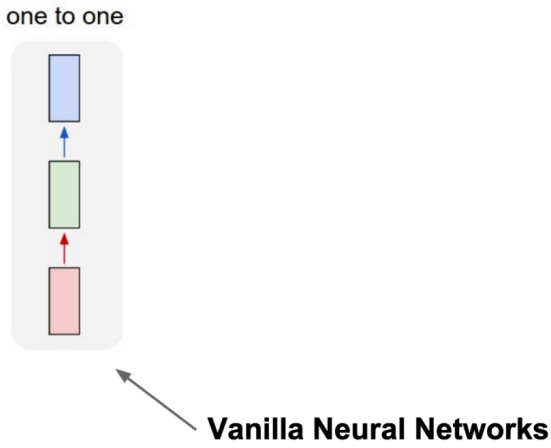
LT2318 H22 AICS: Image Description Generation

Nikolai Ilinykh
Department of Philosophy, Linguistics and Theory of Science
Centre for Linguistic Theory and Studies in Probability (CLASP)
University of Gothenburg, Sweden
{name.surname}@gu.se

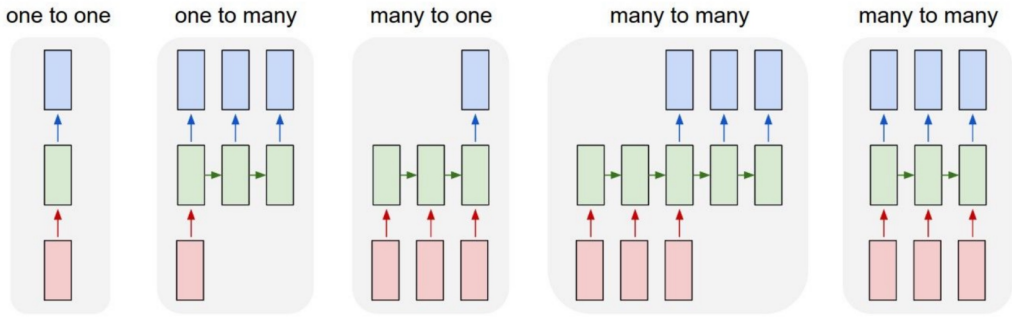
November 28, 2022

Recap: basics of neural networks[1]

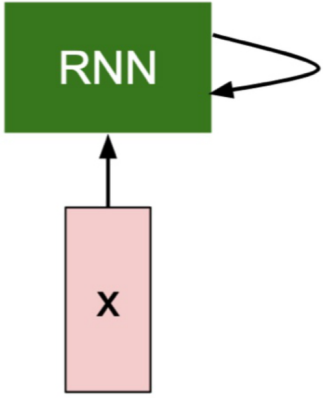


[1] http://karpathy.github.io/2015/05/21/rnn-effectiveness/

Image captioning is sequential



Recurrent networks are used for image captioning



What is inside RNN?

We can process a sequence of vectors **x** by applying a **recurrence formula** at every time step:

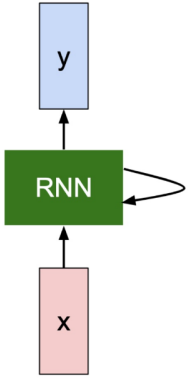
$$h_t = f_W(h_{t-1}, x_t)$$

new state

old state

input vector at some time step

some function with parameters W



RNN in words

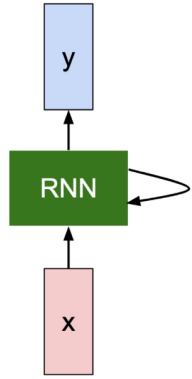
- Think "hidden state = **memory**": it encapsulates information from all previous states into one single representation; during training, the model learns this state.
- **Intuition**: in order to predict the next word, we need to take into account what has been stored in the "memory" of the model, e.g., in its hidden state.
- Why do we need hidden state and why don't we use the model's output? The output state is a concatenation of all hidden states up to the step t, but to predict the output for the particular current state we use hidden state.

