文本自动生成

玖强

Outline

- 文本生成介绍
- 文本到文本生成
- 图像到文本生成
- 展望

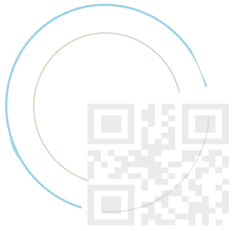
文本生成介绍

文本生成介绍

- 目标：期待未来有一天计算机能够像人类一样会写作，能够撰写出高质量的自然语言文本
- 应用：智能问答与对话、机器翻译、更加智能和自然的人机交互、新闻的自动撰写与发布、学术论文撰写
- 按照输入不同进行分类：
  - **Text-to-text generation**
  - **Image-to-text generation**
  - **Video-to-text generation**
  - **And so on ….**

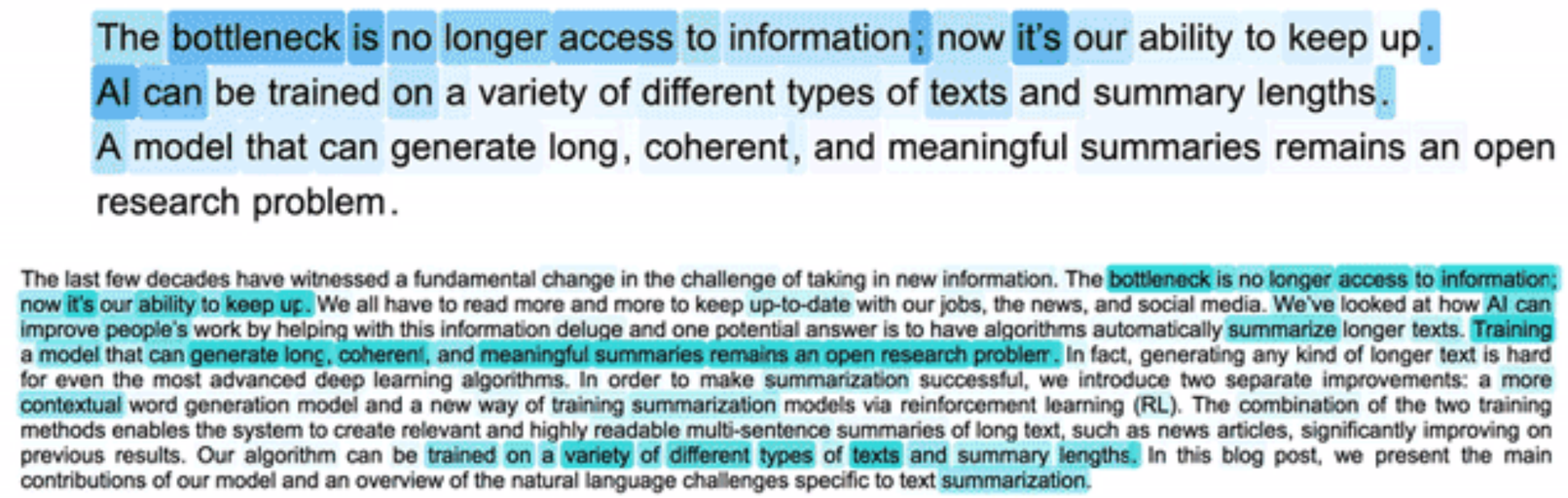
文本到文本生成

- 对给定文本进行变换和处理从而获得新文本的技术，包括有：
  - 文本摘要 (Document Summarization)
  - 句子压缩 (Sentence Compression)
  - 句子融合 (Sentence Fusion)
  - 文本复述(Paraphrase Generation)
- 哪里可以看到最新的研究成果？
  - ACL、EMNLP、NAACL、COLING、AAAI、IJCAI、SIGIR、INLG、ENLG



文本摘要

- 文本摘要技术通过自动分析给定的文档或文档集，摘取其中的要点信息，最终输出一篇短小的摘要，该摘要中的句子可直接出自原文，也可重新撰写所得。
- 目的：通过对原文本进行压缩、提炼，为用户提供简明扼要的内容描述

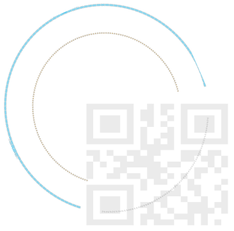


由新闻文章生成多语句摘要。对于每个生成的词，模型重点关注输入的特定词和之前生成的输出。

文本摘要

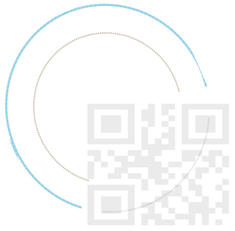
根据不同的划分标准，文档摘要可以主要分为以下几种不同类型：

- 根据处理的文档数量，摘要可以分为单文档摘要和多文档摘要：
  - 单文档摘要只对单篇文档生成摘要；
  - 而多文档摘要则对一个文档集生成摘要。
- 根据是否提供上下文环境，摘要可以分为主题或查询无关的摘要和主题或查询相关的摘要
  - 主题或查询相关的摘要在给定的某个主题或查询下，能够诠释该主题或回答该查询
  - 而主题或查询无关的摘要则指不给定主题和查询的情况下对文档或文档集生成的摘要
- 据摘要所采用的方法，摘要可以分为生成式和抽取式
  - 生成式方法通常需要利用自然语言理解技术对文本进行语法、语义分析，对信息进行融合，利用自然语言生成技术生成新的摘要句子
  - 而抽取式方法则相对比较简单，通常利用不同方法对文档结构单元(句子、段落等)进行评价，对每个结构单元赋予一定权重，然后选择最重要的结构单元组成摘要。抽取式方法应用较为广泛，通常采用的结构单元为句子
- 根据摘要的应用类型，摘要可以分为标题摘要、传记摘要、电影摘要等



文本摘要

- 最早的应用需求来自于图书馆
- 随着信息检索技术的发展，文档自动摘要在信息检索系统中的重要性越来越大，逐渐成为研究热点之一
- 文档自动摘要技术的第一篇论文来自**Luhn (1958)**
- 际上文档自动摘要方面比较著名的几个系统包括:
  - **ISI的NeATS系统**
  - 哥伦比亚大学的**NewsBlaster**系统
  - 密歇根大学的**NewsInEssence**系统
- 常用方法
  - 基于句子抽取的方法，也就是以原文中的句子作为单位进行评估与抽取。这类方法的好处是易于实现，能保证摘要句子具有良好的可读性
  - 压缩式文本摘要方法，考虑对句子进行压缩，以在较短长度限制下让摘要涵盖更多的内容
  - 生成式摘要，也就是通过对原文档进行语义理解，将原文档表示为深层语义形式（例如深层语义图），然后分析获得摘要的深层语义表示（例如深层语义子图），最后由摘要的深层语义表示生成摘要



文本压缩与融合

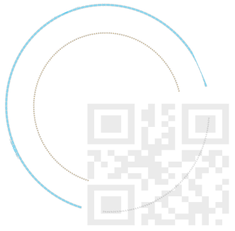
一般用于文本摘要系统中，用于生成信息更加紧凑的摘要，获得更好的摘要效果

- 句子压缩技术基于一个长句子生成一个短句子，要求该短句保留长句中的重要信息，也就是重要信息基本不损失，同时要求该短句是通顺的

But they are still continuing to search the area to try and see if there were, in fact, any further shooting incidents 【原始句子】

They are continuing to search the area to see if there were any further incidents【压缩后】

- 压缩的常用方法：
  - 从句子中删除词语
  - 对句子中的词语进行替换、重排序或插入



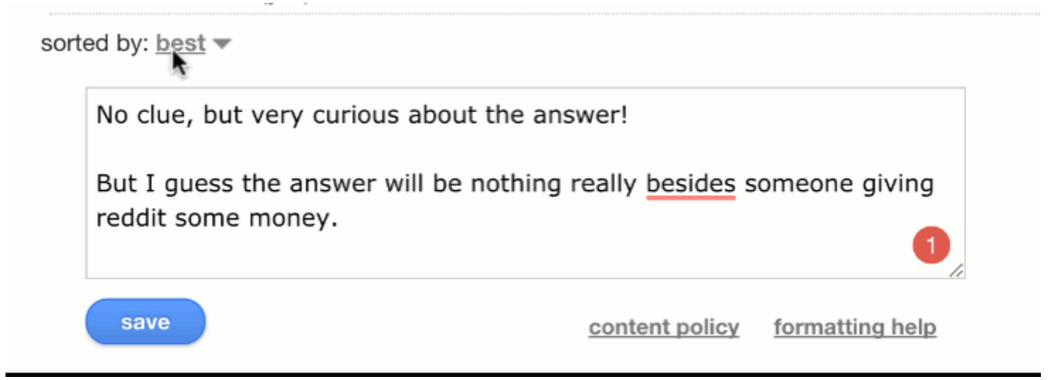
文本压缩与融合

- 句子融合技术则是合并两个或多个包含重叠内容的相关句子得到一个句子
  - 句子融合常见方法
    - 句子融合只保留多个句子中的共同信息，而过滤无关的细节信息（类似于集合运算中的取交集运算）
    - 子融合则只过滤掉多个句子之间的重复内容（类似于集合运算中的取并集运算）

- In 2003, his nomination to the U.S. Court of Appeals for the District of Columbia sailed through the Senate Judiciary Committee on a 16-3 vote. 【句子1】
- He was nominated to the U.S. Court of Appeals for the District of Columbia Circuit in 1992 by the first President Bush and again by the president in 2001.【句子2】
- He was nominated to the U.S. Court of Appeals for the District of Columbia Circuit. 【合并后的句子(取交集)】
- In 2003, his nomination by the first President Bush, and again by the second Bush in 2001 to the U.S. Court of Appeals for the District of Columbia sailed through the Senate Judiciary Committee on a 16-3 vote. 【合并后的句子(取并集)】

文本复述

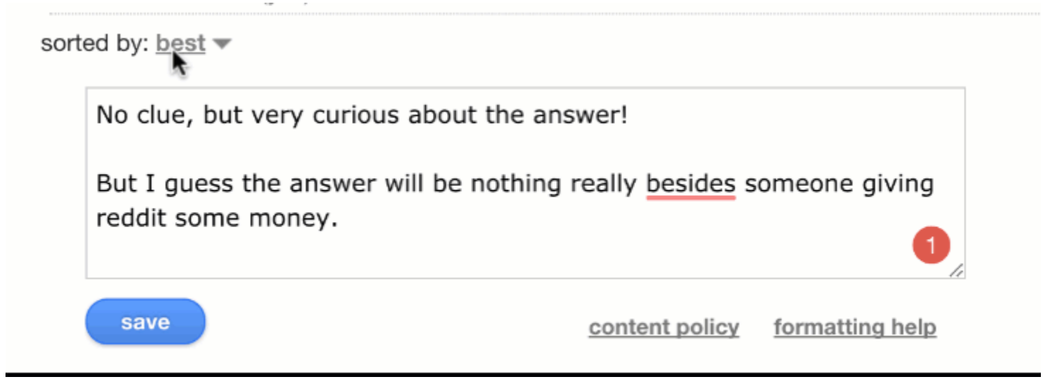
- 文本复述生成技术通过对给定文本进行改写，生成全新的复述文本，一般要求输出文本与输入文本在表达上有所不同，但所表达的意思基本一样
- 应用：
  - 在机器翻译系统中可利用文本复述技术对复杂输入文本进行简化从而方便翻译
  - 信息检索系统中可利用文本复述技术对用户查询进行改写
  - 在儿童教学系统中可利用文本复述技术将难以理解的文本简化为儿童容易理解的文本



- 通过复述生成技术得到的输出文本与原文本相比，可以只是一两个词发生了改变（如例1），也可以是整段文本面目全非（如例2）
  - 例1：all the members of –> all members of
  - 例2：He said there will be major cuts in the salaries of high-level civil servants.=>He claimed to implement huge salary cut to senior civil servants
- 简单的文本复述生成可以通过同义词替换来实现，也可以通过人工或自动构建的复述规则来实现
  - 输入：He booked a single room in Beijing yesterday.
  - 输出：Yesterday, he booked a single room in Beijing.

文本复述

- 文本复述生成技术通过对给定文本进行改写，生成全新的复述文本，一般要求输出文本与输入文本在表达上有所不同，但所表达的意思基本一样
- 应用：
  - 在机器翻译系统中可利用文本复述技术对复杂输入文本进行简化从而方便翻译
  - 信息检索系统中可利用文本复述技术对用户查询进行改写
  - 在儿童教学系统中可利用文本复述技术将难以理解的文本简化为儿童容易理解的文本



- 通过复述生成技术得到的输出文本与原文本相比，可以只是一两个词发生了改变（如例1），也可以是整段文本面目全非（如例2）
  - 例1：all the members of –> all members of
  - 例2：He said there will be major cuts in the salaries of high-level civil servants.=>He claimed to implement huge salary cut to senior civil servants
- 简单的文本复述生成可以通过同义词替换来实现，也可以通过人工或自动构建的复述规则来实现
  - 输入：He booked a single room in Beijing yesterday.
  - 输出：Yesterday, he booked a single room in Beijing.

文本复述

- 实现复杂的文本复述生成，研究人员提出了：
  - 基于自然语言生成的方法
    - 基于自然语言生成的方法模拟人类的思维方式，首先对输入句子进行语义理解，获得该句子的语义表示，然后基于得到的语义表示生成新的句子
  - 基于机器翻译的方法
    - 基于机器翻译的方法则将文本复述生成问题看作是单语言机器翻译问题，从而利用现有机器翻译模型(例如噪声信道模型)来为给定文本生成复述文本
  - 基于支点(Pivot)的方法
    - 基于支点的方法则将当前语言中的输入文本翻译到另一种语言(支点)，然后将翻译得到的文本再次翻译回当前语言。由于每次翻译过程均要求源语言和目标语言中文本的语义保持一致，因此可以预期最后得到的文本在语义上能跟输入文本保持一致

输入英文句子：**What toxins are English most hazardous to expectant mothers?**

翻译后的意大利文句子：**Che tossine sono più pericolose alle donne incinte?**

再次翻译后的英文句子：**What toxins are more dangerous to pregnant women?**

句子简化(Sentence Simplification)可以看作是一类特殊的复述生成问题

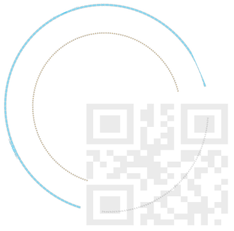
文本摘要研究进展

A DEEP REINFORCED MODEL FOR ABSTRACTIVE SUMMARIZATION

Romain Paulus, Caiming Xiong & Richard Socher
Salesforce Research
172 University Avenue
Palo Alto, CA 94301, USA
{rpaulus,cxiong,rsocher}@salesforce.com

