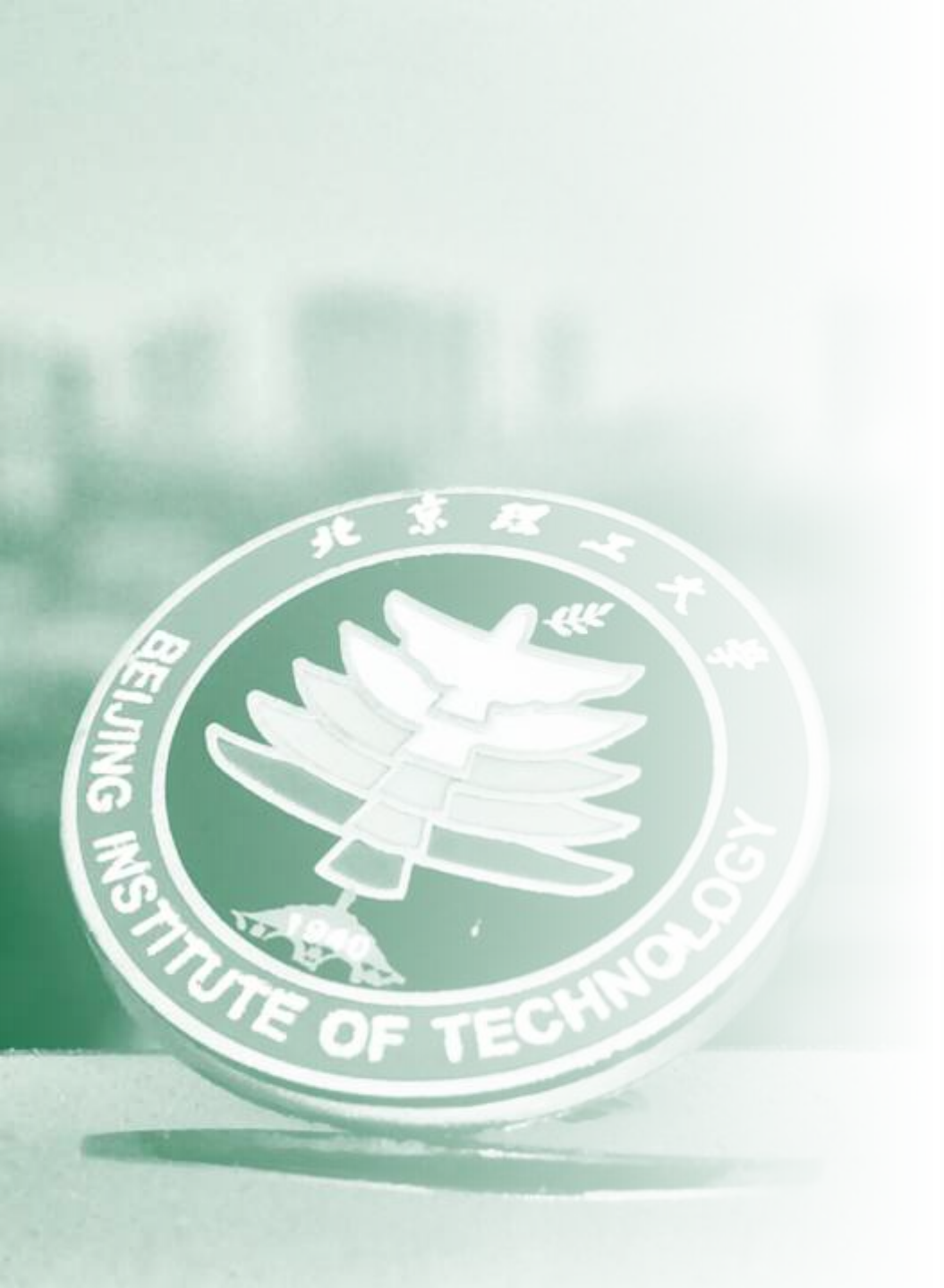
北京理工大學
BEIJING INSTITUTE OF TECHNOLOGY

看 图 说 话

Image Caption

组员：于敬楠、赵嘉旌、薛晓军、陈世艺、宋迎新

时间：2019/12/5



目录 | CONTENTS

1 看图说话的概念及应用

汇报人：于敬楠

看图说话的概念及应用

看图说话是计算机视觉与自然语言处理的结合。要求算法能准确地识别图像内容并将图像内容表达为通顺的、符合情境的句子。

计算机视觉

手工特征+浅层模型：如支持向量机、随机森林

深层模型：以CNN为代表

自然语言处理

统计模型：需要设计模型所需的人工特征与人工特征组合

深层模型：以低维稠密向量和RNN为代表

看图说话的概念及应用

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.