Same parameter set at each step!

The concept of **generalisation** is important: in order to fit all data, we need to have the same set of parameters that are updated across the examples.

We can process a sequence of vectors **x** by applying a **recurrence formula** at every time step:

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

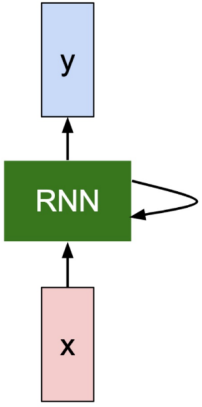


Notice: the same function and the same set of parameters are used at every time step.

Inner workings of RNNs

Think "hidden state = internal blackbox representations", W_{xh} is the input layer, W_{hy} is the output layer, the number of internal hidden layers differs from model to model.

The state consists of a single *"hidden"* vector **h**:



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

$$\downarrow$$

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

$$y_t = W_{hy}h_t$$

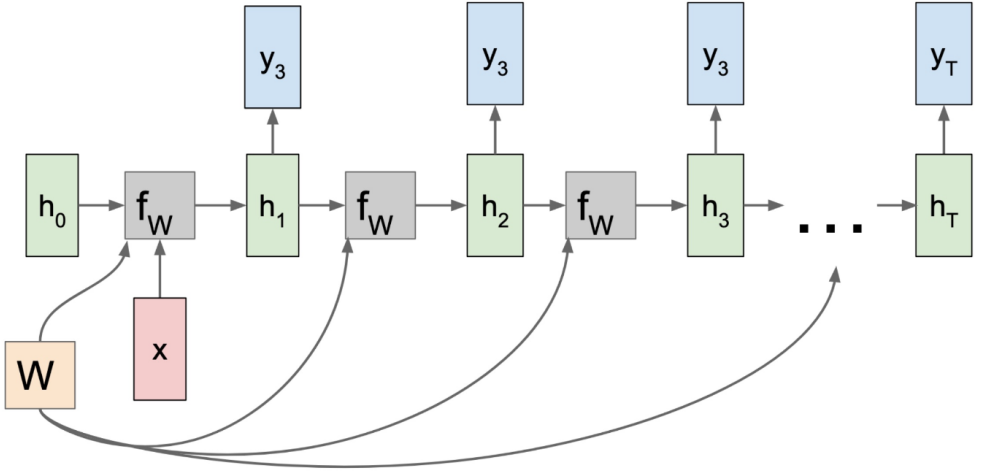
Tanh, ReLU, etc

Why non-linearity?[2]

[2] https://stackoverflow.com/questions/9782071/why-must-a-nonlinear-activation-function-be-used-in-a-backpropagation-neural-net

Generation is sequential

Important concept in any language generation is that every next word depends **ONLY** on previous words.



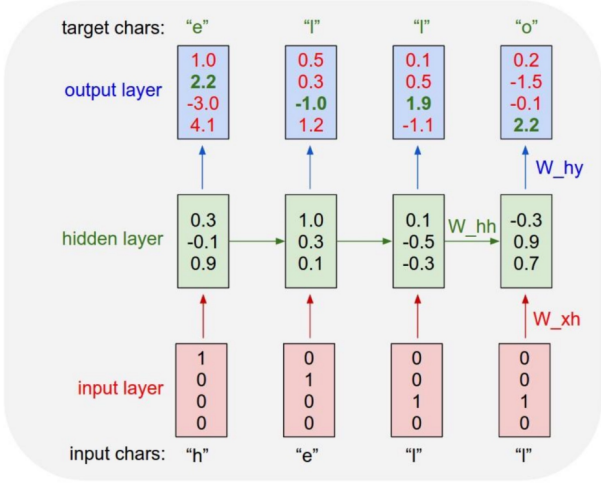
RNNs are trained with teacher-forcing

Ignore model's output, use the ground-truth at each step.