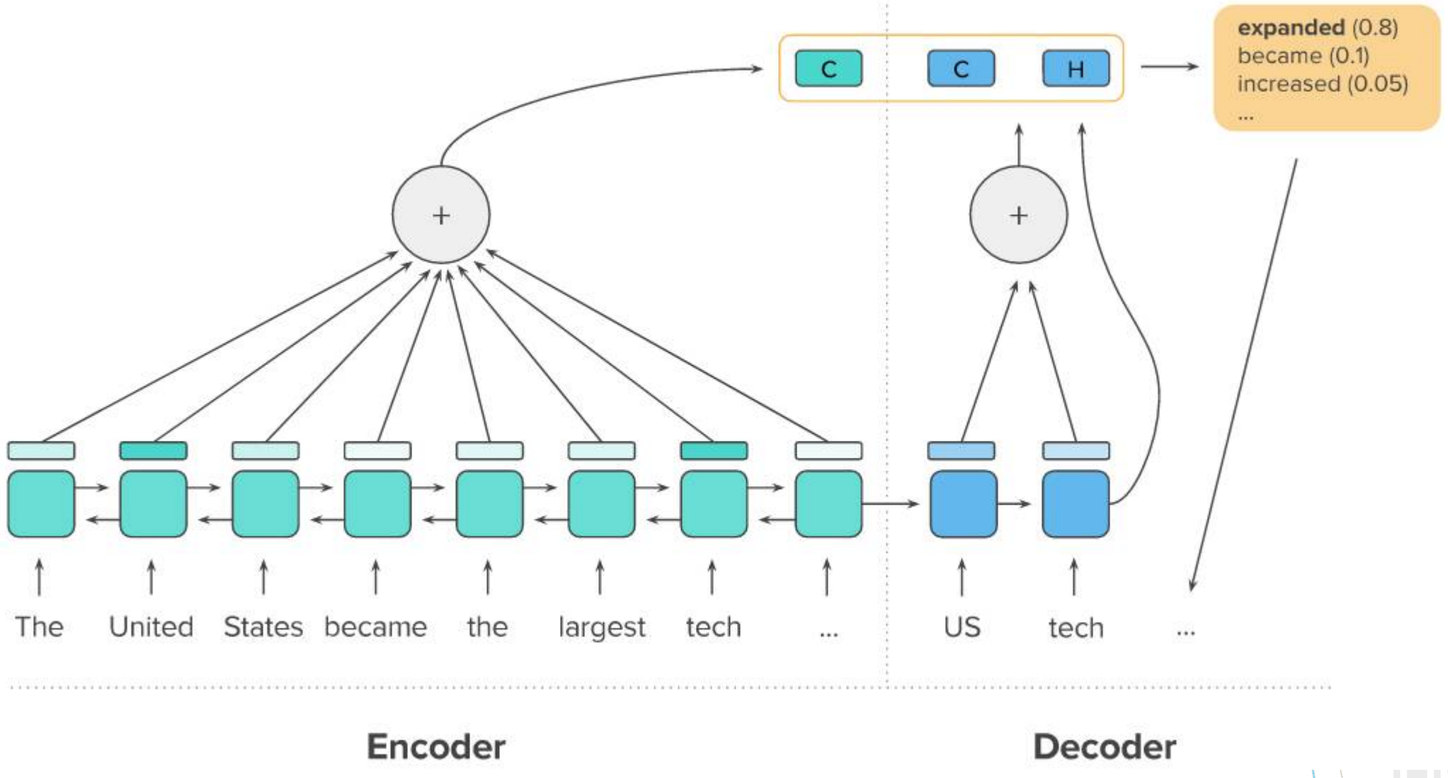
回顾

- 提取式摘要（Extractive Summarization）与抽象式摘要（Abstractive Summarization）
- 自动摘要模型可以通过以下两种方法实现：
  - 通过提取或抽象。提取式模型执行「复制和粘贴」操作：它们选择输入文档的相关短语并连接它们以形成摘要。它们非常稳健，因为它们使用直接从原文中提取的已有自然语言短语，但是由于不能使用新词或连接词，它们缺乏灵活性。它们也不能像人一样改述。
  - 相反，抽象式模型基于实际的「抽象」内容生成摘要：它们可以使用原文中没有出现的词。这使得它们有更多的潜力来产生流畅和连贯的摘要，但因为需要模型生成连贯的短语和连接词，这也是一个更难的问题。
- 利弊：
  - 虽然抽象式模型在理论上更强大，但在实践中也常出现错误。在生成的摘要中，典型的错误包括不连贯、不相关或重复的短语，特别是在尝试创建长文本输出时。从已有模型来看，它们缺乏一般连贯性、意识流动性和可读性。
- 贡献：
  - 他们设计了一个更稳健和更连贯的抽象式摘要模型。



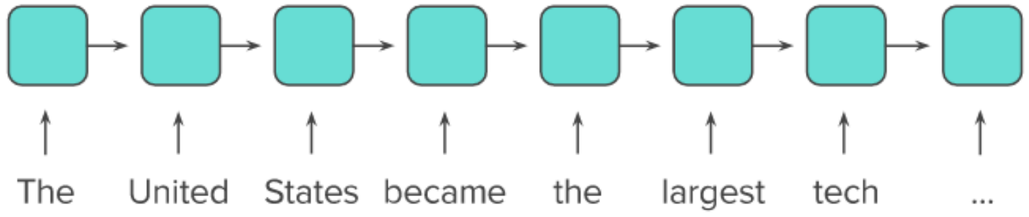
模型

- **Bi-LSTM 作为编码器**
  - **还得及上节课将为什么文本检索的编码器用gru和bi-gru对比吗？**
- **LSTM作为译码器**
- **使用Attention机制进行增强**
- **更好的训练方法**
  - **Global RL-based reward**
  - **Local supervision**



编码器

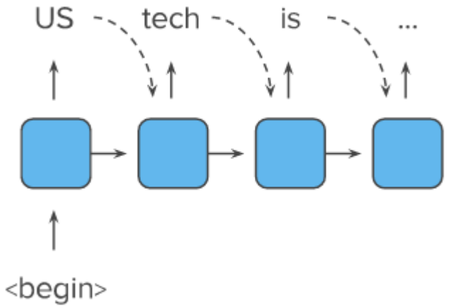
- 循环神经网络（RNN）能够处理可变长度的序列（例如文本），并为每个短语计算有用的表征（或隐藏状态）。网络逐一处理序列的每个元素（在这种情况下，即每个词）；对于序列中的每个新输入，网络通过该输入和之前隐藏状态的函数输出新的隐藏状态。
  - **Bi-RNN**，同时使用时序数据输入历史及未来数据，时序相反两个循环神经网络连接同一输出，输出层可以同时获取历史未来信息。
  - **Language Modeling**，不适合**Bi-RNN**(也有改进的)，目标是通过前文预测下一单词，不能将下文信息传给模型。分类问题，手写文字识别、机器翻译、蛋白结构预测，**Bi-RNN**提升模型效果。



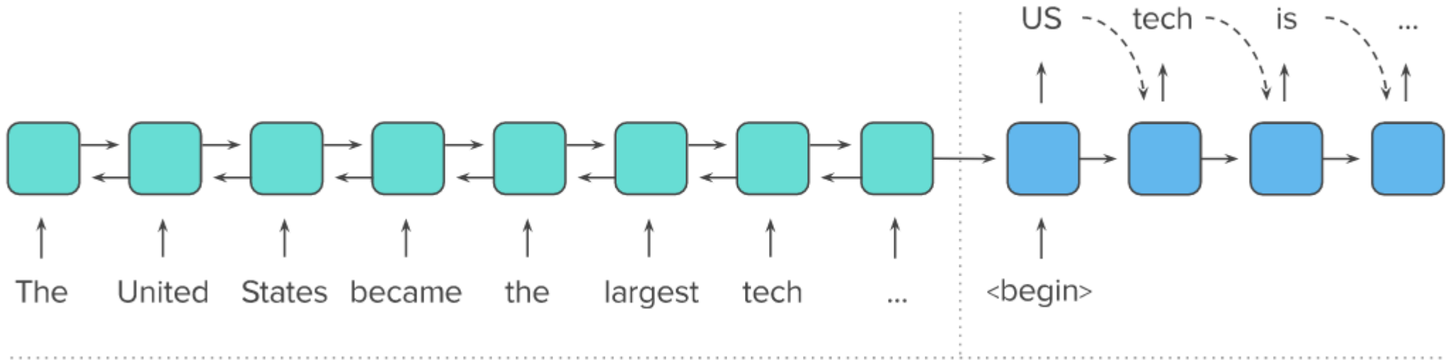
循环神经网络通过对每个词应用相同的函数（绿色）来读取输入语句

编码器+解码器

- **RNN 也可以用类似的方式产生输出序列。在每个步骤中，RNN 隐藏状态用于生成添加到最终输出文本的新词，该词将被用作该模型的下一个输出**



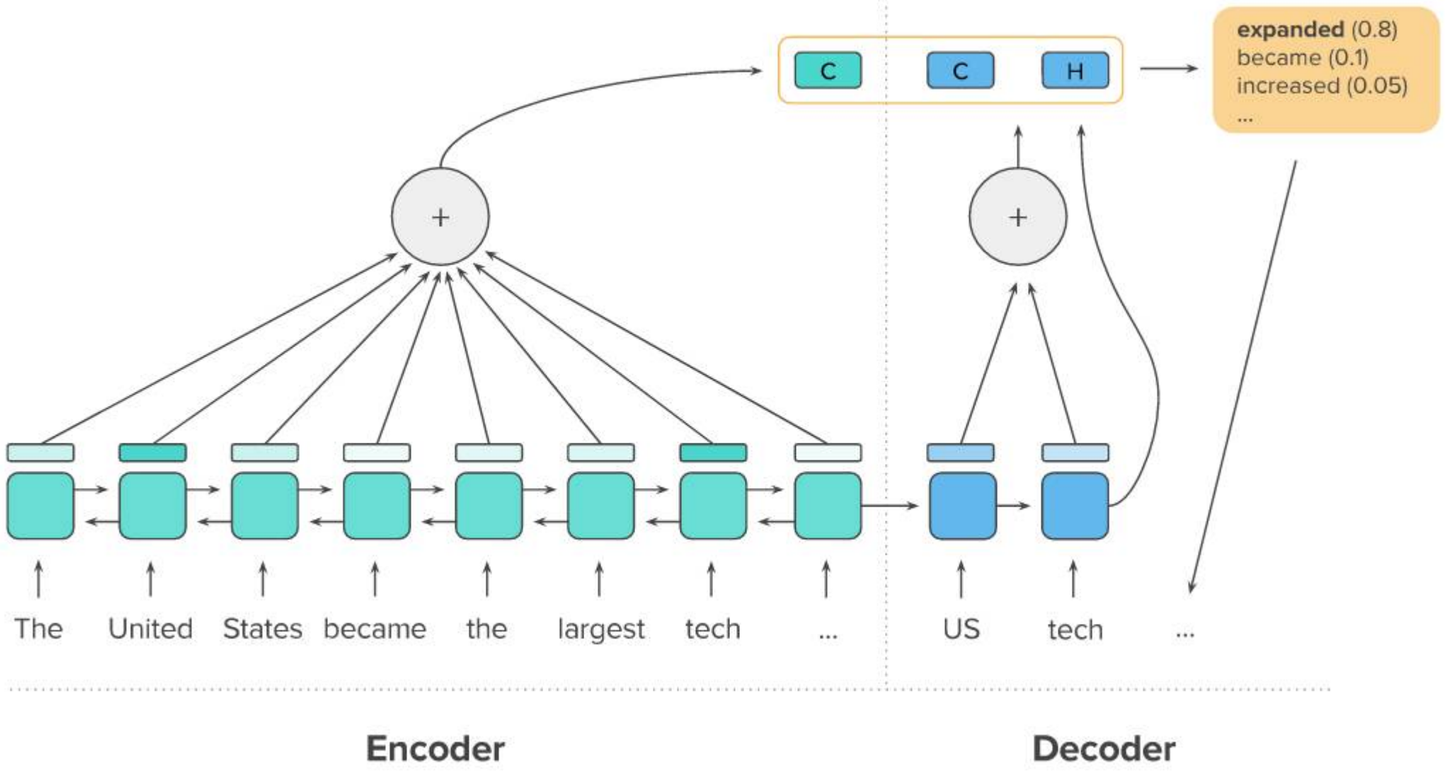
RNN 可以生成输出序列，并重使用输出单词作为下一个函数的输入



编码器-解码器 RNN 模型可用于解决自然语言中的 sequence-to-sequence 任务（如摘要）

贡献1：新的注意及解码机制

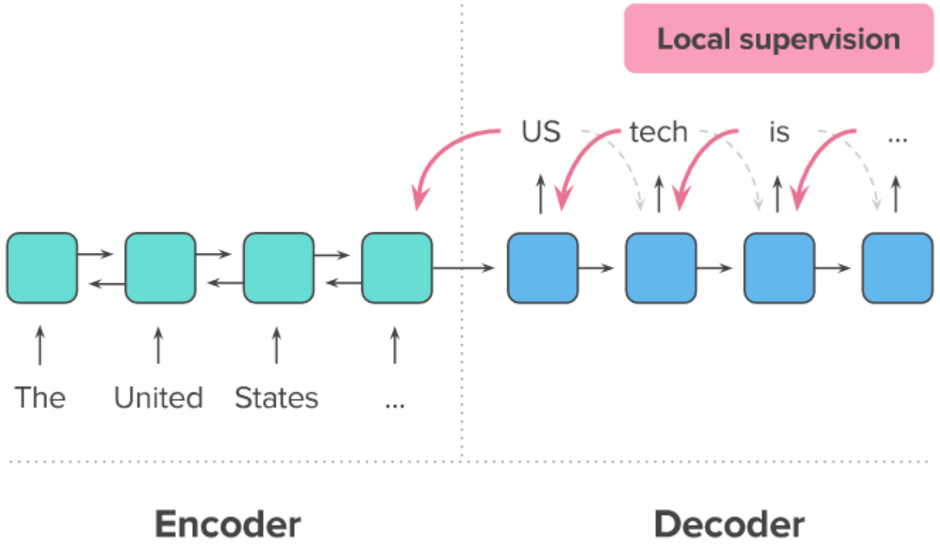
- **Temporal attention**



贡献2：监督式学习 + 强化学习

❑ 监督式学习

- ❑ **Teacher forcing algorithm**：一个模型在生成一个摘要时使用参考摘要（reference summary），并且该模型在每生成一个新单词时会被分配一个逐词误差（word-by-word error，或「局部监督/local supervision」
- ❑ 除了teacher forcing 还有professor learning, self-critical learning ….

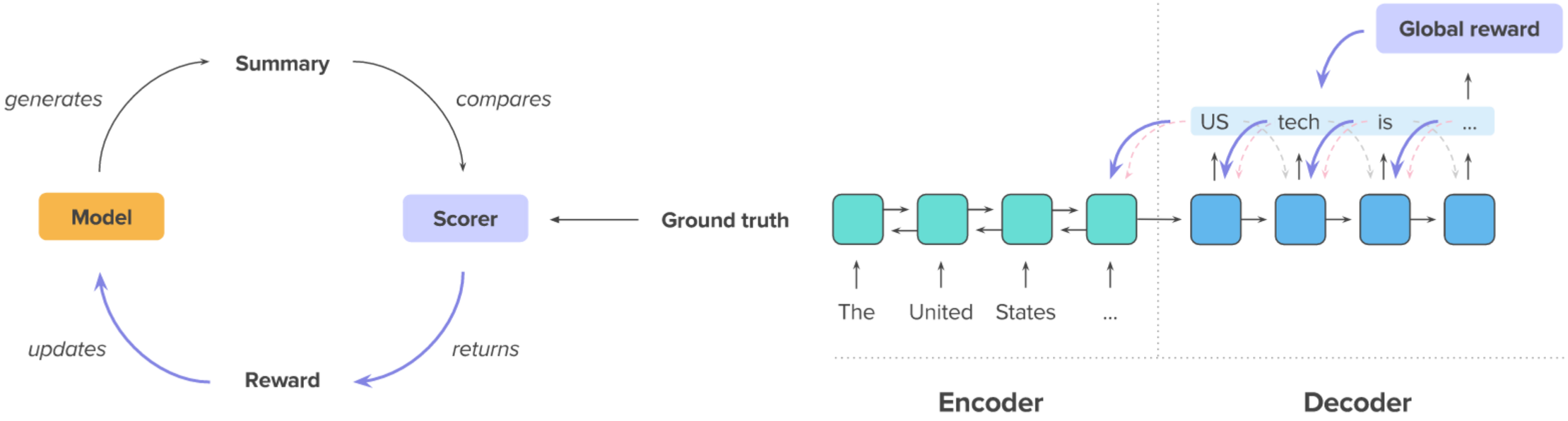


用监督式学习训练模型。每个生成的单词得到一个训练监督信号，通过与同一位置的正确摘要单词进行比较来进行训练。

贡献2：监督式学习 + 强化学习

❑ 强化学习

- ❑ 首先，强化学习算法使模型生成自己的摘要，然后使用外部评分器（scorer）来比较生成的摘要与正确摘要。
- ❑ 这个评分器然后向模型表明生成的摘要有多「好」。
- ❑ 如果分数很高，那么模型进行更新，使得这些摘要更有可能在将来出现。否则，如果得分低，模型将受到惩罚，并改变其生成过程以防止生成类似的摘要。这种强化模型擅长得出用于评估整个序列而不是逐词预测的摘要分数。



结果演示

Article	Summary (ground truth)	Summary (our model)
Google Wallet says it has changed its policy when storing users' funds as they will now be federally-insured (file photo) For those who use Google Wallet, their money just became safer with federal-level insurance. Google confirmed to Yahoo Finance in a statement that its current policy changed - meaning the company will store the balances for users of the mobile transfer service (similar to PayPal and Venmo) in multiple federally-insured banking institutions. This is good news for people who place large amounts of money in their Wallet Balance because the Federal Deposit Insurance Corporation insures funds for banking institutions up to $250,000. Currently, Google's user agreement says funds are not protected by the FDIC. However, a Google spokesperson told Yahoo Finance that the current policy has changed. *(...)*	Google spokesperson confirmed current policy changed meaning funds will be protected by the federal deposit insurance corporation. As a non-banking institution, Google Wallet, along with competitors PayPal and Venmo, is not legally required to be federally insured. With the new change to its policy, funds in wallet balance are protected if anything were to happen to the company like bankruptcy.	Google confirmed to Yahoo Finance in a statement that its current policy changed. The company will store the balances for users of the mobile transfer service (similar to PayPal and Venmo) in multiple federally-insured banking institutions. Google's user agreement says funds are not protected by the federal deposit insurance corporation.
Talk about a chain reaction! This is the moment a billiards player performs a complex trick shot by setting up a domino train to pot four balls. Video footage shows a white ball being rolled down a positioned cue. It then bounces off one side of the red-clothed table and hits the first in a long line of dominoes. One by one the small counters fall down, tapping balls into various pockets as they go. First a yellow, then a blue, then a red. Finally, the last domino gently hits an orange ball, causing it to roll down another positioned cue lying on the table. The orb then knocks a green ball into the center pocket. In less than 30 seconds the stunt comes to a close. *(...)*	The clip was uploaded by youtube user honda4ridered. In another upload the skilled billiards player shows viewers how to pocket four balls in a single shot-and for those who miss it there's a slow motion version.	Video footage shows a white ball being rolled down a jumper. It then bounces off one side of the red-clothed table and hits the first in a long line of dominoes. One by one the small counters fall down, tapping balls into pockets as they go-first a yellow. It comes to a close. The clip was uploaded by youtube user honda4ridered.