- 人类进行看图说话游戏包括三个过程：观察、想象、描述。
- 相应的在计算机领域，看图说话也可以总结为三个过程：识别，上下文语义分析，描述。

- **看图说话的应用**

可以用于准确地将图像表达为语言，能够帮助盲人理解周围环境防止发生危险情况。

为智能机器人引入看图说话，机器人在和人类聊天时可以根据摄像头捕捉到的图像产生更符合特定场景的聊天内容。

2 看图说话的传统方法

基于分类思想的方法

先对图像进行分割，过滤噪声和过分割部分，把每一个语义概念当作一个类别，对分割后的图像进行分类。

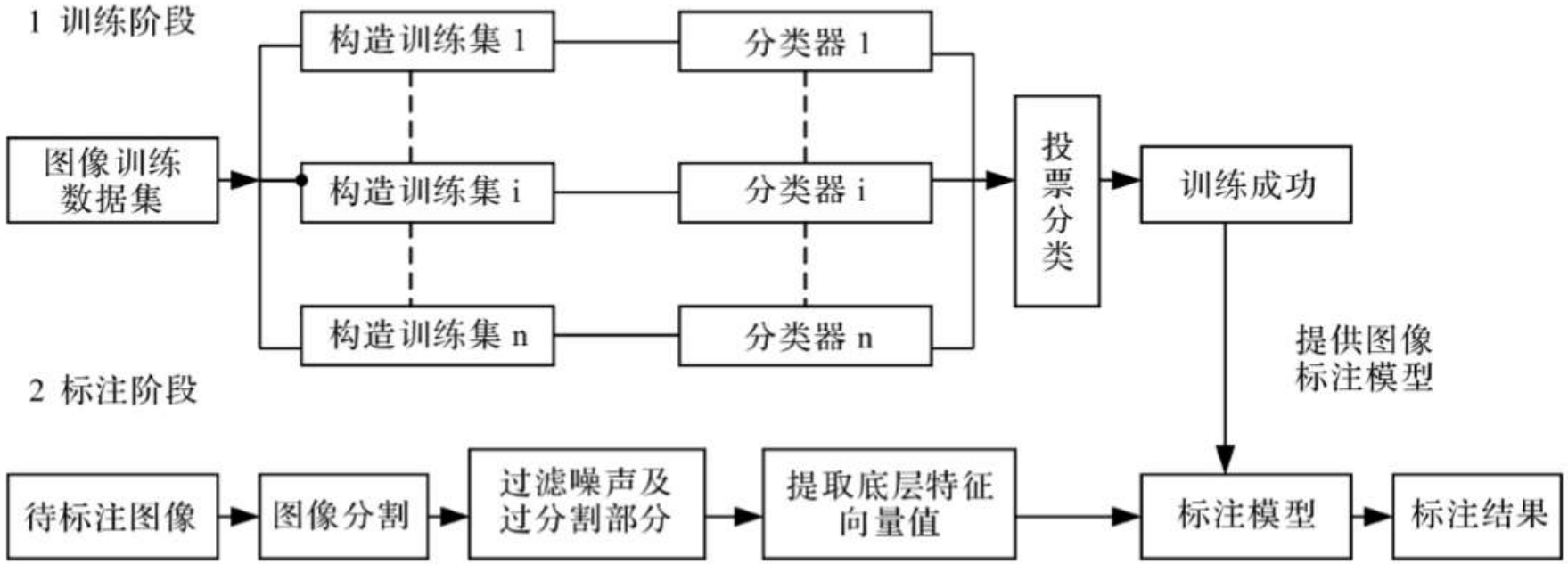
相关模型方法

建立图像与语义关键词之间的概率相关模型。通过关联模型，给待标注图像找到与其相关性概率最大的一组语义关键词来标注图像。

半监督模型方法

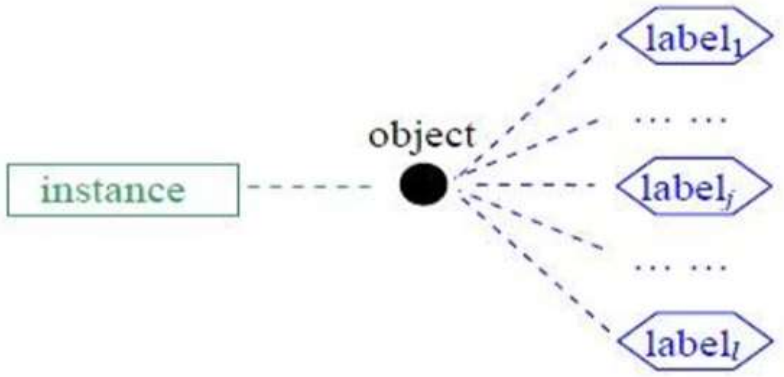
已标注的图像信息和未标注的图像信息都要参与到机器的学习过程中，在学习过程中可以利用的图像信息更多，对信息的了解更加清楚。

- 基于分类思想的方法

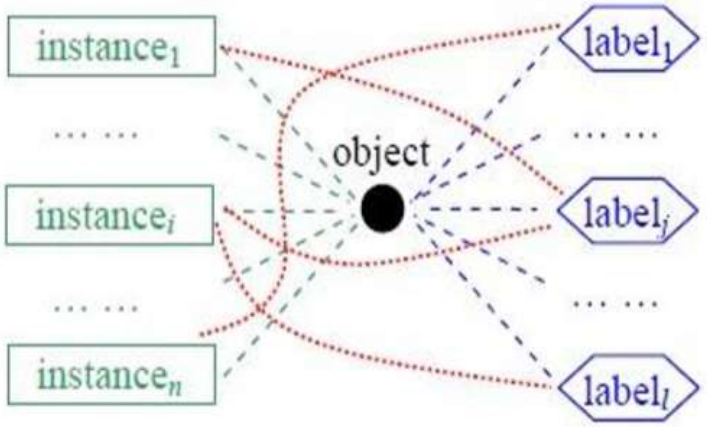


- 分类器训练过程会不断地通过反馈信息调整分类器，使得分类器达到某个精度。

- 多示例多标记方法



（1）单示例多标记对象



（2） 多示例多标记对象的示例与标记间的关系

多样性密度算法（经典）：特征空间中如果某点附近出现来自于不同正包中的示例越多，反包中的示例离得越远，则该点表征了给定关键词语义的概率就越大。

● 多分类标记方法

基于 SVM 的否定概率和法：先建立小规模训练集，库中每个图像标有单一的语义标签，再利用其底层特征，以 SVM 为子分类器，“否定概率和”法为合成方法构建基于成对耦合方式（PWC）的多类分类器，并对未标注的图像进行分类。

● 其他分类标记方法

以上这几种聚类方法，通常都是基于视觉特征；

◆ 从语义约束的聚类算法方面对图像区域进行聚类，然后进行图像标注。

◆ 利用深度学习的思想，设计实现了深度生成模型完成特征学习。

相关模型方法

1.对已经标注过的图像集进行分割，使其成为比较小的图像区域。

2.利用软约束的半监督图像聚类算法进行语义聚类，每个子类称为 blobs。

3.结合概率相关模型和排序学习算法，建立语义概念和blobs之间的概率关系。

4.针对未标注的图像，通过判断其区域所属的blob，利用概率关系进行自动标注。

● 半监督模型方法

图像间关系

两幅图像之间由视觉特征所决定的相关性

词间关系

两个词对于一幅图像的适合程度

图像到词的关系

通过图像产生语义关键词的可能性

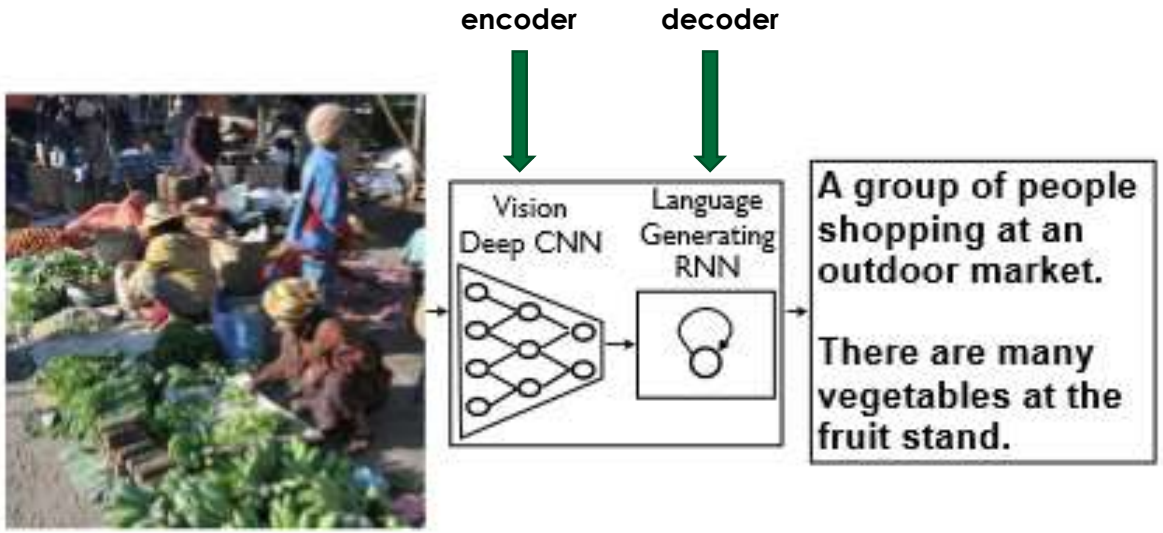
词到图像的关系

通过给定的语义关键词来取得产生图像的后验概率

图节点可以用每幅图像或者标注关键词表示，边可以用标注关键词之间或者图像之间的相似关系来表示，通过图学习算法来实现标注。

3 看图说话的深度学习方法

汇报人：赵嘉旌



- 由于CNN强大的图像特征提取能力，在看图说话任务中使用深度CNN网络作为图像特征**编码器**成为主流的做法

- 使用RNN作为**解码器**接收CNN提取出的图像特征，其中RNN也可替换为LSTM或GRU等

[Vinyals et al. Show and tell: A neural image caption generator. CVPR 2015: 3156-3164]

看图说话任务主要有3种生成文字的方法：

- **基于模板填充的方法**：在人为规定的一系列句法模板中留出部分空白，然后再基于提取出的图像特征获得目标，动作及属性，将它们填充进入空白，从而获得对某一图像的描述。这种方法保证了语义和句法正确性。然而，完全确定的模板无法产生多样性的输出，故现在这种方法使用较少。

- **基于检索的方法**：基于检索的方法指的是将大量的图片描述存于一个集合，再通过比较待描述图片和训练集中图片的相似性获得一个待选句集，再从中选取该图片的描述。这种方法保证了句法正确性，然而无法保证语义正确性，也无法对新图片进行准确的描述。