Example: Character-level Language Model

Vocabulary: [h,e,l,o]

Example training sequence: **"hello"**



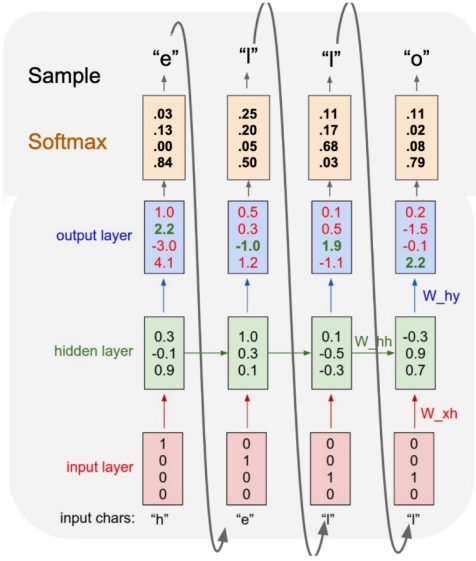
RNNs are tested WITHOUT teacher-forcing

Every next input is the previous output.

Example:
Character-level
Language Model
Sampling

Vocabulary:
[h,e,l,o]

At test-time sample characters one at a time, feed back to model



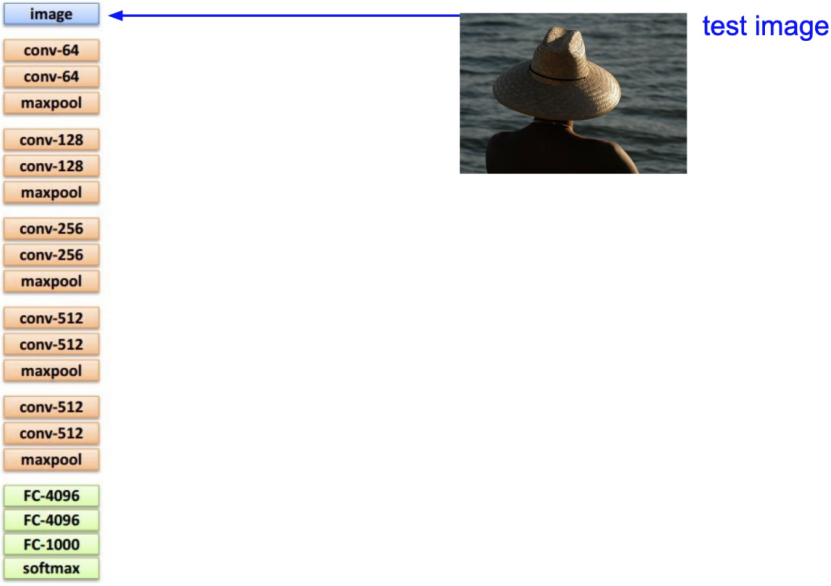
Generating image descriptions

So, how do we generate image captions? (Vinyals et al., 2015; Karpathy and Fei-Fei, 2015)

