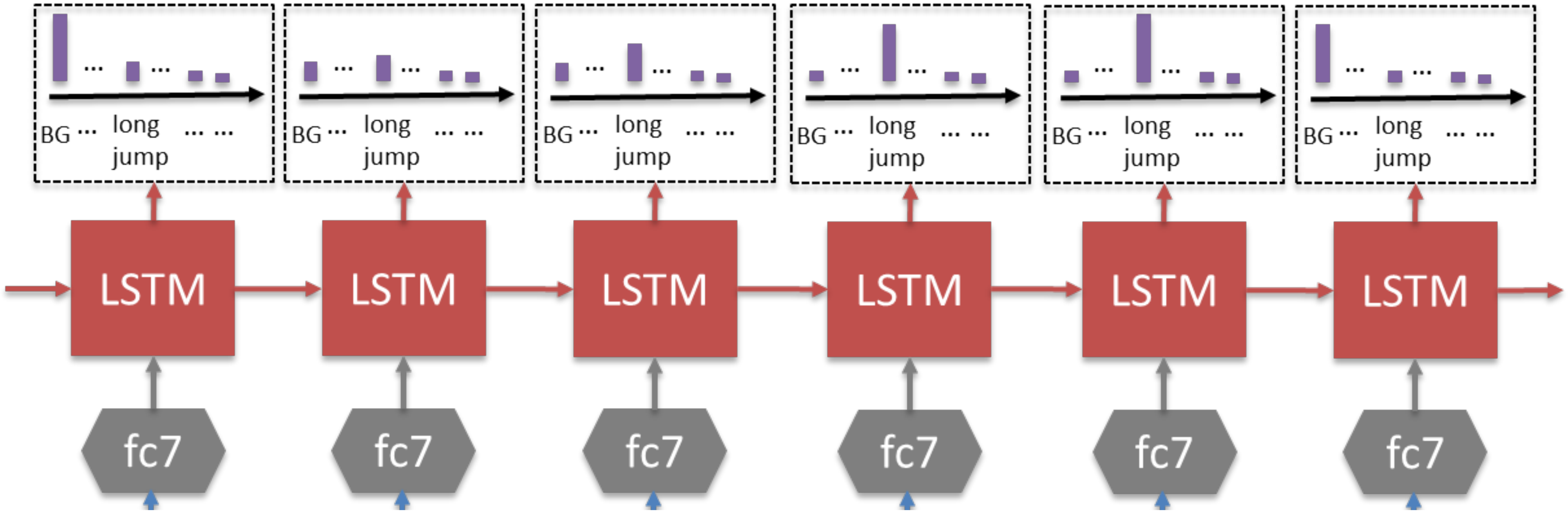
Penalty at every time step is the same

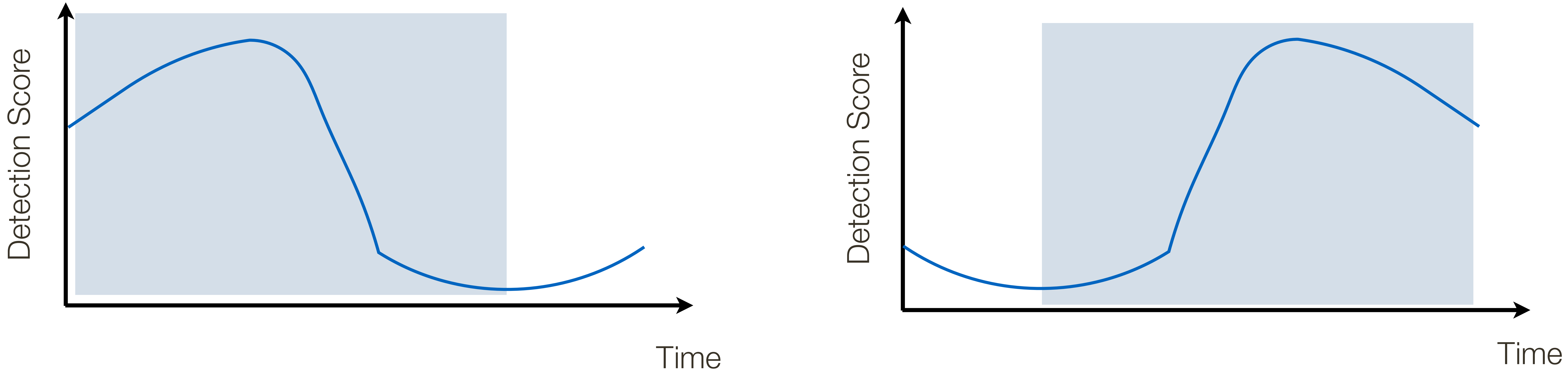


[Ma et al., 2014]