结果演示

Article

Tony Blair has said he does not want to retire until he is 91 – as he unveiled plans to set up a 'cadre' of ex-leaders to advise governments around the world. The defiant 61-year-old former Prime Minister said he had 'decades' still in him and joked that he would 'turn to drink' if he ever stepped down from his multitude of global roles. He told Newsweek magazine that his latest ambition was to recruit former heads of government to go round the world to advise presidents and prime ministers on how to run their countries. In an interview with the magazine Newsweek Mr Blair said he did not want to retire until he was 91 years old Mr Blair said his latest ambition is to recruit former heads of government to advise presidents and prime ministers on how to run their countries Mr Blair said he himself had been 'mentored' by US president Bill Clinton when he took office in 1997. And he said he wanted to build up his organisations, such as his Faith Foundation, so they are 'capable of changing global policy'. Last night, Tory MPs expressed horror at the prospect of Mr Blair remaining in public life for another 30 years. Andrew Bridgen said: 'We all know weak Ed Miliband's called on Tony to give his flailing campaign a boost, but the attention's clearly gone to his head.' (...)

Summary (ground truth, written by a human)

The former Prime Minister claimed he has 'decades' of work left in him. Joked he would 'turn to drink' if he ever stepped down from global roles. Wants to recruit former government heads to advise current leaders. He was 'mentored' by US president Bill Clinton when he started in 1997.

Summary (our model)

Blair said he did not want to retire until he was 91 years old. 61-year-old former prime minister said he would 'turn to drink' if he ever stepped down from his own. He said he wanted to build up his charity to advise presidents and prime ministers on how to run their countries. Mr Blair says he is to recruit former heads of government to go round the world to advise ministers. He says he wants to emulate ex-Israeli president Shimon Peres.

Summary (without intra-attention and reinforcement learning)

61-year-old former prime minister said he did not want to retire until he was 91 years old. He said he wanted to build up his organisations, such as his Faith Foundation. He said he wanted to emulate ex-Israeli president Shimon Peres. Mr Blair said he wanted to emulate ex-Israeli President Shimon Peres. He said he wanted to be seeing someone when he took office in 1997. Mr Blair said he wanted to be seeing someone when he took office in 1997. Mr Blair said he wanted to

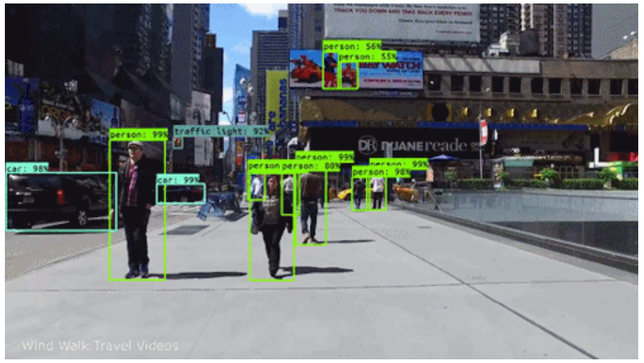
图像到文本生成介绍

Computer vision

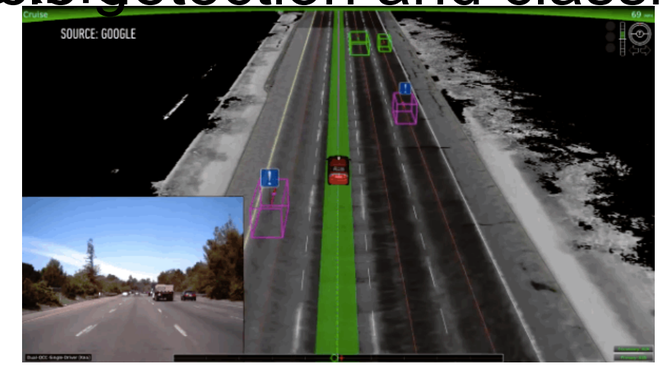
Natural language processing

What is Computer Vision

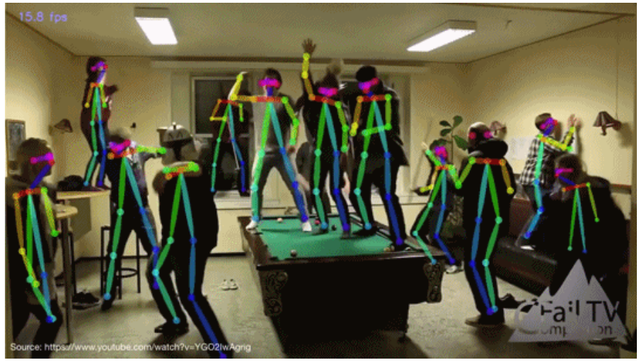
Pedestrian detection and tracking



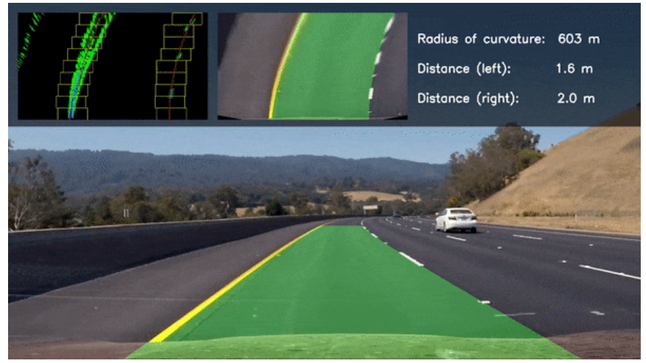
Vehicle detection and classification



Pose estimation

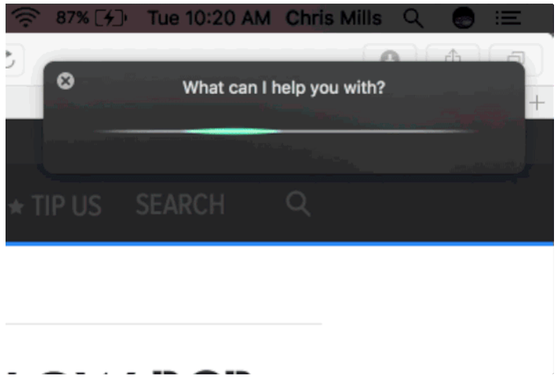


Road segmentation



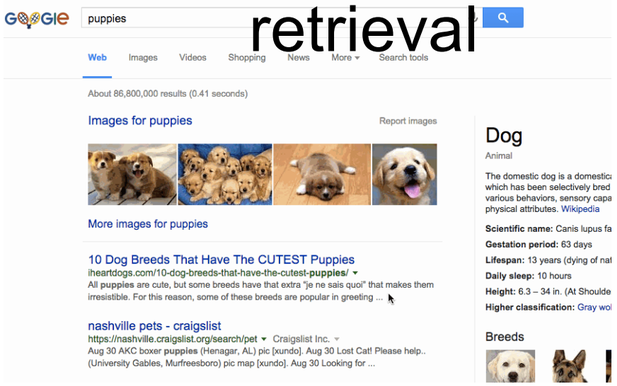
What is Natural Language Processing

Dialog

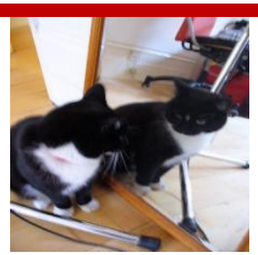


Optical character recognition

Text-based image retrieval



What is Image Captioning?



Visual to Text Translator

A black and white cat looking at itself in a mirror

Image Encoder (CNN etc.,)

Text Decoder (RNN, GRU, etc.,)

*The goal of image captioning is to "**translate**" a query image into a sentence that describes the image.*

图像到文本的生成

- 图像到文本的生成技术是指根据给定的图像生成描述该图像内容的自然语言文本，例如新闻图像附带的标题、医学图像附属的说明、儿童教育中常见的看图说话、以及用户在微博等互联网应用中上传图片时提供的说明文字。

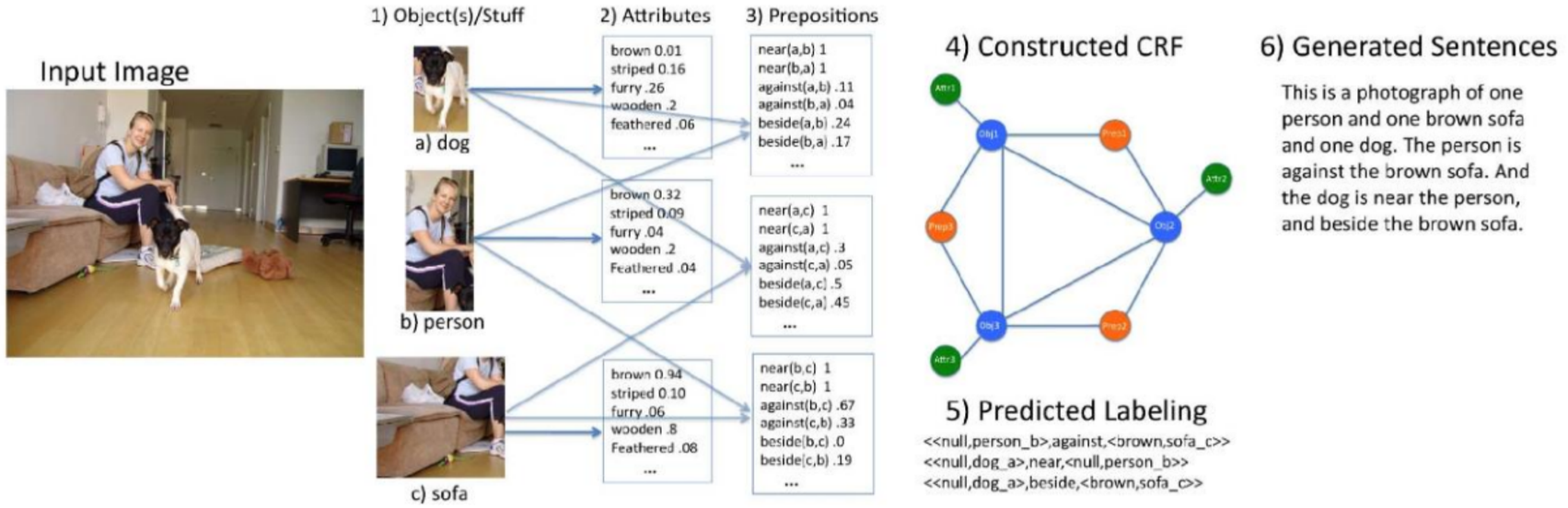
- 对于图像到文本的自动生成这一任务，人类可以毫不费力地理解图像内容，并按具体需求以自然语言句子的形式表述出来；然而对于计算机而言，则需要综合运用图像处理，计算机视觉和自然语言处理等几大领域的研究成果。作为一项标志性的交叉领域研究任务，图像到文本的自动生成吸引着来自不同领域研究者的关注。

- 自2010年起，自然语言处理界的知名国际会议和期刊ACL、TACL和EMNLP中都有相关论文的发表；而自2013年起，模式识别与人工智能领域顶级国际期刊IEEE TPAMI以及计算机视觉领域顶级国际期刊IJCV也开始刊登相关工作的研究进展，至2015年，计算机视觉领域的知名国际会议CVPR中，更是有近10篇相关工作的论文发表，同时机器学习领域知名国际会议ICML中也有2篇相关论文发表。图像到文本的自动生成任务已被认为是人工智能领域中的一项基本挑战。

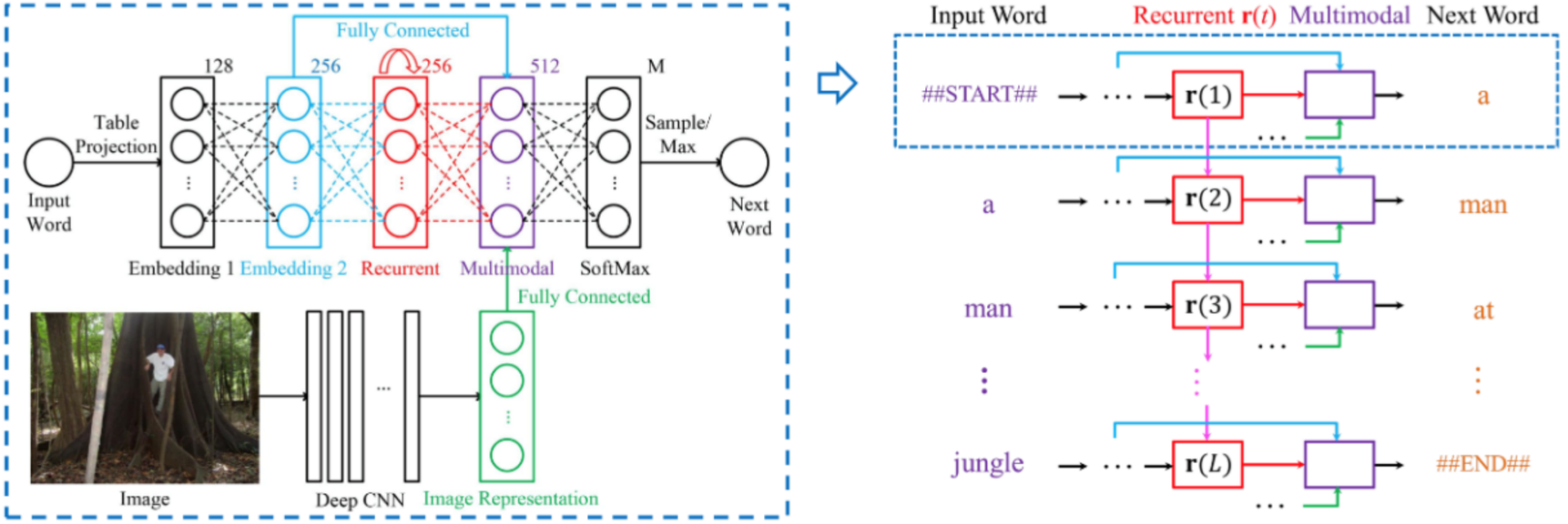
图像到文本的生成

- 与一般的文本生成问题类似，解决图像到文本的自动生成问题也需要遵循三阶段流水线模型
  - 在内容抽取方面，需要从图像中抽取物体、方位、动作、场景等概念，其中物体可以具体定位到图像中的某一具体区域，而其他概念则需要进行语义标引。这部分主要依靠模式识别和计算机视觉技术。
  - 在句子内容选择方面，需要依据应用场景，选择最重要（如图像画面中最突出的，或与应用场景最相关的），且意义表述连贯的概念。这部分需要综合运用计算机视觉与自然语言处理技术
  - 最后，在句子实现部分，根据实际应用特点选取适当的表述方式将所选择的概念梳理为合乎语法习惯的自然语言句子。这部分主要依靠自然语言处理技术

