Human Eye - Task 7

2017202090 Liu Yifan

Problem Definition

Given a picture I, the purpose of this problem is to generate a series of words $X = \{x_1, x_2, x_3, \ldots, x_N\}$ which can describe the given picture I reasonably.

Baseline Method

The baseline idea is based on *Show & Tell： A Neural Image Caption Generator*. The method showed in the paper is a supervised learning method by using large dataset which has pairs of images and descriptions to train a generative model. The general idea is to extract features from image using CNN and then put the features and word embeddings of the corresponding description into a RNN.

Dataset

1. MS-COCO 2017

   Training Set: 118,287 images (each image with 5 descriptions)

   Validation Set: 5000 images

2. Google conceptual captions

   Training Set: 3,318,333 images (each image with 1 description)

Data Preprocess

We need to apply preprocess to both image data and text data(the caption of the related image) in both training phase and inference phase.

- For **Image**, apply data augmentation(such as random crop, random flip and so on)
- For **Text**:
    1. Remove infrequent words (frequency < 5) to build vocabulary.
    2. Word embedding to get the representation of each word. (Embedding Size = 300, the default embedding size of *spaCy* language model)
    3. Do sequence padding.

Model

The model has two parts, the first one is a convolution neural network serving as an **Encoder** which is used to extract features from images, and the second part is a recurrent neural network serving as a **Decoder** to generate the reasonable sequence.

Encoder

Use inception_v3 pretrained model with the top layer retrained based on training dataset.

Trainable parameters: weights of the top fully connected layer.

Decoder

Use 1-layer LSTM/GRU and one linear layer as the decoder

Input sequence:

x_0 is the extracted features from the encoder above.

$X = \{x_1, x_2, x_3 \dots x_N\}$ is the embedding sequence of the given image.

Trainable parameters: LSTM/GRU parameters (the embeddings are derived from spaCy lanuage model and here are set to be not trainable)

Training Phase

The goal of training phase is to maximize the joint probability which describes the occurrence of the given text sequence.

Loss function: use cross entropy loss

Optimizer: Adam optimizer

Inference Phase

The goal of inference phase is to generate sequence of words to describe the given picture.

Input & Output:

x_0 is the extracted features from encoder.

x_1 is the output of decoder when input is x_0.

x_2 is the output of decoder when input is x_1.

...

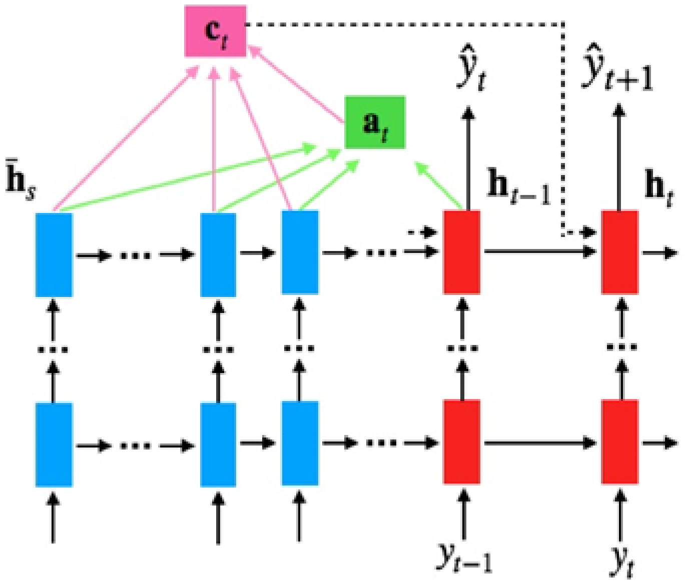
Until the inferred sequence meet `<END>` symbol or the length of the inferred sequence meets `max_length`.

As for the criterion for deciding which word to choose for next inference step, now this method uses **Greedy Search** (just choose the word having maximum occurrence probability). The future method will apply **Beam Search** as well.

