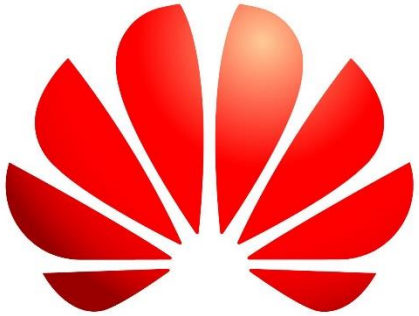


HUAWEI

Обзор методов решения задачи image captioning и поиска изображений по текстовому запросу

21.11.2019, UNN, Nizhny Novgorod

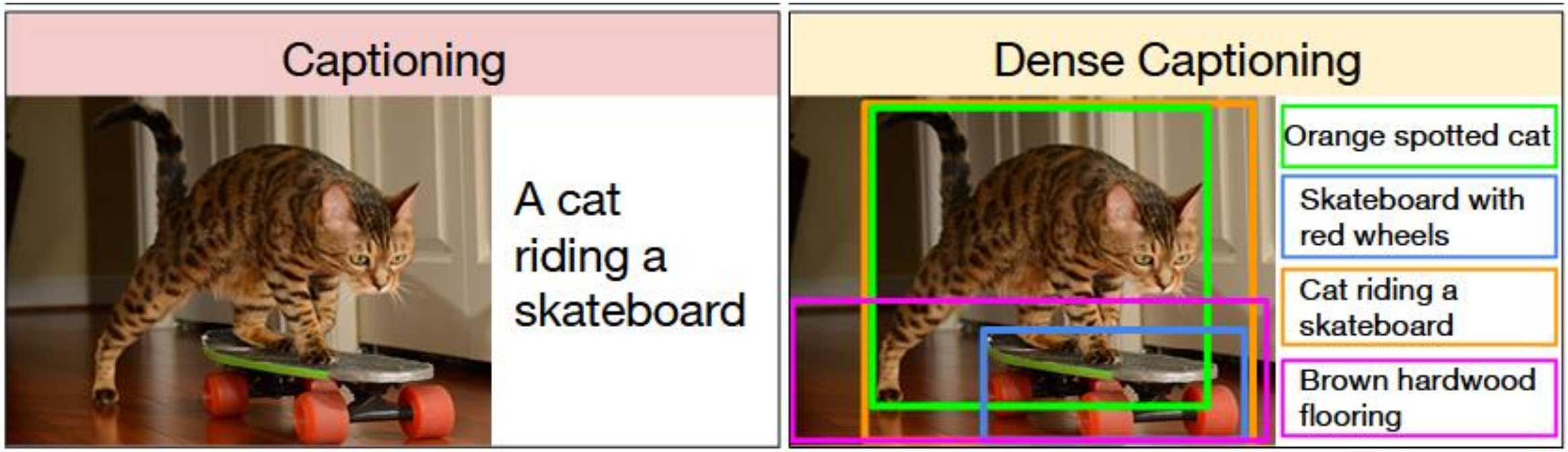
Huawei Research Center, Nizhny Novgorod, Russia

Sergey Kosolapov, kosolapov.sergey@huawei.com

Agenda

1. Image Captioning
    - Datasets
    - Metrics
    - Algorithms
    - Examples
2. Image-Text matching
    - Approaches
    - Metrics
    - Demo

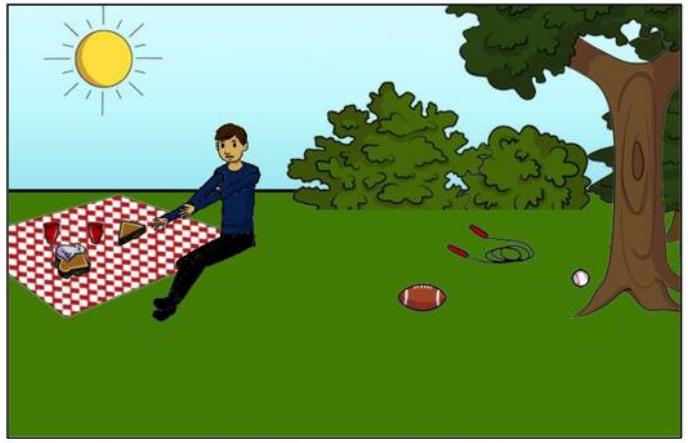
Image Captioning, Dense Captioning



Visual Question Answering

What color are her eyes?
What is the mustache made of?

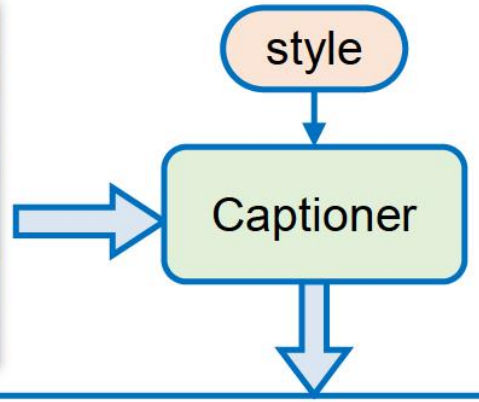
How many slices of pizza are there?
Is this a vegetarian pizza?



Is this person expecting company?
What is just under the tree?

Does it appear to be rainy?
Does this person have 20/20 vision?

Stylized Captioning



Factual:
A brown dog drinks from a body of water.
Humorous:
A dog putting his legs into a pond, but scared of the water.
Romantic:
A brown dog steps into murky water,
careful to swim back to his master.
Positive:
A cuddly dog is drinking from a body of tranquil water.
Negative:
A black ugly dog drinks from a body of dirty water.

Datasets

- MS COCO
- Flickr30K (Flickr8K)
- Google's Conceptual Captions
- Visual Genome (108k images, dense captioning)
- FlickrStyle10k
- Instagram (1.1m images, hashtag prediction and post-generation tasks)

Metrics

1. BLEU@N (Bilingual evaluation understudy)

- Compares n-grams and their frequencies
- https://www.aclweb.org/anthology/P02-1040

2. CIDEr (Consensus-based Image Description Evaluation)

- Proposed for image description evaluation. Cosine similarity of stemmed n-grams weighted by TF-IDF.
- https://arxiv.org/pdf/1411.5726.pdf

3. ROUGE-L (Recall-Oriented Understudy for Gisting Evaluation)

- Proposed for the task of text summarization. Calculates F score on longest matching sequence of words
- https://www.aclweb.org/anthology/W04-1013

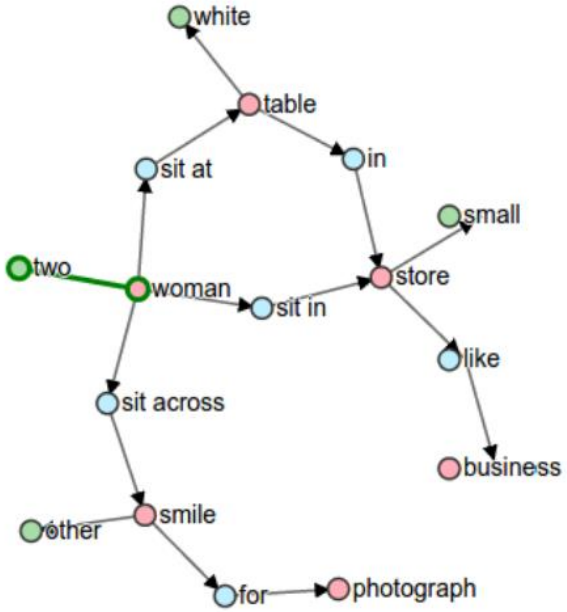
Metrics

4. METEOR

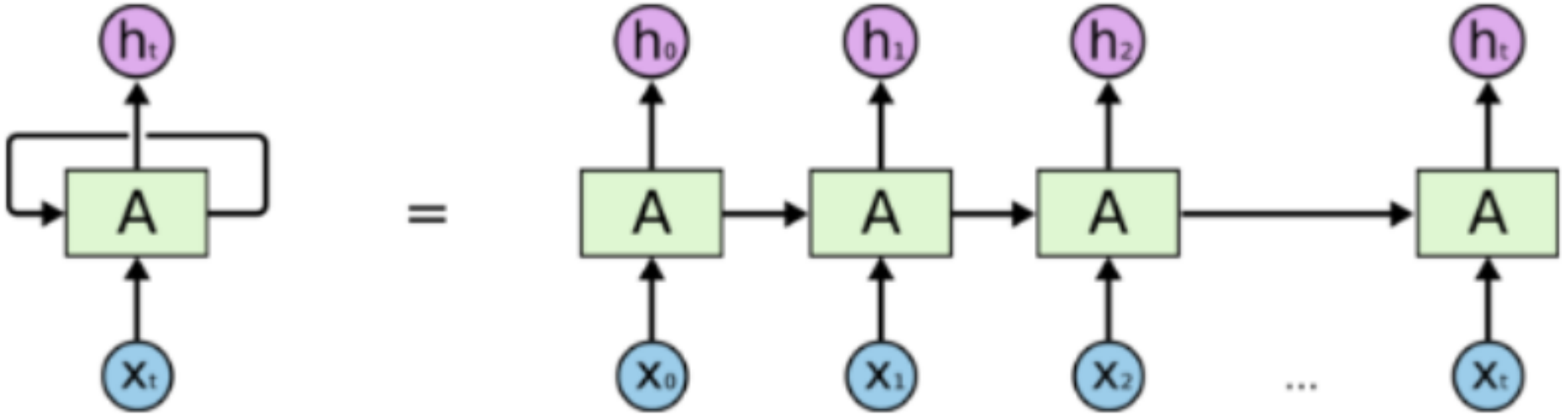
- Alignments are based on exact, stem, synonym, and paraphrase matches between words and phrases
- http://www.cs.cmu.edu/~alavie/METEOR/