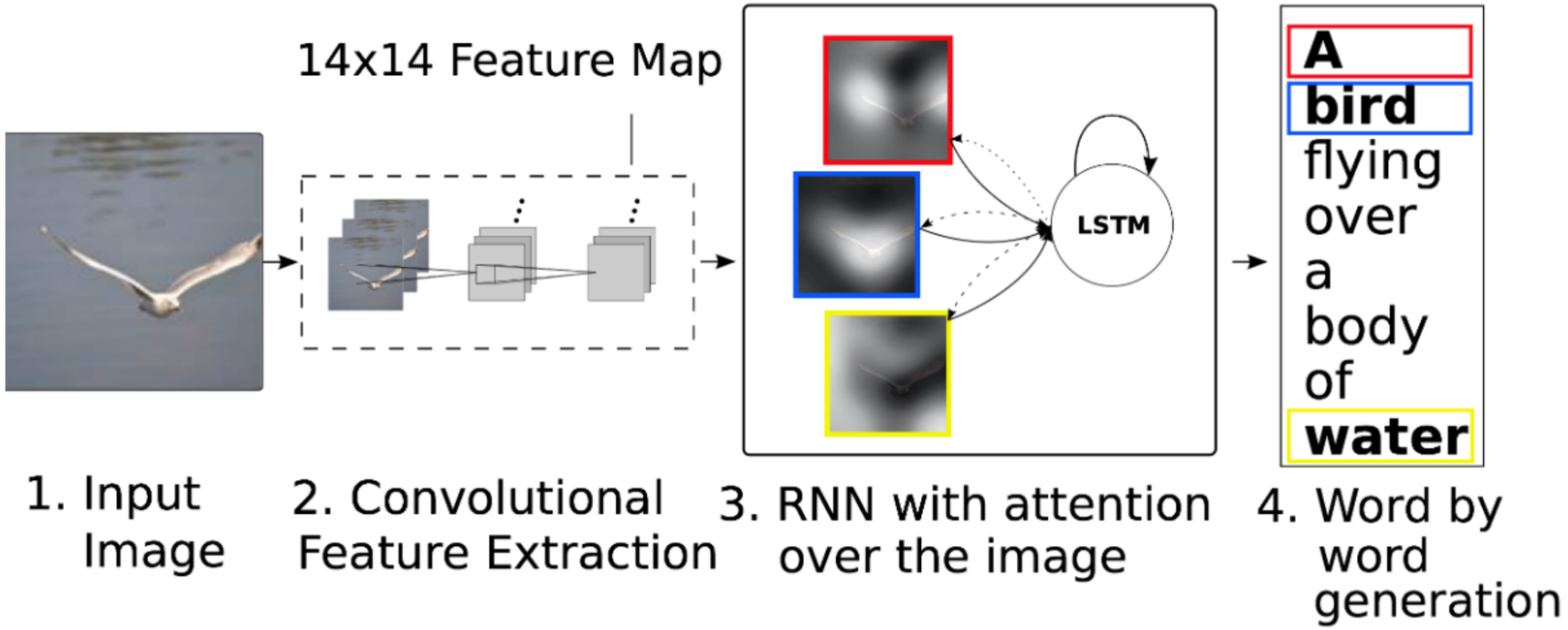
模板方法


Input Image
1) Object(s)/Stuff
2) Attributes
3) Prepositions
a) dog
brown 0.01
striped 0.16
furry .26
wooden .2
feathered .06
...
near(a,b) 1
near(b,a) 1
against(a,b) .11
against(b,a) .04
beside(a,b) .24
beside(b,a) .17
...
b) person
brown 0.32
striped 0.09
furry .04
wooden .2
Feathered .04
...
near(a,c) 1
near(c,a) 1
against(a,c) .3
against(c,a) .05
beside(a,c) .5
beside(c,a) .45
...
c) sofa
brown 0.94
striped 0.10
furry .06
wooden .8
Feathered .08
...
near(b,c) 1
near(c,b) 1
against(b,c) .67
against(c,b) .33
beside(b,c) .0
beside(c,b) .19
...
4) Constructed CRF
5) Predicted Labeling
<<null,person_b>,against,<brown,sofa_c>>
<<null,dog_a>,near,<null,person_b>>
<<null,dog_a>,beside,<brown,sofa_c>>
6) Generated Sentences
This is a photograph of one person and one brown sofa and one dog. The person is against the brown sofa. And the dog is near the person, and beside the brown sofa.

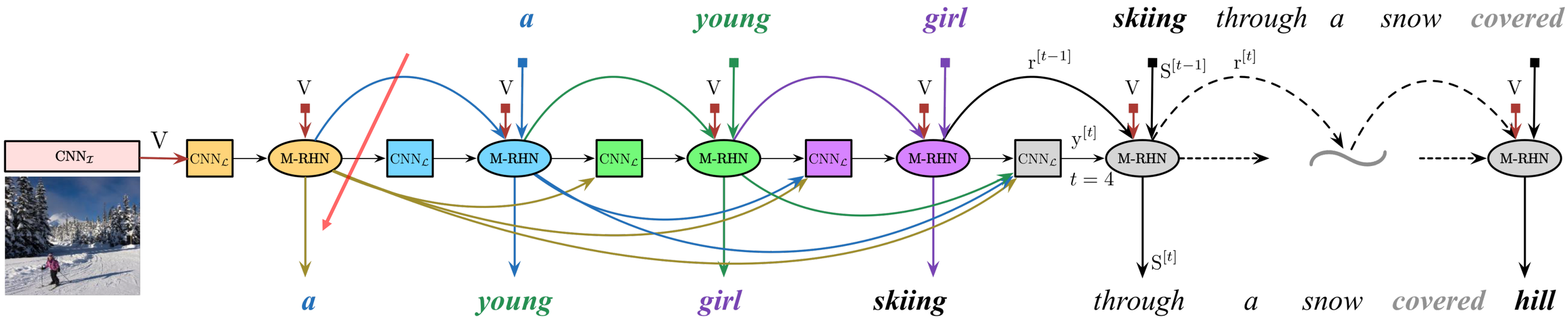
RNN方法

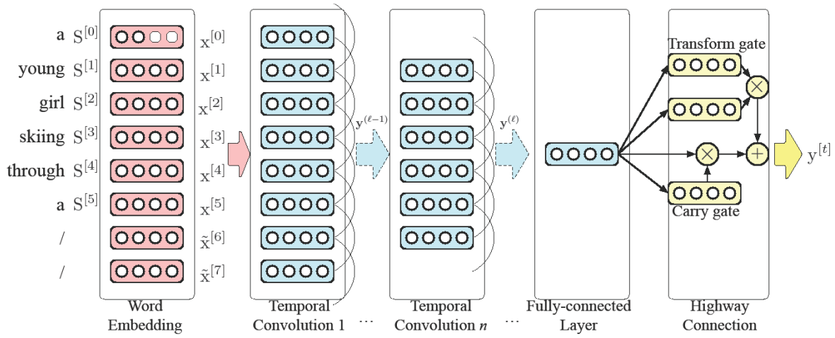

Fully Connected
128
256
256
512
M
Table Projection
Input Word
Sample/ Max
Next Word
Embedding 1
Embedding 2
Recurrent
Multimodal
SoftMax
Fully Connected
Image
Deep CNN
Image Representation
Input Word
Recurrent r(t)
Multimodal
Next Word
##START##
r(1)
a
a
r(2)
man
man
r(3)
at
jungle
r(L)
##END##

Attention+RNN方法



CNN+RNN方法


a young girl skiing through a snow covered
CNN$_\mathcal{I}$
V
CNN$_\mathcal{L}$
M-RHN
$r^{[t-1]}$
$S^{[t-1]}$
$r^{[t]}$
$y^{[t]}$
$t = 4$
$S^{[t]}$
a young girl skiing through a snow covered hill

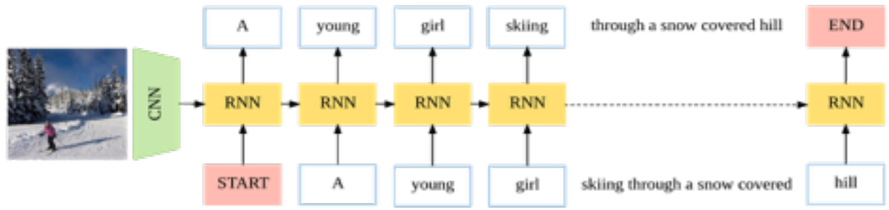

a $S^{[0]}$
young $S^{[1]}$
girl $S^{[2]}$
skiing $S^{[3]}$
through $S^{[4]}$
a $S^{[5]}$
/
/
$x^{[0]}$
$x^{[1]}$
$x^{[2]}$
$x^{[3]}$
$x^{[4]}$
$x^{[5]}$
$\tilde{x}^{[6]}$
$\tilde{x}^{[7]}$
Transform gate
Carry gate
$y^{[t]}$
Word Embedding
Temporal Convolution 1
Temporal Convolution n
Fully-connected Layer
Highway Connection

Challenges

- **Long-term dependency**
  - The history-summarizing hidden states of RNNs are updated at each time, which render the **long-term memory rather difficult**.
  - Although models like LSTM networks have **memory cells** which aim to memorize history information for long-term, they are still limited to several time steps because **long-term information is gradually diluted at every time step**.
- **Hierarchical structure of word sequences**
  - LSTM networks cannot explicitly model the **hierarchical representation of words**.
  - Even with multi-layer LSTM networks, such hierarchical structure is still hard to be captured due to the **more complex model and higher risk of over-fitting**.

The classical **encoder-decoder** framework for image captioning (first proposed in Vinyals et.al., CVPR 2015).

Vinyals, Oriol, et al. "Show and tell: A neural image caption generator." CVPR**. 2015.**



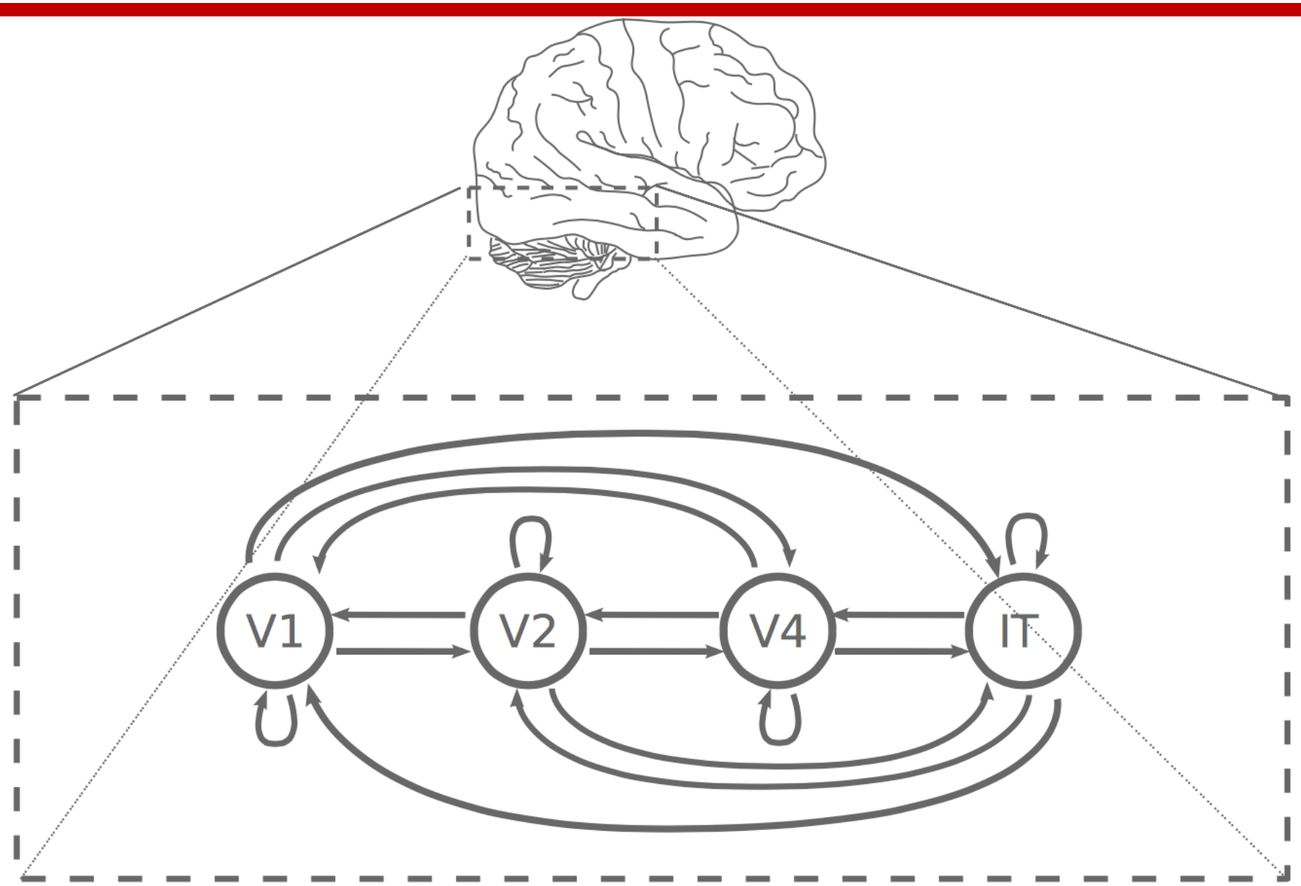
$$v = \mathrm{CNN}(I)$$

$$h_t = \mathrm{RNN}(h_{t-1}, \mathbf{W}_e w_{t-1}; v)$$

$$w_t \sim \arg\max \mathrm{Softmax}(\mathbf{W}_o h_t + \mathbf{b}_o)$$

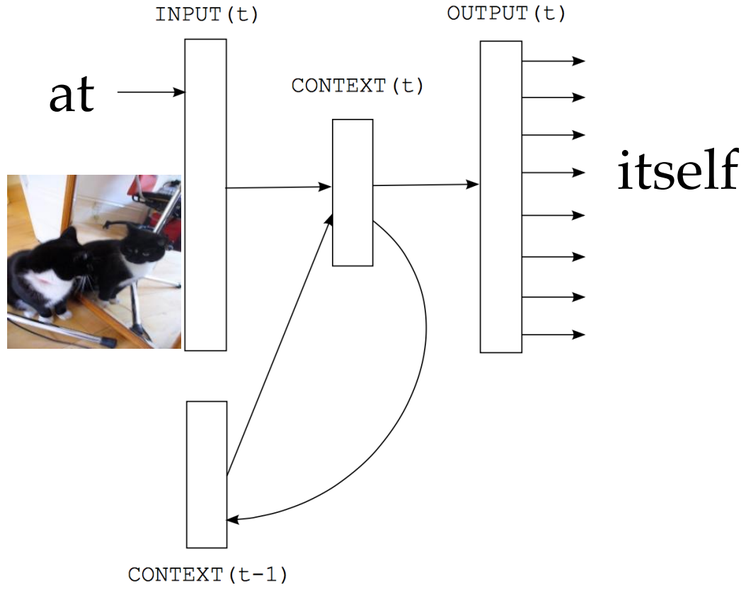
$$\mathcal{L}_{XE} = -\sum_S \log P(w_t | w_0, \cdots, w_{t-1}; I)$$

Motivation

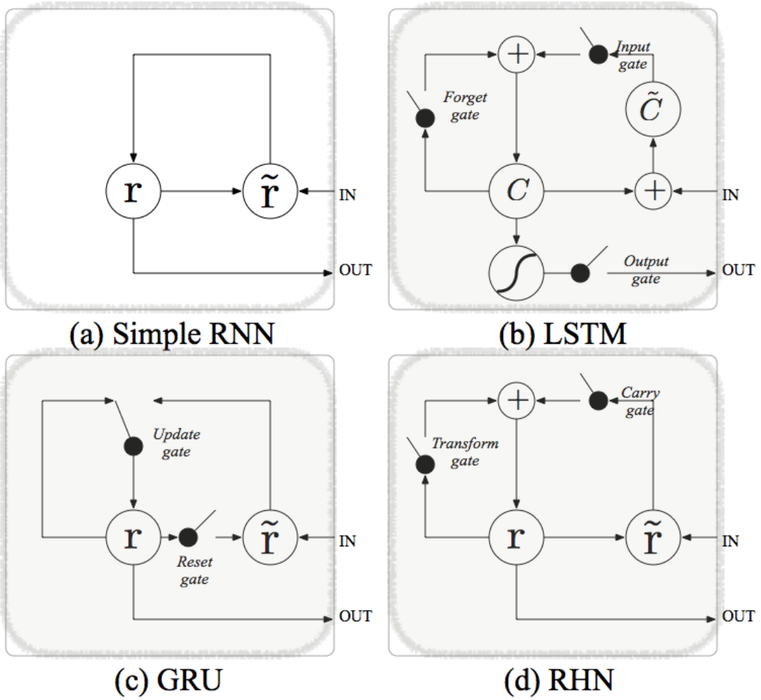


Explicitly model long-term dependencies

Motivation

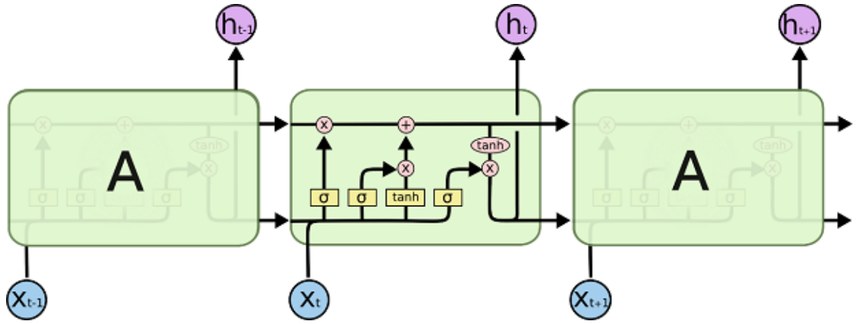


- **What is the Role of Recurrent Neural Networks (RNNs) in an Image Caption Generator?**
  - In neural image captioning systems, a recurrent neural network (RNN) is typically viewed as the primary 'generation' component.
  - The goal of statistical language modeling is to predict the next word in textual data given context.
  - Thus we are dealing with sequential data prediction problem when constructing image captioning models.
  - It is well known that humans can exploit longer context with great success. However, it is also often claimed that learning long-term dependencies by stochastic gradient descent can be quite difficult.
  - Many variants of RNN have been proposed, e.g., GRU, LSTM, GRH, etc.
- **Which one is the best?**
  - **The best RNNs are the ones that work well for you ！ ！ ！**
  - We should select the appropriate recurrent network for our application (LSTM not always suitable)
- **Why we need RNNs?**
  - The reason we choose RNN because we need the recurrent path.
  - Each time step those improved RNNs will drop some information. Maybe not useable for current step, but may be useful for future time steps.