Applications: Activity Detection

Penalty at every time step is the same





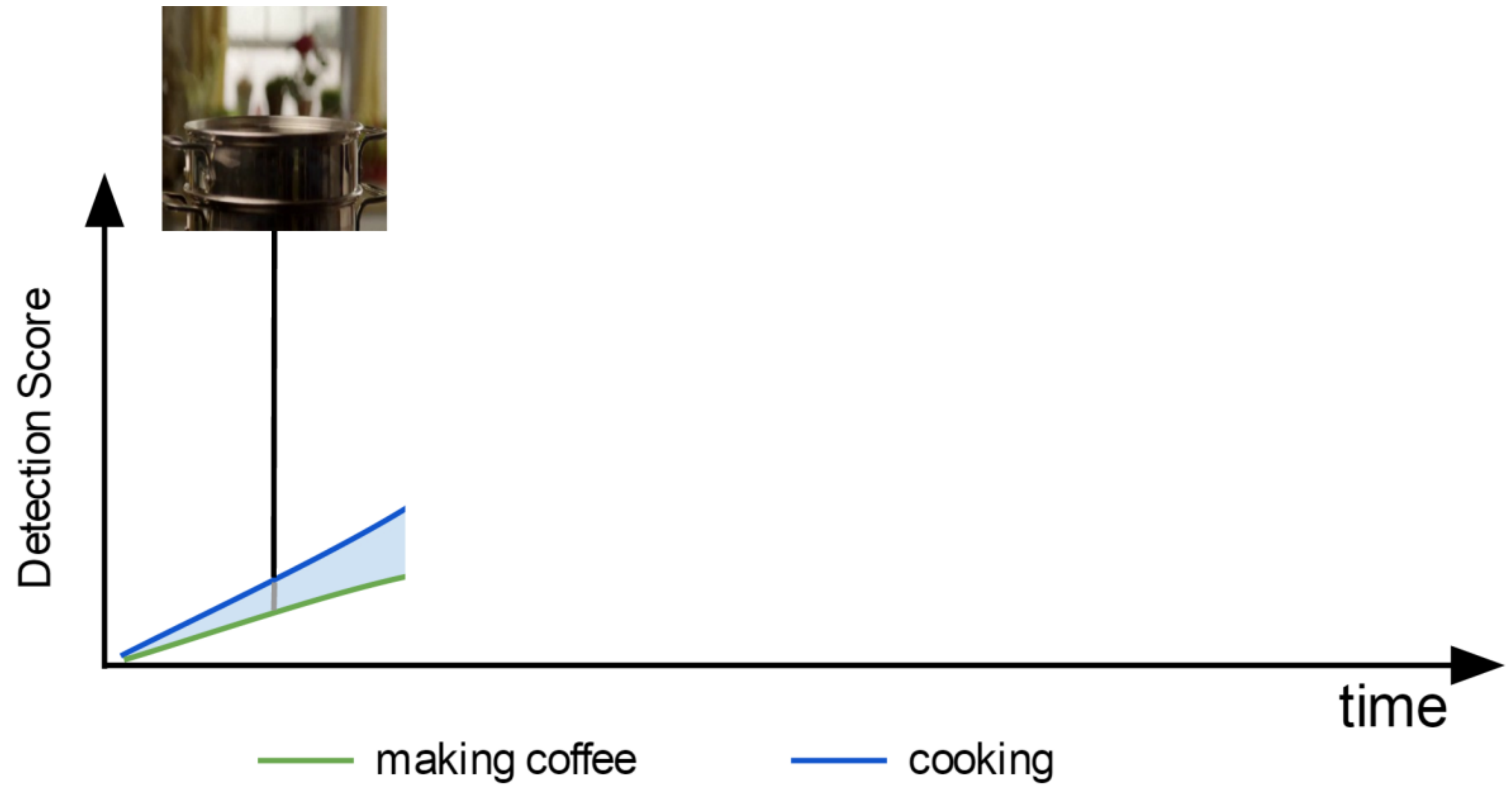
[Ma et al., 2014]

Applications: Activity Detection

As the detector sees more of an action, it should become more confident of

— Detecting the correct action class

— More confident that it is not the incorrect action class



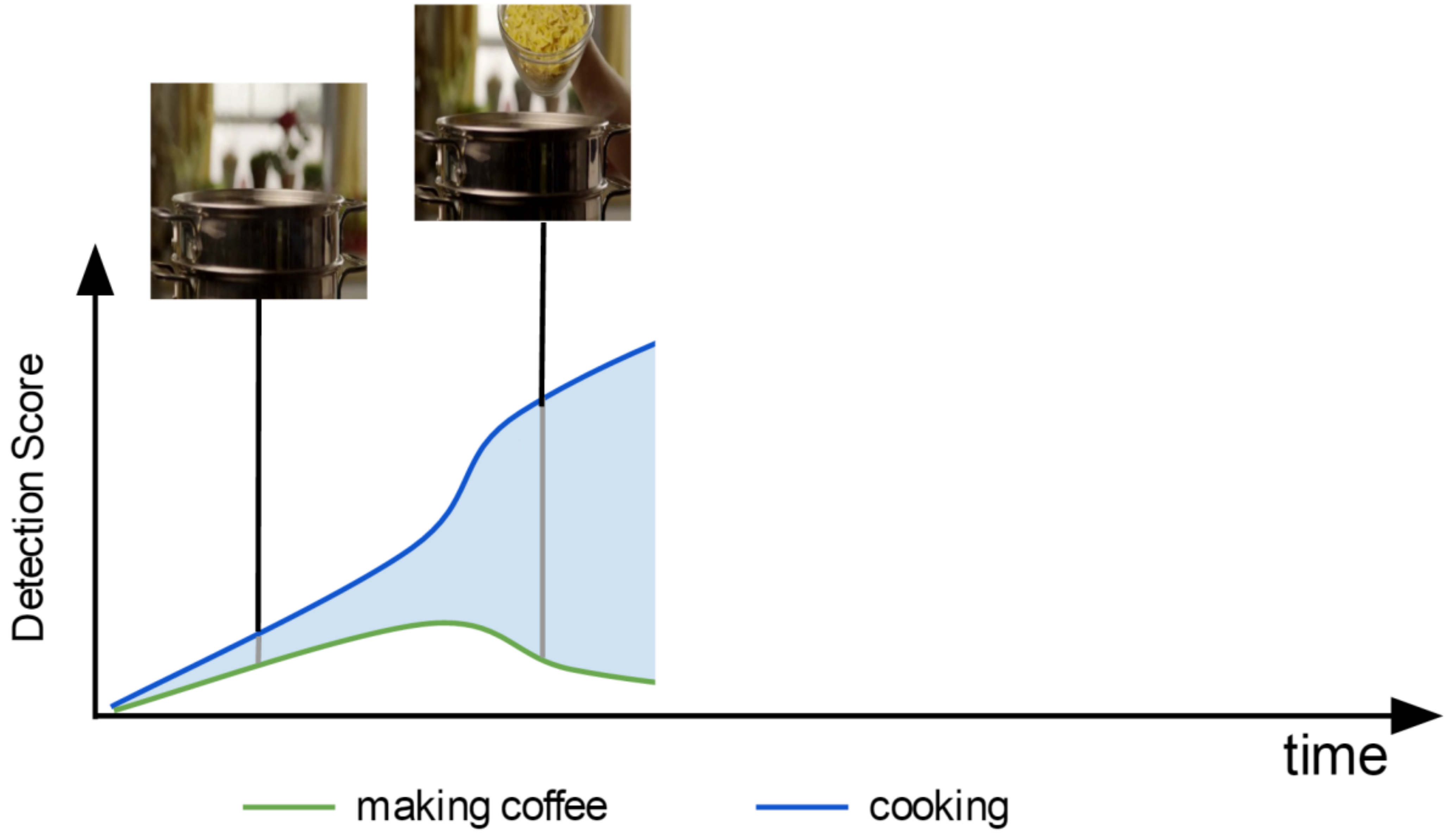
[Ma et al., 2014]

