南京大学NLP夏令营

Image Caption小组 结题汇报

NLP Summer Camp of Nanjing University
Concluding Report of Image Captioning Group

指导老师：张建兵
组长：马征
组员：程瞰之 张雨 朱鑫浩

目录

CONTENTS

任务介绍

Task Introduction

What is Image Captioning?

- Generating descriptive caption(s) for a picture
- Combining NLP and CV
- Being easy for human, while hard for machine because:
  1. It needs to detect objects in a picture.
  2. It needs to figure out interactions between objects.
  3. It needs to use natural language to describe them.

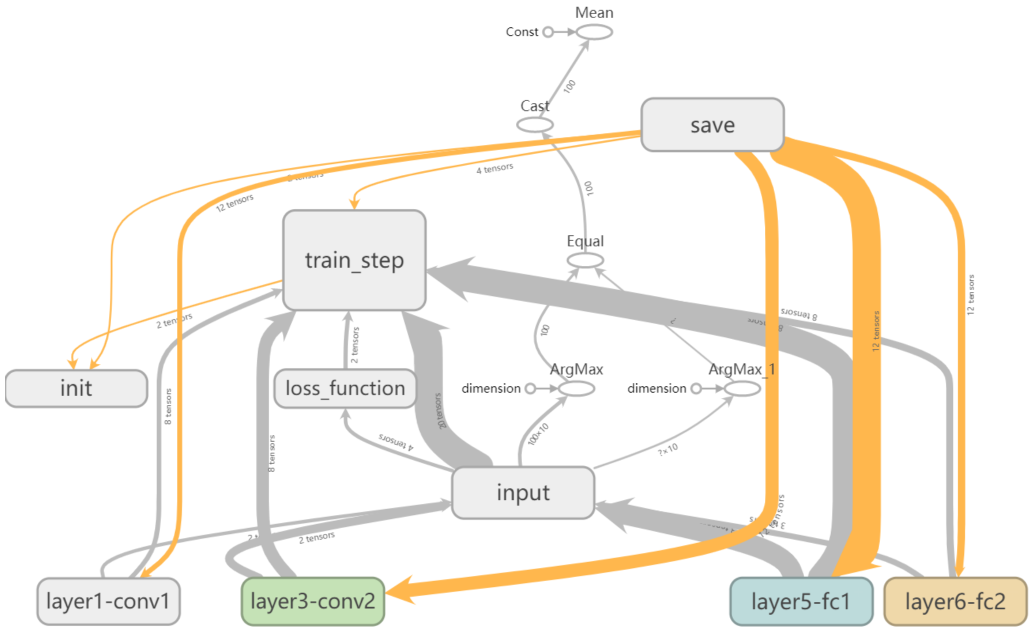
A person riding a motorcycle on a dirt road.

项目历程：第一周

Details about program: Week 1

Getting Started: CNN & MNIST

1. Basic knowledge of neural networks (DNN, CNN)
2. Introduction to Pytorch
3. Solution to MNIST & CIFAR-10 image classification task (based on DNN & CNN)



项目历程：第一周

Details about program: Week 1

Part of week 1 report

图像输入部分

1. 数据读取：用CPU进行所有的图像读取与预处理工作，用GPU进行模型的训练工作，提高模型训练的效率。在训练开始时，预处理20000张处理过的CIFAR图像填充到随机化处理队列中，避免图像I/O过程影响模型的训练速度。
2. 图像增强（训练过程中）：对原始图像进行随机切割，翻转，调整（随机失真），增大训练样本的数据量。
    1. 切割：略小于原始图像，增加训练数据量并减小计算量
    2. 翻转：对图像随机进行左右翻转
    3. 亮度调整：对图像随机进行亮度调整（在一定范围内）
    4. 对比度调整：对比度增大阈值大于减小阈值（高对比度常常有助于识别）
3. 图像增强（测试过程中）：
    1. 切割：从原图像中心进行切割，防止影响图像主体
    2. 标准化：对原图像的RGB值进行线性标准化，使模型对图像的动态范围变化不敏感。

模型预测部分

该网络在AlexNet的基础上进行了一定修改，其模型结构如下。

层名称	说明
conv1	采用5x5卷积核，步长为1，全0填充，过滤器深度为64，激活函数为ReLU
pool1	采用3x3最大池，步长为2*
norm1	LRN层，对同一层响应较小的神经元进行抑制
conv2	采用5x5卷积核，步长为1，全0填充，过滤器深度为64，激活函数为ReLU
norm2	LRN层，对同一层响应较小的神经元进行抑制
pool2	采用3x3最大池，步长为2
local3	含有384个节点的全连接层，激活函数为ReLU
local4	含有192个节点的全连接层，激活函数为ReLU
softmax_linear	生成最终结果的softmax层

*：重叠池化（Overlapping Pooling），步长小于池化范围，可以抽取更强的特征表达，但增大了计算量。

模型调整

训练机配置如下：

CPU : I7-8700 (6C12T)
GPU: RTX2060 (6G)
Tensorflow-gpu = 1.14.0
CUDA = v10.0
cuDNN = v7.3.1

原始参数如下：

```
Batch_size = 128
Steps = 20000
Moving_Average_Decay = 0.9999
Num_Epochs_per_decay = 350
Learning_Rate_Decay_Factor = 0.1
nitial_Learning_Rate = 0.1
```