5. SPICE (Semantic Propositional Image Caption Evaluation)

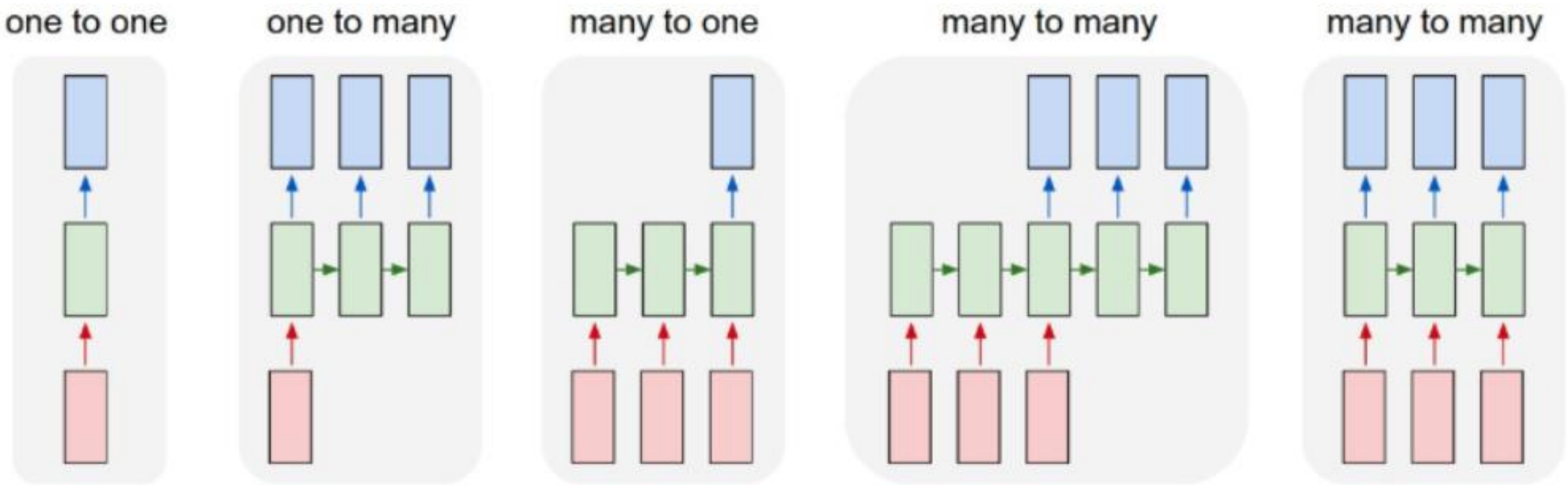
- scene graphs — encoding the objects (red), attributes (green), and relations (blue) present.
- https://arxiv.org/pdf/1607.08822.pdf