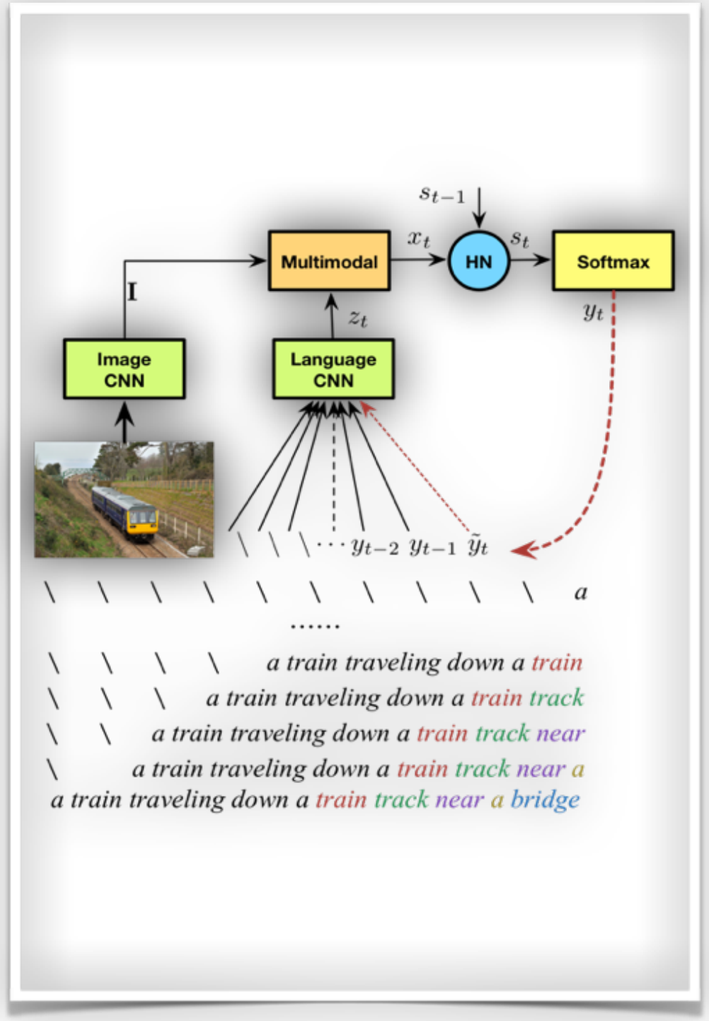
Motivation

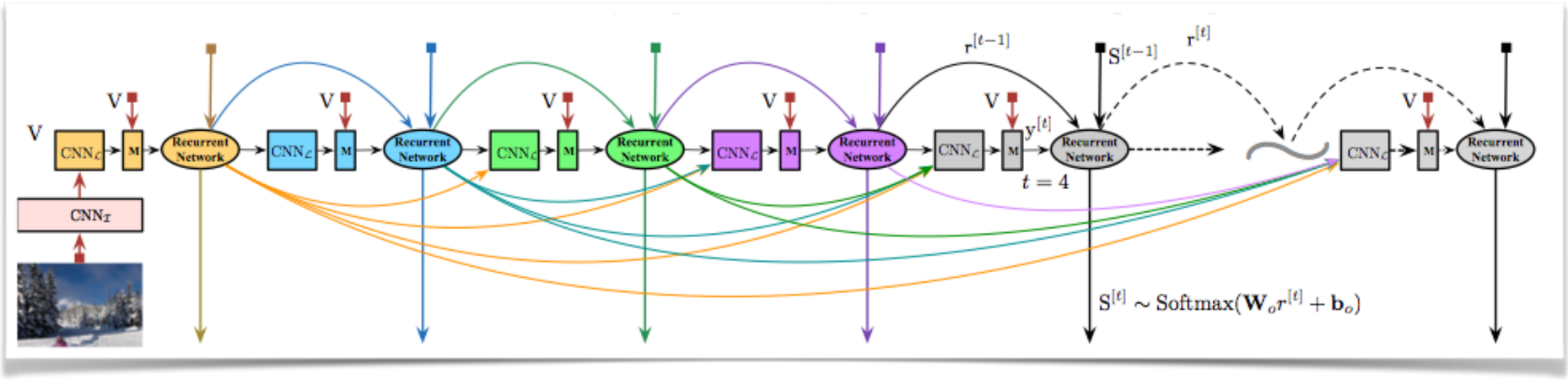
- **Let's take a close look at LSTM.**
  - The full name of LSTM is **Long Short-Term Memory** (LSTM).
    - Long-Term
      - The expression long short-term refers to the fact that LSTM is a model has a memory which can last for a long period of time.
      - Long-term dependencies are hard to learn, especially when we have limited number of data.
      - However, we can not trust LSTM completely, as the information will be dropped each time step (input gate, forget gate, and output gate)
    - Short-Term
      - Each word prediction is highly depended on their adjacent words (n-gram model).
      - Modeling the dynamic temporal behaviour.
- **LSTM sometime works, but not always, there is no strict theoretical proof ！**
- **We need a network which can learn both long-term and short-term behaviour.**
- **Let's propose a new network !!!**

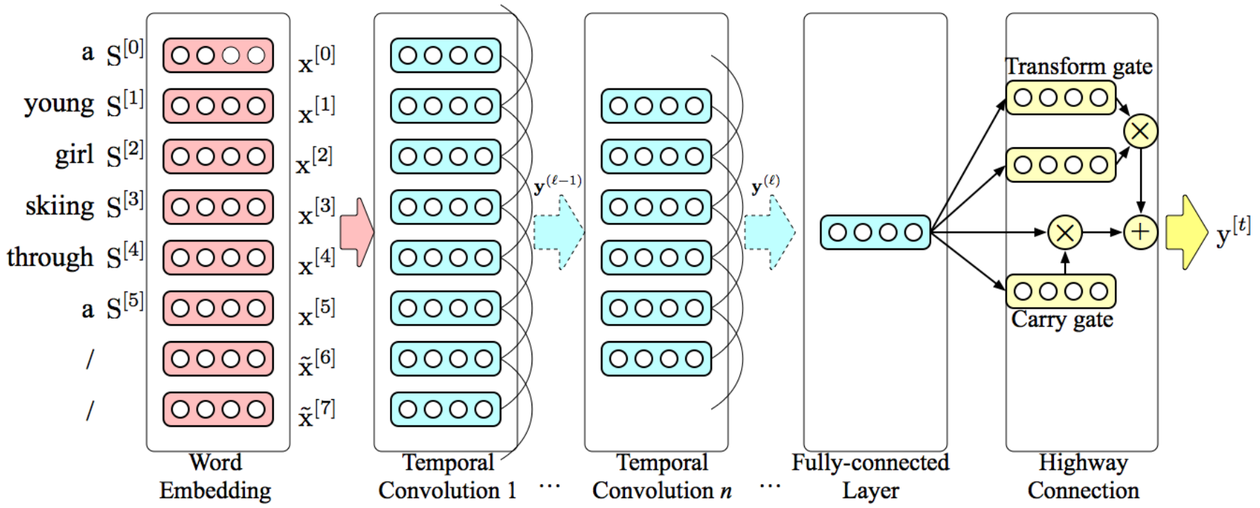


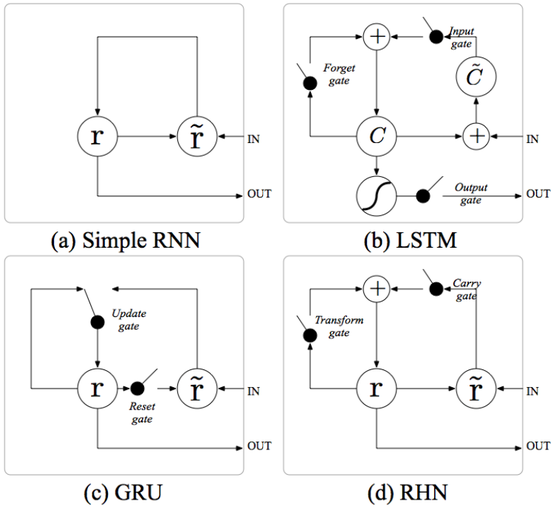
Motivation

- **Convolutional Neural Networks**
  - ConvNets can be stacked to extract hierarchical features over long-range contexts.
  - CNNs have shown their powerful abilities for image and text representation.
  - To model words with long-term dependencies, we adopt a language CNN with a hierarchical structure to capture the long-range dependencies between the input words.
- **Dynamic temporal behaviour**
  - Only using language CNN may miss the important temporal information because it extracts the holistic features from the whole sequence of words.
  - Temporal recurrence of RNNs is still crucial for modelling the short-term contextual information across words in the sentence.
  - To overcome this limitation, we combine our language CNN with four types of recurrent networks: Simple RNN, LSTM network, GRU, and Recurrent Highway Network.









(a) Simple RNN (b) LSTM (c) GRU (d) RHN

Our model estimates the probability distribution of the next word given previous words and image. It consists of four parts: a CNN_I for image feature extraction, a deep CNN_L for language modelling, a multimodal layer (M) that connects the CNN_I and CNN_L, and a Recurrent Network for word prediction. The weights are shared among all time frames.

** Our model is trained with cross-entropy loss*

Language CNN ($\text{CNN}_{\mathcal{L}}$)

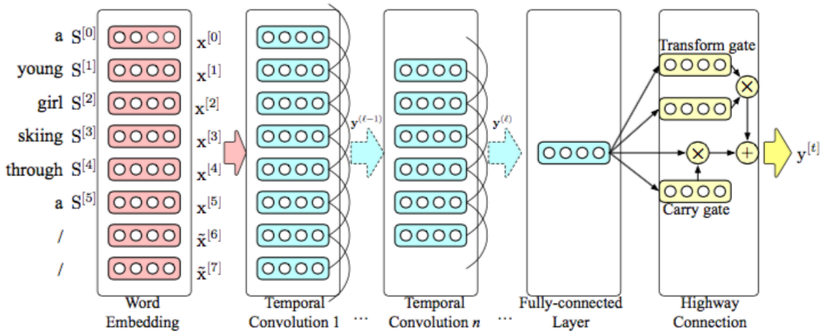


Figure 1: The architecture of language CNN for sentence modeling.

❶ The first layer of $\text{CNN}_{\mathcal{L}}$ is a word embedding layer.
- $\mathbf{x} = [\mathrm{x}^{[0]}, \mathrm{x}^{[1]}, \cdots, \mathrm{x}^{[t-1]}]^T, \mathbf{x} \in \mathbb{R}^{t \times K}$
- $\mathrm{x}^{[t]} = \mathbf{W}_e \mathrm{S}^{[t]}$, where $\mathbf{W}_e \in \mathbb{R}^{K \times V}$ is a word embedding matrix.
- $\mathbf{S} = \{\mathrm{S}^{[0]}, \mathrm{S}^{[1]}, \cdots, \mathrm{S}^{[t-1]}\}$ are input words.

❷ The following layers are temporal convolution layers.
- The output feature map $\mathbf{y}^{(\ell)} \in \mathbb{R}^{M_\ell \times K}$ of layer-ℓ will be:

$$\mathrm{y}_i^{(\ell)}(\mathbf{x}) = \sigma(\mathbf{w}_L^{(l)} \mathbf{y}_i^{(\ell-1)} + \mathrm{b}_L^{(\ell)}) \tag{1}$$

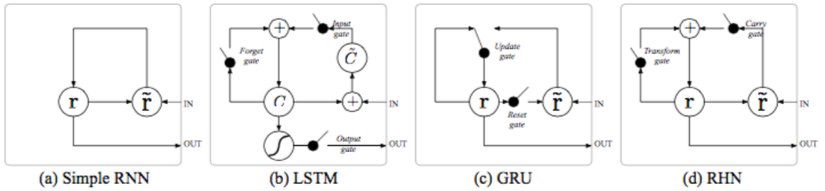
- $\mathbf{y}^{(\ell-1)} \in \mathbb{R}^{M_{\ell-1} \times K}$ is the input feature map of Layer-ℓ.
- $\mathbf{y}^{(0)}$ is the concatenation of t word embedding:

$$\mathbf{y}^{(0)} \stackrel{\text{def}}{=} \begin{cases} [\mathrm{x}^{[t-L_{\mathcal{L}}]}, \cdots, \mathrm{x}^{[t-1]}]^T, & \text{if } t \geq L_{\mathcal{L}} \\ [\mathrm{x}^{[0]}, \cdots, \mathrm{x}^{[t-1]}, \tilde{\mathrm{x}}^{[t]}, \cdots, \tilde{\mathrm{x}}^{[L_{\mathcal{L}}-1]}]^T & \text{otherwise} \end{cases} \tag{2}$$

- When $t \geq L_{\mathcal{L}}$, the input sentence will be truncated.
- When $t < L_{\mathcal{L}}$, the input sentence will be padded with $\tilde{\mathrm{x}}^{[:]}$.
- When $t = 0$, $\tilde{\mathrm{x}}^{[:]}$ are the image features V, otherwise $\tilde{\mathrm{x}}^{[:]}$ are the zero vectors that have the same dimension as $\mathrm{x}^{[:]}$.

❸ $\text{CNN}_{\mathcal{L}}$ with larger window size can better utilize contextual information and learn better word embedding representation.

Recurrent Networks



(a) Simple RNN (b) LSTM (c) GRU (d) RHN

Figure 2: Combination $\text{CNN}_{\mathcal{L}}$ with 4 types of RNNs: Simple RNN, LSTM, GRU, and RHN.