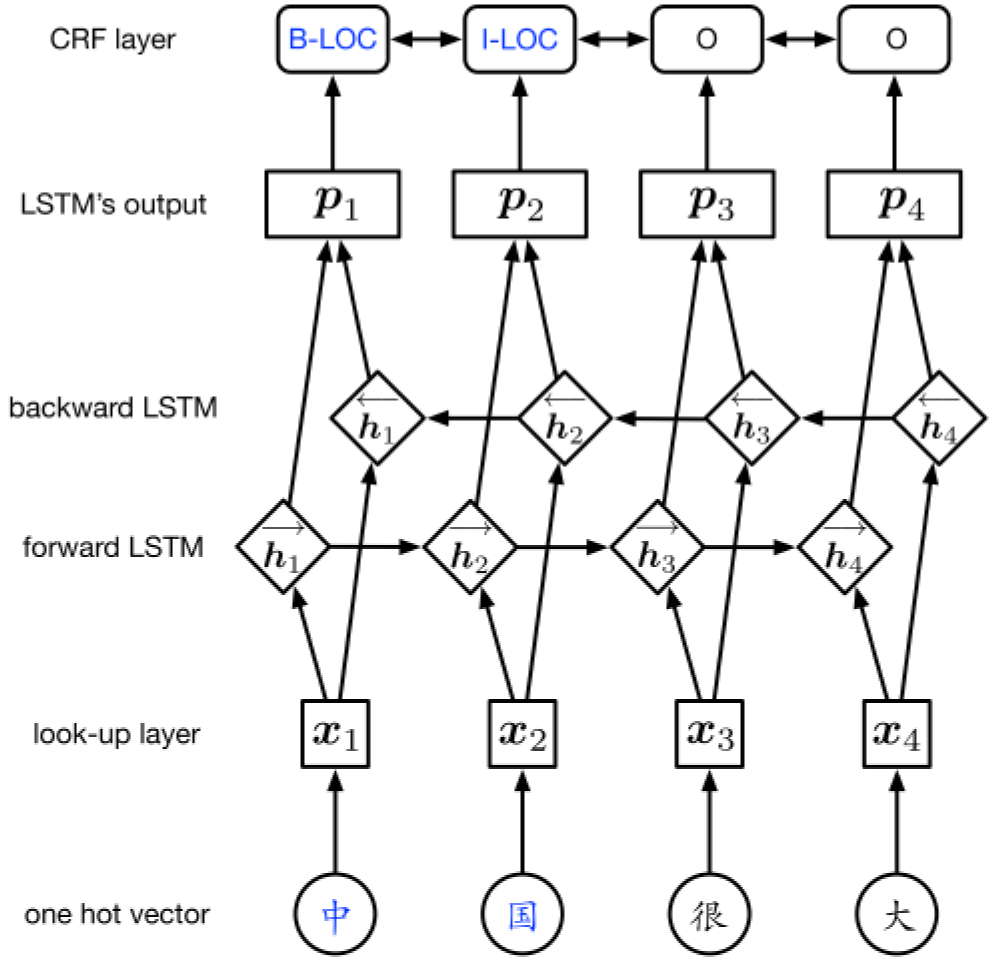
改动描述	训练速度	测试集正确率
原始参数	9600 ex./s, 0.013 sec/batch	89.9%
Batch_size = 256 (增大一倍)	11000 ex./s, 0.023 sec/batch	92.7%
不使用滑动平均	9800 ex./s, 0.013sec/batch	87.6%
删去两个LRN层	13000 ex./s, 0.010sec/batch	90.0%
在local4与softmax_linear间加入84神经元的全连接层	9200 ex./s, 0.014sec/batch	89.4%
Batch_size = 256, 训练次数= 50k	11000 ex./s, 0.023sec/batch	96.1%

项目历程：第二周

Details about program: Week 2

RNN (LSTM) & NER Task

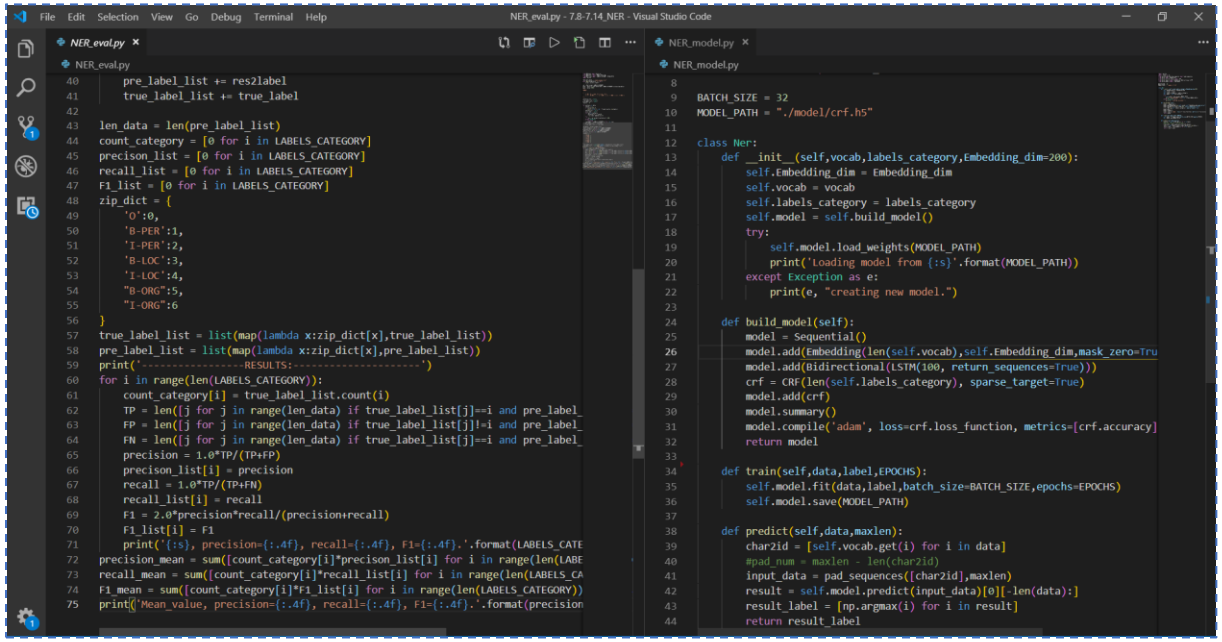
1. Basic knowledge of RNN (LSTM)
2. Introduction to Named Entity Recognition, NER
3. Solution to NER (based on CNN & Bi-LSTM & CRF)