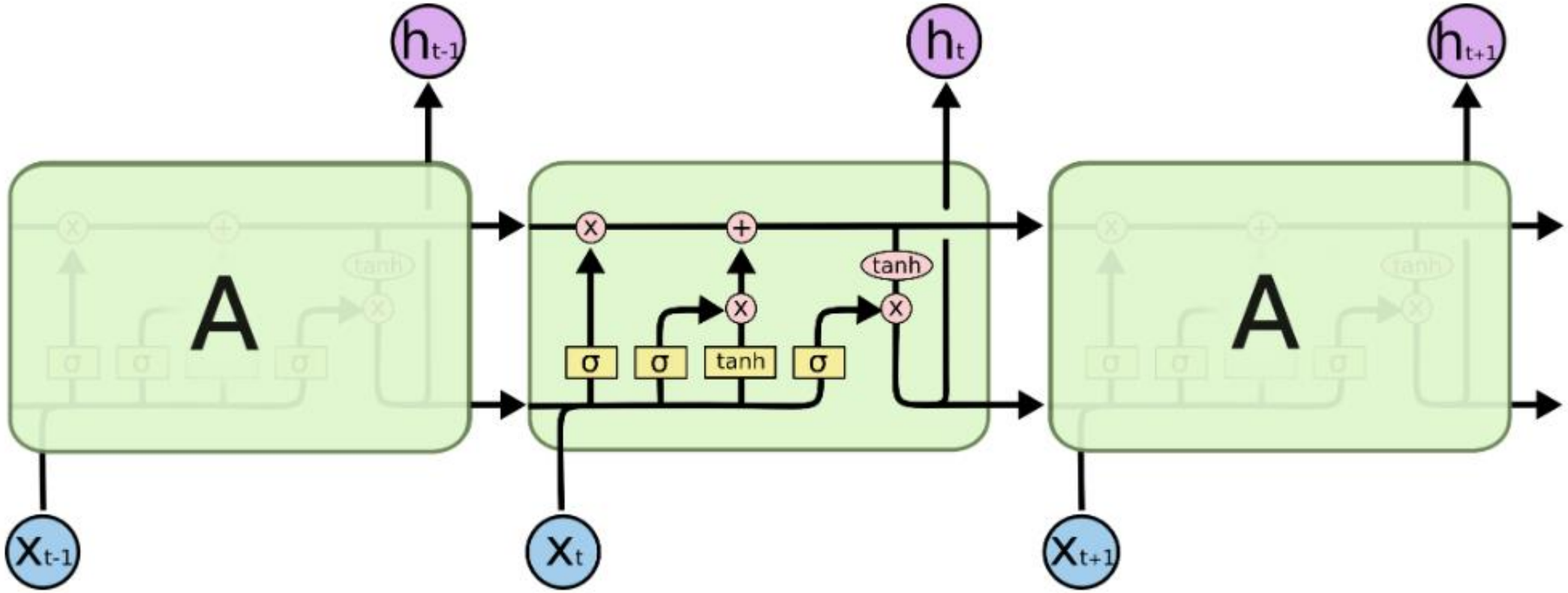
Recurrent neural network



Recurrent neural network

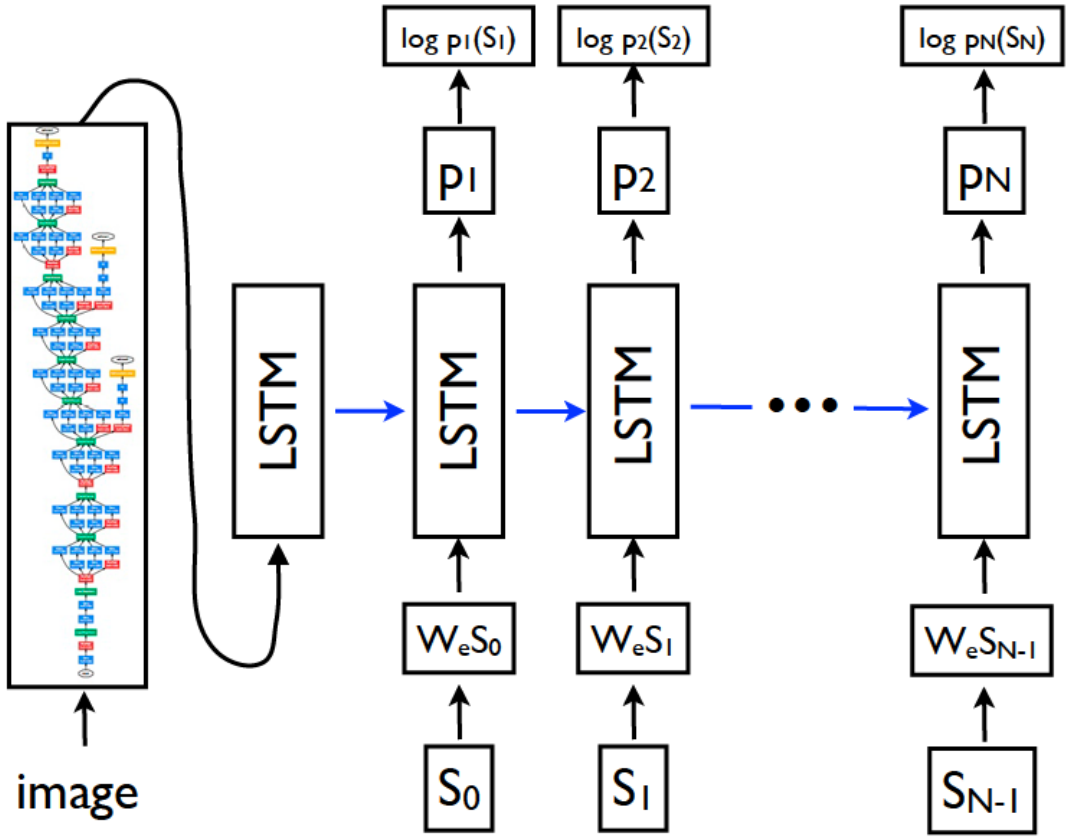


LSTM



https://colah.github.io/posts/2015-08-Understanding-LSTMs/

Show and Tell: A Neural Image Caption Generator



- BB – GoogLeNet
- Last FC layer features are extracted
- CNN parameters fine-tuning
- Won 1st place MSCOCO contest in 2015

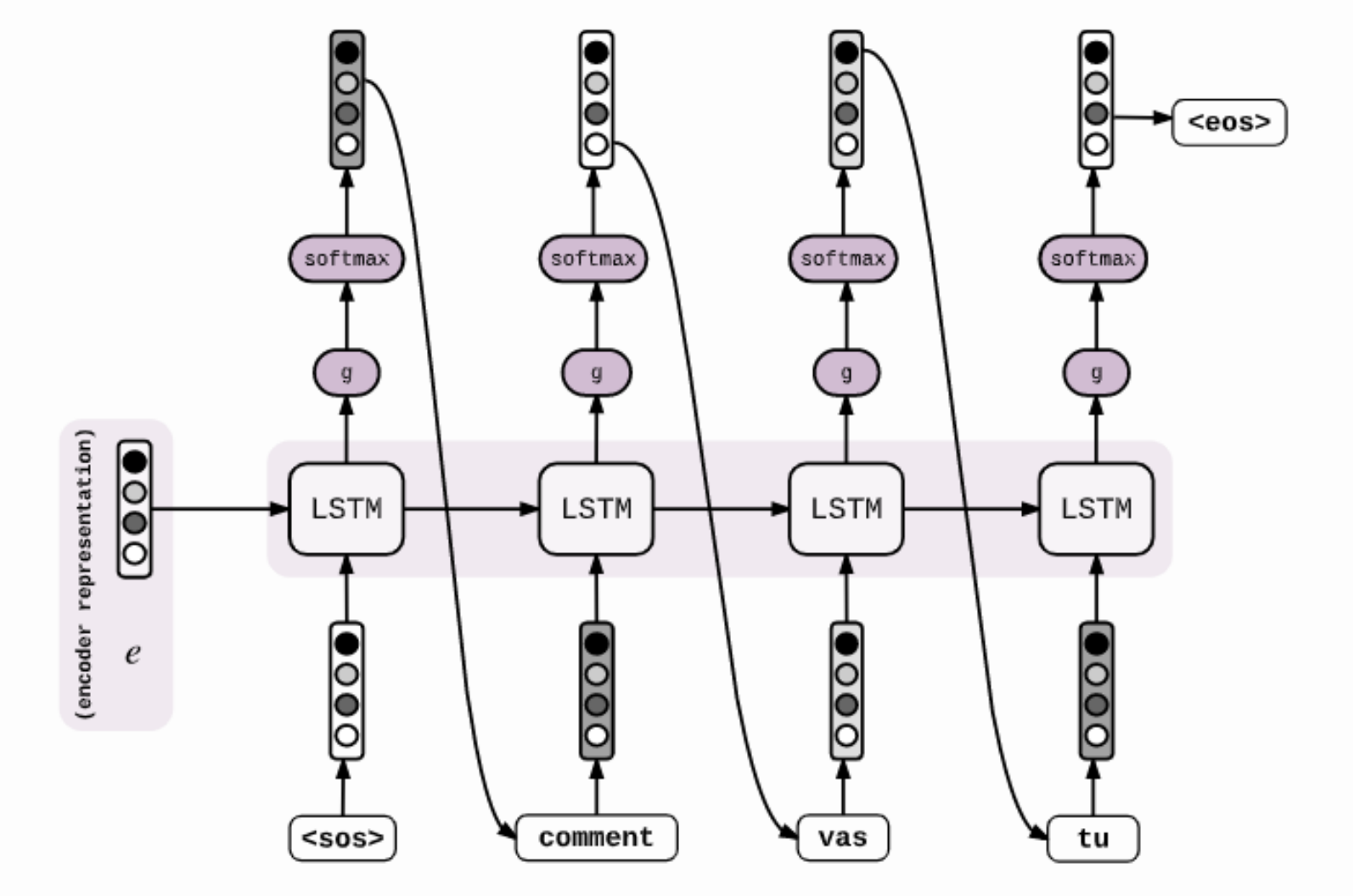
https://arxiv.org/abs/1411.4555

Show and tell. Examples

A person riding a motorcycle on a dirt road.

Two dogs play in the grass.