Applications: Activity Detection

As the detector sees more of an action, it should become more confident of

— Detecting the correct action class

— More confident that it is not the incorrect action class



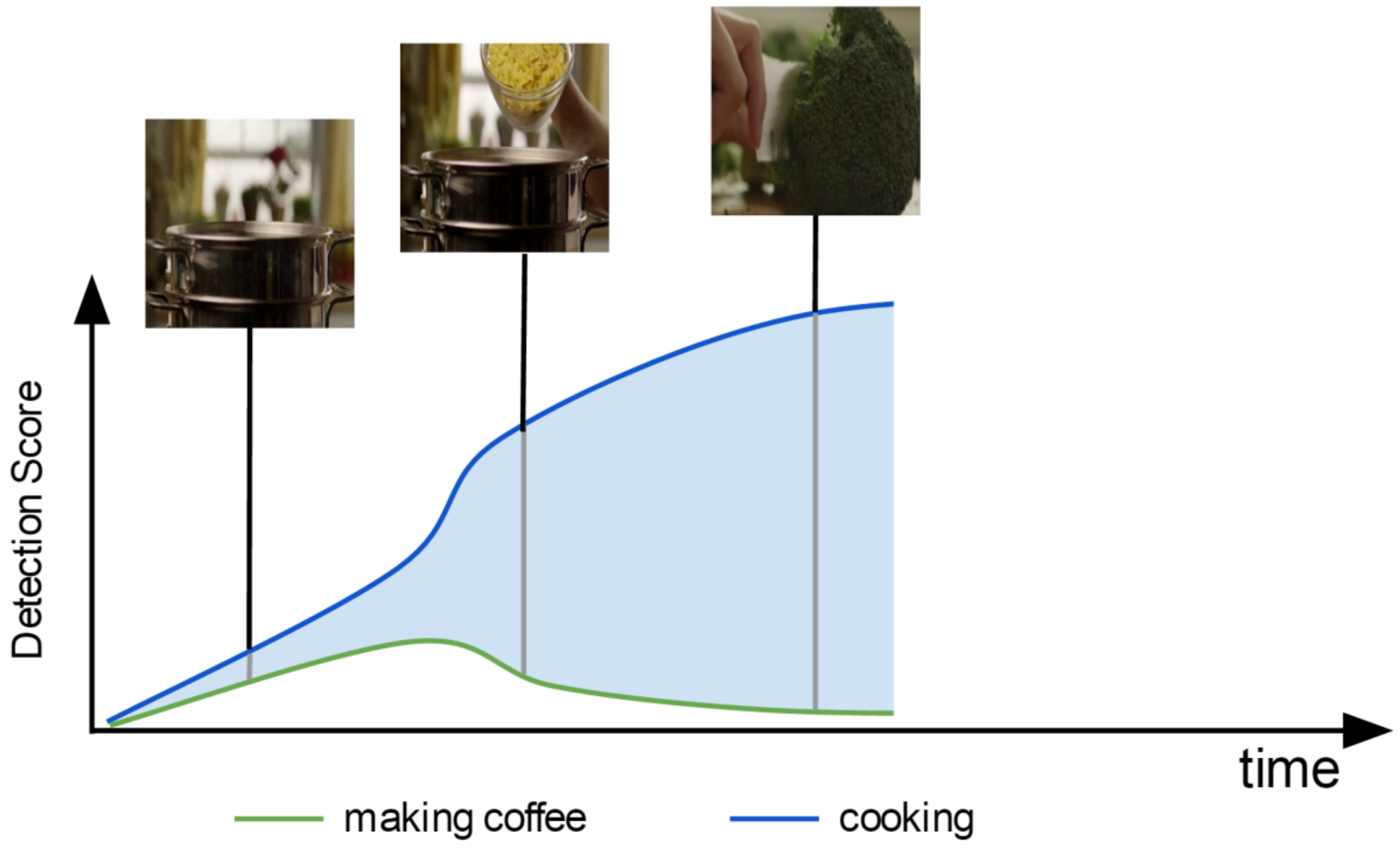
[Ma et al., 2014]