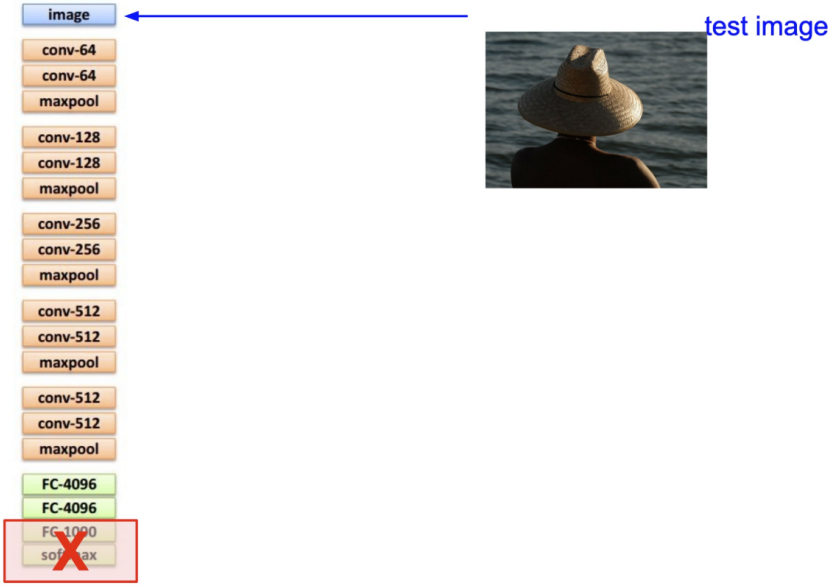
Take input image

This image is CC0 public domain

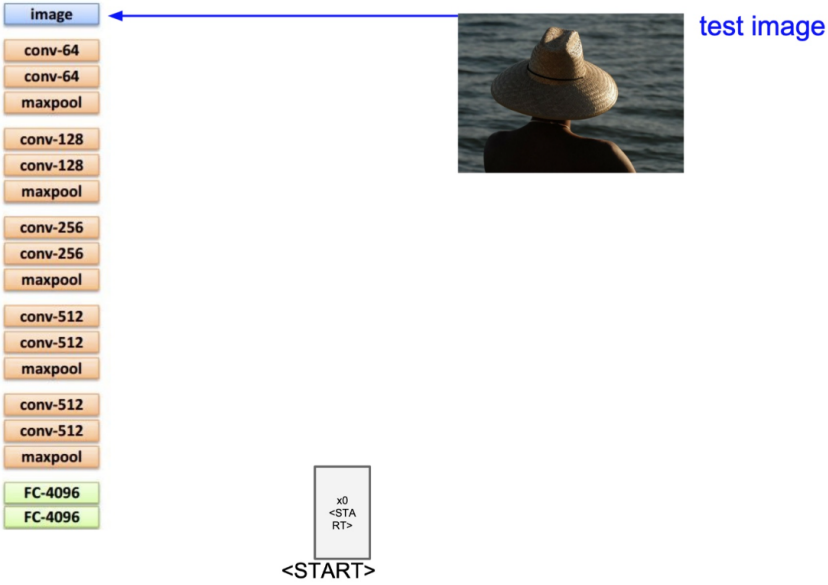
Pass it to a pre-trained CNN



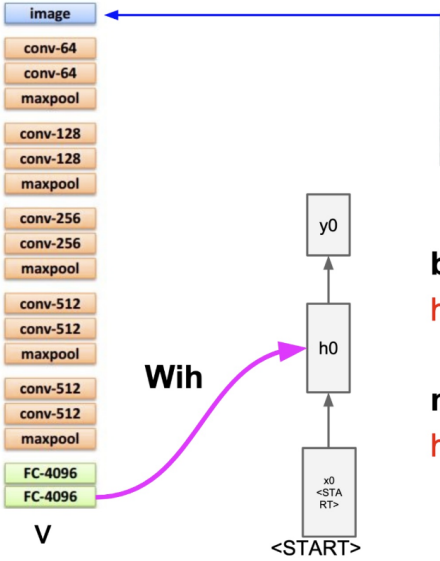
Extract visual features