❶ The transition equations of the RNNs can be formulated as:

$$\mathrm{r}^{[t]} = f_{\text{recurrent}}(\mathrm{r}^{[t-1]}, \mathrm{x}^{[t-1]}, \mathrm{m}^{[t]}) \tag{3}$$

$$\mathrm{S}^{[t]} \sim \arg\max_{S} \text{Softmax}(\mathbf{W}_o \mathrm{r}^{[t]} + \mathbf{b}_o) \tag{4}$$

❷ Traditionally, the simple RNN updates the recurrent state $\mathrm{r}^{[t]}$ as:

$$\mathrm{r}^{[t]} = \tanh(\mathbf{W}_r \mathrm{r}^{[t-1]} + \mathbf{W}_z \mathrm{z}^{[t]} + \mathbf{b}) \tag{5}$$

❸ $\mathrm{z}^{[t]} \in \mathbb{R}^{2K}$ denotes the concatenation of two vectors: $\mathrm{m}^{[t]}$ and $\mathrm{x}^{[t-1]}$:

$$\mathrm{z}^{[t]} = [f_{\text{multimodal}}(\text{CNN}_{\mathcal{L}}(\mathrm{x}^{[0,\cdots,t-1]}), \mathrm{V}); \mathrm{x}^{[t-1]}] \tag{6}$$

Multimodal Fusion Layer

❶ Fusing the image features and word representation at each t:

$$\mathrm{m}^{[t]} = \sigma\left(f_{\mathrm{y}}(\mathrm{y}^{[t]}; \mathbf{W}_{\mathrm{Y}}, \mathbf{b}_{\mathrm{Y}}) + g_{\mathrm{v}}(\mathrm{V}; \mathbf{W}_{\mathrm{V}}, \mathbf{b}_{\mathrm{V}})\right) \tag{7}$$

❷ V is the image representation from $\text{CNN}_{\mathcal{I}}$.

❸ $\mathrm{y}^{[t]}$ the bottom-up words representation from $\text{CNN}_{\mathcal{L}}$.

❹ $f_{\mathrm{y}}(\cdot)$ and $g_{\mathrm{v}}(\cdot)$ are linear mapping functions.

❺ $\sigma(\cdot)$ is the scaled tanh function.

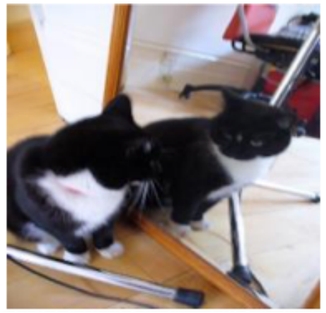
❻ $\mathrm{m}^{[t]}$ is the multimodal layer output feature vector.

Training

❶ Maximizing the probability of the target words by using:

$$\mathcal{L}(\mathbf{S}, \mathbf{I}) = -\sum_{t=0}^{N-1} \log P(\mathrm{S}^{[t]} | \mathrm{S}^{[0]}, \cdots, \mathrm{S}^{[t-1]}, \mathbf{I}) \tag{8}$$

Qualitative Results



CNNL+RHN **:** a black and white cat looking at itself in a mirror
CNNL+RNN **:** a black and white cat sitting in front of a mirror
GRU **:** a black and white cat standing next to a mirror
LSTM **:** a black and white cat sitting in a bathroom sink
RNN **:** a cat sitting on the floor in a bathroom

- there is a black tuxedo cat looking in the mirror
- two cats sitting on top of a wooden floor
- a cat looking at itself in the mirror next to a tripod
- a cat and a tripod sitting in front of a mirror
- a close up of a cat in a mirror

CNNL+RHN **:** a man standing next to a child on a snow covered slope
CNNL+RNN **:** a man and a woman standing on a snow covered slope
GRU **:** a man and a child standing on a snow covered slope
LSTM **:** a man and a child are standing in the snow
RNN **:** a man and a woman are skiing on the snow

- a woman and child in ski gear next to a lodge
- a man and a child are smiling while standing on skiis
- a young man poses with a little kid in the snow
- an adult and a small child dressed for skiing
- a man and a little girl in skis stand in front of a mountain lodge

CNNL+RHN **:** a man talking on a cell phone while walking down a street
CNNL+RNN **:** a man is talking on a cell phone
GRU **:** a man is talking on a cell phone in the street
LSTM **:** a man is talking on his cell phone
RNN **:** a man standing next to a woman talking on a cell phone

- a man talking on the phone in front of a blue car
- a man on a telephone holds his hand up to his other ear as he walks
- a man standing next to a car with a cellphone
- a man is talking on a cell phone next to a city street
- a man standing on the side of the street with a cell phone up to his

CNNL+RHN **:** a cat looking at a dog in a door
CNNL+RNN **:** a cat is looking at a dog in front of a window
GRU **:** a cat standing next to a door looking out a window
LSTM **:** a dog and a cat are standing in front of a window
RNN **:** a cat sitting on the side of the road

- a dog looking at a cat through a glass window
- a cat is outside looking through in at a dog
- the dog wants to go outside with the cat
- a cat sitting outside of a door next to a dog
- a cat sitting at a sliding glass door

Quantitative Results

⋆ B@n are short for BLEU-n, M is short for METEOR, and S is short for SPICE, C is short for CIDEr. All values are reported as percentage.

Analysis of $\text{CNN}_{\mathcal{L}}$ on MS COCO.

Different hisotry encoding approaches

Approach	B@4	C		B@4	C
$\text{Avg}_{\text{history}}$+RHN	30.1	95.8	$\text{CNN}_{\mathcal{L}_{2\text{ words}}}$+RHN	29.2	93.8
$\text{CNN}_{\mathcal{L}_{16\text{ words}}}$+RHN	28.9	91.9	$\text{CNN}_{\mathcal{L}_{4\text{ words}}}$+RHN	29.5	95.8
$\text{CNN}_{\mathcal{L}}$+RHN	**30.6**	**98.9**	$\text{CNN}_{\mathcal{L}_{8\text{ words}}}$+RHN	30.0	95.9

Different architectures

Approach	Params	B@4	C		Params	B@4	C
Simple RNN	5.4M	27.0	87.0	LSTM	7.0M	29.2	92.6
$\text{CNN}_{\mathcal{L}}$	6.3M	18.4	56.8	LSTM_2	9.1M	**29.7**	93.2
$\text{CNN}_{\mathcal{L}}$+RNN	**11.7M**	29.5	**95.2**	LSTM_3	11.2M	29.3	92.9

Results Using $\text{CNN}_{\mathcal{L}}$ on MS COCO and Flickr30K.

Performance comparison on MS COCO

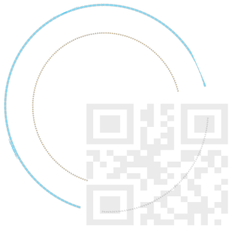
Approach	B@1	B@2	B@3	B@4	M	C	S
Simple RNN	70.1	52.1	37.6	27.0	23.2	87.0	16.0
$\text{CNN}_{\mathcal{L}}$+RNN	72.2	55.0	40.7	29.5	24.5	95.2	17.6
RHN	70.5	52.7	37.8	27.0	24.0	90.6	17.2
$\text{CNN}_{\mathcal{L}}$+RHN	72.3	55.3	**41.3**	**30.6**	**25.2**	98.9	**18.3**
LSTM	70.8	53.6	39.5	29.2	24.5	92.6	17.1
$\text{CNN}_{\mathcal{L}}$+LSTM	72.1	54.6	40.9	30.4	25.1	**99.1**	18.0
GRU	71.6	54.1	39.7	28.9	24.3	93.3	17.2
$\text{CNN}_{\mathcal{L}}$+GRU	**72.6**	**55.4**	41.1	30.3	24.6	96.1	17.6

Performance comparison on Flickr30k

Approach	B@1	B@2	B@3	B@4	M	C	S
Simple RNN	60.5	41.3	28.0	19.1	17.1	32.5	10.5
$\text{CNN}_{\mathcal{L}}$+RNN	71.3	53.8	39.6	28.7	**22.6**	**65.4**	**15.6**
RHN	62.1	43.1	29.4	20.0	17.7	38.4	11.4
$\text{CNN}_{\mathcal{L}}$+RHN	**73.8**	**56.3**	**41.9**	**30.7**	21.6	61.8	15.0
LSTM	60.9	41.8	28.3	19.3	17.6	35.0	11.1
$\text{CNN}_{\mathcal{L}}$+LSTM	64.5	45.8	32.2	22.4	19.0	45.0	12.5
GRU	61.4	42.5	29.1	20.0	18.1	39.5	11.4
$\text{CNN}_{\mathcal{L}}$+GRU	71.4	54.0	39.5	28.2	21.1	57.9	14.5

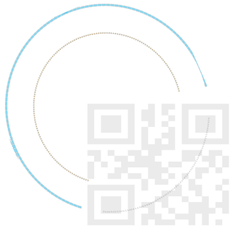
Comparison with State-of-the-art Methods.

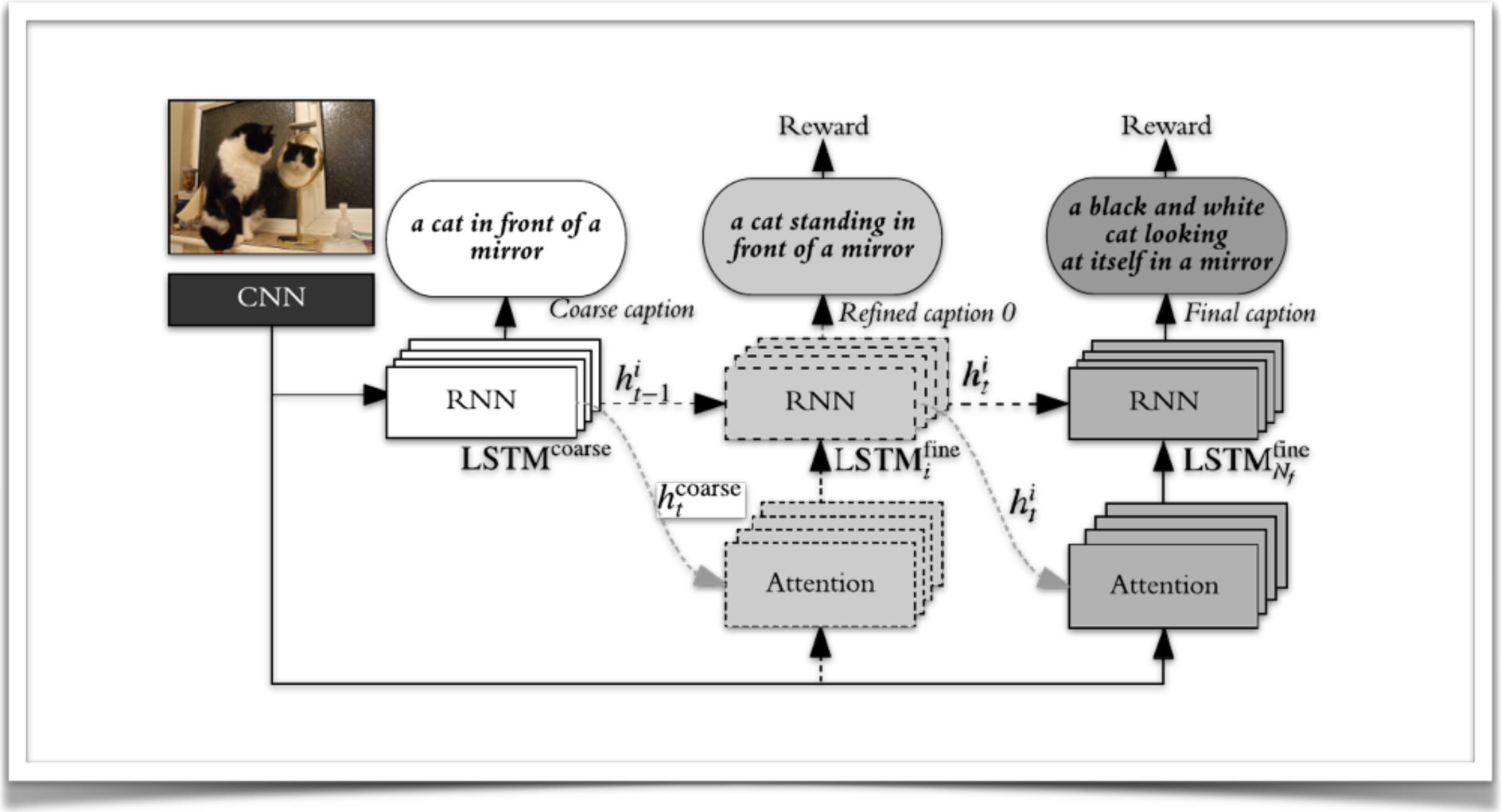
	Flickr30k					MS COCO					
Approach	B@1	B@2	B@3	B@4	M	B@1	B@2	B@3	B@4	M	C
m-RNN (Mao et.al., ICLR 2015)	60.0	41.0	28.0	19.0	—	67.0	49.0	35.0	25.0	—	—
Soft-Attention (Xu et.al., ICML 2015)	66.7	43.4	28.8	19.1	18.5	71.8	50.4	35.7	25.0	23.0	—
ATT-FCN (You et.al, CVPR 2016)	64.7	46.0	32.4	23.0	18.9	70.9	53.7	40.2	30.4	24.3	—
VAE (Pu et.al., NIPS 2016)	72.0	53.0	38.0	25.0	—	72.0	52.0	37.0	28.0	24.0	90.0
Google NICv2 (Vinyals et.al., PAMI 2017)	—	—	—	—	—	—	—	—	32.1	25.7	99.8
Attributes-CNN+RNN (Wu et.al., CVPR 2016)	73.0	55.0	40.0	28.0	—	74.0	56.0	42.0	31.0	26.0	94.0
$\text{CNN}_{\mathcal{L}}$+RNN	71.3	53.8	39.6	28.7	22.6	72.2	55.0	40.7	29.5	24.5	95.2
$\text{CNN}_{\mathcal{L}}$+RHN	73.8	56.3	41.9	30.7	21.6	72.3	55.3	41.3	30.6	25.2	98.9
$\text{CNN}_{\mathcal{L}}$+LSTM	64.5	45.8	32.2	22.4	19.0	72.1	54.6	40.9	30.4	25.1	99.1
$\text{CNN}_{\mathcal{L}}$+GRU	71.4	54.0	39.5	28.2	21.1	72.6	55.4	41.1	30.3	24.6	96.1



Drawbacks

- Without strict theoretical proof (that's the reason why we titled as "An Empirical Study … ").
- Rich fine-grained descriptions.
  - The existing image captioning approaches typically train a one-stage sentence decoder, which is difficult to generate rich fine-grained descriptions.
  - On the other hand, multi-stage image caption model is hard to train due to the vanishing gradient problem.
- Exposure bias problem.
  - The sentence decoder is trained to predict a word given the previous ground-truth words, while at testing time, the caption generation is accomplished by greedy search or with beam search, which predicts the next word based on the previously generated words (that is different from the training mode). Since the model has never been exposed to its own predictions, it will result in error accumulation at test time.
- Loss-evaluation mismatch problem.
  - Specifically, language models are usually trained to minimize the cross-entropy loss at each time-step, while at testing time, we evaluate the generated captions with the sentence-level evaluation metrics, e.g., BLEU-n, CIDEr, SPICE, etc., which are non-differentiable and cannot be directly used as training loss.