项目历程：第二周

Details about program: Week 2

Part of week 2 code & report



Batch_size = 16, Epoches = 3

类别	Precision	Recall	F1-score
O	98.36%	98.91%	0.9863
B-PER	61.76%	70.00%	0.6562
I-PER	59.65%	73.91%	0.6602
B-LOC	76.00%	71.25%	0.7355
I-LOC	68.03%	66.94%	0.6748
B-ORG	67.14%	74.60%	0.7068
I-ORG	77.94%	49.53%	0.6057
Mean	96.97%	97.08%	0.9697

Batch_size = 32, Epoches = 4

类别	Precision	Recall	F1-score
O	98.58%	98.68%	0.9863
B-PER	57.14%	70.59%	0.6316
I-PER	53.12%	68.00%	0.5965
B-LOC	68.75%	57.89%	0.6286
I-LOC	69.23%	49.09%	0.5745
B-ORG	47.62%	47.62%	0.4762
I-ORG	67.14%	74.60%	0.7068
Mean	96.88%	96.87%	0.9684

项目历程：第三、四周

Details about the program: Week 3&4

Show and Tell

1. Advanced usage of Pytorch
2. Reading Material : Show and Tell: A Neural Image Caption Generator, CVPR 2015.
3. Repetition of this paper

Part of week 3 report

- 论文内容
  - 首先介绍了整个工作，然后提及了其他研究人员的相关工作
  - 接着介绍了model的建立，借鉴于机器翻译的发展, 提出了最大化给定图片生成正确描述的概率，用RNN对概率进行建模，基于LSTM生成文本。
  - 文中提出的模型NIC，由于研究已经证明，CNN可以从输入图像中充分地提取特征并嵌入到一个定长的向量中，所以用CNN作一个编码器，并且进行预训练，然后将其最后一层隐藏层作为作为RNN的输入。
  - 随后介绍实验的评价标准，在数据集上得到的数据结果，对生成的图像结果是否有多样性新颖性判断，各种排名的结果比较。并且有对词嵌入进行分析。
  - 最后是总结

- 细节内容
  - 难点：既有视觉分析还有语言模型，要体现出检测出的图像中的物体之间的关系，以前的经验是作为两个子问题处理，而本文中是作为一个整体处理的。
  - 参考的机器翻译相关工作：使用一个RNN为encoder输入源语句，然后转换为长度固定的特征向量，紧接着这些向量作为decoder的RNN的初始隐藏层状态。最后使用该RNN来生成target语句。
  - NIC模型：在机器翻译中，使用有一个编码RNN、一个解码RNN，这里把编码RNN替换成CNN。也就是说，用CNN作一个编码器，并且在ImageNet上进行预训练，然后将其最后一层隐藏层作为RNN的输入。
  - 模型的具体建模
  - 使用LSTM进行解码：为了避免RNN的梯度爆炸与弥散问题，LSTM的定义及更新规则，以及训练过程，在此略过。就是之前学习的LSTM的内容
  - NIC推理的方法： Sampling方法, 即每次只选择概率最大的值生成单词； BeamSearch方法， 每次选择概率最大的k个值
  - 防止过拟合：使用预训练权重（ImageNet）来初始化CNN，dropout，集成学习等
  - 词嵌入分析

项目历程：第五周

Details about the program: Week 5

Discussing & Reading

1. Discussion on Image Captioning
2. Reading Materials:
    1. 程瞰之：

       (GAN) Towards Diverse and Natural Image Descriptions via a Conditional GAN, ICCV 2017
    2. 张雨：

       (Attention) Show, Attend and Tell: Neural Image Caption Generation with Visual Attention, ICML 2015
    3. 朱鑫浩：

       (Dense Captioning) Dense Captioning with Joint Inference and Visual Context, CVPR 2017

       (RL) Actor-Critic Sequence Training for Image Captioning, CVPR 2017

项目历程：第五周

Details about the program: Week 5

Part of week 5 report

soft attention 和 hard attention 小结

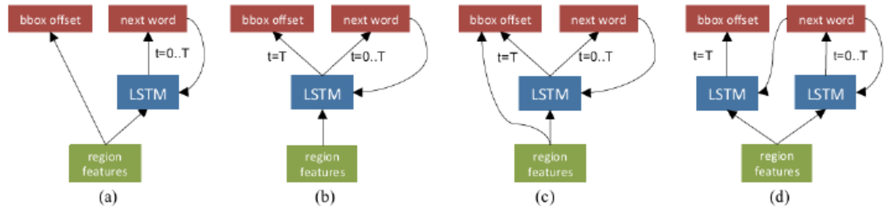
- attention
  - 把输入X编码成一个固定的长度，对于句子中每个词都赋予相同的权重，这样是不合理的，没有区分度往往使模型性能下降。因此提出Attention Mechanism，用于对输入X的不同部分赋予不同的权重，进而实现软区分的目的
  - 2015年发表论文《Show, Attend and Tell: Neural Image Caption Generation with Visual Attention》，在Image Caption中引入了Attention，当生成第i个关于图片内容描述的词时，用Attention来关联与i个词相关的图片的区域
- Soft Attention
  - Soft Attention是所有的数据都会注意，都会计算出相应的注意力权值，不会设置筛选条件
  - 可以直接求导，进行梯度反向传播
- Hard Attention
  - Hard Attention会在生成注意力权重后筛选掉一部分不符合条件的注意力，让它的注意力权值为0，即可以理解为不再注意这些不符合条件的部分
  - Hard Attention是一个随机的过程。Hard Attention不会选择整个encoder的隐层输出做为其输入，Hard Attention会依概率 S_i 来采样输入端的隐状态一部分来进行计算，而不是整个encoder的隐状态。不可微，不能后向传播，因为采样梯度为0，为了实现梯度的反向传播，需要采用蒙特卡洛采样的方法来估计模块的梯度