Show and tell. Vanila decoder



Show and tell. Beam search

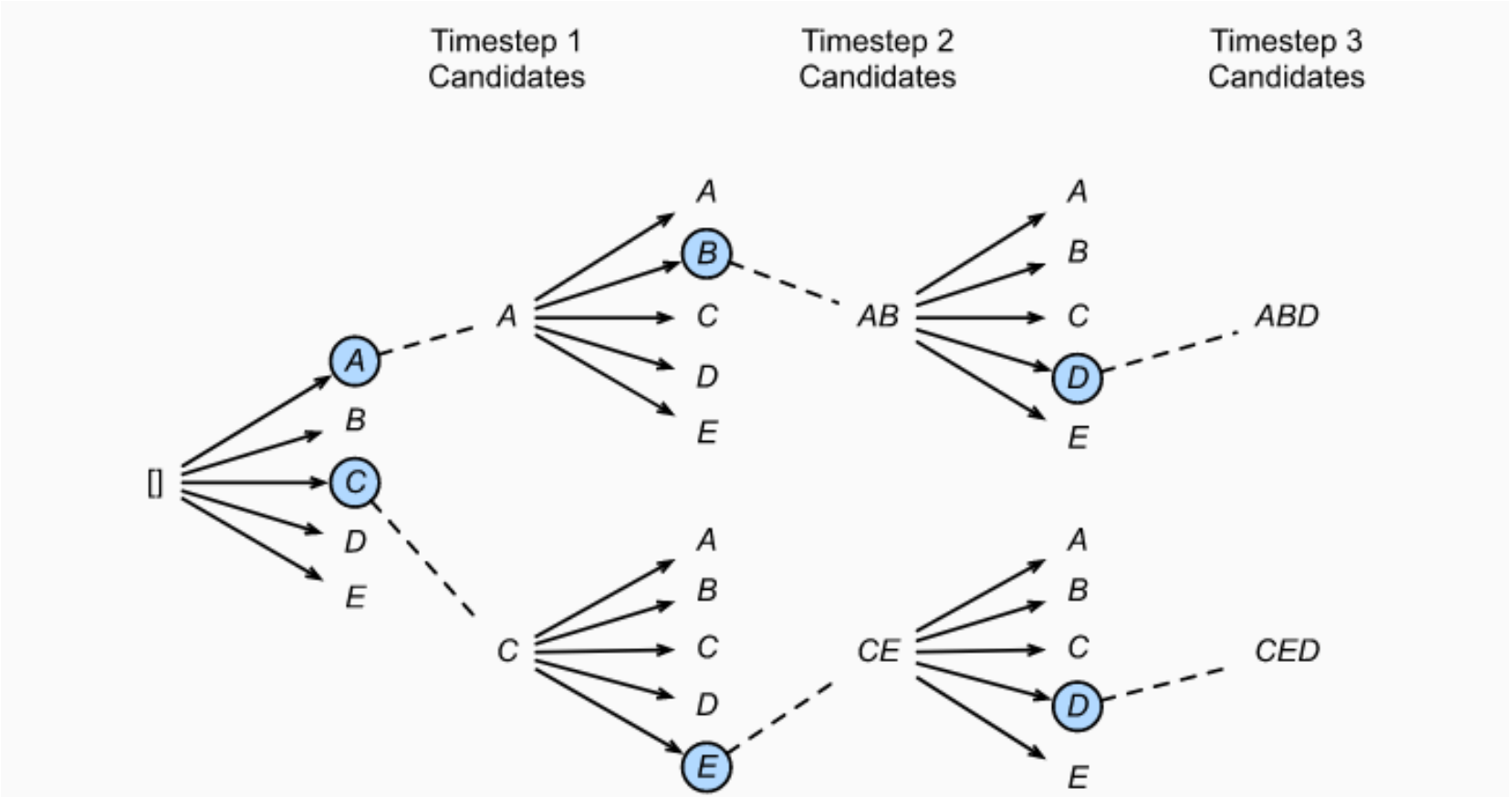
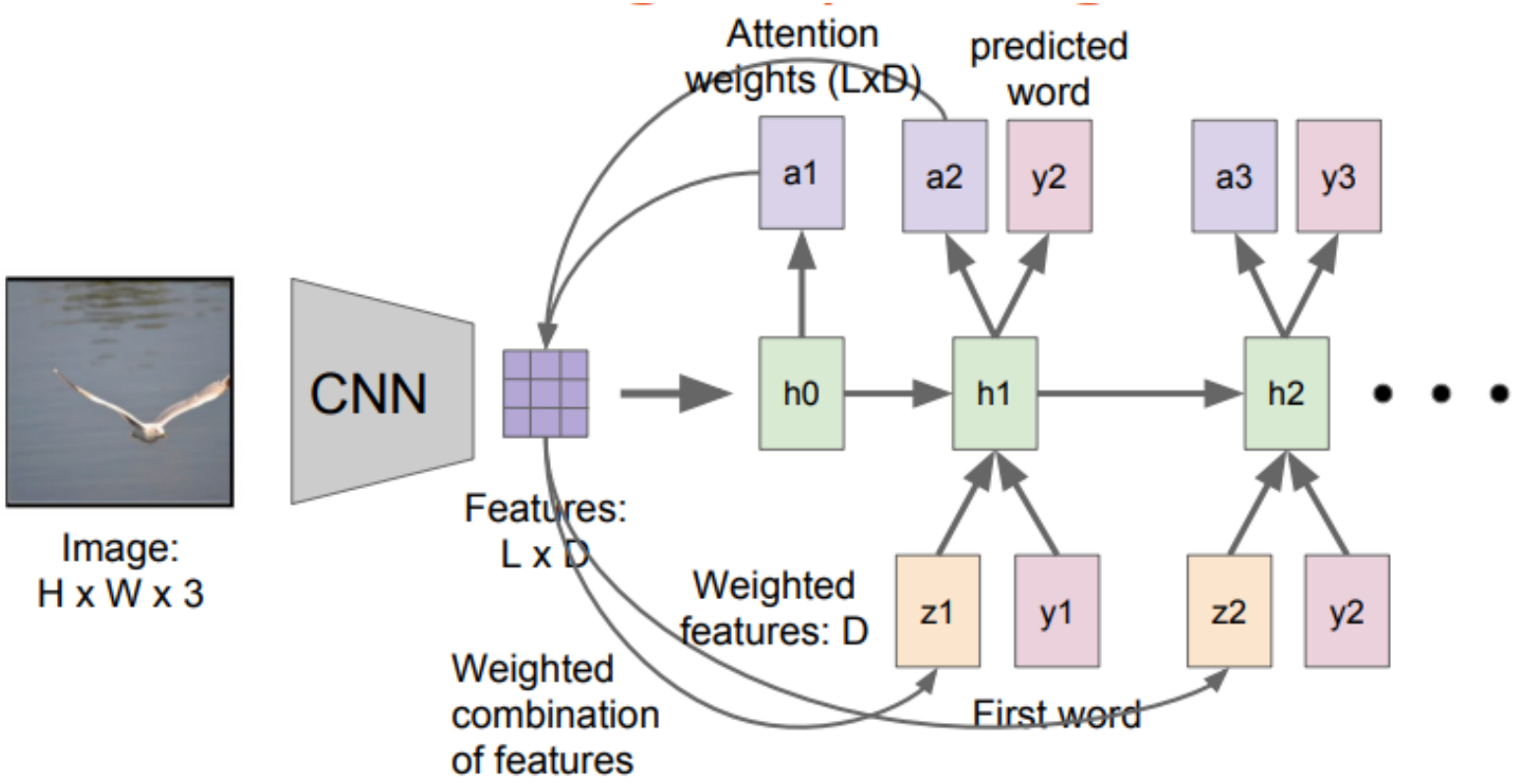


Fig. 7.15.3 The beam search process. The beam size is 2 and the maximum length of the output sequence is 3. The candidate output sequences are A, C, AB, CE, ABD, and CED.

Show and tell. Beam search Example

- **Normal Max search:** A dog is jumping in the air to catch something .
- **Beam Search, k=3:** A brown dog is jumping in the air .
- **Beam Search, k=5:** A dog is jumping in the air to catch something .
- **Beam Search, k=7:** A dog in a harness holds a stick in his mouth while standing in the grass .

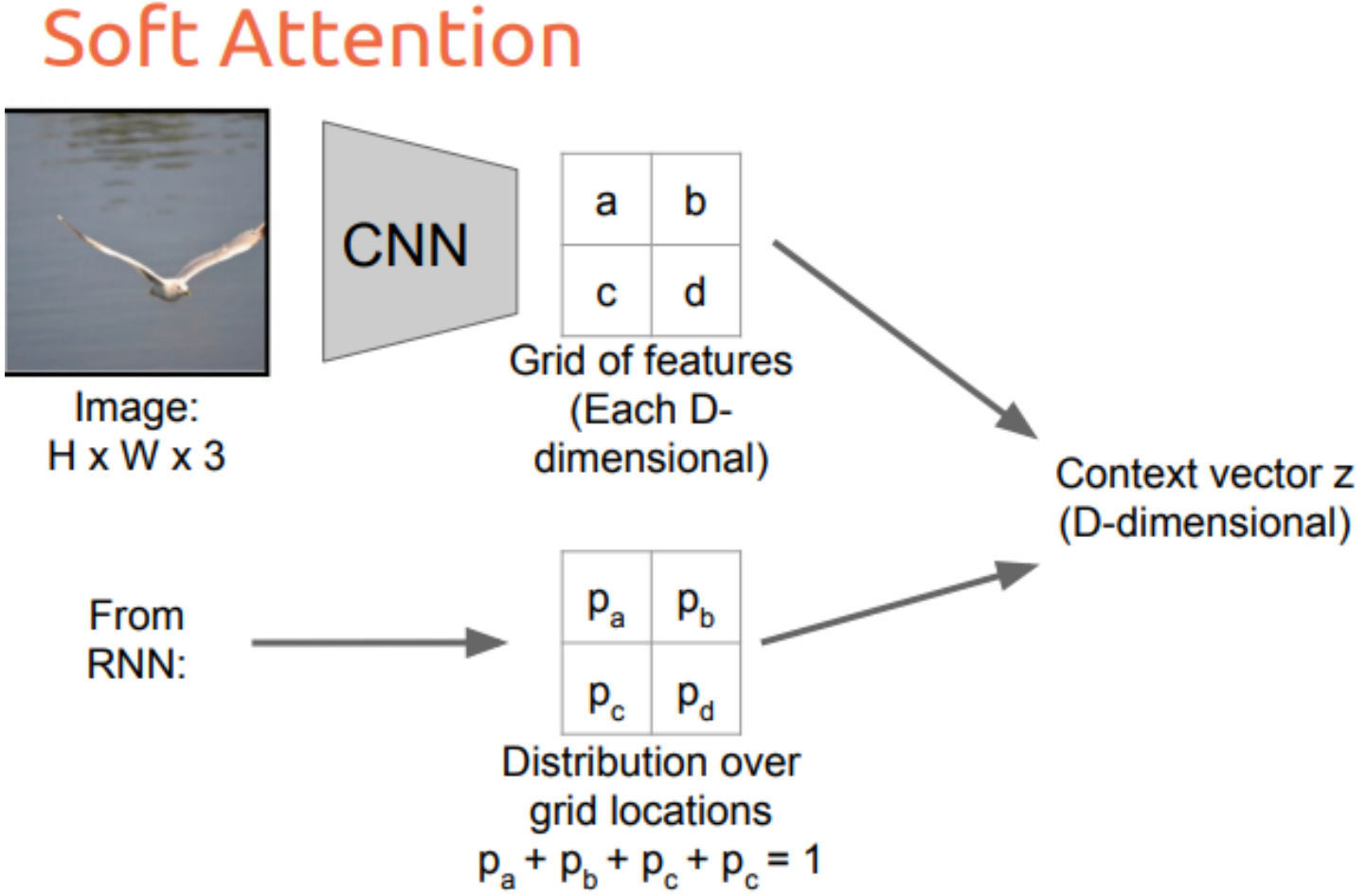
Attention



- BB – GoogLeNet /VGG
- 14th Convolutional layer is extracted
- Soft and Hard attention mechanism