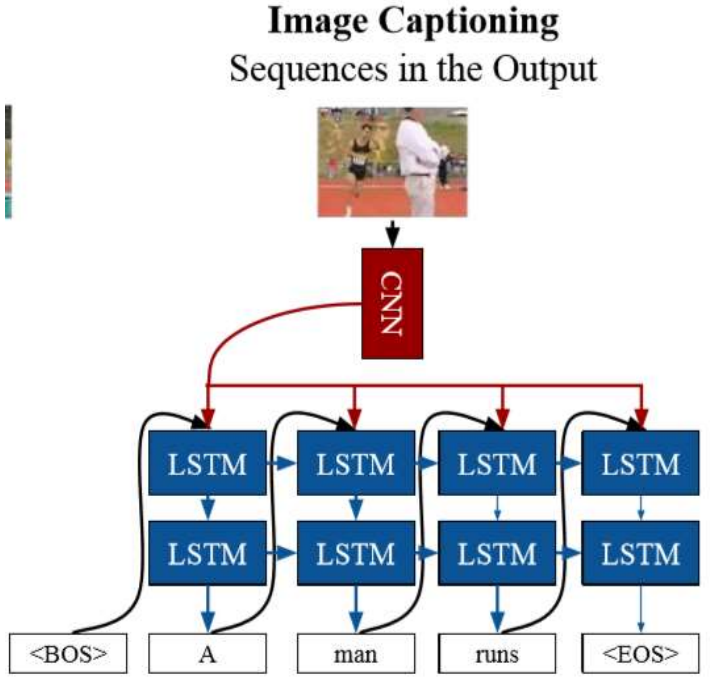
- **基于生成的方法**：先将图像信息编码后作为输入送入语言模型，再利用语言模型产生全新的描述。绝大部分基于深度学习的看图说话技术都使用基于生成的方法，也是目前效果最好的普遍应用方法。它在句法正确性，语义准确性和对新图片的泛化能力上都达到了较好的效果。

北京理工大学 BEIJING INSTITUTE OF TECHNOLOGY

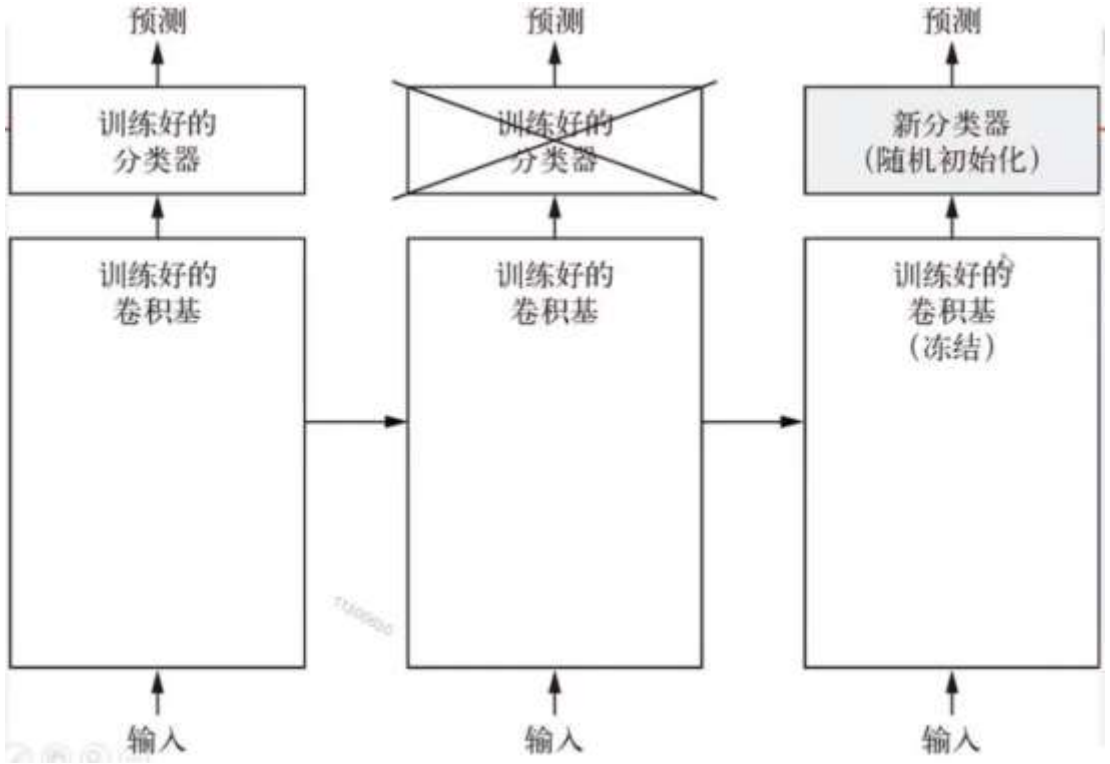
编码器：VGG Net　　输入图像；输出图像特征

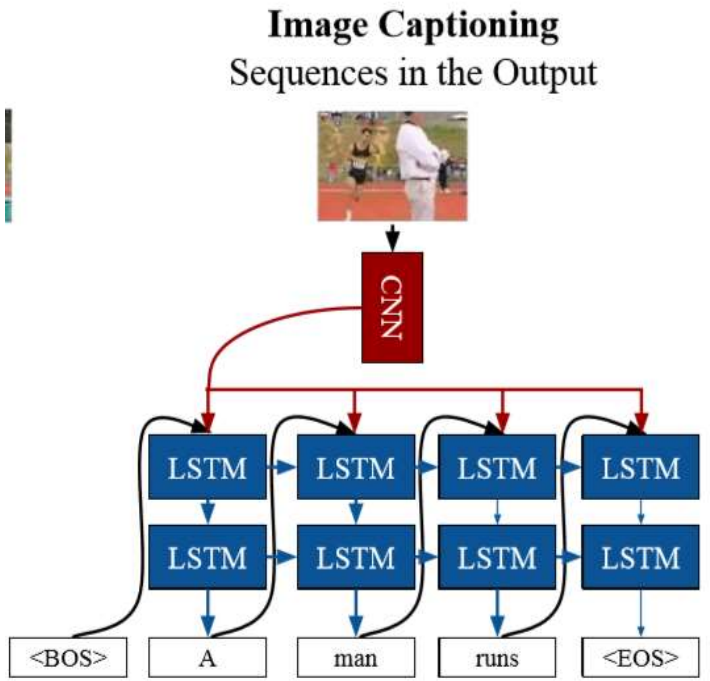
ConvNet Configuration					
A	A-LRN	B	C	D	E
11 weight layers	11 weight layers	13 weight layers	16 weight layers	16 weight layers	19 weight layers
input (224 × 224 RGB image)					
conv3-64	conv3-64 **LRN**	conv3-64 **conv3-64**	conv3-64 conv3-64	conv3-64 conv3-64	conv3-64 conv3-64
maxpool					
conv3-128	conv3-128	conv3-128 **conv3-128**	conv3-128 conv3-128	conv3-128 conv3-128	conv3-128 conv3-128
maxpool					
conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256 **conv1-256**	conv3-256 conv3-256 **conv3-256**	conv3-256 conv3-256 conv3-256 **conv3-256**
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 **conv1-512**	conv3-512 conv3-512 **conv3-512**	conv3-512 conv3-512 conv3-512 **conv3-512**
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 **conv1-512**	conv3-512 conv3-512 **conv3-512**	conv3-512 conv3-512 conv3-512 **conv3-512**
maxpool					
FC-4096					
FC-4096					
FC-1000					
soft-max					

卷积层、池化层、全连接层、最后softmax层的简单堆叠



[Simonyan et al. Very Deep Convolutional Networks for Large-Scale Image Recognition. ICLR 2015]