Applications: Activity Detection

As the detector sees more of an action, it should become more confident of

— Detecting the correct action class

— More confident that it is not the incorrect action class



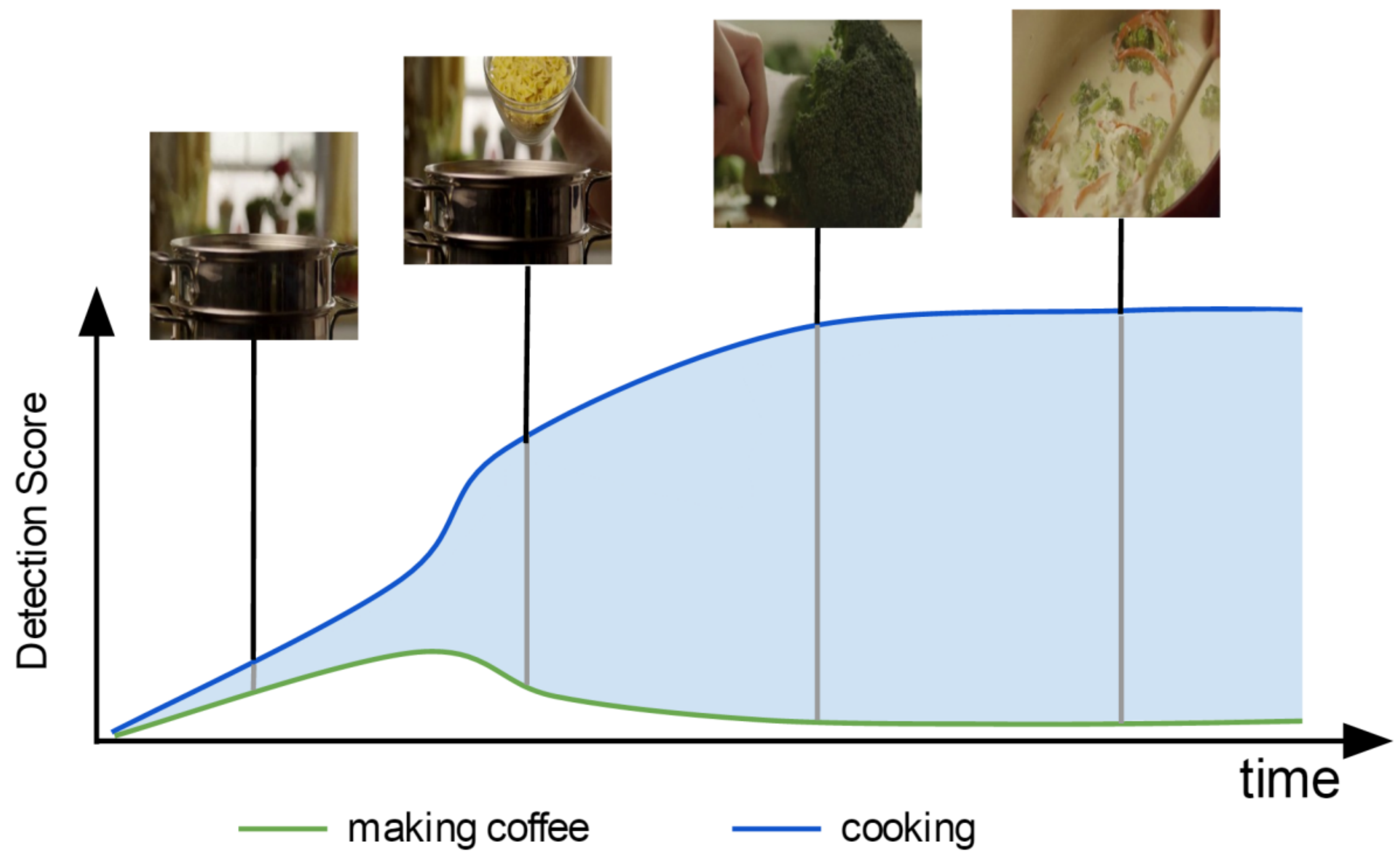
[Ma et al., 2014]

Applications: Activity Detection

As the detector sees more of an action, it should become more confident of

— Detecting the correct action class

— More confident that it is not the incorrect action class



[Ma et al., 2014]

New Class of Loss Functions

Classification loss at time t

Training loss at time t: $\mathcal{L}^t = \boxed{\mathcal{L}_c^t} + \lambda_r \boxed{\mathcal{L}_r^t}$

Ranking loss at time t

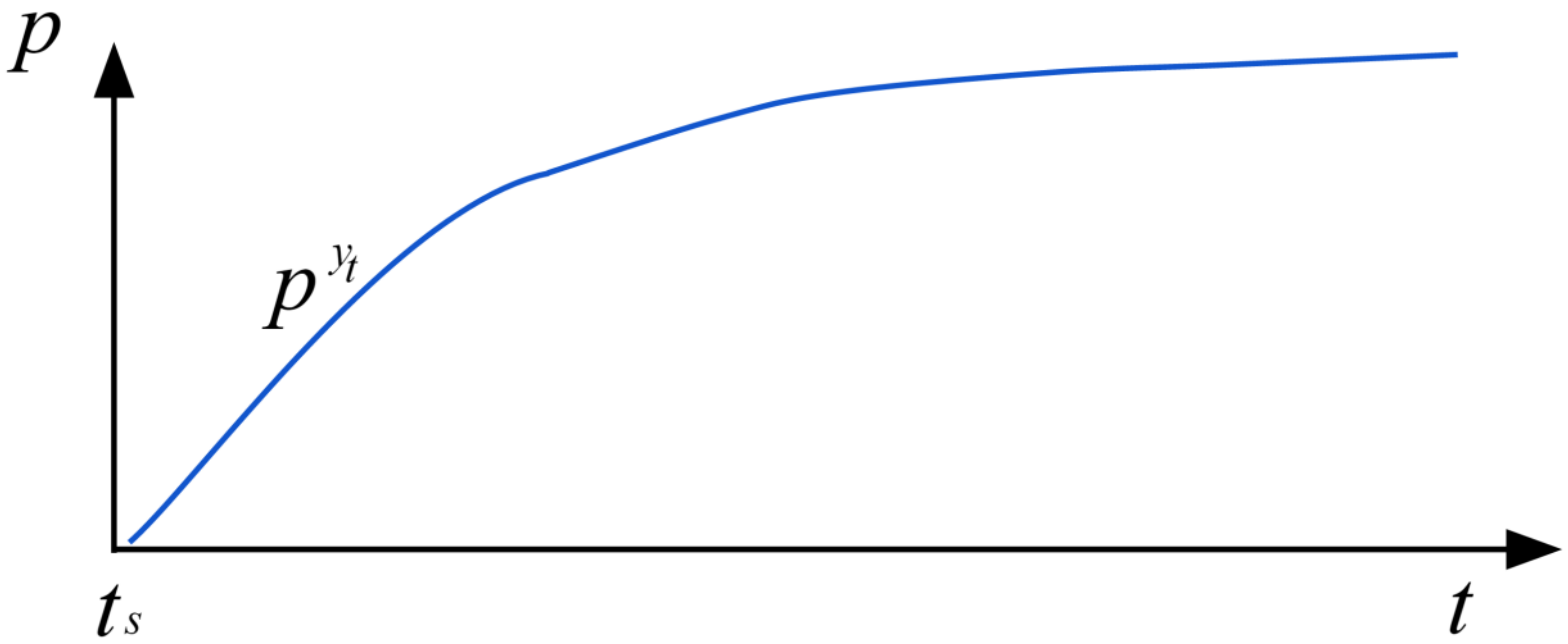
$\mathcal{L}_r^t$ is one of the following:

- $\mathcal{L}_s^t$ ranking loss on detection score
- $\mathcal{L}_m^t$ ranking loss on discriminative margin

[Ma et al., 2014]

Ranking Loss on Detection Score $\mathcal{L}_s^t$

Ideally what we want:

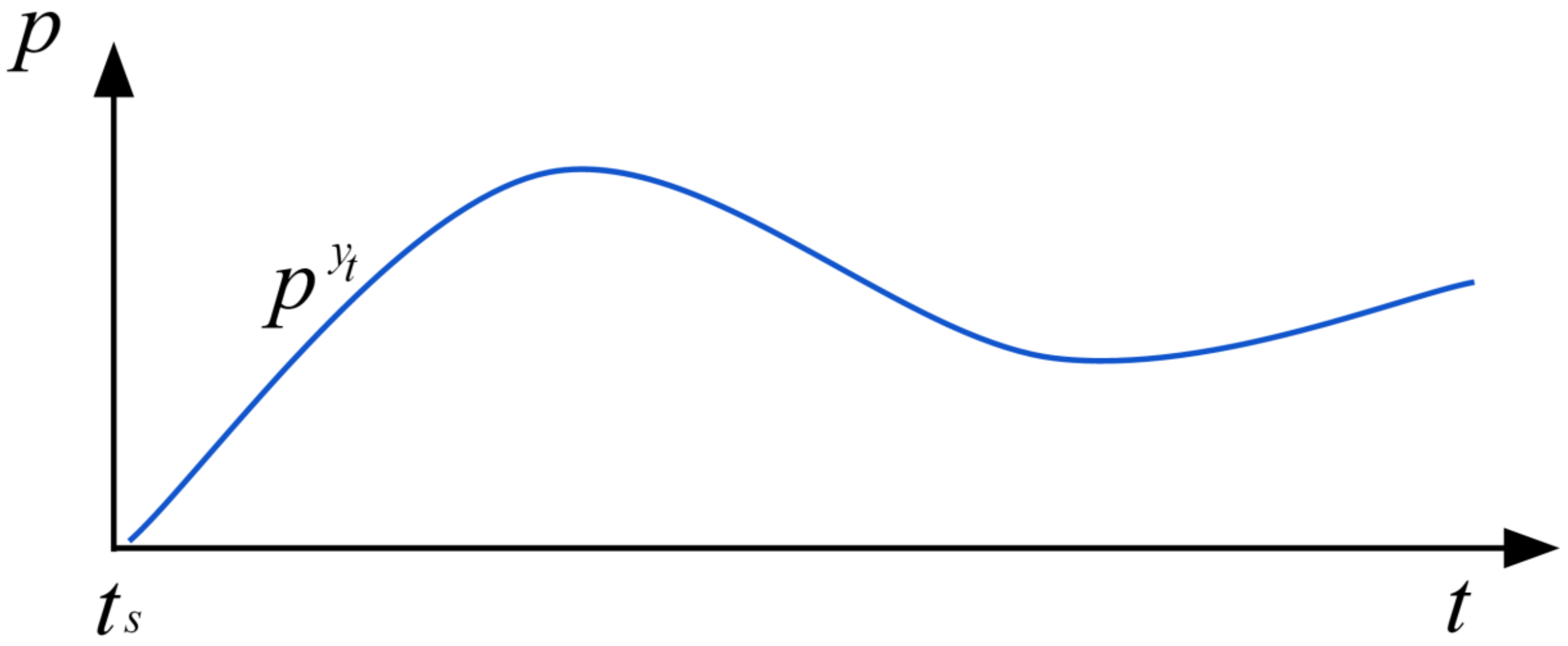


Prediction score of the ground truth action label

[Ma et al., 2014]

Ranking Loss on Detection Score $\mathcal{L}_s^t$

In **Practice:**

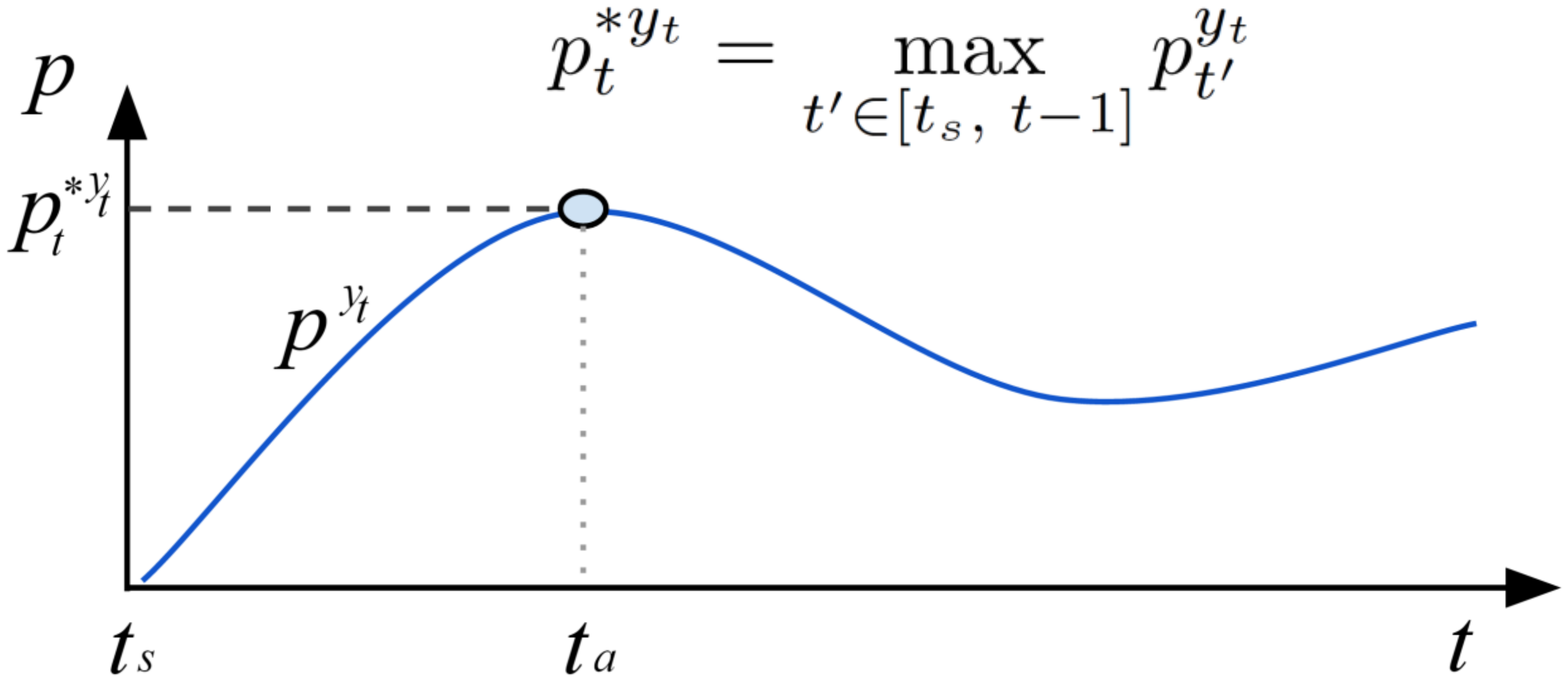


Prediction score of the ground truth action label

[Ma et al., 2014]