解决该问题的模型由**两部分**组成：Region Detection Network（生成RoI，提取RoI特征与环境特征）与 Localization and Captioning Network（生成检测得分，描述短语和偏移量）

对于模型的第一部分，我们使用基于CNN的**RPN**（受到 **faster R-CNN** 的启发）。以下将对模型的第二部分进行详细描述。

Joint Inference：获取准确的localization

Localization包括生成RoI与bbox offset两部分，而在此部分**bbox offset**是我们的主要关注对象。本文共提出了4种Joint Inference方法来生成bbox offset：



(a) : Baseline model —— bbox offset仅由RoI特征生成

(b) : Shared-LSTM (S-LSTM) —— 用已有LSTM层的最后一步输出生成offset

(c) : Shared-Concatenation-LSTM (SC-LSTM) —— 合并LSTM的输出与RoI特征生成offset（类似于ResNet）

(d) : Twin-LSTM (T-LSTM) —— 用两个LSTM网络分别生成caption与location。当caption完成时，location-LSTM此时收到完整的图像描述信息，并用整个caption作为输入生成offset。

项目历程：第六、七周

Details about the program: Week 6&7

Adaptive Attention

1. Reading Material:
   Knowing When to Look - Adaptive Attention via A Visual Sentinel for Image Captioning, CVPR 2017
2. Repetition of this paper
3. Building & Deployment of a visualization training system

Part of week 6 report

Encoder-Decoder结构

对于给定的图像与对应的Caption文本，Encoder-Decoder结构直接优化如下的目标函数：

$$\theta^* = arg\ max\ \theta \sum_{(I,y)} log\ p(y|I;\theta)$$

其中 θ 为模型的参数，I 为图像，$y = \{y_1, \dots, y_n\}$ 为对应的Caption文本。

根据链式法则，联合概率分布的对数似然可以被分解成如下的有序条件概率：

$$log\ p(y) = \sum_{t=1}^{T} log\ p(y_t|y_1, \dots, y_{t-1}, I)$$

为了方便起见，我们此处暂时不考虑对模型参数的依赖关系。

在Encoder-Decoder结构中，每个词语的条件概率可以被表示为：

$$log\ p(y_t|y_1, \dots, y_{t-1}, I) = f(h_t, c_t)$$

其中 f 是输出 y_t 概率的一个非线性函数，c_t 是在时刻 t 从图像 I 中提取出的视觉特征向量，h_t 是时刻 t RNN的hidden-state。

我们在本文中采用LSTM作为RNN的实际模型。对于LSTM，h_t 可以被如下表示：

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

其中 x_t 为输入向量，m_{t-1} 是在 $t-1$ 时刻的记忆单元。

- 基于注意力的视觉神经编码-译码模型的研究，引入注意力机制，生成一个空间图spatial map，标识了与每个生成的词语相关的图像区域
- 标注里不是所有的词都有对应的视觉信息，并且语言之间的关联性会使预测过程不怎么需要视觉信息。非视觉词汇的梯度，会误导和减弱视觉信息在控制标注语句生成过程的整体效果。
- 提出一个自适应注意力编码-译码框架，能够自动决定何时依赖视觉信息、何时依赖语言模型。在依赖视觉信息时，模型也决定了具体应该关注图像的哪块区域，为了提取空间图像特征，提出了一个新型的空间关注模型。采用了一个新的LSTM扩展方法，能够生成一个额外的视觉哨兵向量，而不是一个单一的隐藏状态。视觉哨兵是一个额外的对译码器存储的隐式表示，进一步设计一个新的哨兵门，决定译码器在生成下一词语时从图像中获取信息的多与少。
- 自适应注意力模型做了扩展分析，包括词语的视觉基础概率visual grounding probabilities和生成的注意力图attention maps的弱监督定位weakly supervised localization。

论文介绍：Show and Tell

Show and Tell: A Neural Image Caption Generator

Image Captioning

Input: Image (224x224x3 for MS-COCO dataset)
Output: Caption sequence (unfixed length)

Features

Image as input: use CNN to extract features from image
Sequence as output: use RNN to generate word sequence (1vsN)

Inspired by Seq2Seq model (RNN+RNN) in Machine Translation, we employ a similar encoder-decoder (CNN+RNN) framework to cope with this task.