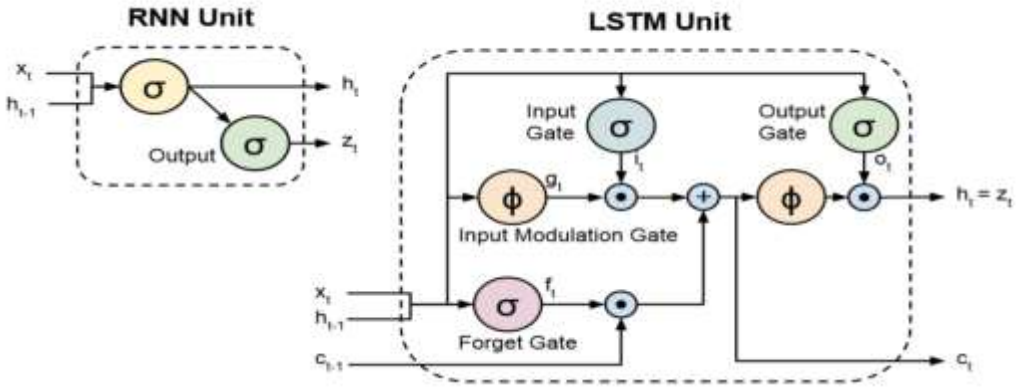


使用预训练网络(也叫迁移学习)就是，以VGG为例，使用它训练好的卷积基来提取特征，在该特征的基础上处理下游任务

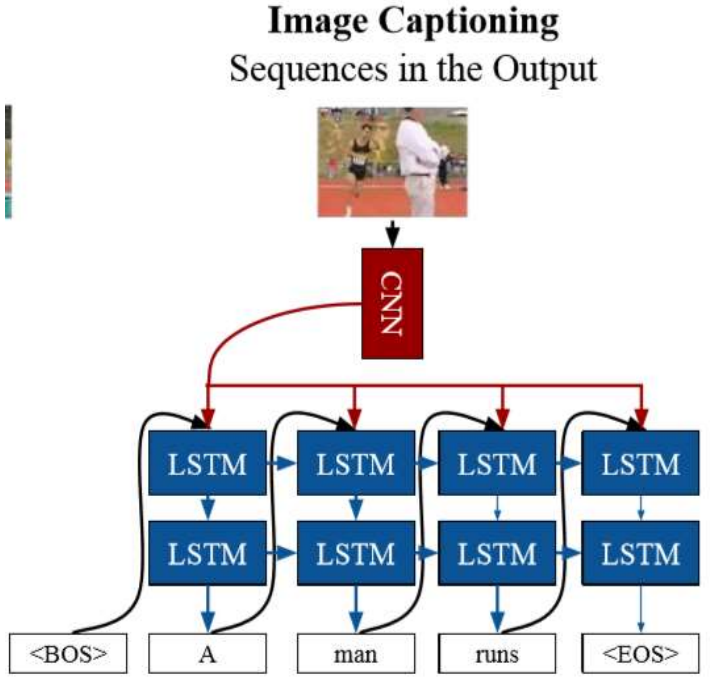
```
conv_base=tf.keras.applications.VGG16(weights='imagenet',include_top=False)
conv_base.trainable=False #将VGG里的权重设置为不可训练
```

[Donahue et al. Long-term recurrent convolutional networks for visual recognition and description. CVPR 2015: 2625-2634]

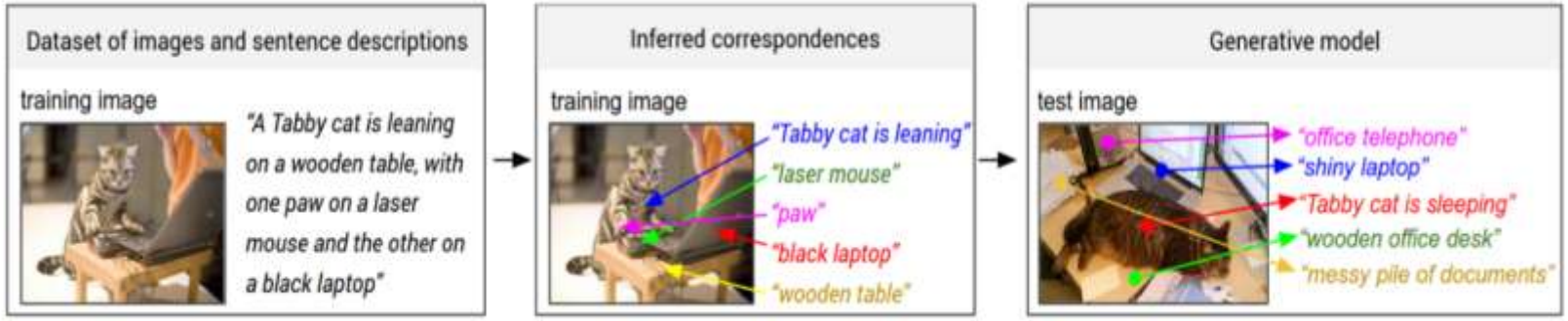
解码器：LSTM 输入图像特征和前一个词；输出文本



$$
\begin{aligned}
i_t &= \sigma(W_{xi}x_t + W_{hi}h_{t-1} + b_i) \\
f_t &= \sigma(W_{xf}x_t + W_{hf}h_{t-1} + b_f) \\
o_t &= \sigma(W_{xo}x_t + W_{ho}h_{t-1} + b_o) \\
g_t &= \tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c) \\
c_t &= f_t \odot c_{t-1} + i_t \odot g_t \\
h_t &= o_t \odot \tanh(c_t) \\
\hat{y}_t &= W_z z_t + b_z \\
P(y_t = c) &= \text{softmax}(\hat{y}_t) = \frac{\exp(\hat{y}_{t,c})}{\sum_{c' \in C} \exp(\hat{y}_{t,c'})}
\end{aligned}
$$



[Donahue et al. Long-term recurrent convolutional networks for visual recognition and description. CVPR 2015: 2625-2634]



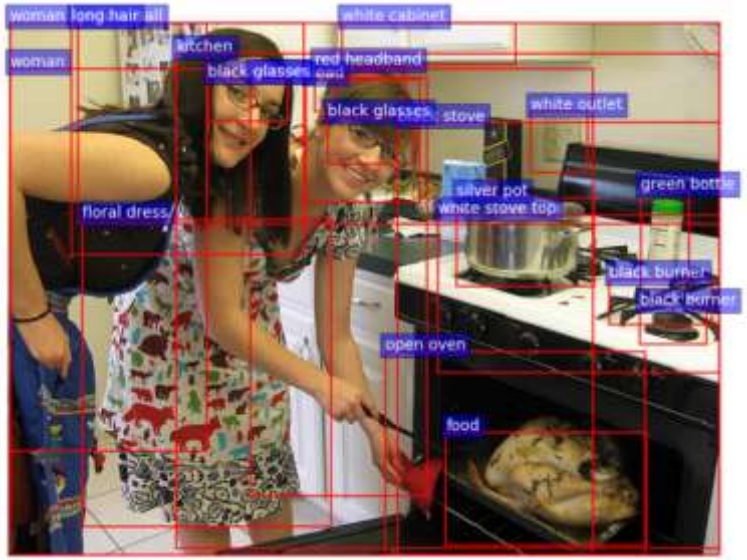
- 训练时输入的是图片和人工标注的图片描述
- 数据经过网络运算后，将最后模型的输出结果与训练数据的真实标签计算交叉熵损失，并通过反向传播算法来不断调整网络权重，最终学习得到一个较优的模型

[Karpathy et al. Deep visual-semantic alignments for generating image descriptions. CVPR 2015: 3128-3137]

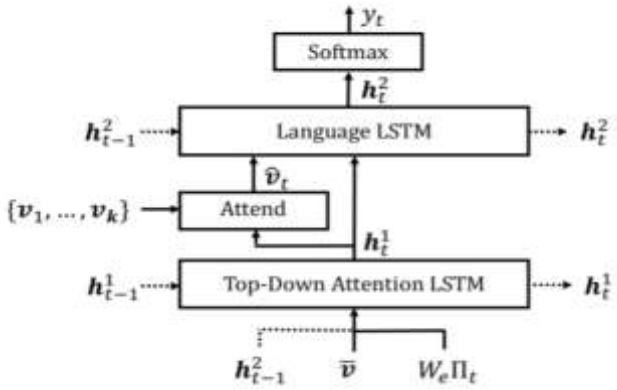
北京理工大学 BEIJING INSTITUTE OF TECHNOLOGY

在编码器端的创新：

- 使用bottom-up机制进行目标检测，提取图像中的兴趣区域，获得对应检测目标的边界框和标签，并且通过设定的阈值允许兴趣框的重叠，这样可以更有效的理解图像内容。