Stack-Captioning: Coarse-to-Fine Learning for Image Captioning

Jiuxiang Gu, Jianfei Cai, Gang Wang, Tsuhan Chen

*Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence (**AAAI**). New Orleans, USA. February 2–7, 2018.*

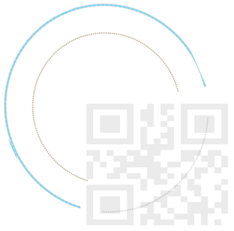
Challenges

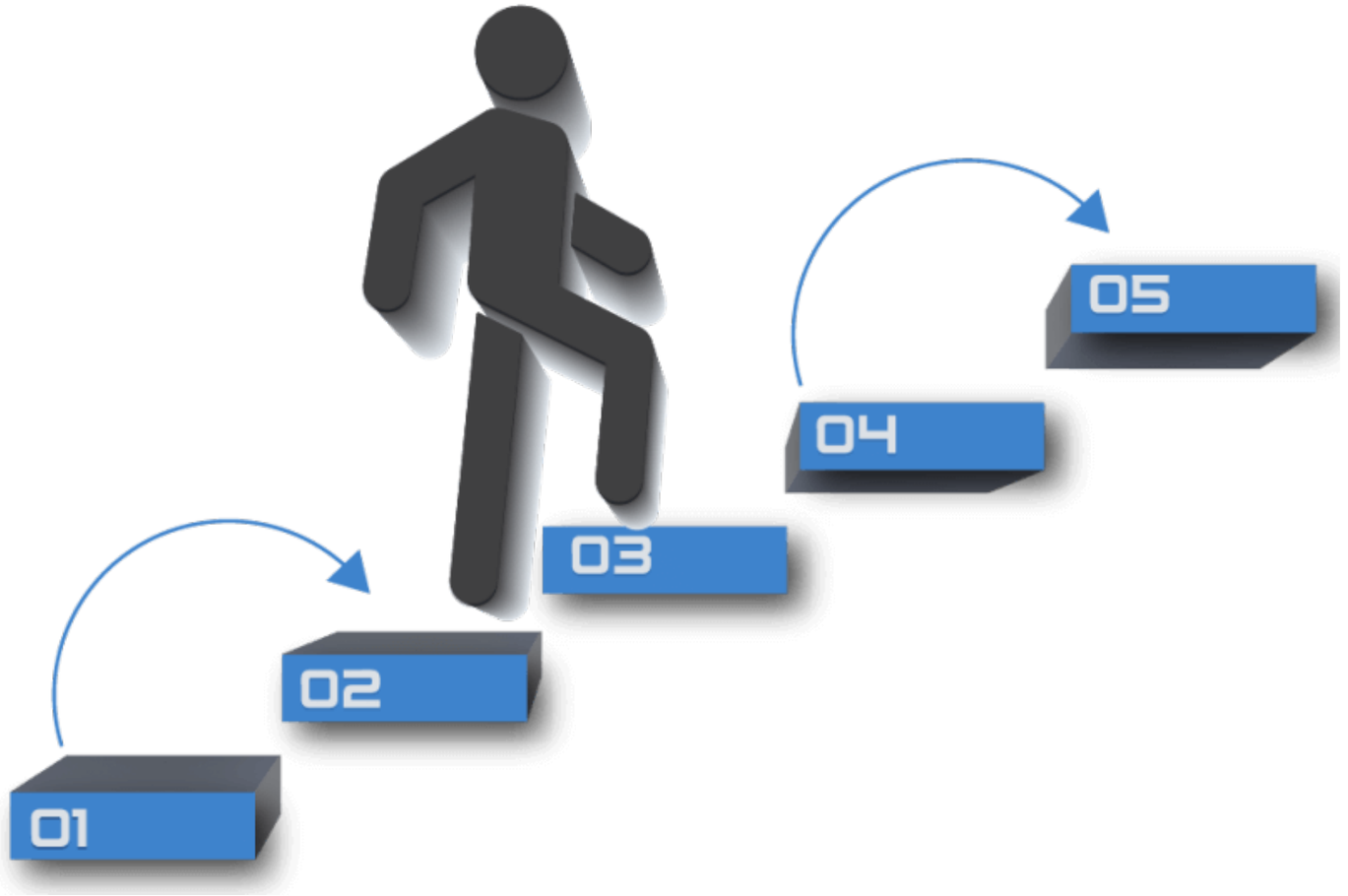
1. One-Stage image captioning approaches

- It is extremely hard for them to generate rich fine-grained descriptions.
  - *High-complexity* models
  - Back-propagated *gradients diminish*

2. Exposure bias between the training and the testing

- **Scheduled Sampling** (*Training*) vs. **Greedy Decoding / Beam Search** (*Testing*)
  - In training, sentence decoder is trained to predict a word *given the previous ground-truth words*;
  - In testing, sentence decoder predicts the next word based on the *previously generated words* that is different from the training mode.
- **Cross-Entropy Loss** (*Training*) vs. **Sentence-level Scores**(*Testing*)
  - Language models are usually trained to minimize the *cross-entropy loss* at each time-step, while at testing time, we evaluate the generated captions with the *sentence-level evaluation metrics*.



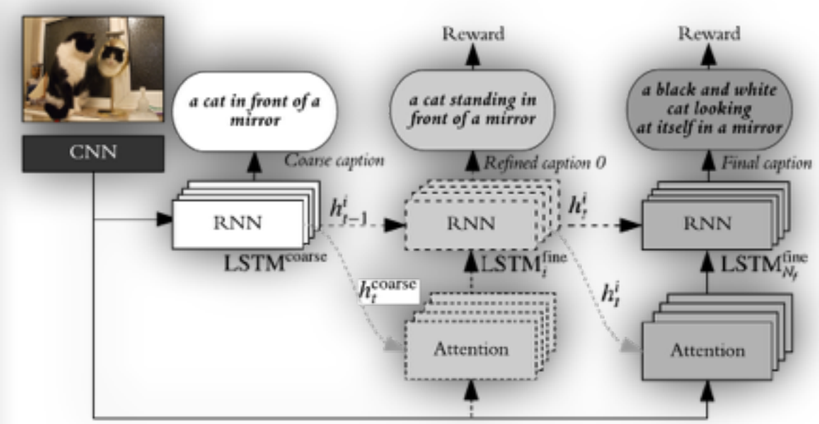


1. Step-by-Step/Coarse-to-Fine

2. Self-Critical Training/Reinforcement Learning

Motivation

- **Coarse-to-fine learning**
  - Our approach via the stacked attention can consider visual information in the image from coarse to fine, aligning well with the human visual system, where we usually use a coarse-to-fine procedure to understand pictures.
  - A stacked attention model to filter out noises gradually and pinpoint the regions that are highly relevant to the word prediction.
  - Our proposed learning approach addresses the difficulty of vanishing gradients during training by providing a learning objective function that enforces intermediate supervisions.
- **Reinforcement learning**
  - A reinforcement learning method that directly optimizes model with the normalized intermediate rewards.



$$o_t^0, h_t^0 = \mathrm{LSTM}_{\mathrm{coarse}}(h_{t-1}^0, i_t^0, y_{t-1}) \quad (1)$$

$$i_t^0 = [f(\mathbf{V}); h_{t-1}^{N_f}] \quad (2)$$

$$o_t^i, h_t^i = \mathrm{LSTM}_{\mathrm{fine}}^i(h_{t-1}^i, i_t^i, y_{t-1}) \quad (3)$$

$$i_t^i = [g(\mathbf{V}, \boldsymbol{\alpha}_t^{i-1}, h_t^{i-1}); h_t^{i-1}] \quad (4)$$

$$g(\mathbf{V}, \boldsymbol{\alpha}_t^{i-1}, h_t^{i-1}) = \sum_{n=0}^{k\times k-1} \alpha_t^{i,n} \cdot (\mathbf{W}_{v\alpha}^i V_n + \mathbf{b}_{v\alpha}^i) \quad (5)$$

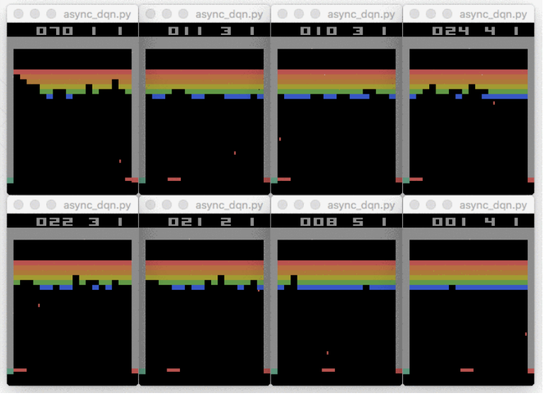
MOTIVATION

❖ Coarse-to-Fine

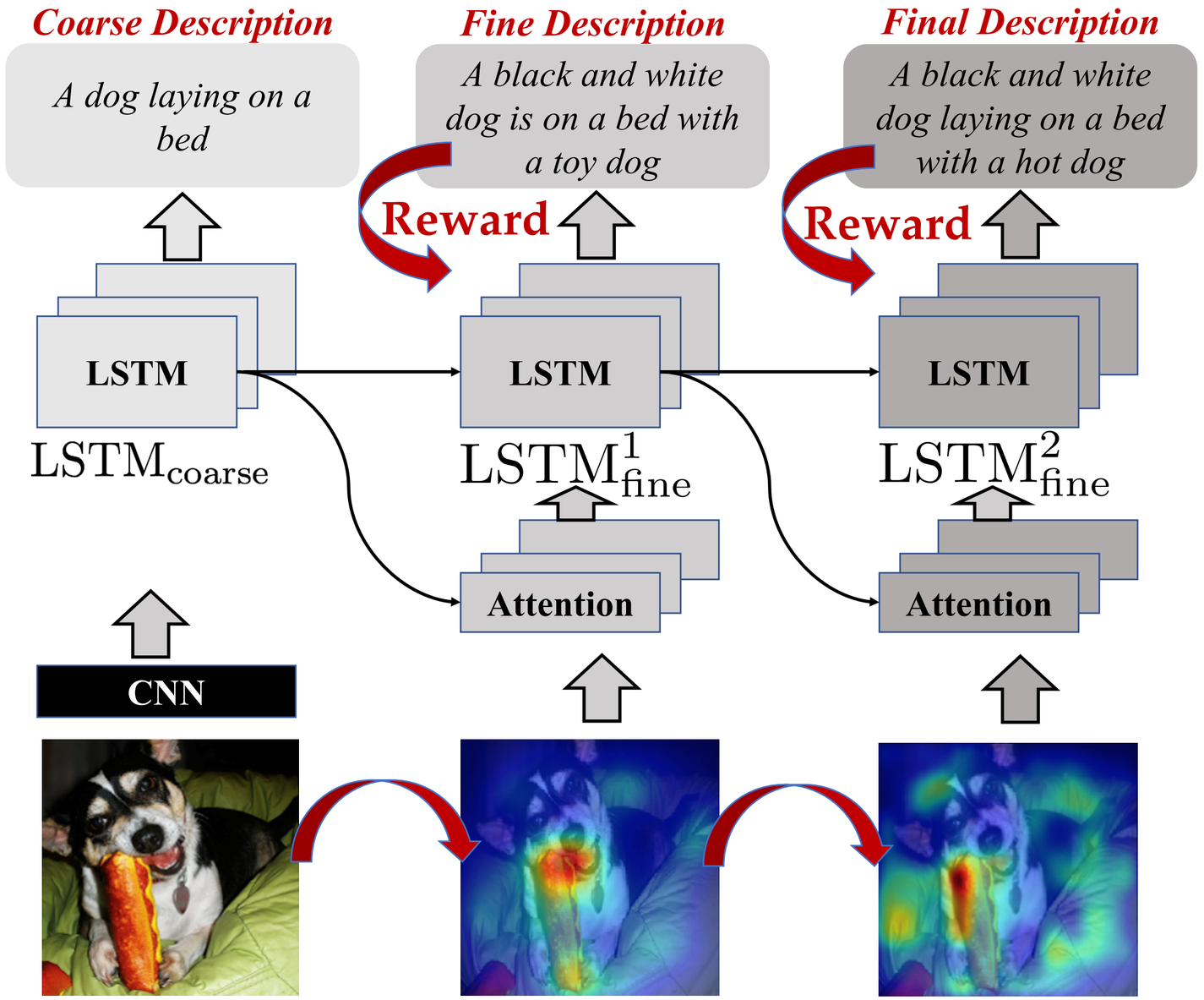
- Our approach via the stacked attention can consider visual information in the image from coarse to fine, aligning well with the *human visual system*, where we usually use a *coarse-to-fine procedure to understand pictures*.
- A stacked attention model to *filter out noises gradually* and pinpoint the regions that are highly relevant to the word prediction.
- Our proposed learning approach addresses the difficulty of vanishing gradients during training by providing *a learning objective function that enforces intermediate supervisions*.

❖ Reinforcement learning

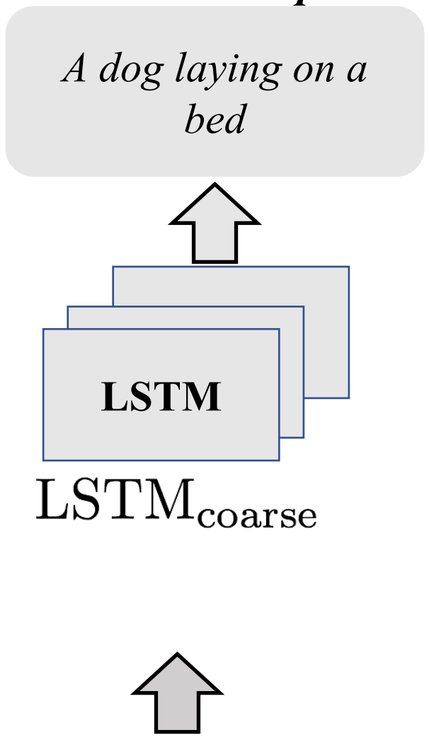
- A reinforcement learning method that *directly* optimizes model with the normalized *intermediate rewards.*



Methodology



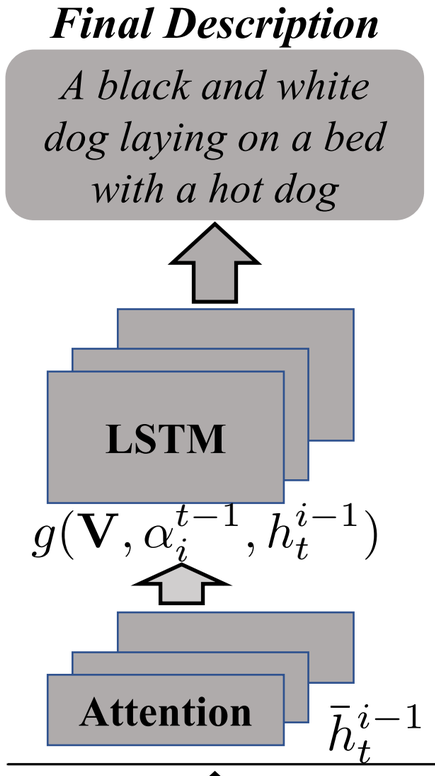
Methodology (Coarse-Decoder)

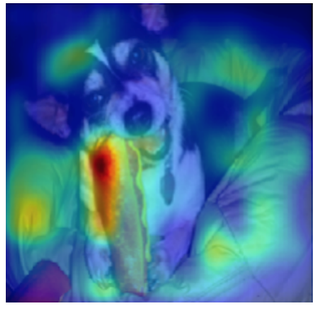


- **Coarse Decoder**
  - We start by decoding in a *coarse search space* in the first stage;
  - The operation of the LSTM$_{coarse}$ can be described as:
    $$o_t^0, h_t^0 = \text{LSTM}_{coarse}(h_{t-1}^0, i_t^0, y_{t-1})$$
    $$i_t^0 = [f(\mathbf{V}); h_{t-1}^{N_f}]$$
  - where h_{t-1}^0 and $h_{t-1}^{N_f}$ are the hidden states;
  - $y_{t-1} = \boldsymbol{W}_e\, Y_{t-1}$ is the embedding of previous word Y_{t-1};
  - $\hat{Y}_t^0 \sim \text{Softmax}(\boldsymbol{W}_o^0 o_t^0 + b_o^0)$ is *decoded word drawn from the dictionary according* to the o_t^0.

- **Encoder**
  - For coarse-stage, we take a *mean-pooling* over the *spatial image features*. The global image feature is:
    $$f(\mathbf{V}) = \frac{1}{k \times k} \sum_{i=0}^{k \times k - 1} V_i$$
  - where $\mathbf{V} = \text{CNN}(\mathbf{I})$.