Initiate generation



Map visual features and produce output



test image

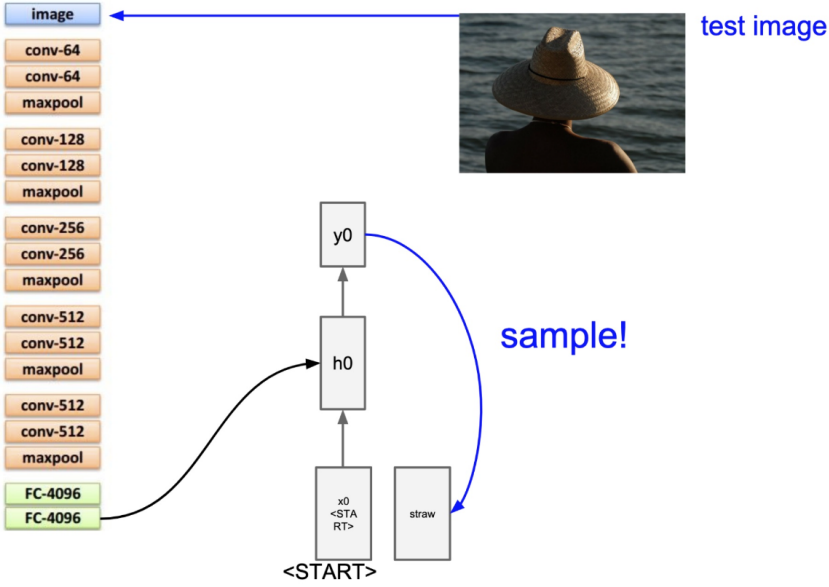
before:

$h = \tanh(W_{xh} * x + W_{hh} * h)$

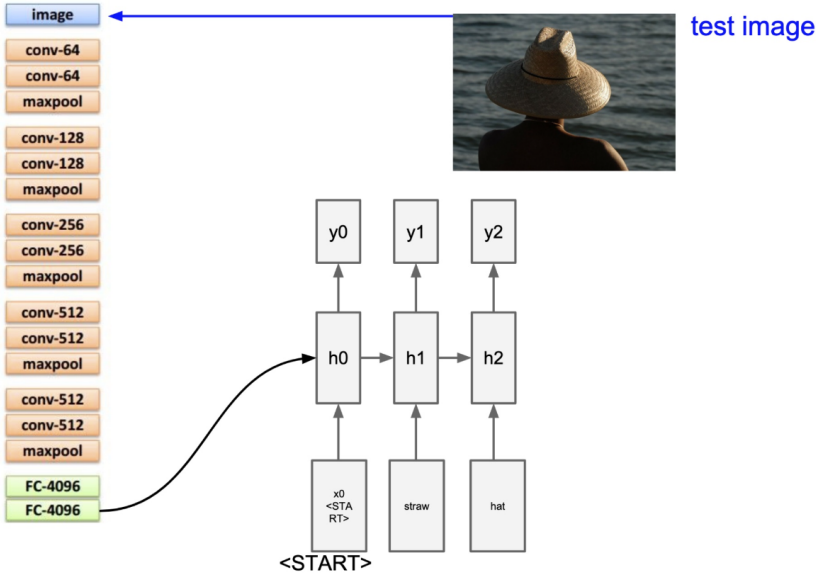
now:

$h = \tanh(W_{xh} * x + W_{hh} * h + W_{ih} * v)$

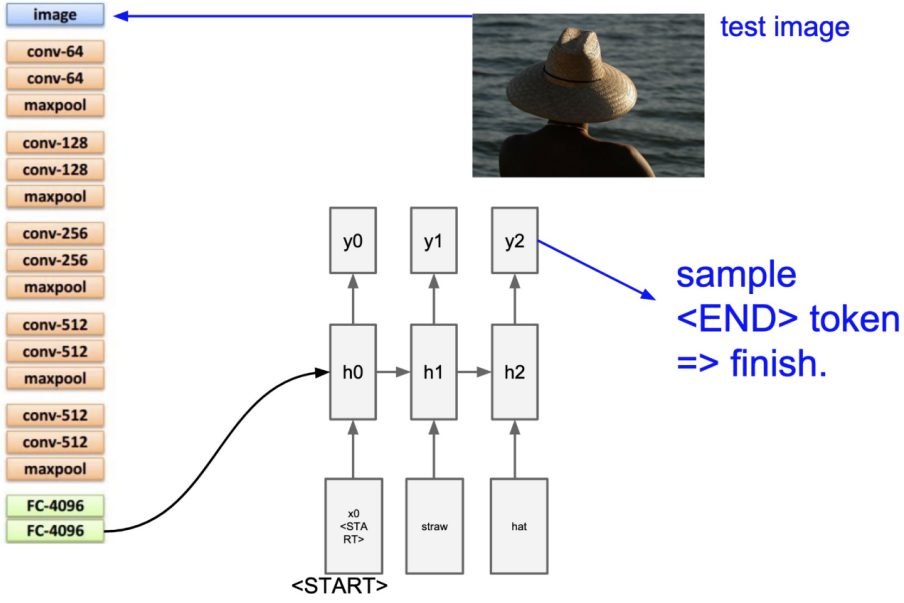
Get next textual input



Generate sequence of outputs



Complete generation



Good captions

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

Bad captions

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

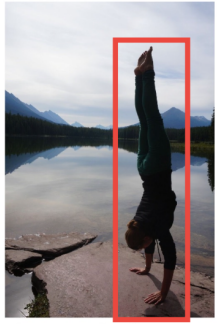
A man in a baseball uniform throwing a ball

Why captions can be bad?

Because descriptions are *situated* in the image, the describer, the receiver, the environment: **the situation**.

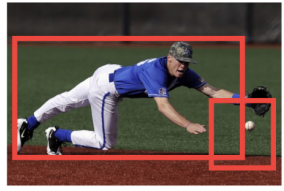
A woman is holding a cat in her hand

A person holding a computer mouse on a desk



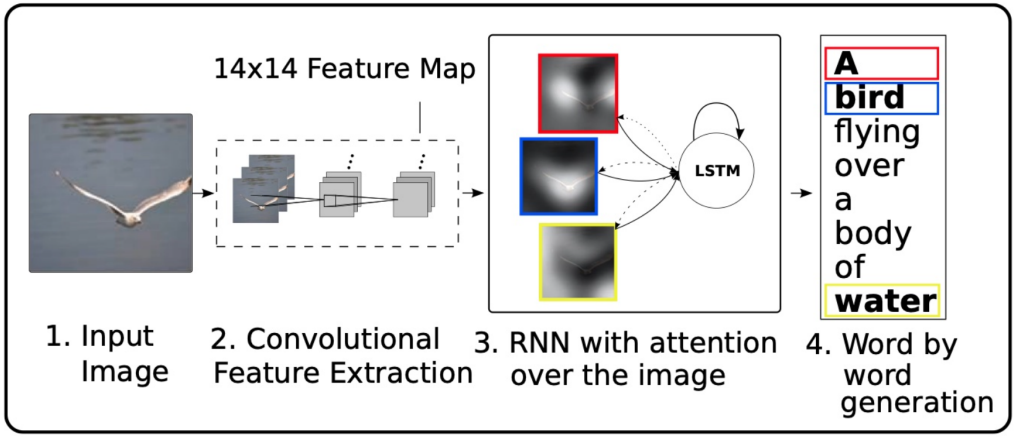
A woman standing on a beach holding a surfboard

A bird is perched on a tree branch

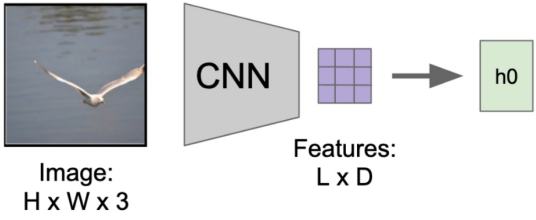


A man in a baseball uniform throwing a ball

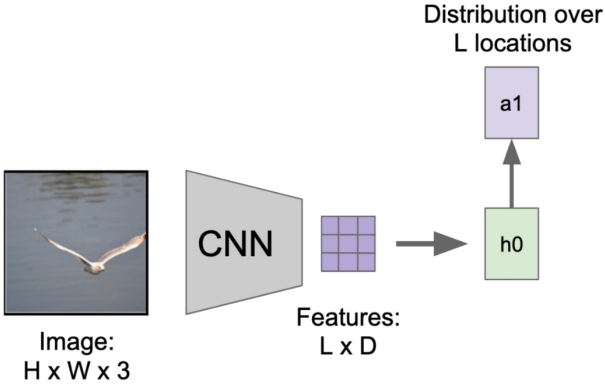
Attention: focus on specific visual features (Xu et al., 2015)