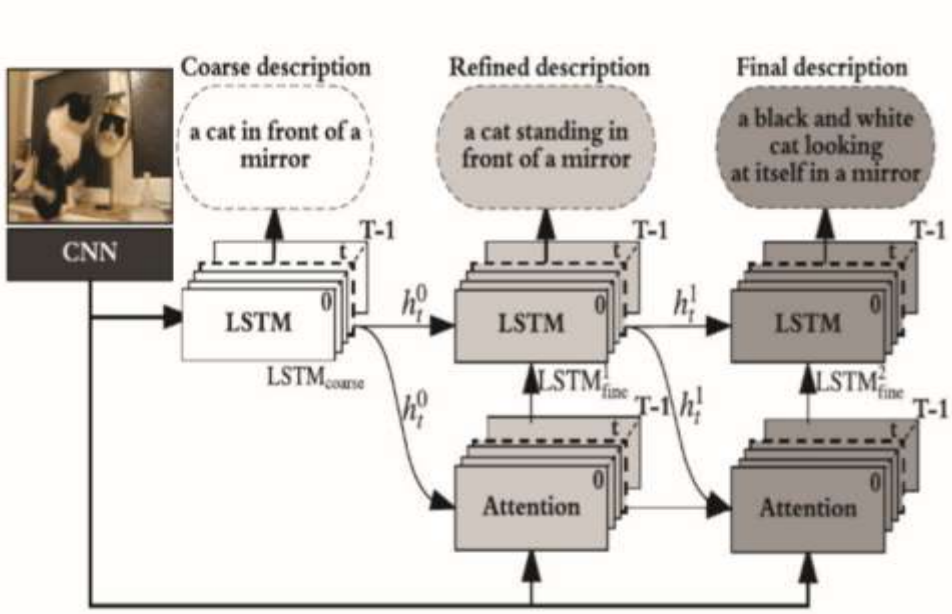
- 使用top-down机制对输入的图片特征根据输出的语言进行实时的注意力调整。这种attention机制指对于图像中明显和重要的目标进行更多关注



[Anderson et al. Bottom-Up and Top-Down Attention for Image Captioning and Visual Question Answering. CVPR 2018: 6077-6086]

在解码器端的创新：

- 提出了逐步求精的stack caption的思想
- 使用一个粗粒度的解码器和多个细粒度的解码器:粗粒度解码器接受图像特征作为输入，输出描述结果。接下来在每一个阶段都有一个细粒度的解码器进行更精细的解码，其输入来自于上一阶段解码器的输出结果和图像特征
- 使用attention机制，从而使得细粒度解码器在每一阶段对粗粒度产生结果的不同方面进行扩展，最终获得较详细的结果



[Gu et al. Stack-Captioning: Coarse-to-Fine Learning for Image Captioning. AAAI 2018: 6837-6844]

未来展望

- 编码器端/解码器端结构的优化和创新
- 在看图说话中通过知识图谱引入外部知识来提高语义丰富度
- 在看图说话中引入生成对抗网络结构（GAN）
- 跳出Encoder-Decoder的主流架构

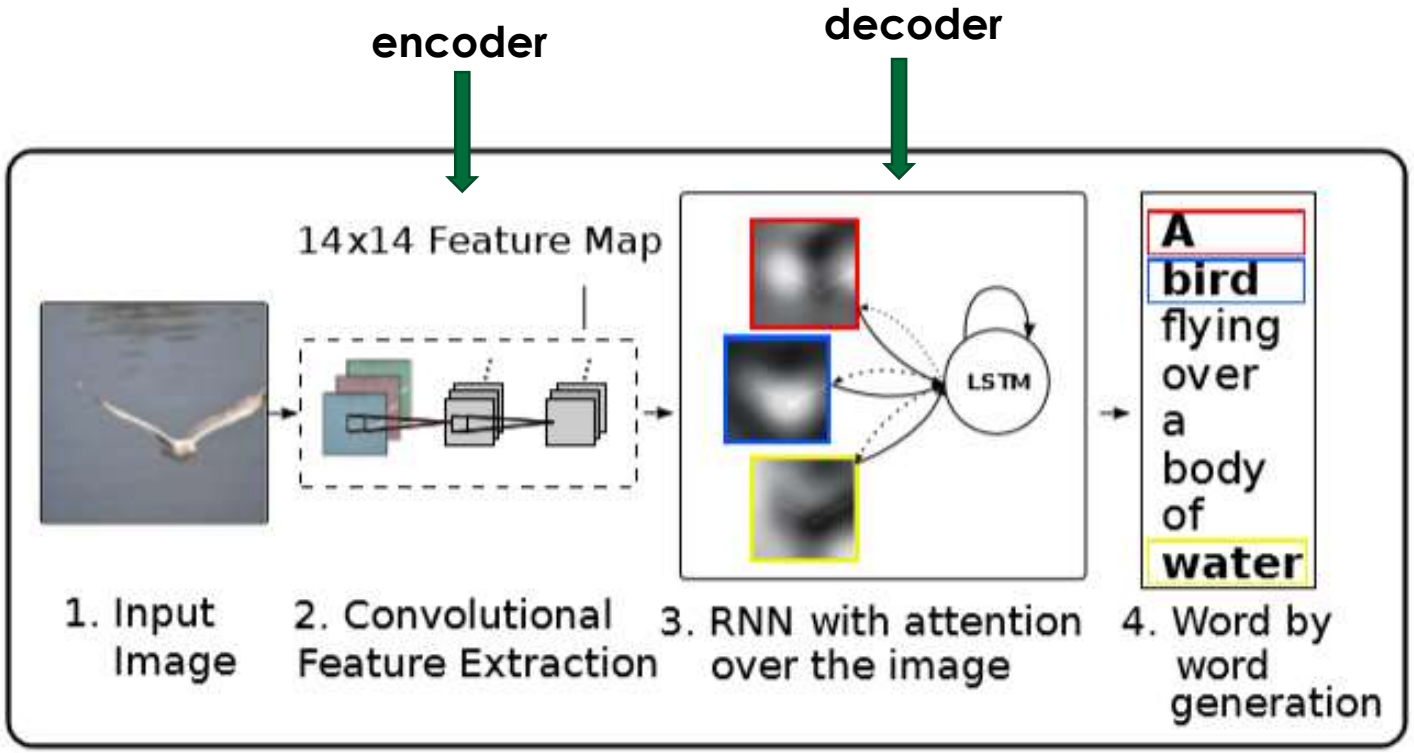
……

4 模型实现及实验

汇报人：薛晓军

(1)CNN特征提取

(2)带有Attention机制的RNN解码特征



[Vinyals et al. Show and tell: A neural image caption generator. CVPR 2015: 3156-3164]

注意力机制

Attention:

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

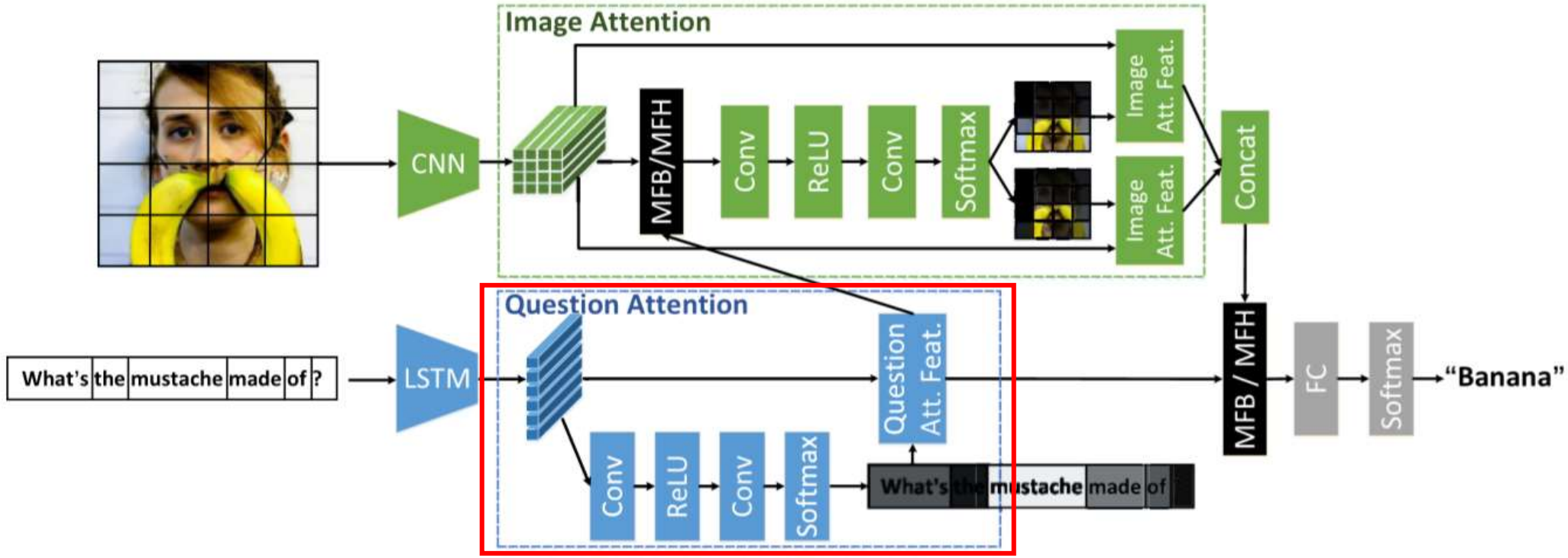
$$\alpha_{ti} = \frac{\exp(e_{ti})}{\sum_{k=1}^{L} \exp(e_{tk})}$$

Hard Attention:

$$\hat{\mathbf{z}}_t = \sum_i s_{t,i}\mathbf{a}_i$$

Soft Attention:

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



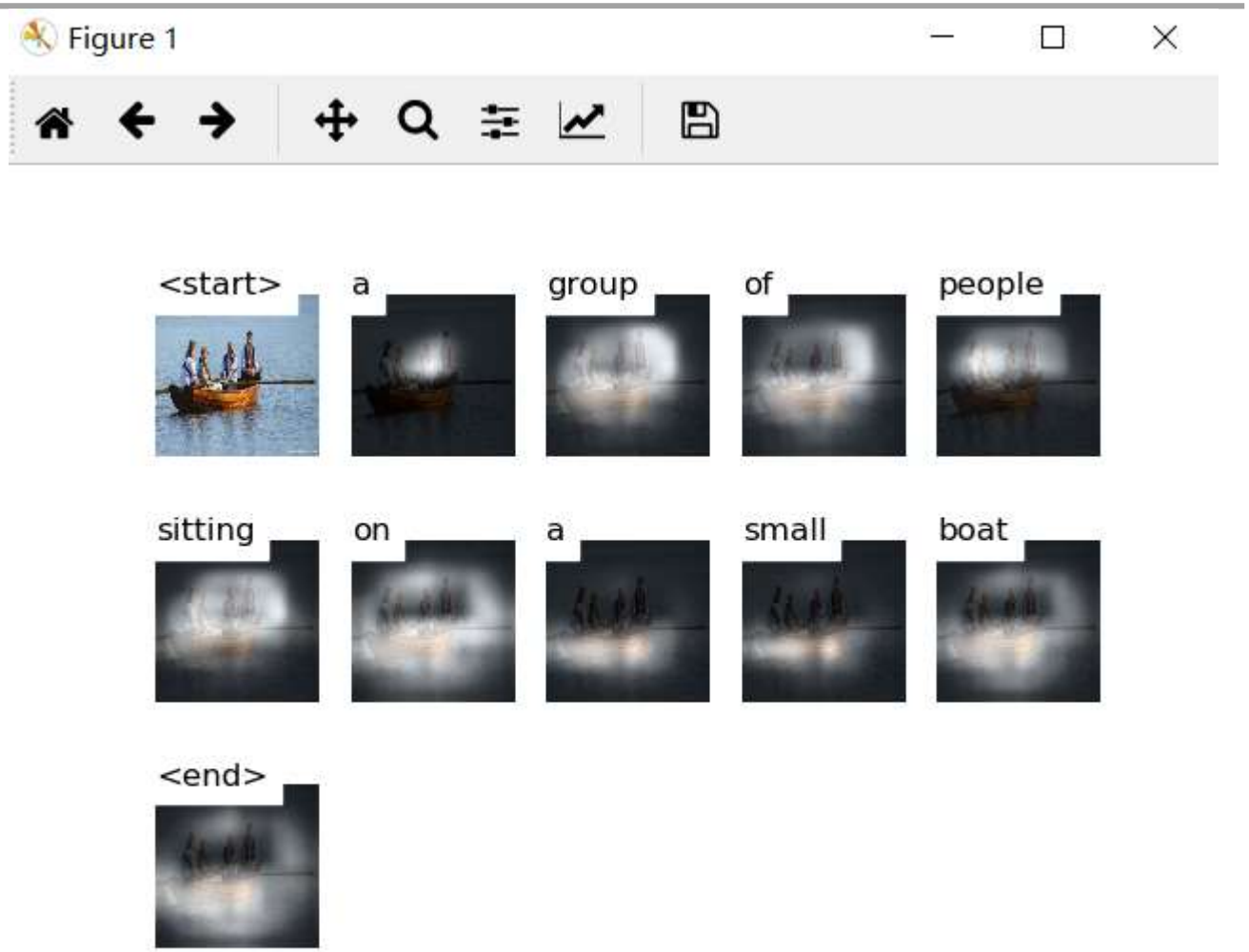
Encoder

```
def forward(self, images):
    """
    Forward propagation.
    :param images: images, a tens
    :return: encoded images
    """
    out = self.resnet(images)  #
    out = self.adaptive_pool(out)
    out = out.permute(0, 2, 3, 1)
    return out
```

Decoder

```
for t in range(max(decode_lengths)):
    batch_size_t = sum([l > t for l in decode_lengths])
    attention_weighted_encoding, alpha = self.attention(encoder_out[:batch_size_t],
                                                        h[:batch_size_t])
    gate = self.sigmoid(self.f_beta(h[:batch_size_t]))  # gating scalar, (batch_size_t, e
    attention_weighted_encoding = gate * attention_weighted_encoding
    h, c = self.decode_step(
        torch.cat([embeddings[:batch_size_t, t, :], attention_weighted_encoding], dim=1),
        (h[:batch_size_t], c[:batch_size_t]))  # (batch_size_t, decoder_dim)
    preds = self.fc(self.dropout(h))  # (batch_size_t, vocab_size)
    predictions[:batch_size_t, t, :] = preds
    alphas[:batch_size_t, t, :] = alpha
```