论文介绍：Show and Tell

Show and Tell: A Neural Image Caption Generator

Optimizing

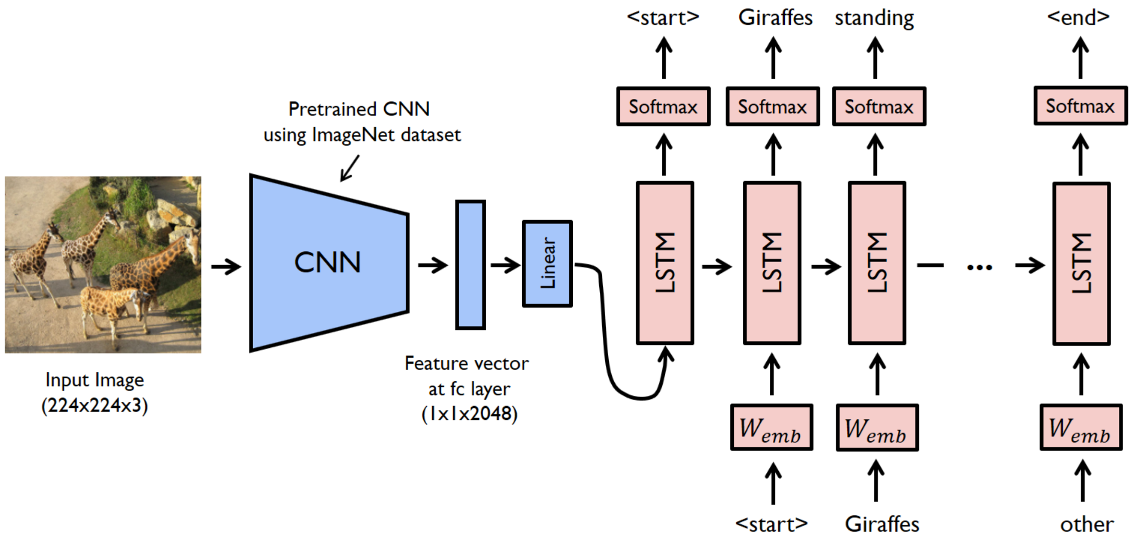
Training objective:

$$\theta^* = \arg\max \sum_{(I,y)} \log p(y|I;\theta)$$

Where θ is the parameters of model,
I is the given image,
and y is the given caption.

CNN: pretrained ResNet152 model (on ImageNet), all layers but the last one.
RNN: LSTM cells.