https://arxiv.org/abs/1502.03044

Soft attention



Soft attention:
Summarize ALL locations
$z = p_a a + p_b b + p_c c + p_d d$

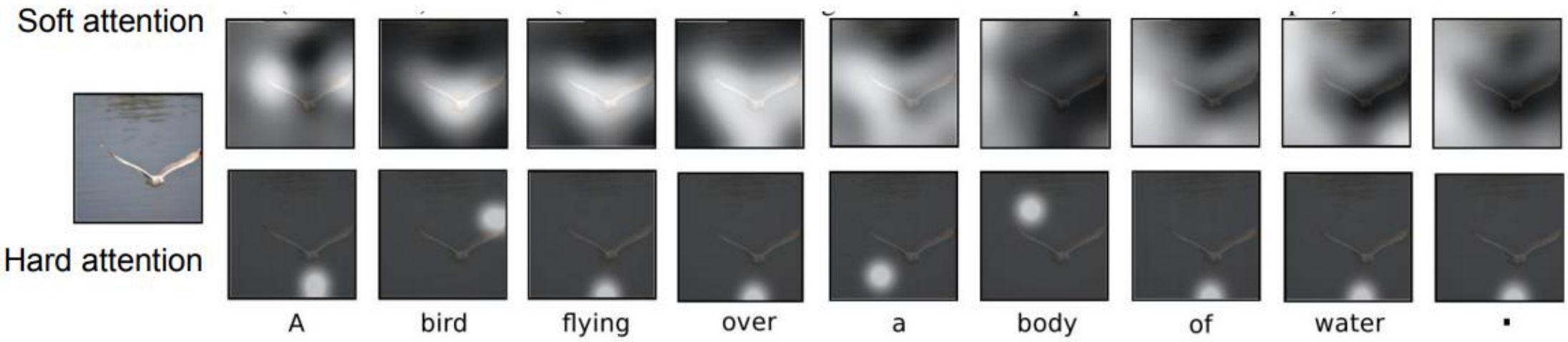
Derivative dz/dp is nice!
Train with gradient descent

Soft attention. Examples

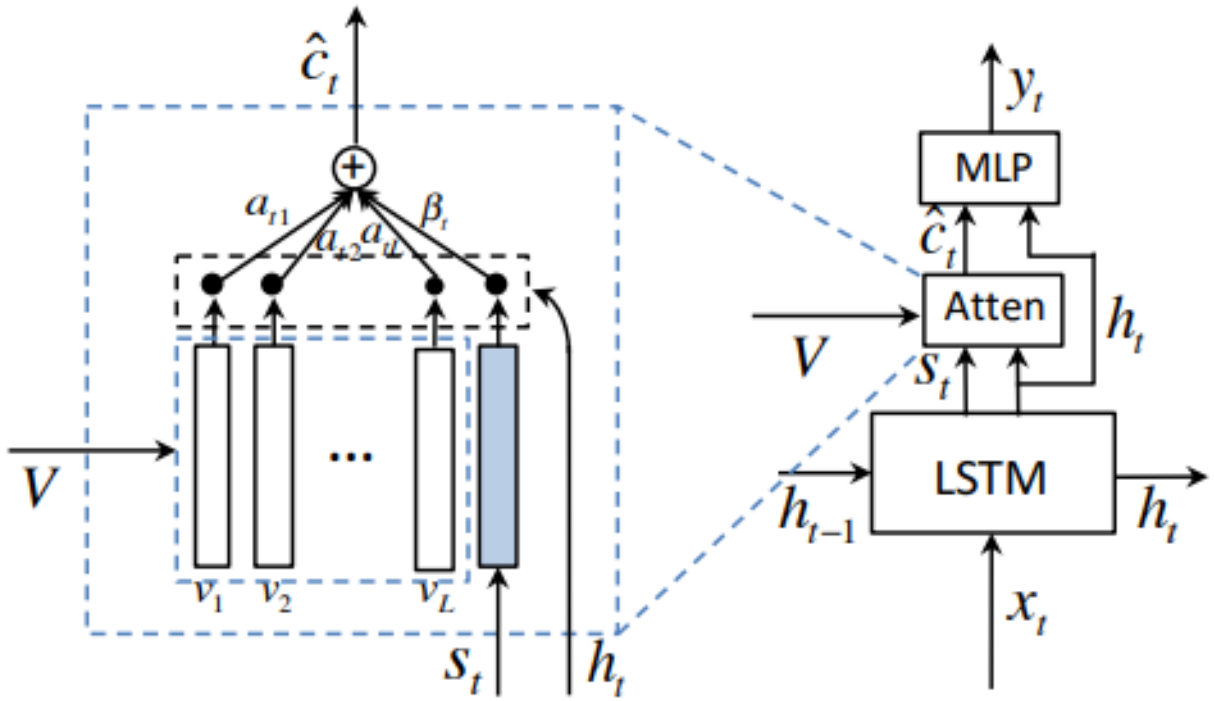
A woman is throwing a frisbee in a park.

A dog is standing on a hardwood floor.

Soft vs Hard attention



Adaptive attention



$$\boldsymbol{g}_t = \sigma\left(\boldsymbol{W}_x \boldsymbol{x}_t + \boldsymbol{W}_h \boldsymbol{h}_{t-1}\right)$$

$$\boldsymbol{s}_t = \boldsymbol{g}_t \odot \tanh\left(\boldsymbol{m}_t\right)$$

$$\hat{\boldsymbol{c}}_t = \beta_t \boldsymbol{s}_t + (1 - \beta_t)\boldsymbol{c}_t$$

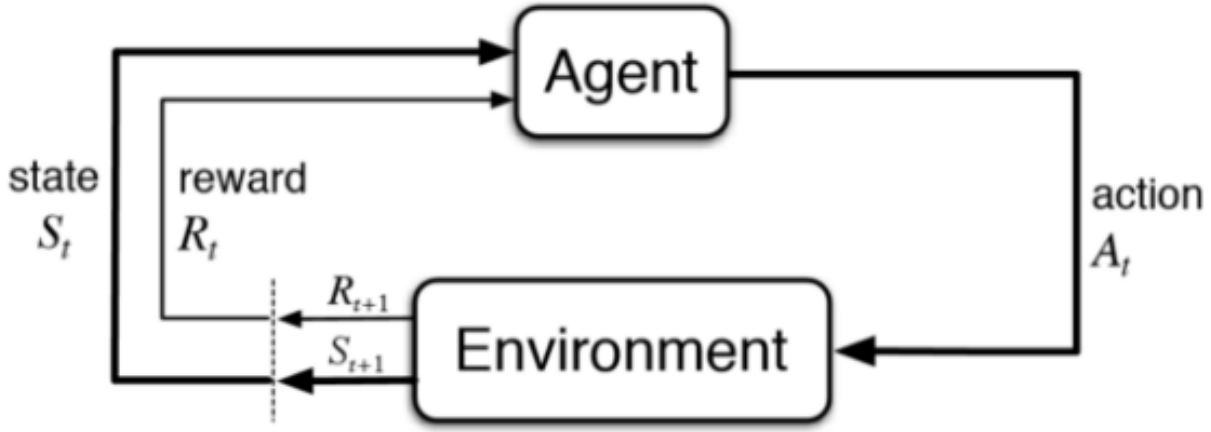
https://arxiv.org/abs/1612.01887

Adaptive attention. Examples

a man riding a bike down a road next to a body of water.

an elephant standing next to rock wall.