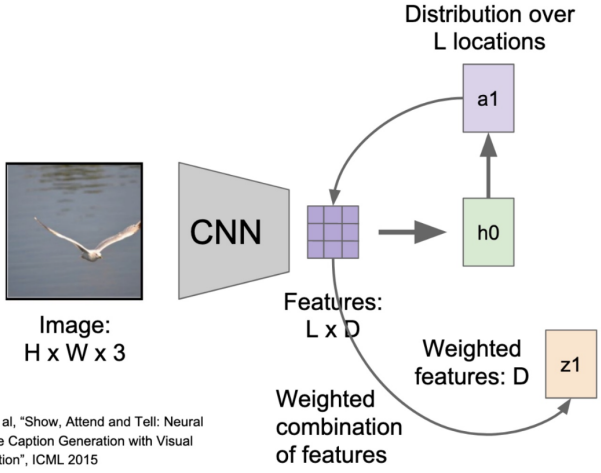
Use visual features to initialise the language model



Learn importance (probability scores) of visual features



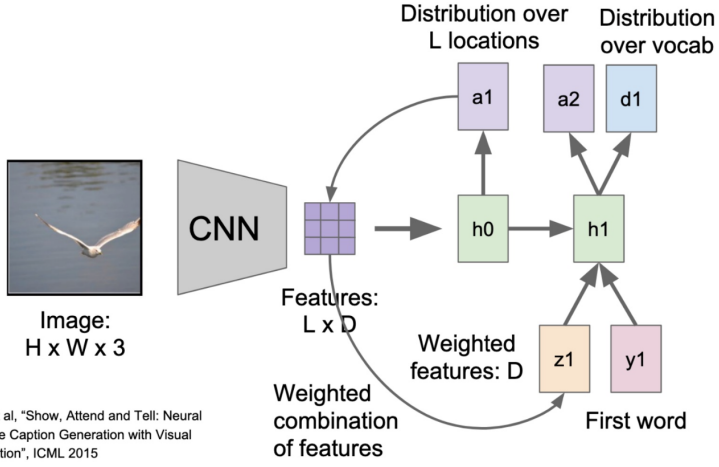
Weight visual features



Xu et al, "Show, Attend and Tell: Neural Image Caption Generation with Visual Attention", ICML 2015