Ranking Loss on Detection Score $\mathcal{L}_s^t$

In **Practice:**

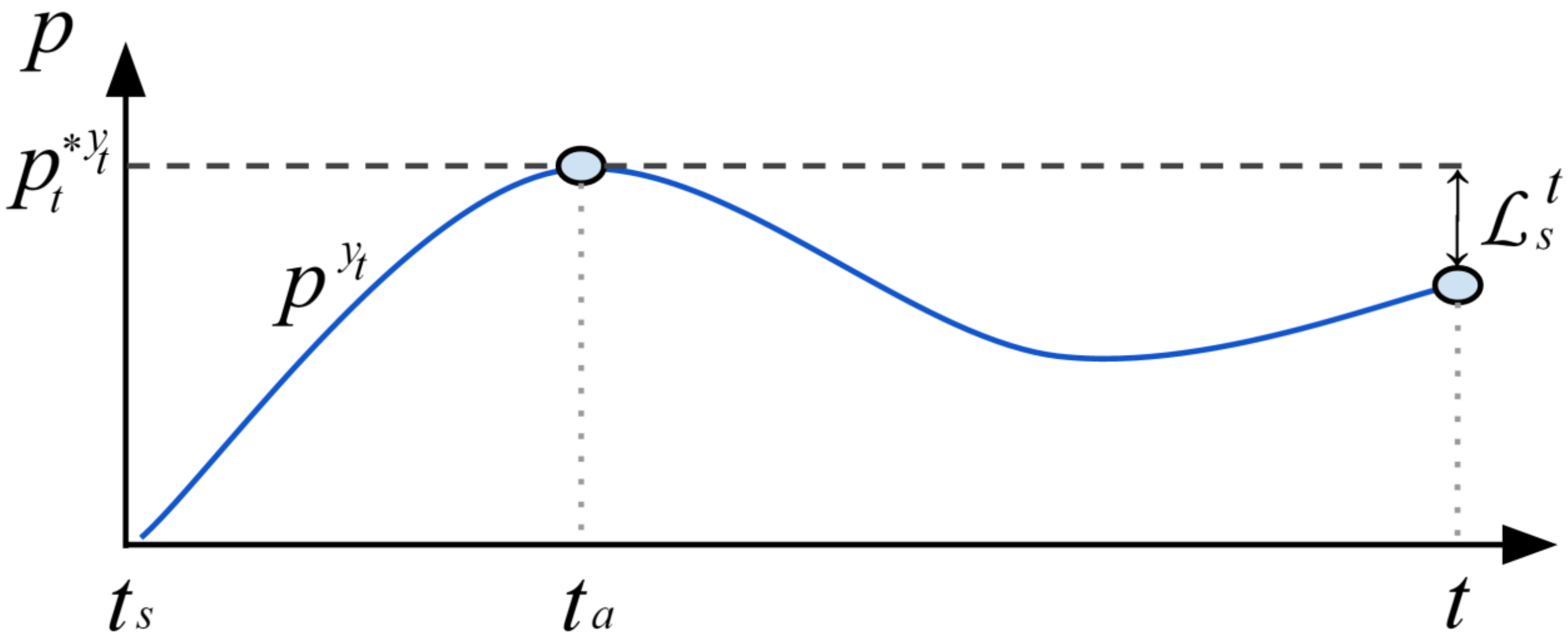


Prediction score of the ground truth action label

[Ma et al., 2014]

Ranking Loss on Detection Score $\mathcal{L}_s^t$

In **Practice:**



Prediction score of the ground truth action label

[Ma et al., 2014]

Applications: Activity Detection

Activity detection performance measured in mAP at different IOU thresholds

Model	$\alpha = 0.1$	$\alpha = 0.2$	$\alpha = 0.3$	$\alpha = 0.4$	$\alpha = 0.5$	$\alpha = 0.6$	$\alpha = 0.7$	$\alpha = 0.8$
Heilbron *et al.*	12.5%	11.9%	11.1%	10.4%	9.7%	-	-	-
CNN	30.1%	26.9%	23.4%	21.2%	18.9%	17.5%	16.5%	15.8%
LSTM	48.1%	44.3%	40.6%	35.6%	31.3%	28.3%	26.0%	24.6%
LSTM-m	52.6%	48.9%	45.1%	40.1%	35.1%	31.8%	29.1%	27.2%
LSTM-s	**54.0%**	**50.1%**	**46.3%**	**41.2%**	**36.4%**	**33.0%**	**30.4%**	**28.7%**

LSTM-m LSTM trained using both classification loss and rank loss on *discriminative margin*.

LSTM-s LSTM trained using both classification loss and rank loss on *detection score*.

[Ma et al., 2014]

Applications: Early Activity Detection

Activity early detection performance measured in mAP at different IOU thresholds

Model	$\alpha = 0.1$	$\alpha = 0.2$	$\alpha = 0.3$	$\alpha = 0.4$	$\alpha = 0.5$	$\alpha = 0.6$	$\alpha = 0.7$	$\alpha = 0.8$
CNN	27.0%	23.4%	20.4%	17.2%	14.6%	12.3%	11.0%	10.3%
LSTM	49.5%	44.7%	38.8%	33.9%	29.6%	25.6%	23.5%	22.4%
LSTM-m	52.6%	47.9%	41.5%	36.2%	31.4%	27.1%	24.8%	23.5%
LSTM-s	**55.1%**	**50.3%**	**44.0%**	**38.9%**	**34.1%**	**29.8%**	**27.4%**	**26.1%**

Note: first 3/10 of activity is seen by a detector

LSTM-m LSTM trained using both classification loss and rank loss on *discriminative margin*.