Attention

Bahdanau Attention

$$\mathbf{h}_{t-1} \rightarrow \mathbf{a}_t \rightarrow \mathbf{c}_t \rightarrow \mathbf{h}_t$$



The picture above shows **Bahdanau Attention** applied in Machine Translation scene. The blue blocks refer to sentence in source language and the red blocks refer to target language.

In image captioning scene, we divide each image into several regions, and extract features of each region. These features can be seen as h_s. The y_t in the picture above represents the input word vector of target sentence.

This process can be described as (soft attention):

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

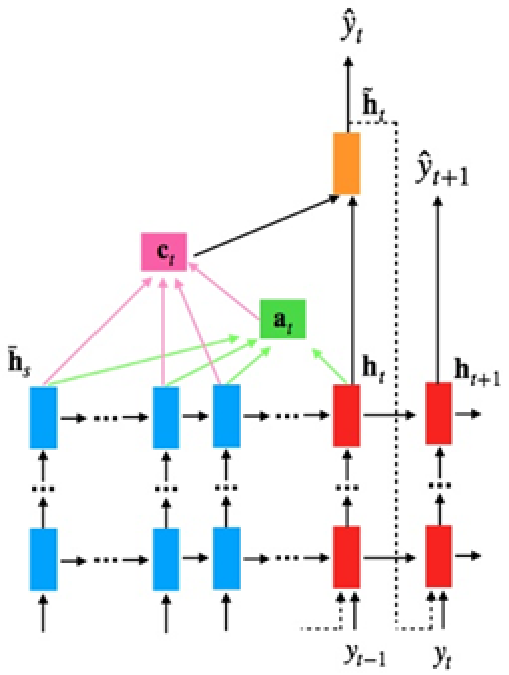
$$\alpha_{ti} = \frac{\exp(e_{ti})}{\sum_{k=1}^{L} \exp(e_{tk})}.$$

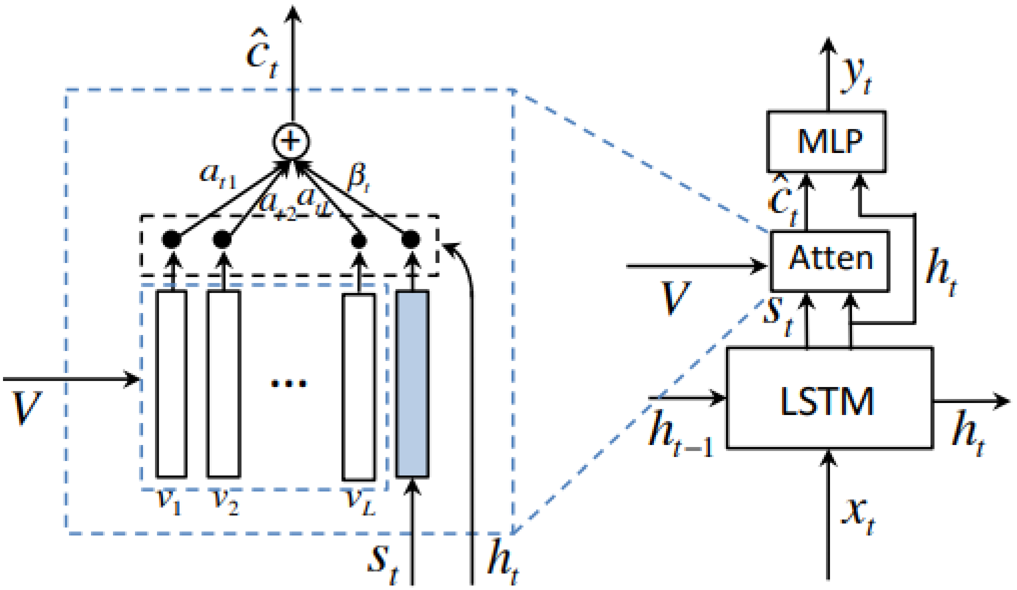
$$\hat{\mathbf{z}}_t = \phi\left(\{\mathbf{a}_i\}, \{\alpha_i\}\right)$$

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

Luong Attention + Visual Sentinel

$$\mathbf{h}_t \rightarrow \mathbf{a}_t \rightarrow \mathbf{c}_t \rightarrow \tilde{\mathbf{h}}_t$$





Bahdanau Attention use h_{t-1} to apply attention to calculate context vector, but actually we care more about the which region in the image is the most important part related to the **current** word but not the last word, so the Luong Attenion was proposed to improve the attention mechanism.