Using beam search to generate words

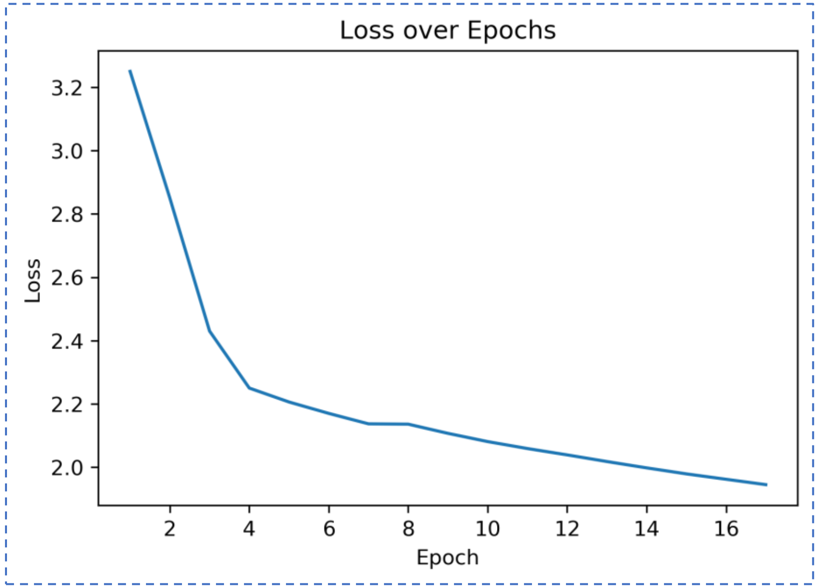


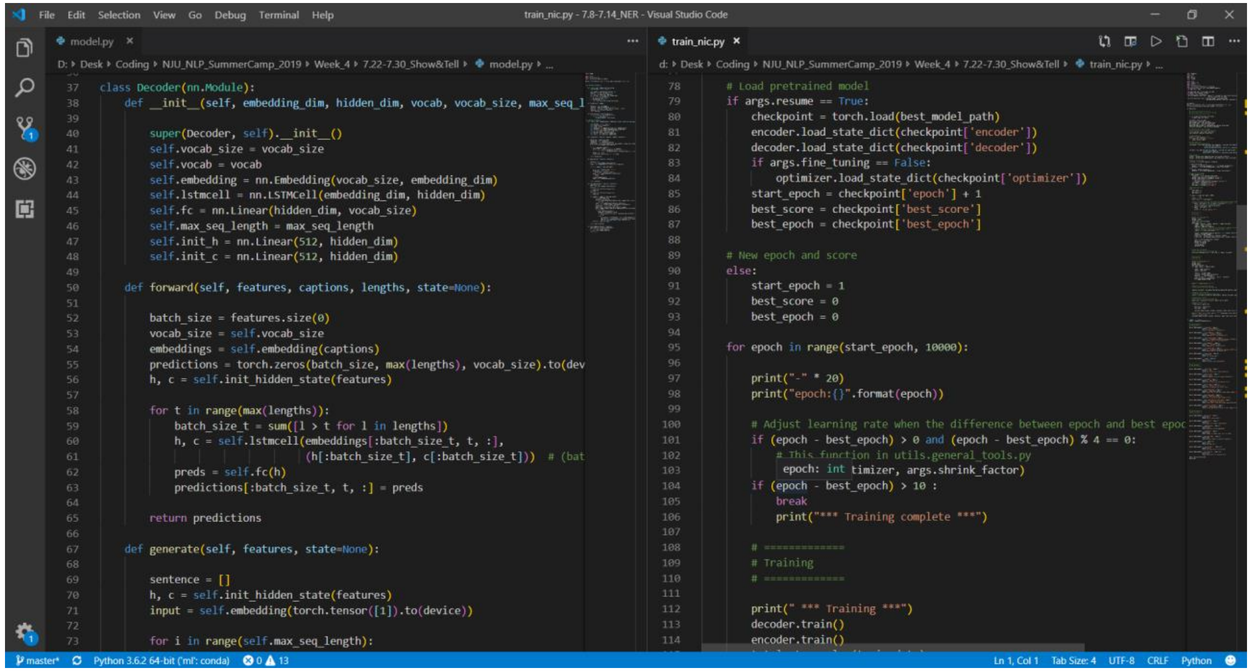
论文复现：Show and Tell

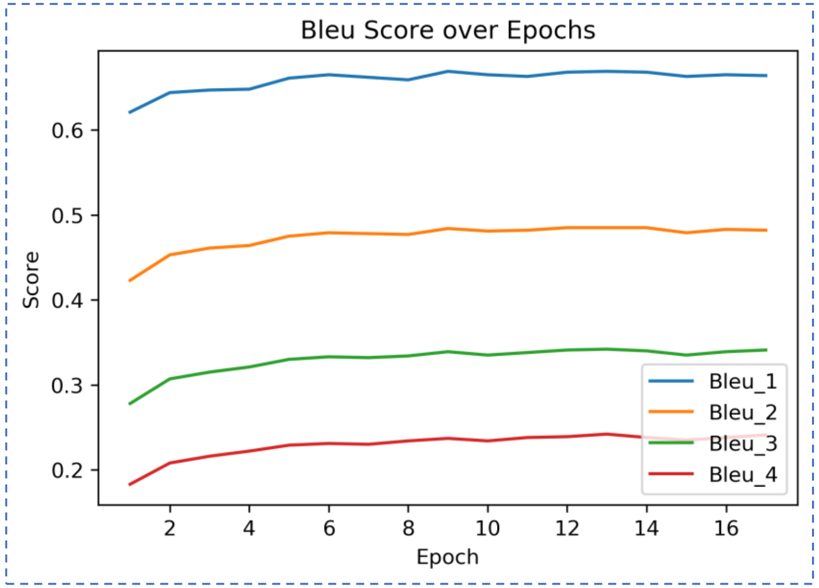
Show and Tell: A Neural Image Caption Generator

Dataset: MS-COCO 2014

Training: 2014 Contest Train images [83K images/13GB]
Validation: 2014 Contest Val images [41K images/6GB]
Test: 2014 Contest Test images [41K images/6GB]







论文介绍：Show, Attend and Tell

Show, Attend and Tell - Neural Image Caption Generation with Visual Attention

Attention-based-method

Attention made it possible for model to focus on a certain specific object in the picture when necessary.

Two attention mechanism: Hard attention and Soft Attention are introduced to cope with this task.

Attention can help visualizing and explaining the results of model.

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

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

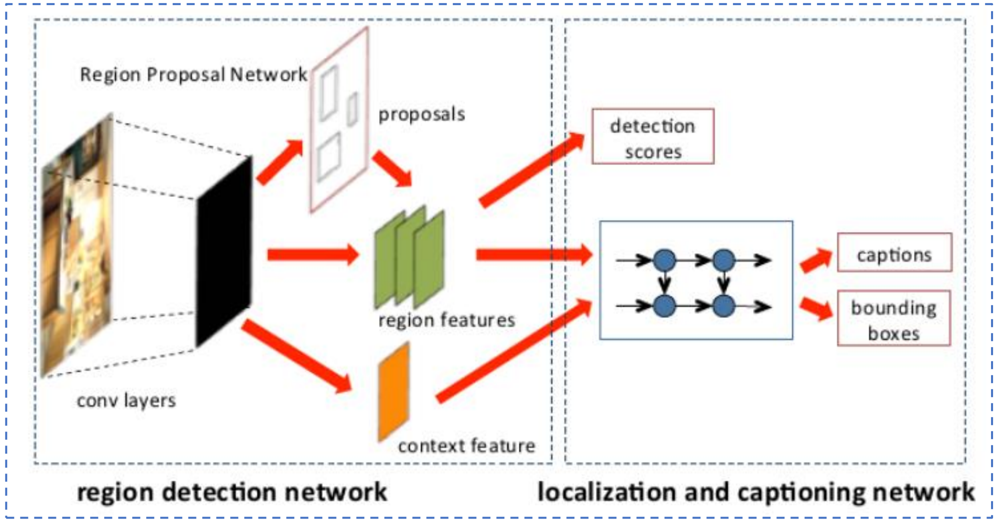
$$\alpha_{ti} = \frac{\exp(e_{ti})}{\sum_{k=1}^{L} \exp(e_{tk})}.$$

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

A large white bird standing in a forest.