看图说话示例



看图说话示例


toopen shifuyy

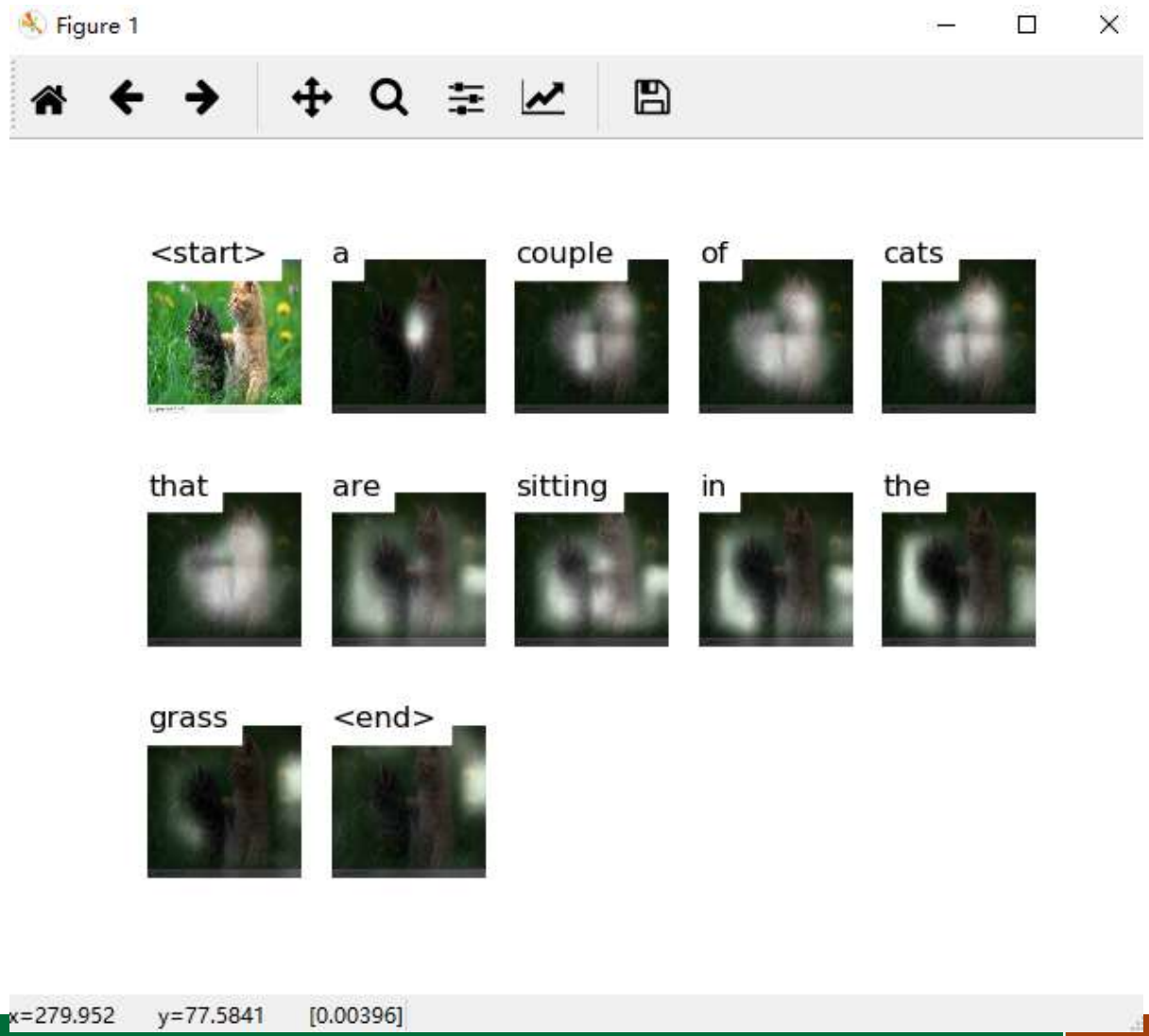


Figure 1
<start>
a
couple
of
cats
that
are
sitting
in
the
grass
<end>
x=279.952 y=77.5841 [0.00396]

看图说话示例

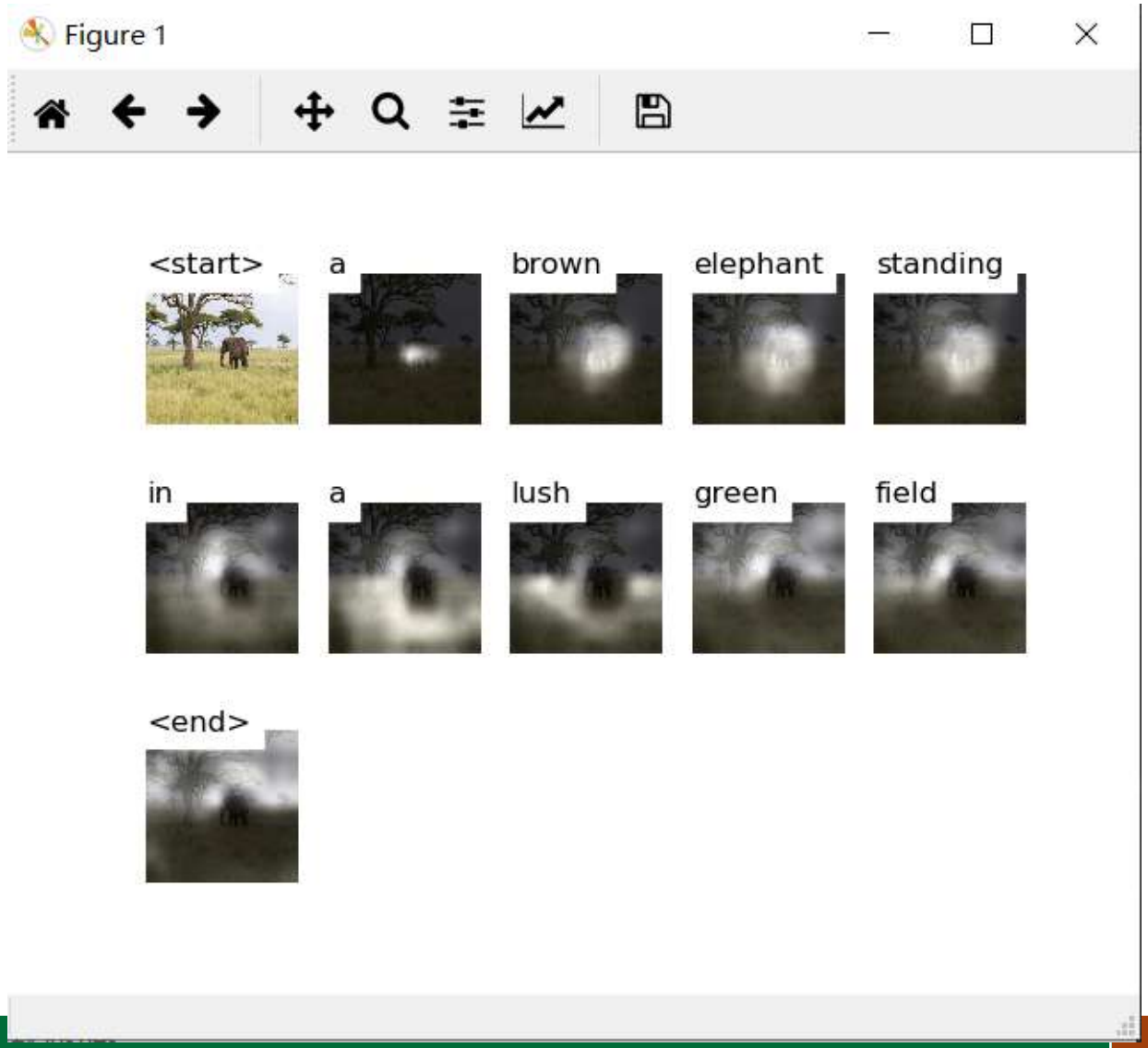


Figure 1
<start>
a
brown
elephant
standing
in
a
lush
green
field
<end>

数据集

Flickr8k：共8000张图片，每张图片有5个相关的句子。

Flickr30k：共30000张图片，每张图片有5个相关的句子。

Microsoft COCO：共82,783张图片，每张图片选取5个相关的句子。

		BLEU				
Dataset	Model	BLEU-1	BLEU-2	BLEU-3	BLEU-4	METEOR
Flickr8k	Google NIC(Vinyals et al., 2014)†Σ	63	41	27	—	—
	Log Bilinear (Kiros et al., 2014a)°	65.6	42.4	27.7	17.7	17.31
	Soft-Attention	**67**	44.8	29.9	19.5	18.93
	Hard-Attention	**67**	**45.7**	**31.4**	**21.3**	**20.30**
Flickr30k	Google NIC†°Σ	66.3	42.3	27.7	18.3	—
	Log Bilinear	60.0	38	25.4	17.1	16.88
	Soft-Attention	66.7	43.4	28.8	19.1	**18.49**
	Hard-Attention	**66.9**	**43.9**	**29.6**	**19.9**	18.46
COCO	CMU/MS Research (Chen & Zitnick, 2014)a	—	—	—	—	20.41
	MS Research (Fang et al., 2014)†a	—	—	—	—	20.71
	BRNN (Karpathy & Li, 2014)°	64.2	45.1	30.4	20.3	—
	Google NIC†°Σ	66.6	46.1	32.9	24.6	—
	Log Bilinear°	70.8	48.9	34.4	24.3	20.03
	Soft-Attention	70.7	49.2	34.4	24.3	**23.90**
	Hard-Attention	**71.8**	**50.4**	**35.7**	**25.0**	23.04