Results of the method using Bahdanau Attention shows that when the input word is not related to visual conceptions or meaningless such as `'is', 'the', 'a'`, the attention does not work well, thus results in repeating of meaningless words.

Based on this problem, some work proposed **Visual Sentinel**, a module serving as a trade-of-tool to decide the weight of attended context vector and LSTM/GRU state vector. So if the state in memory cell is more important, the generation of the current word will depend on the sequence before more on the given image, which is more reasonable for the words not related to certain visual conceptions.

The process above can be described as:

$$\begin{aligned}\boldsymbol{z}_t &= \boldsymbol{w}_h^T \tanh(\boldsymbol{W}_v \boldsymbol{V} + (\boldsymbol{W}_g \boldsymbol{h}_t)\mathbb{1}^T) \\ \boldsymbol{\alpha}_t &= \operatorname{softmax}(\boldsymbol{z}_t)\end{aligned}$$

$$\boldsymbol{c}_t = \sum_{i=1}^{k} \boldsymbol{\alpha}_{ti} \boldsymbol{v}_{ti}$$

$$\begin{aligned}\boldsymbol{g}_t &= \sigma\left(\boldsymbol{W}_x \boldsymbol{x}_t + \boldsymbol{W}_h \boldsymbol{h}_{t-1}\right) \\ \boldsymbol{s}_t &= \boldsymbol{g}_t \odot \tanh\left(\boldsymbol{m}_t\right)\end{aligned}$$

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

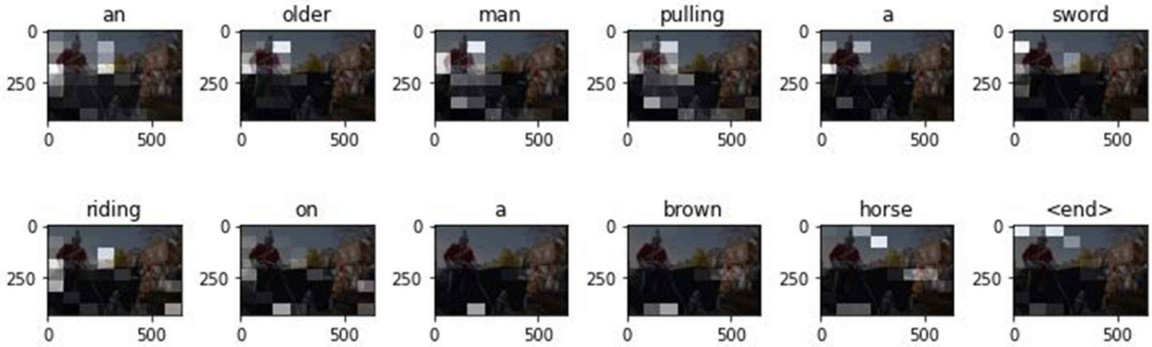
$$\hat{\boldsymbol{\alpha}}_t = \operatorname{softmax}([\boldsymbol{z}_t; \boldsymbol{w}_h^T \tanh(\boldsymbol{W}_s \boldsymbol{s}_t + (\boldsymbol{W}_g \boldsymbol{h}_t))])$$