Reinforcement learning



Agent (e.g. RNN, LSTM or GRU)
Environment (image features, hidden states, and previous words)
Action (the prediction of the next word)
After generating a complete sentence, the agent will observe a sentence-level reward and update its internal state

Reinforcement learning

Policy Gradients

$p(x|s,\theta)$ - policy function ; f(x) - reward

$$\begin{aligned}
\nabla_\theta E_x[f(x)] &= \nabla_\theta \sum_x p(x) f(x) && \text{definition of expectation} \\
&= \sum_x \nabla_\theta p(x) f(x) && \text{swap sum and gradient} \\
&= \sum_x p(x) \frac{\nabla_\theta p(x)}{p(x)} f(x) && \text{both multiply and divide by } p(x) \\
&= \sum_x p(x) \nabla_\theta \log p(x) f(x) && \text{use the fact that } \nabla_\theta \log(z) = \frac{1}{z} \nabla_\theta z \\
&= E_x[f(x) \nabla_\theta \log p(x)] && \text{definition of expectation}
\end{aligned}$$

http://karpathy.github.io/2016/05/31/rl/

$$Loss = \sum_i f_i * \log(p(x_i))$$

Metrics comparison

Model	BLUE-1	BLUE-2	BLUE-3	BLUE-4	METEOR	ROUGE-L	CIDEr	SPICE
Show and tell: A Neural Image Caption Generator	0.713	0.542	0.407	0.309	0.254	0.530	0.943	
Soft/Hard attention	0.705	0.528	0.383	0.277	0.241	0.516	0.865	
Adaptive attention	0.748	0.584	0.444	0.336	0.264	0.550	1.042	0.197
RL	0.786	0.625	0.479	0.361	0.274	0.569	1.120	0.209
SOTA	0.819	0.666	0.521	0.401	0.293	0.594	1.290	

Our examples

Soft attention:
A group of people sitting on a bench

Adaptive Attention:
A group of people sitting at a table together

RL:
A group of people sitting at a table

Our examples

Soft Attention:
A living room with a couch and a couch

Adaptive Attention:
A living room with a couch and a table

RL:
A living room with a couch and table

Our examples

Soft Attention:
A man sitting on a bench in a park

Adaptive Attention:
A man in a red hat is on a blue and white surfboard

RL:
A man playing a frisbee in the water

Image-text matching

Task

Image retrieval by query

Types of queries:

- **General**
  Example: young boy in the forest
- Sightseeing
  Example: a group of people near the Kremlin
- Persons
  Example: My wife/friend in the bar

etc.