(1) 使用效果更好的注意力机制

Attention:

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

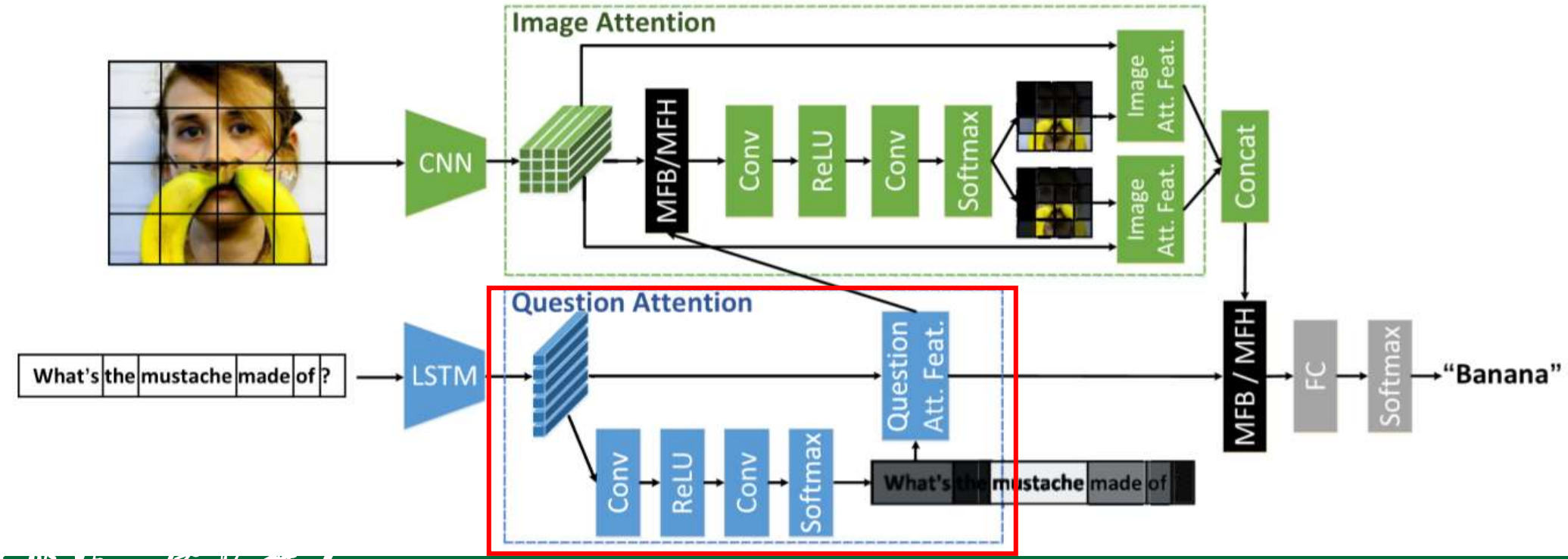
$$\alpha_{ti} = \frac{\exp(e_{ti})}{\sum_{k=1}^{L} \exp(e_{tk})}$$

Hard Attention:

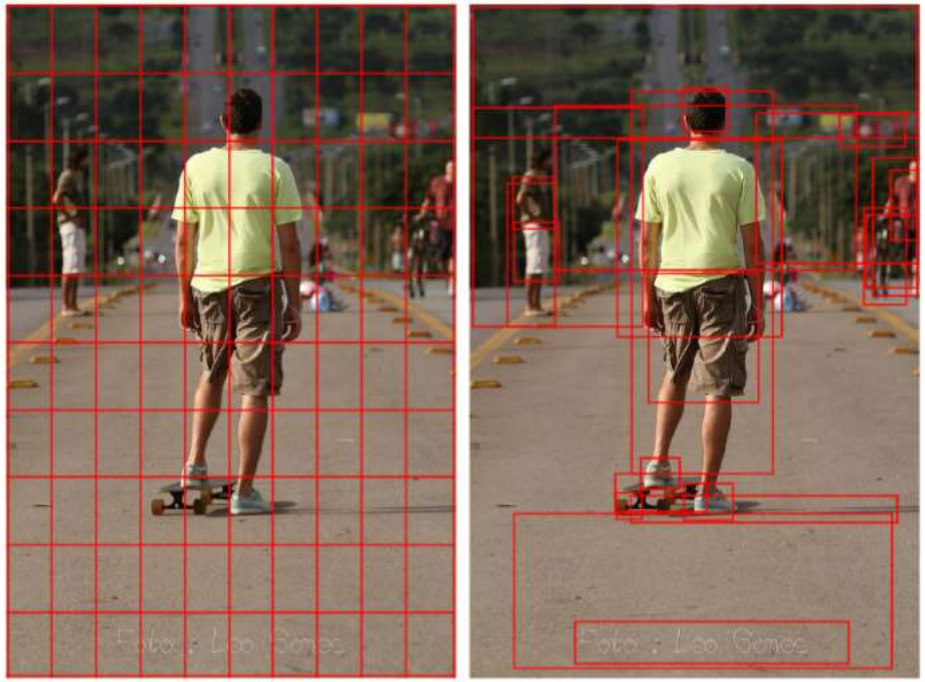
$$\hat{\mathbf{z}}_t = \sum_i s_{t,i} \mathbf{a}_i$$

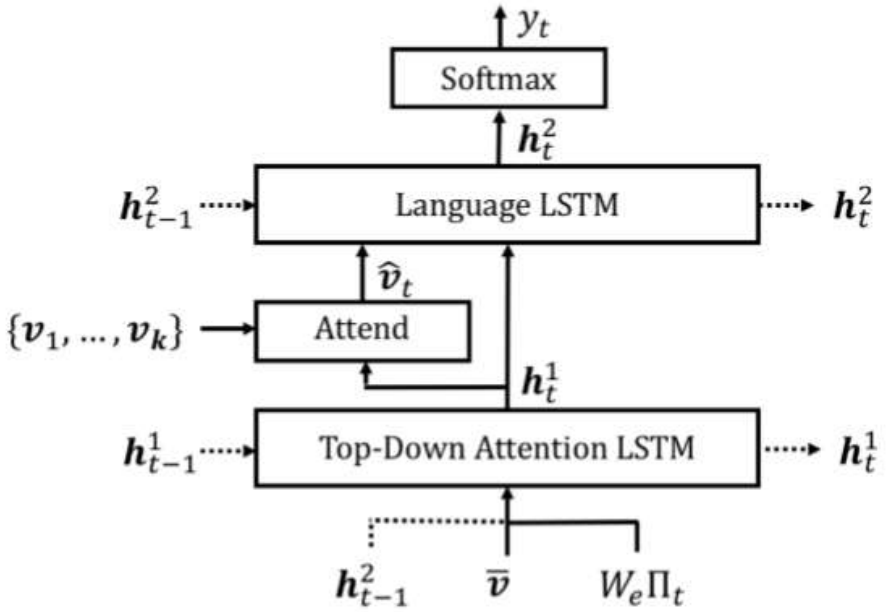
Soft Attention:

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



(2)在图像特征提取方面改进
(3)使用效果更好的decoder模型





北京理工大学
BEIJING INSTITUTE OF TECHNOLOGY

谢谢观看
敬请老师批评指正