LSTM-s LSTM trained using both classification loss and rank loss on *detection score*.

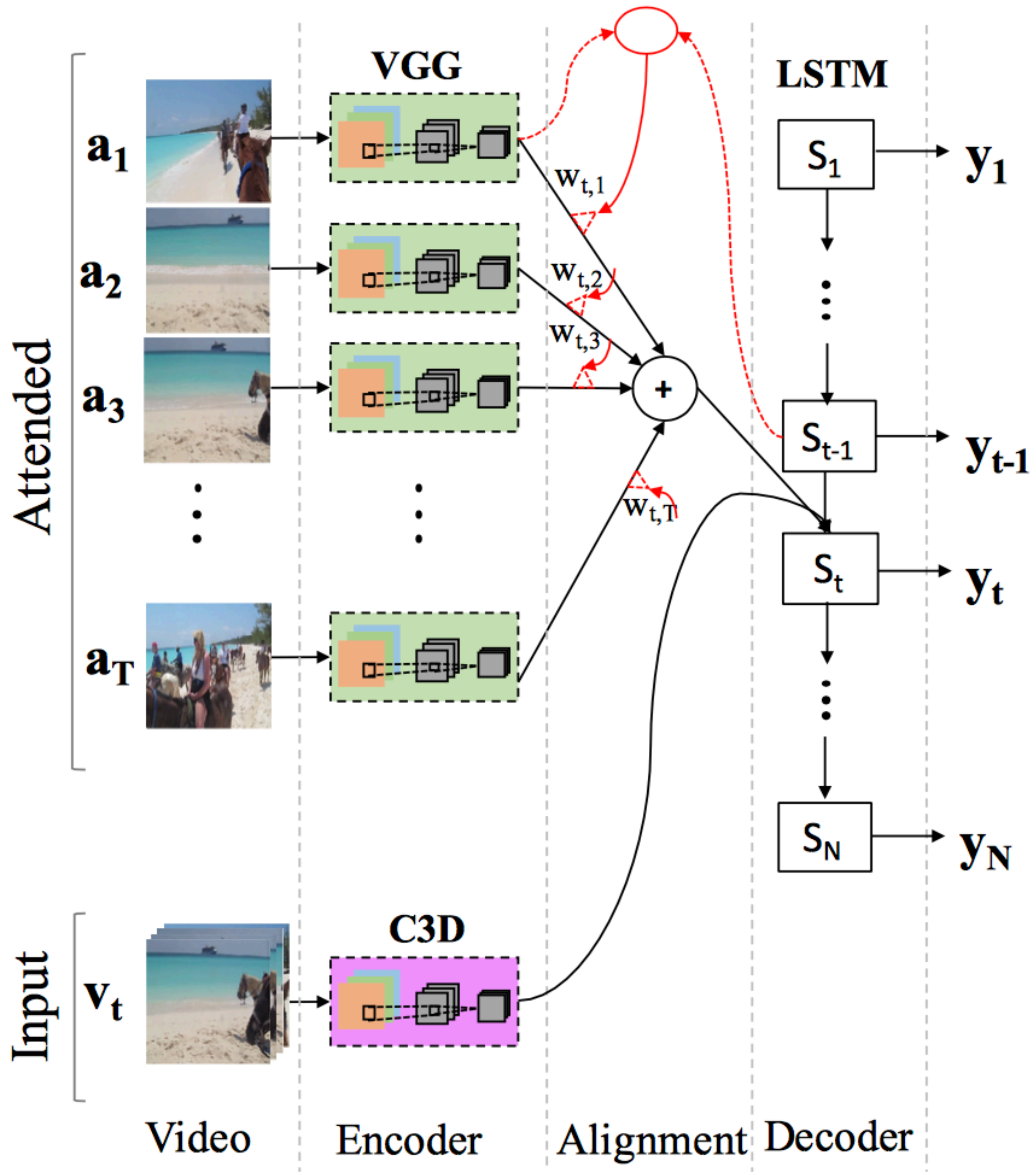
Take home: Early detection is only 1-3% worse than sewing the whole sequence