Dataset: COCO

Approaches

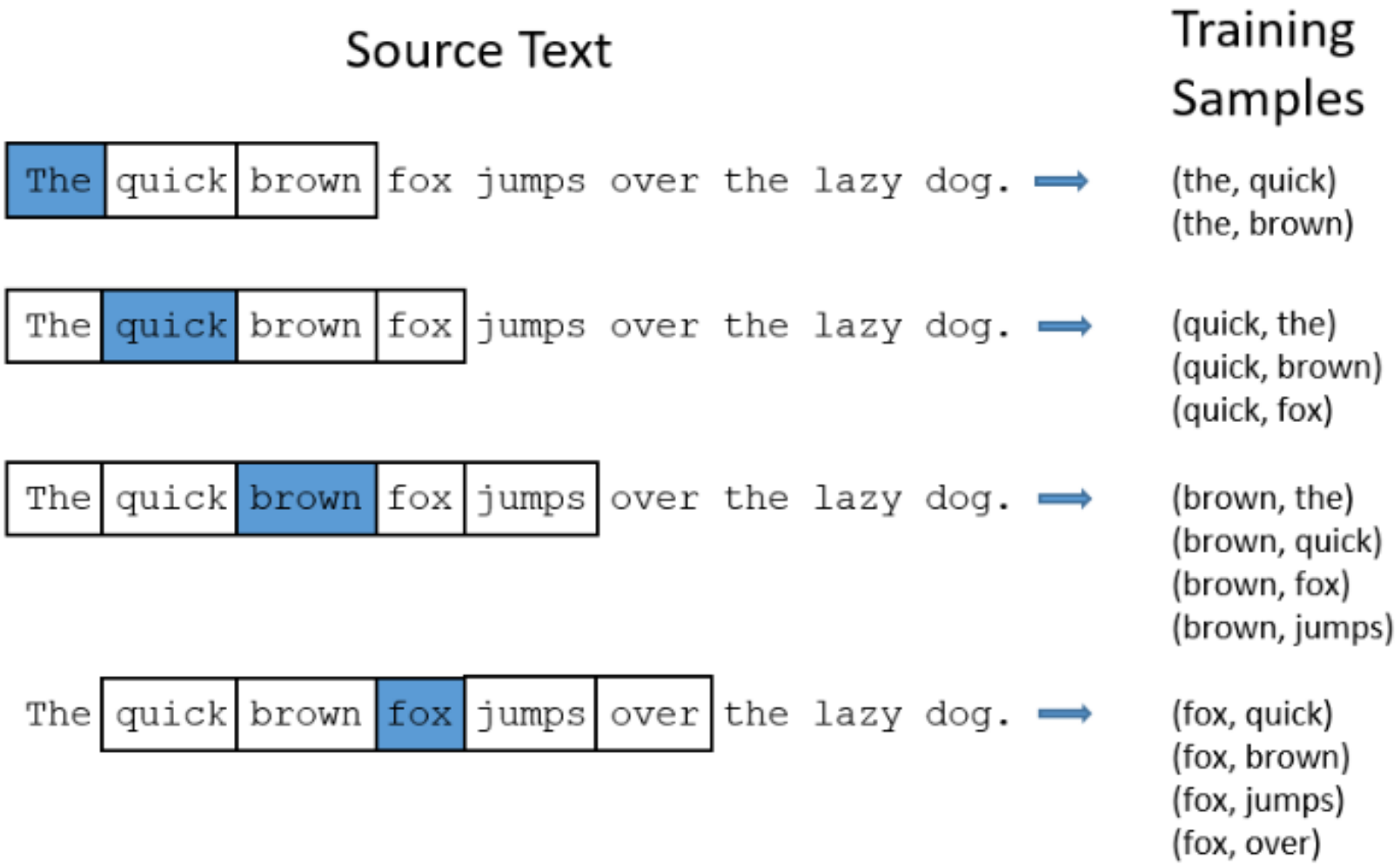
1. Captioning + Search

- Embedding comparison with Word2Vec
- Image Search with ElasticSearch

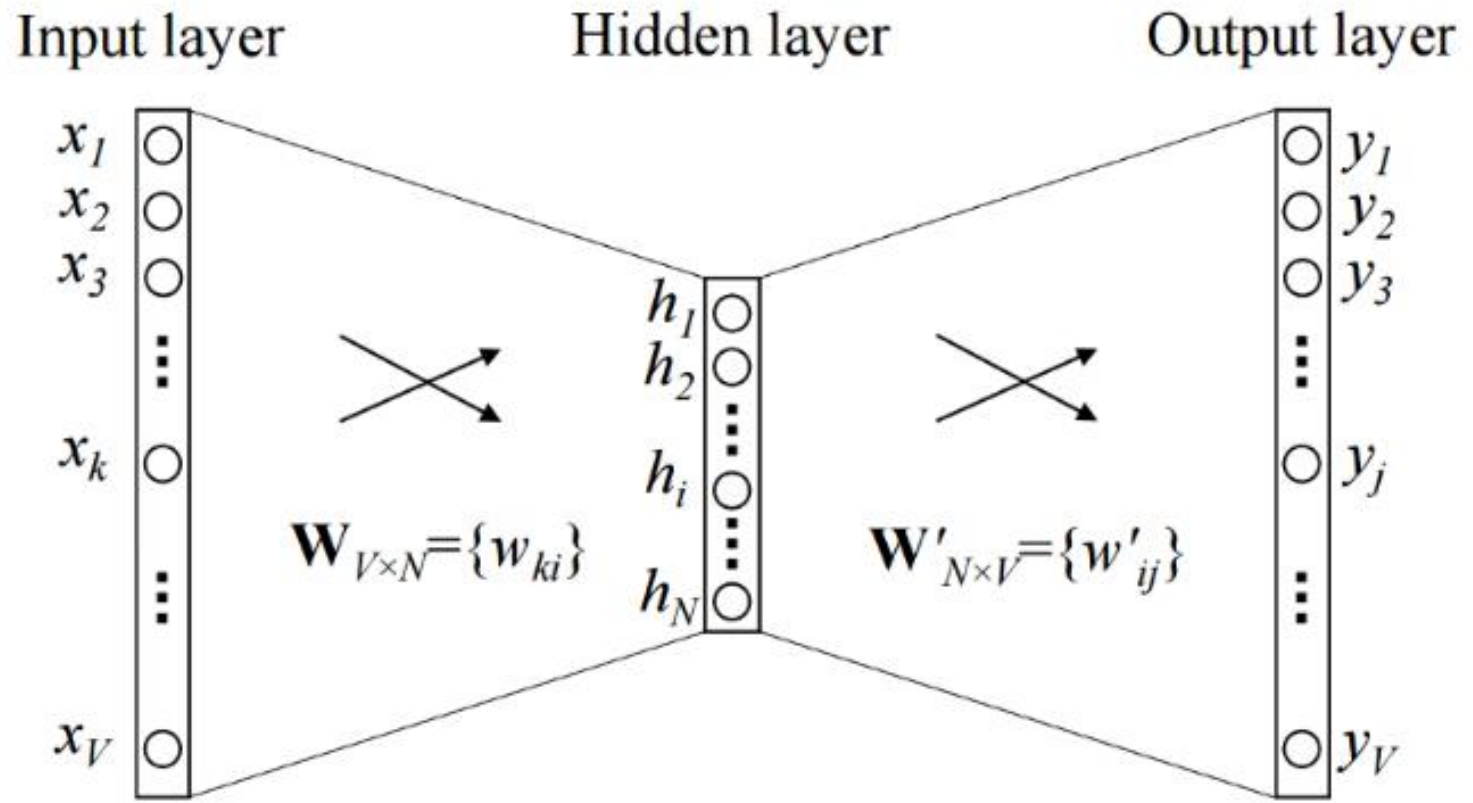
2. Text-image matching

- Cross-Modal-Projection-Learning (CMPL)
- Stacked Cross Attention (Microsoft paper)

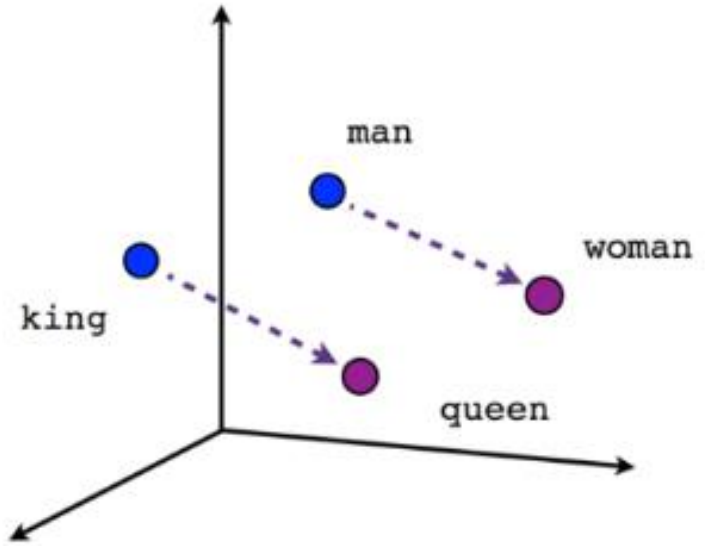
Word2vec


Source Text
Training Samples
The quick brown fox jumps over the lazy dog.
(the, quick)
(the, brown)
The quick brown fox jumps over the lazy dog.
(quick, the)
(quick, brown)
(quick, fox)
The quick brown fox jumps over the lazy dog.
(brown, the)
(brown, quick)
(brown, fox)
(brown, jumps)
The quick brown fox jumps over the lazy dog.
(fox, quick)
(fox, brown)
(fox, jumps)
(fox, over)

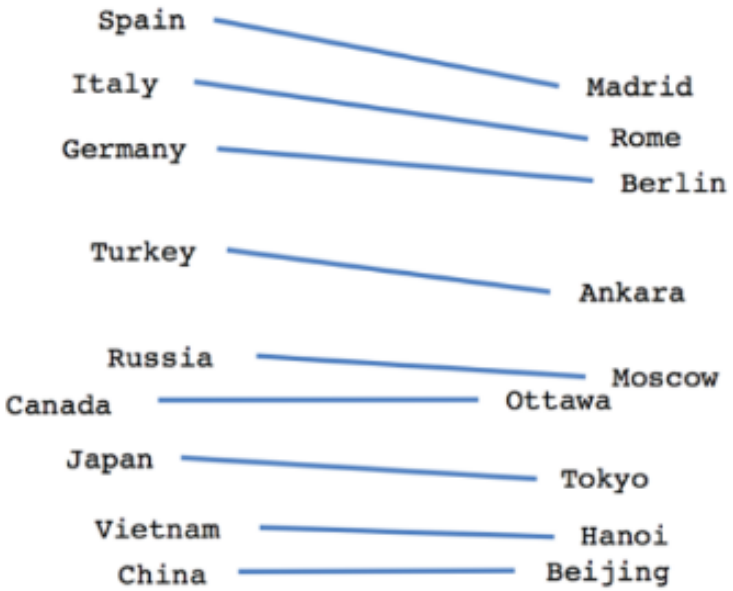
Word2vec


Input layer
Hidden layer
Output layer
x_1
x_2
x_3
x_k
x_V
h_1
h_2
h_i
h_N
y_1
y_2
y_3
y_j
y_V
$\mathbf{W}_{V\times N}=\{w_{ki}\}$
$\mathbf{W}'_{N\times V}=\{w'_{ij}\}$

Word2vec. Examples



Male-Female



Country-Capital

“man” – “woman” + “queen” = “king”

Embedding comparison with Word2Vec

Query: a group of people under the rain

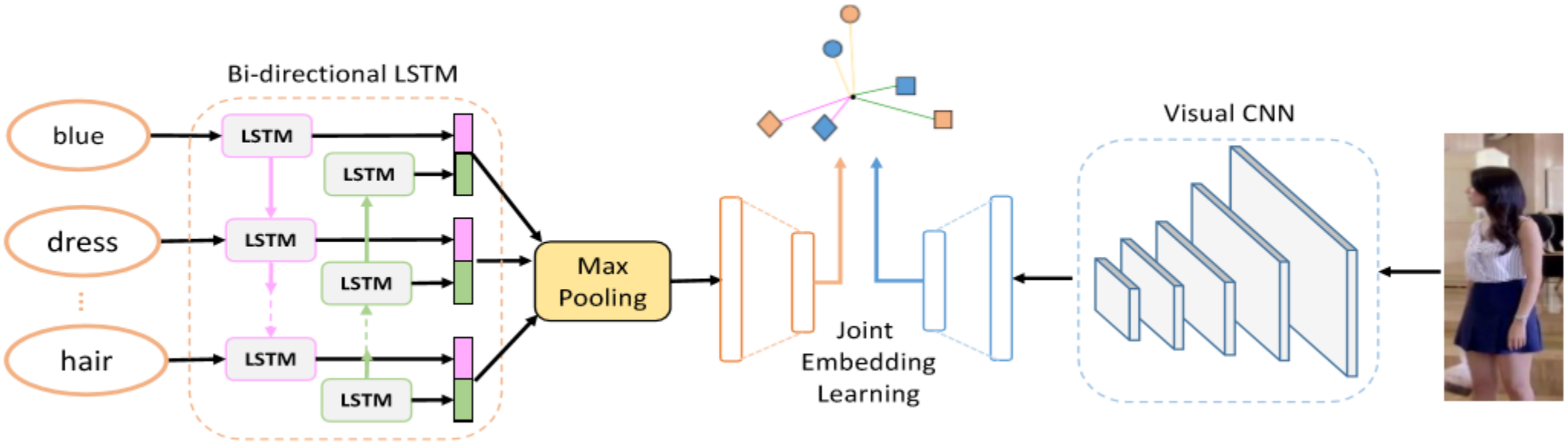
Image Search with ElasticSearch

- Query: a group of people under the rain

a group of people standing under umbrellas in the rain

a group of people sitting under umbrellas on a beach

Cross-Modal-Projection-Learning (CMPL)



https://drive.google.com/file/d/1Xp285WFwTZIE6nVu5Hi54ar4fodKsmjy/view

CMPL

Given a mini-batch with n image and text samples, for each image $\boldsymbol{x}_i$ the image-text pairs are constructed as $\{(\boldsymbol{x}_i, \boldsymbol{z}_j), y_{i,j}\}_{j=1}^{n}$, where $y_{i,j} = 1$ means that $(\boldsymbol{x}_i, \boldsymbol{z}_j)$ is a matched pair, while $y_{i,j} = 0$ indicates the unmatched ones. The probability of matching $\boldsymbol{x}_i$ to $\boldsymbol{z}_j$ is defined as