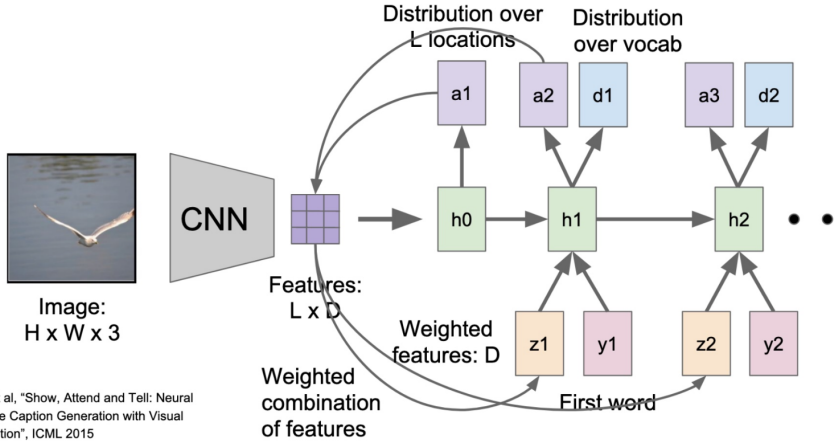
$$z = \sum_{i=1}^{L} p_i v_i$$

Repeat attention for every word that is generated

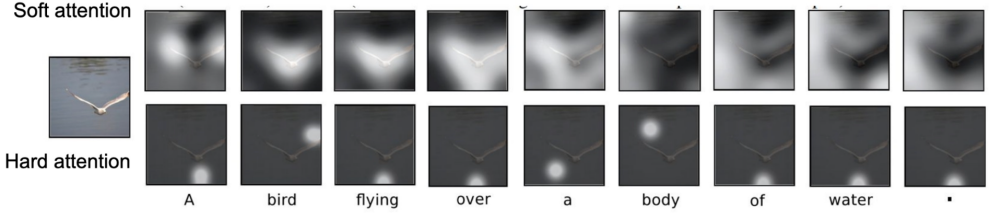


:u et al, "Show, Attend and Tell: Neural
nage Caption Generation with Visual
ttention", ICML 2015

Continue until the last word is generated

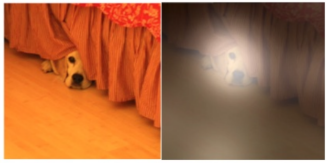


The model looks at specific image parts more now



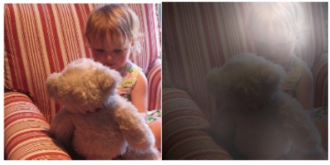
Attention works!

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.

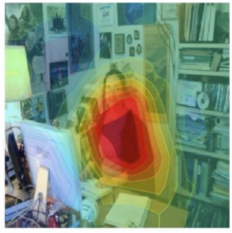
A giraffe standing in a forest with trees in the background.

Well, not always...

Wrong

Baseline:
*A **man** sitting at a desk with a laptop computer.*

Right for the Right Reasons

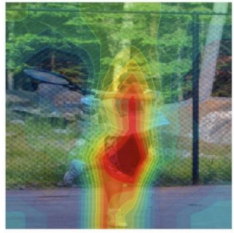


Our Model:
*A **woman** sitting in front of a laptop computer.*

Right for the Wrong Reasons

Baseline:
*A **man** holding a tennis racquet on a tennis court.*

Right for the Right Reasons

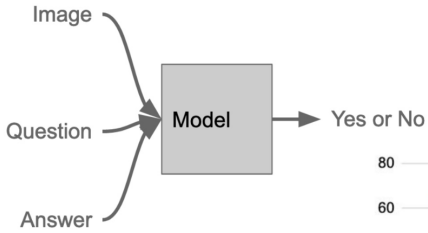


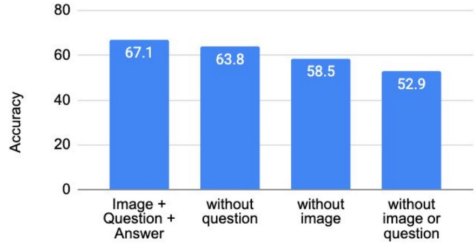
Our Model:
*A **man** holding a tennis racquet on a tennis court.*

Biases and problems: language is stronger than vision (Ilinykh et al., 2022)

What is the dog playing with?

Frisbee





References I

Nikolai Ilinykh, Yasmeen Emampoor, and Simon Dobnik. 2022. Look and answer the question: On the role of vision in embodied question answering. In *Proceedings of the 15th International Conference on Natural Language Generation*, pages 236–245, Waterville, Maine, USA and virtual meeting. Association for Computational Linguistics.

Andrej Karpathy and Li Fei-Fei. 2015. Deep visual-semantic alignments for generating image descriptions.

Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. 2015. Show and tell: A neural image caption generator.

Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. 2015. Show, attend and tell: Neural image caption generation with visual attention. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 2048–2057, Lille, France. PMLR.