论文介绍：Dense Captioning

Dense Captioning with Joint Inference and Visual Context



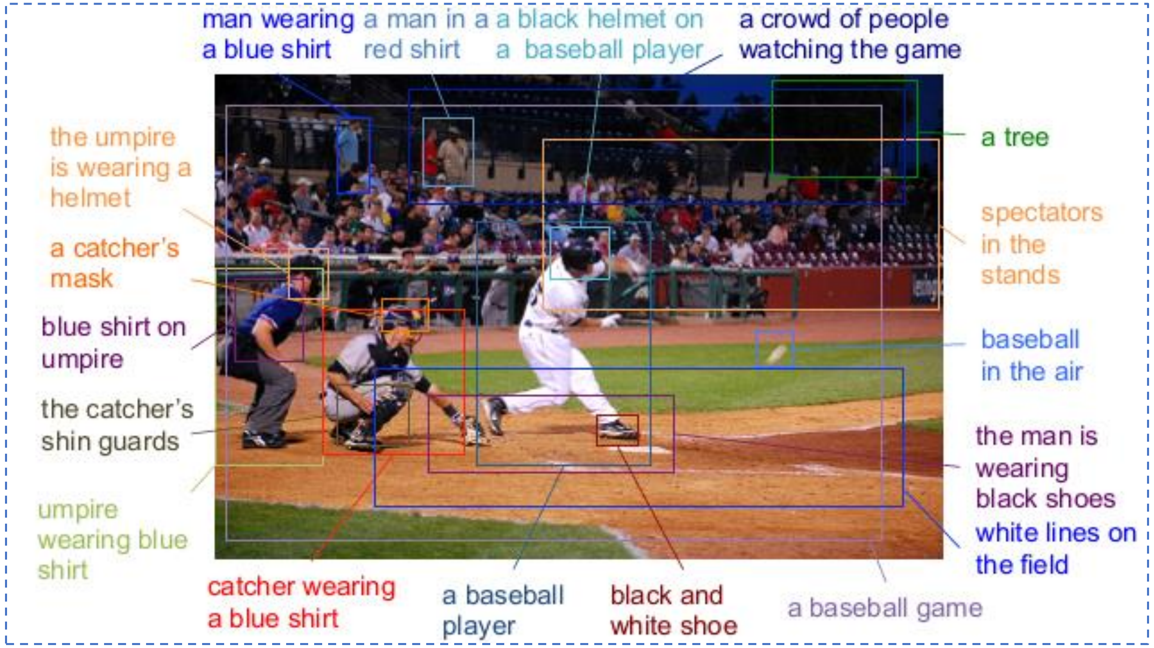
Dense Captioning method

Problems of current model:

1. Captions of current datasets focus on conspicuous objects in a picture, or just describe it in general, which is not a comprehensive understanding of image.
2. Captions tend to be subjective, which keeps evaluation from being accurate.

Contribution of this paper:

It uses a region proposal network (RPN) to figure out possible regions of interest (ROI), followed by captioning and region adjusting.



论文介绍：Actor-Critic Sequence Training

Actor-Critic Sequence Training for Image Captioning

RL-based method

This paper focuses on the optimization of decoder.

Problems of current model:

1. Exposure bias: on training stage, ground truth y_{t-1} is sent into RNN, not model-generated y_{t-1} on evaluating stage, which worsens the evaluation score.
2. Indirective optimization: on training stage, cross entropy is minimized by the process, which not necessarily agrees with standards like BLEU.

Contribution of this paper:

It constructs an actor (decision-making) & critic (evaluation) structure to optimize evaluation standards directly, and trains the two parts successively to improve its overall performance.

Human : A motorcycle carrying many wheels is parked.

XE : a motorcycle parked next to a yellow wall.

Ours : a yellow motorcycle parked in front of a street.

论文介绍：Adaptive Attention

Knowing When to Look - Adaptive Attention via A Visual Sentinel for Image Captioning

Problems of existing methods:

Not all words in a caption have corresponding visual features in image (non-visual words)

Examples:

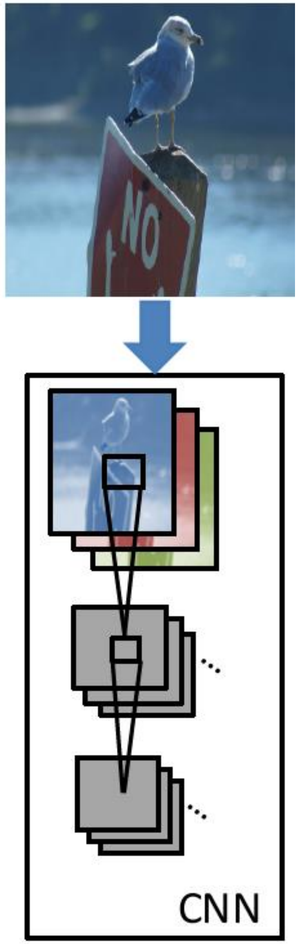
A white bird perched on top of a red stop sign.

“of” corresponds to no visual features.

“sign” after “stop” can be predicted without referring to image.

Consequence:

Gradients of non-visual words will mislead the model, resulting in difficulty in convergence



论文介绍：Adaptive Attention

Knowing When to Look - Adaptive Attention via A Visual Sentinel for Image Captioning

Contributions:

1. Put forward an adaptive encoder-decoder framework, which features the following:
   a) Encoder (where to look): Use spatial attention mechanism to determine which part of image the model should focus on at each time step.
   b) Decoder (when to look): Use visual sentinel mechanism to determine the model should rely on whether the visual features extracted from image, or the language model at each time step.
2. Achieved state-of-the-art performance on several benchmarks like MS-COCO and Flickr30k.

Our repetition:

	B-1	B-2	B-3	B-4	METEOR	CIDEr
Theoretical Score	0.742	0.580	0.439	0.332	0.266	1.085
Our Score	0.727	0.556	0.418	0.315	0.253	0.997

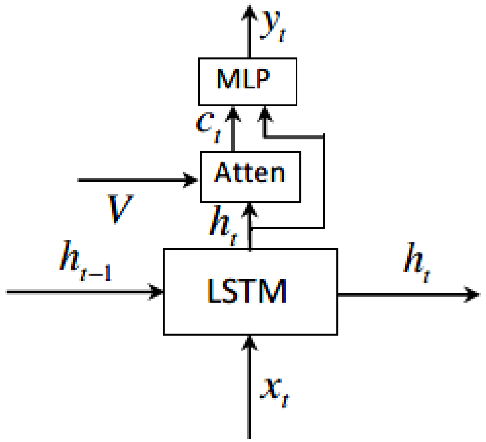
论文介绍：Adaptive Attention

Knowing When to Look - Adaptive Attention via A Visual Sentinel for Image Captioning