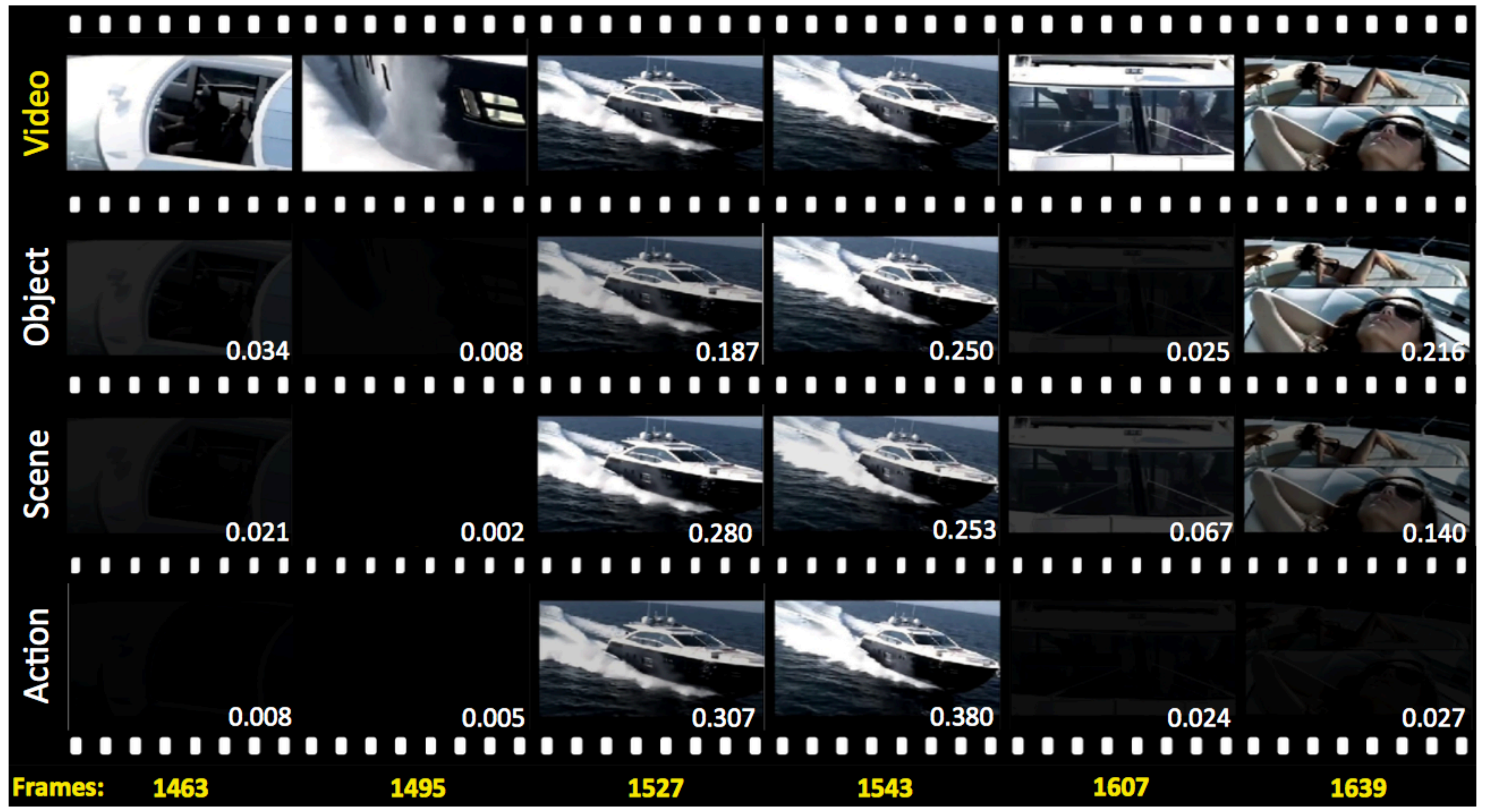
[Ma et al., 2014]

Applications: Activity Detection

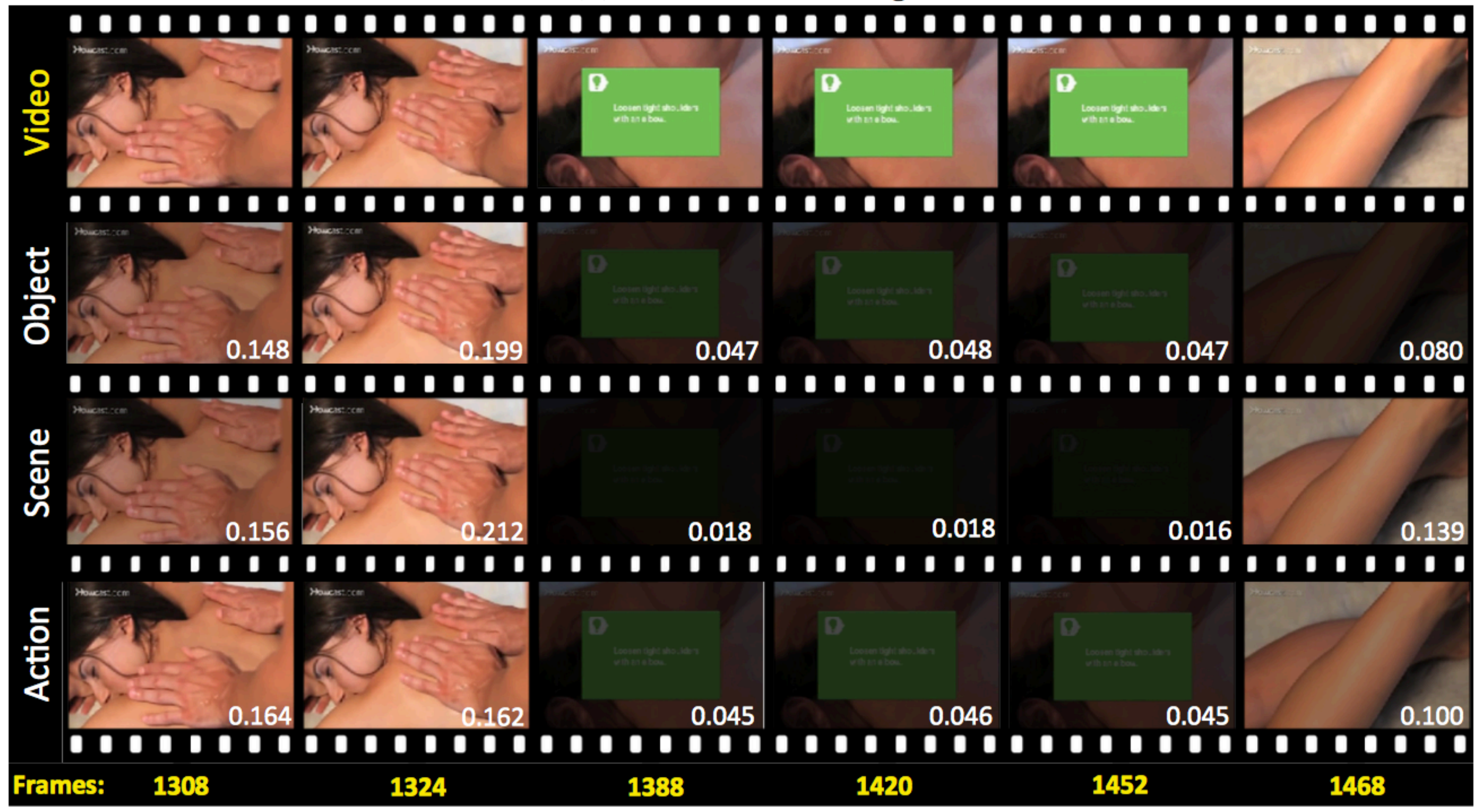
[Ma et al., 2014]

Attention Models for Action Highlighting





(a) Action: *Yatching*



(b) Action: *Having Massage*

[Torabi & Sigal, 2017]