$$p_{i,j} = \frac{\exp(\boldsymbol{x}_i^\top \bar{\boldsymbol{z}}_j)}{\sum_{k=1}^{n} \exp(\boldsymbol{x}_i^\top \bar{\boldsymbol{z}}_k)} \quad s.t.\ \bar{\boldsymbol{z}}_j = \frac{\boldsymbol{z}_j}{\|\boldsymbol{z}_j\|} \tag{1}$$

where $\bar{\boldsymbol{z}}_j$ denotes the normalized text feature

Considering the fact that there might be more than one matched text samples for $\boldsymbol{x}_i$ in a mini-batch, we normalize the true matching probability of $(\boldsymbol{x}_i, \boldsymbol{z}_j)$ as

$$q_{i,j} = \frac{y_{i,j}}{\sum_{k=1}^{n} y_{i,k}} \tag{2}$$

CMPL

The matching loss of associating $\boldsymbol{x}_i$ with correctly matched text samples is defined as

$$\mathcal{L}_i = \sum_{j=1}^{n} p_{i,j} \log \frac{p_{i,j}}{q_{i,j}+\epsilon} \tag{3}$$

where ϵ is a small number to avoid numerical problems, and the matching loss from image to text in a mini-batch is computed by

$$\mathcal{L}_{i2t} = \frac{1}{n} \sum_{i=1}^{n} \mathcal{L}_i \tag{4}$$

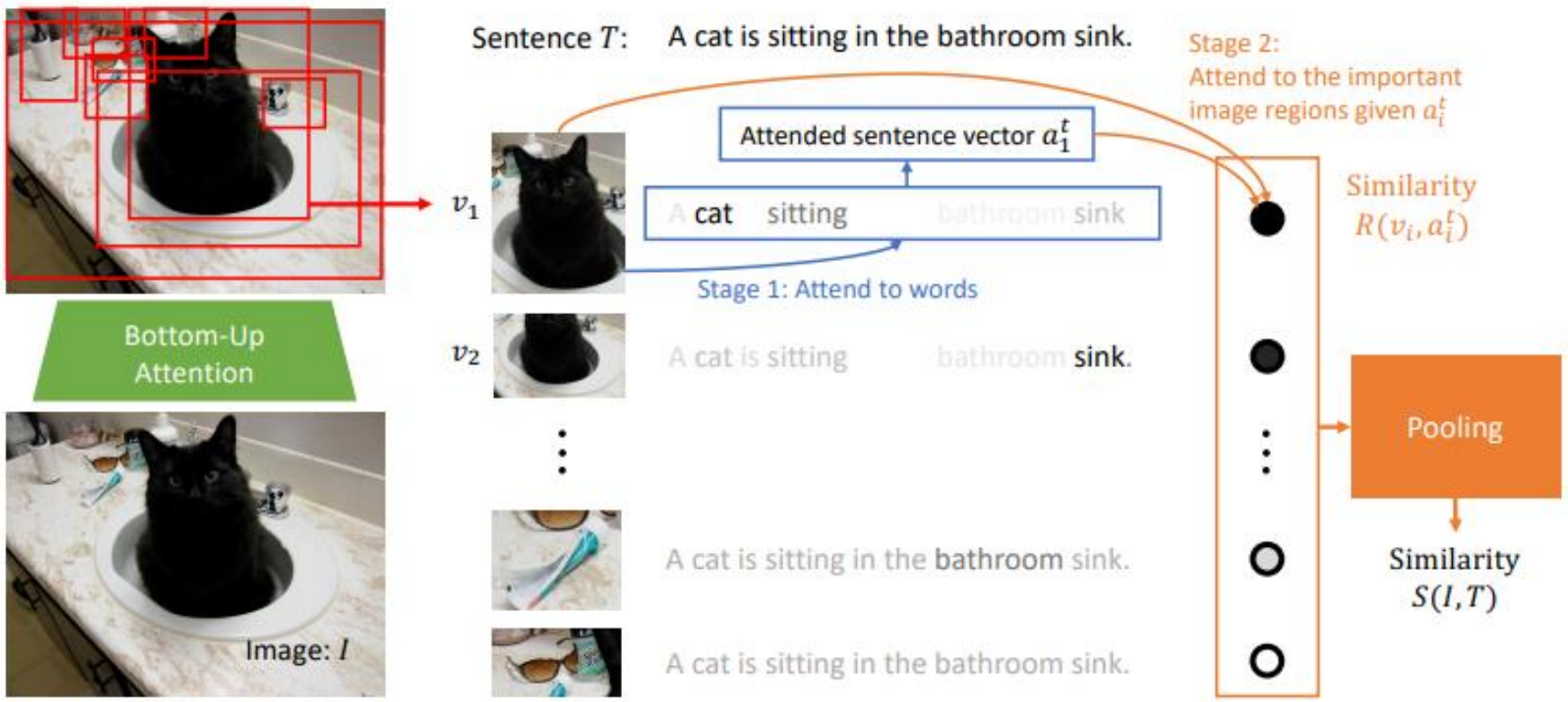
Note that Eq. 3 actually represents the KL divergence from distribution $\boldsymbol{q}_i$ to $\boldsymbol{p}_i$, and minimizing $KL(\boldsymbol{p}_i \| \boldsymbol{q}_i)$ attempts to select a $\boldsymbol{p}_i$ that has low probability where $\boldsymbol{q}_i$ has low probability [4]. Fig. 2 (b) illustrates the proposed matching

$$\mathcal{L}_{cmpm} = \mathcal{L}_{i2t} + \mathcal{L}_{t2i}$$

CMPL
Example

- View from car

Stacked Cross Attention (SCAN)



https://arxiv.org/abs/1803.08024

Image-Text Stacked Cross Attention

$$s_{ij} = \frac{v_i^T e_j}{||v_i|| ||e_j||}, i \in [1,k], j \in [1,n].$$

k – count of regions, n-count of words

$$\bar{s}_{ij} = [s_{ij}]_+ / \sqrt{\sum_{i=1}^{k} [s_{ij}]_+^2}, \text{ where } [x]_+ \equiv max(x,0).$$

$$\alpha_{ij} = \frac{exp(\lambda_1 \bar{s}_{ij})}{\sum_{j=1}^{n} exp(\lambda_1 \bar{s}_{ij})},$$

$$a_i^t = \sum_{j=1}^{n} \alpha_{ij} e_j,$$

$$R(v_i, a_i^t) = \frac{v_i^T a_i^t}{||v_i|| ||a_i^t||}.$$

$$S_{LSE}(I,T) = log(\sum_{i=1}^{k} exp(\lambda_2 R(v_i, a_i^t)))^{(1/\lambda_2)},$$

$$S_{AVG}(I,T) = \frac{\sum_{i=1}^{k} R(v_i, a_i^t)}{k}.$$

Stacked Cross Attention

Loss

In this study, we focus on the hardest negatives in a mini-batch following Fagphri *et al.* [10]. For a positive pair (I, T), the hardest negatives are given by $\hat{I}_h = argmax_{m \neq I} S(m, T)$ and $\hat{T}_h = argmax_{d \neq T} S(I, d)$. We therefore define our triplet loss as

$$l_{hard}(I, T) = [\alpha - S(I, T) + S(I, \hat{T}_h)]_+ + [\alpha - S(I, T) + S(\hat{I}_h, T)]_+. \quad (12)$$

where $[x]_+ \equiv max(x, 0)$ and S is a similarity score function ($e.g.$ S_{LSE}).

Example

Query: cup of coffe

score = 0.43

score = 0.41

score = 0.38

Metrics

- Recall@K (1, 5, 10)

  The percentage of the queries where at least one ground-truth is retrieved among the top-K results

- AP@K (average precision)

  The percent of top-K scoring images whose class matches that of the text query, averaged over all the test classes

Metrics comparison (1k dataset)

- Image retrieval (text to image)

	r@1	r@5	r@10
SCAN	55.2	86.9	94.2
CMPL	40.9	73.9	85.2

- Sentence retrieval (image to text)

	r@1	r@5	r@10
SCAN	69.7	94.4	97.8
CMPL	51.4	80.8	89.8

Validation set

- 5905 images in Chinese manager gallery
- 120 queries (each query has 10 marked images)

nDCG	Mean rank	Median rank
0.88	133	9.5

nDCG (normalized Discounted Cumulative Gain)

$$\mathrm{DCG_p} = \sum_{i=1}^{p} \frac{rel_i}{\log_2(i+1)} = rel_1 + \sum_{i=2}^{p} \frac{rel_i}{\log_2(i+1)}$$

$$\mathrm{nDCG_p} = \frac{DCG_p}{IDCG_p},$$

where IDCG is ideal discounted cumulative gain,

Demo

- Flickr8k demo


MAKE it
POSSIBLE