Spatial Attention:

$$c_t = g(V, h_t)$$

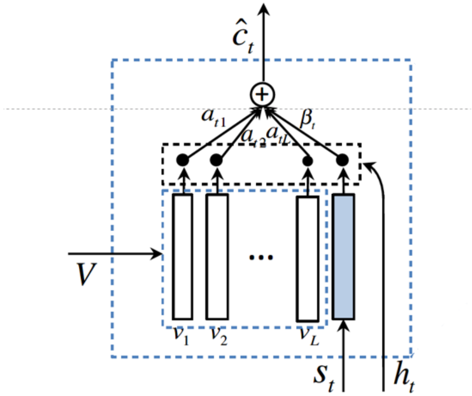
Where c_t is the visual content (which will be sent into decoder) at time t,
g is the Attention function,
V is the visual features of image (output of CNN),
and h_t is the hidden state of LSTM cell at time t.



Visual Sentinel:

It is made up of two concepts:

1. Visual Sentinel Vector s_t: Information related to language model
2. Visual sentinel Gate β_t: the degree of which the result relies on language model



论文介绍：Adaptive Attention

Knowing When to Look - Adaptive Attention via A Visual Sentinel for Image Captioning

Computing s_t:

$$s_t = \sigma(W_x x_t + W_h h_{t-1}) \odot \tanh(m_t)$$

Where σ is the sigmoid function,

W_x and W_h are two parameters to be learned,

x_t is the input of LSTM at time t,

h_{t-1} is the hidden state of LSTM at time $t-1$,

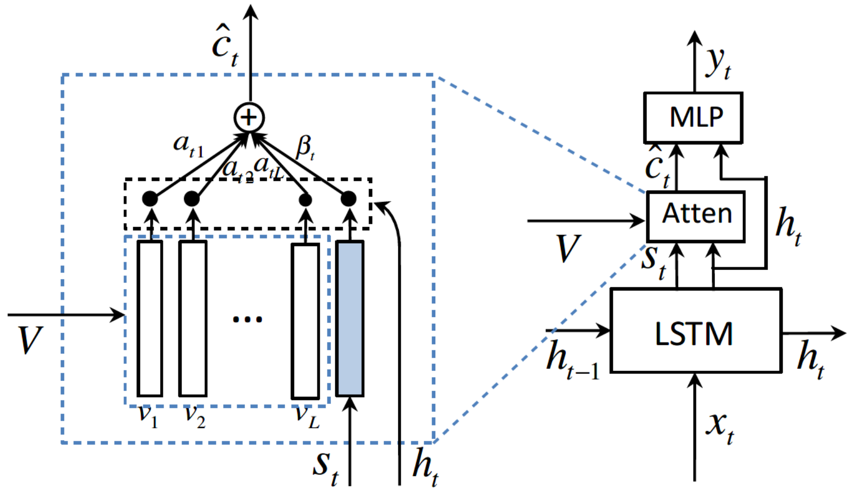
m_t is the cell state of LSTM at time t.

Computing β_t:

$$\beta_t = softmax([z_t; w_t^T \tanh(W_s s_t + W_g h_t)])[k+1]$$

Context Vector $\hat{c}_t$:

$$\hat{c}_t = \beta_t s_t + (1-\beta) c_t$$

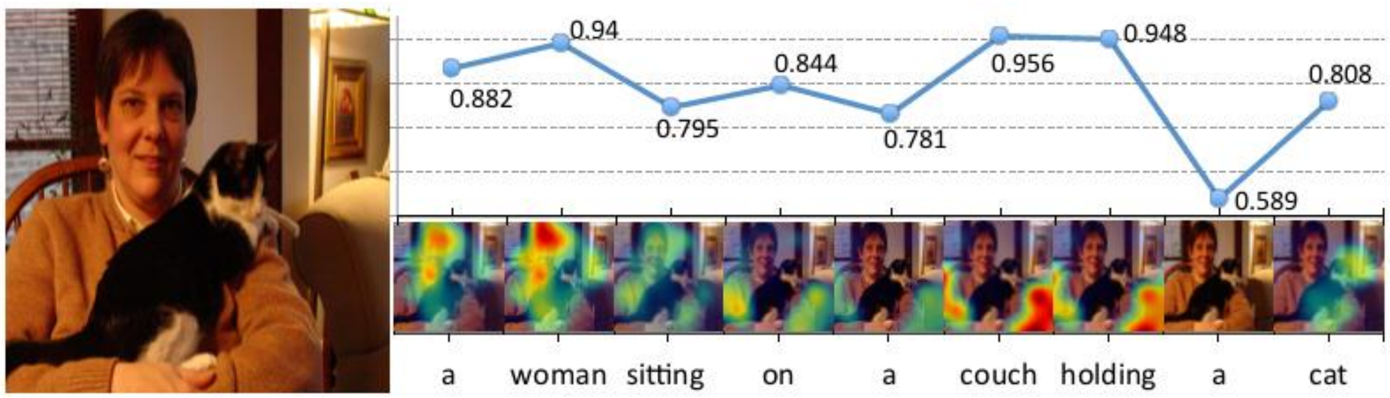


论文介绍：Adaptive Attention

Knowing When to Look - Adaptive Attention via A Visual Sentinel for Image Captioning

Visual Grounding:

In equation $\hat{c}_t = \beta_t s_t + (1-\beta)c_t$, the term $1-\beta$ is called visual grounding, stands for the extent to which we rely on image features (1 for pure image, 0 for pure language model).



Generation of words:

$\hat{c}_t$ and h_t are concatenated and sent into MLP (fully connected layers) to produce y_t, which is the word index predicted at time t.