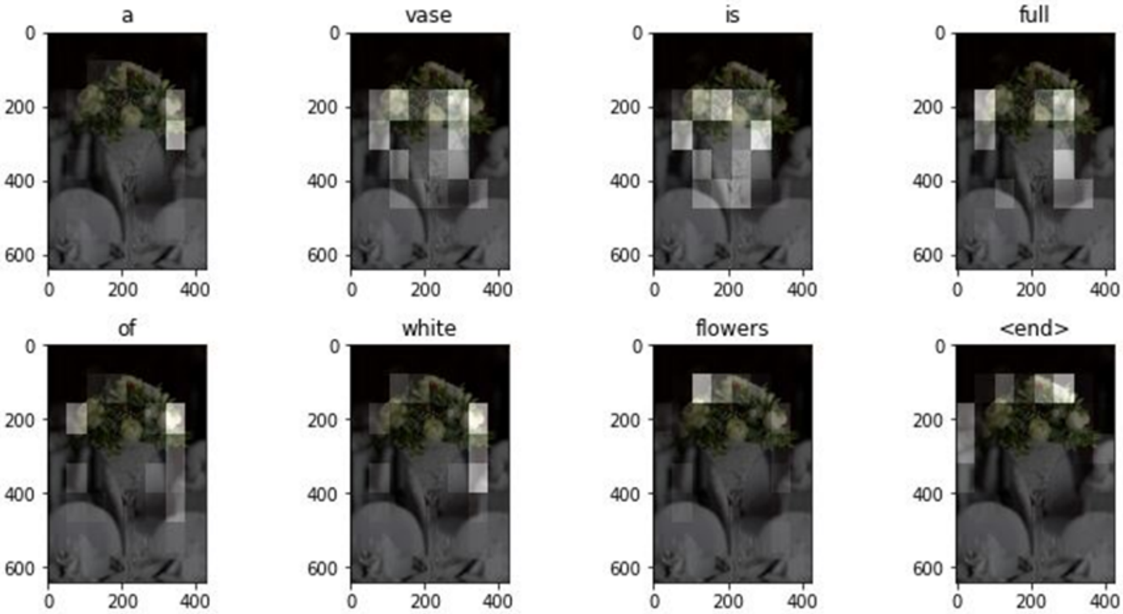
$$\beta_t = \boldsymbol{\alpha}_t[k+1].$$

where V is the features of regions in the image, h_t is the hidden output of LSTM/GRU, $\hat{c}_t$ is the adapted context vector.

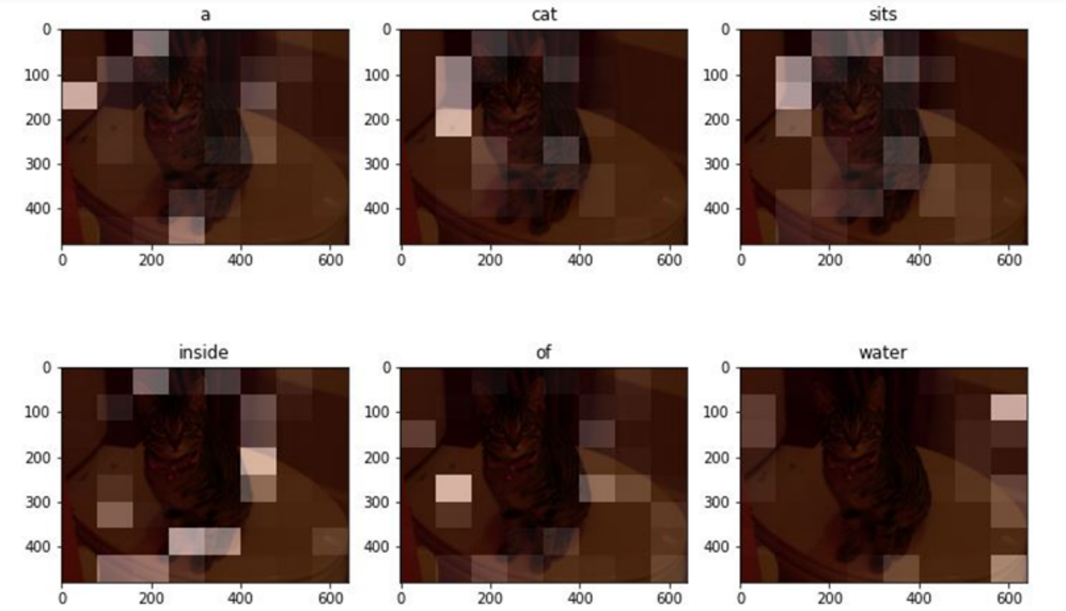
Result

- good example





- fair example



Integrating knowledge and reasoning in image understanding