Methodology (Fine-Decoder)



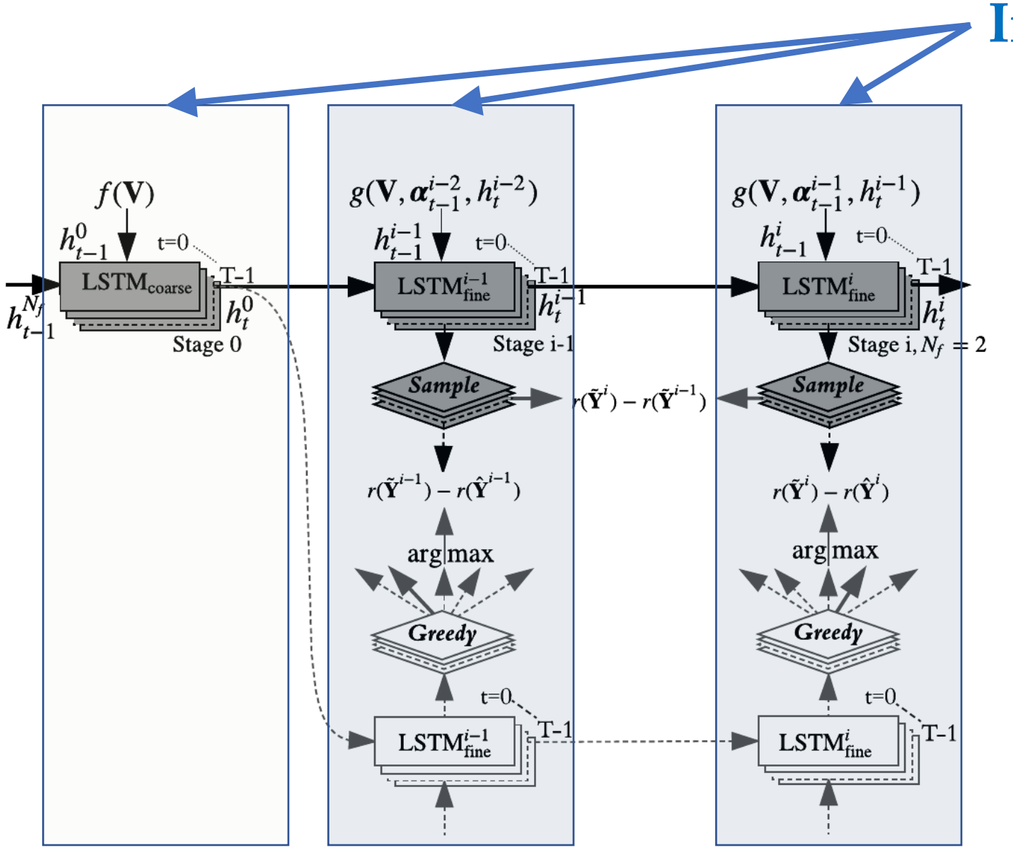


- **Fine Decoder**
  - Each *fine decoder* consists of an $\text{LSTM}_{\text{fine}}$ network and an *attention* model;
  - The updating procedure of $\text{LSTM}_{\text{fine}}$ can be written as: $o_t^i, h_t^i = \text{LSTM}_{\text{fine}}^i(h_{t-1}^i, i_t^i, y_{t-1})$
    $i_t^i = [\text{g}(\mathbf{V}, \boldsymbol{\alpha}_i^{t-1}, h_t^{i-1}); h_{t-1}^{i-1}]$
  - where h_{t-1}^{i-1} is are the hidden state of fine decoder;
  - In each fine stage i, our attention model operates on both image features V and *attention weights* $\boldsymbol{\alpha}_i^{t-1}$ from the preceding stage, the *new* attended feature is $\text{g}(\mathbf{V}, \boldsymbol{\alpha}_i^{t-1}, h_t^{i-1})$.

- **Stack-Attention**
  - $\text{g}(\mathbf{V}, \boldsymbol{\alpha}_i^{t-1}, h_t^{i-1}) = \sum_{n=0}^{k\times k-1} \alpha_t^{i,n}(\mathbf{W}_{v\alpha}^i V_n + \mathbf{b}_{v\alpha}^i)$;
  - *Attention probability*: $\boldsymbol{\alpha}_i^i = \text{Softmax}(\mathbf{W}_\alpha^i \boldsymbol{A}_t^i + \mathbf{b}_\alpha^i)$
  - $A_t^{i,n} = \tanh(\mathbf{W}_{va}^i V_n + \mathbf{W}_{ha}^i \bar{h}_t^{i-1})$
  - $\bar{h}_t^{i-1} = h_t^{i-1} + \sum_{n=0}^{k\times k-1} \alpha_t^{i-1,n}(\mathbf{W}_{v\alpha}^{i-1} V_n + \mathbf{b}_{v\alpha}^{i-1})$

Training (1. cross-entropy loss)

- We first *incorporate supervised training objectives to the intermediate layers*.
- Each stage of the coarse-to-fine sentence decoder is trained to *predict the words repeatedly*.



- We first training the network by defining a *loss function for each stage* i that minimizes the ***cross-entropy (XE) loss***:

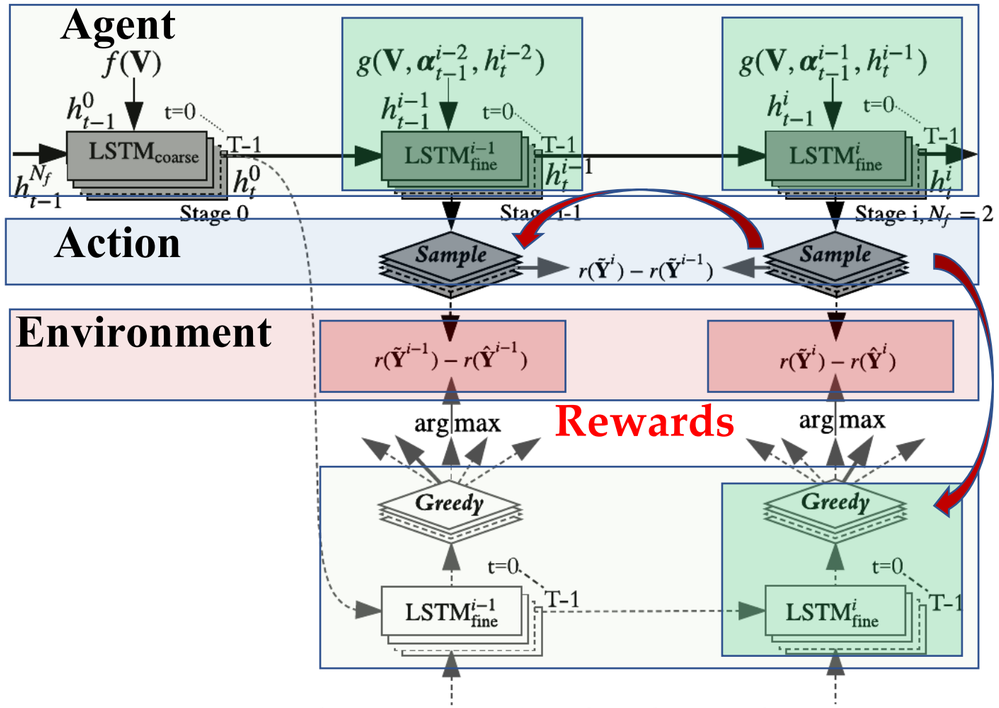
$$\mathcal{L}_{\text{XE}}^{i}(\theta_{0:i}) = -\sum_{t=0}^{T-1} \log(p_{\theta_{0:i}}(Y_t \mid Y_{0:t-1}, \mathbf{I})),$$

$$\mathcal{L}_{\text{XE}}(\theta) = \sum_{i=0}^{N_f} \mathcal{L}_{\text{XE}}^{i}(\theta_{0:i})$$

$$= -\sum_{i=0}^{N_f} \sum_{t=0}^{T-1} \log(p_{\theta_{0:i}}(Y_t \mid Y_{0:t-1}, \mathbf{I}))$$

Training (2. REINFORCE-based approach)

❑ After XE training, we run the proposed RL-based approach on the just trained model to be optimized for the ***CIDEr rewards***, where rewards are introduced at each stage as *intermediate* supervision.



- We ***suppress*** those ***samples*** that have the worse scores than the ***greedy decoding*** results.
- The second term ***increases*** the probability of the samples from ***stage i*** that outperform the samples from ***stage i-1***, and ***suppresses*** the ***inferior*** samples.

- Then, the goal of RL-based training is to minimize the ***negative expected rewards (punishments)*** of multi-stages, :

$$\mathcal{L}_{\text{RL}}(\theta) = -\sum_{i=1}^{N_f} \mathbb{E}_{\tilde{\mathbf{Y}}^i \sim p_{\theta_{0:i}}}[r(\tilde{\mathbf{Y}}^i)] \approx -\sum_{i=1}^{N_f} r(\tilde{\mathbf{Y}}^i)$$

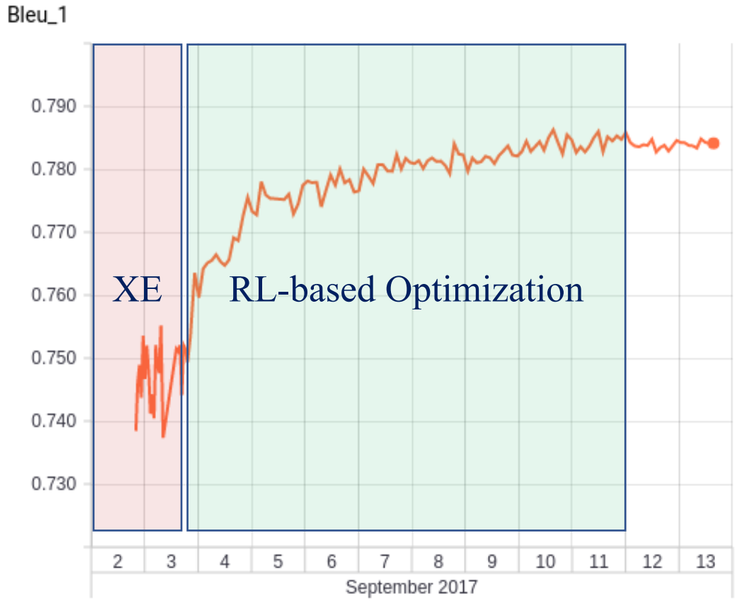
$$\nabla_\theta \mathcal{L}_{\text{RL}}(\theta) = \sum_{i=1}^{N_f} \nabla_{\theta_{0:i}} \mathcal{L}_{\text{RL}}(\theta_{0:i})$$

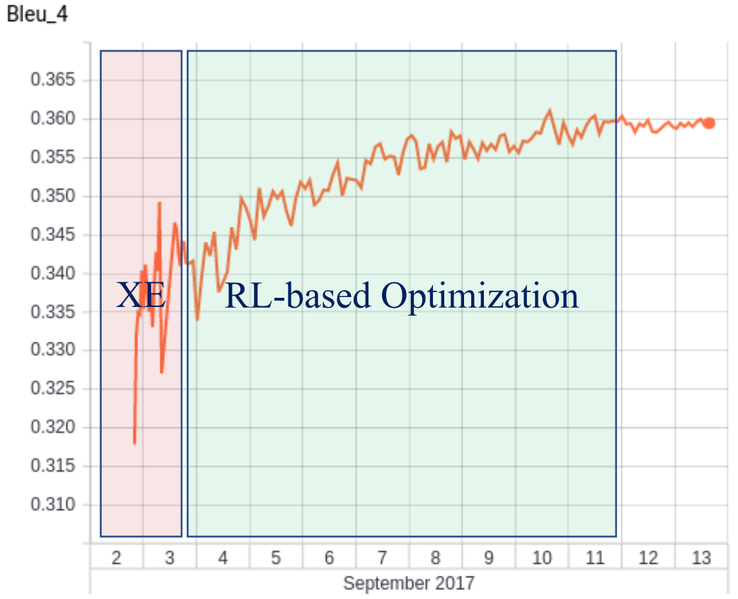
$$\approx -\sum_{i=1}^{N_f} r(\tilde{\mathbf{Y}}^i) \cdot \nabla_{\theta_{0:i}} \log p_{\theta_{0:i}}(\tilde{\mathbf{Y}}^i)$$

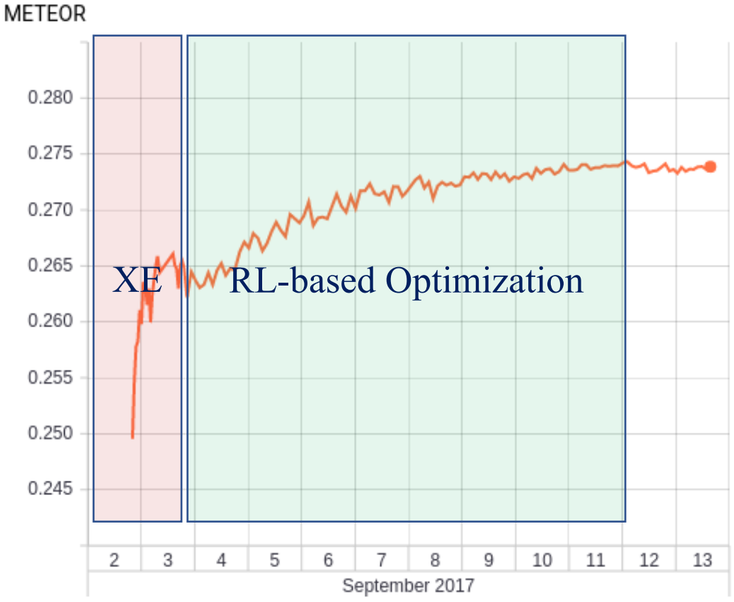
$$\nabla_\theta \mathcal{L}_{\text{RL}}(\theta) \approx -\sum_{i=1}^{N_f} \Delta r(\tilde{\mathbf{Y}}^i) \cdot \nabla_{\theta_{0:i}} \log p_{\theta_{0:i}}(\tilde{\mathbf{Y}}^i)$$

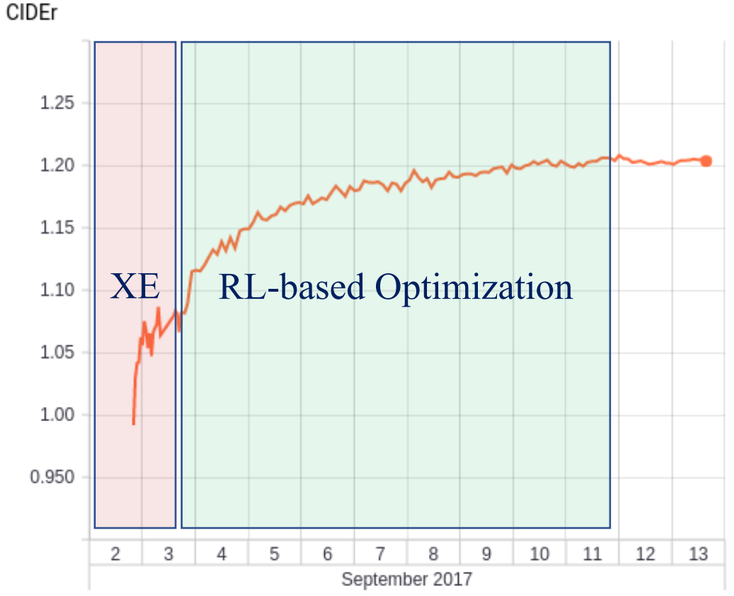
$$\Delta r(\tilde{\mathbf{Y}}^i) = \left[r(\tilde{\mathbf{Y}}^i) - r(\hat{\mathbf{Y}}^i) \right] + \left[r(\tilde{\mathbf{Y}}^i) - r(\tilde{\mathbf{Y}}^{i-1}) \right]$$

Experiments







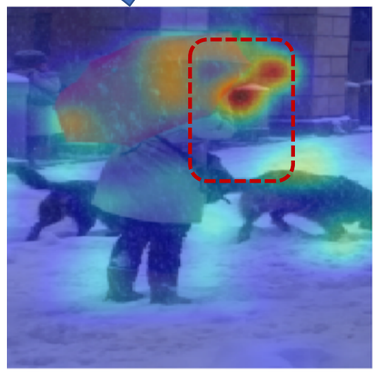


Experiments (Qualitative Analysis)

- *a woman with a red umbrella is walking two dogs in the snow*
- *a person with a white umbrella with two dogs*
- *a woman is walking her dogs on the city sidewalks through the newly fallen snow*
- *a person with an umbrella and two dogs walking in the snow*
- *a woman is walking two dogs in the snow*

a woman walking in the snow

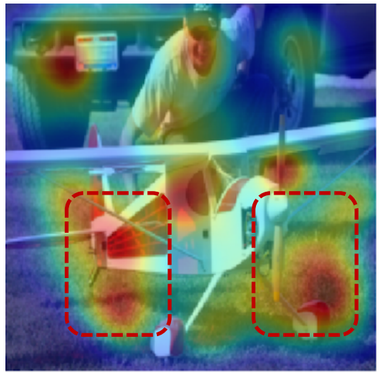
a woman walking a dog in the snow



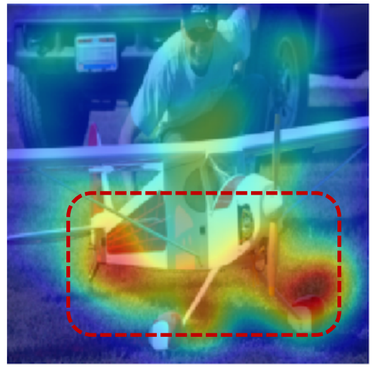
a woman walking a dog in the snow with an umbrella

- *man bending down to check out a model plain that is parked in the grass*
- *man behind a radio operated model airplane on the ground*
- *the man is crouched down with a small airplane model*
- *a small red and white toy plane in the grass*
- *a man smiles while kneeling beside a miniature airplane*

a man is on front grass with a toy



a man is on front grass with a toy truck



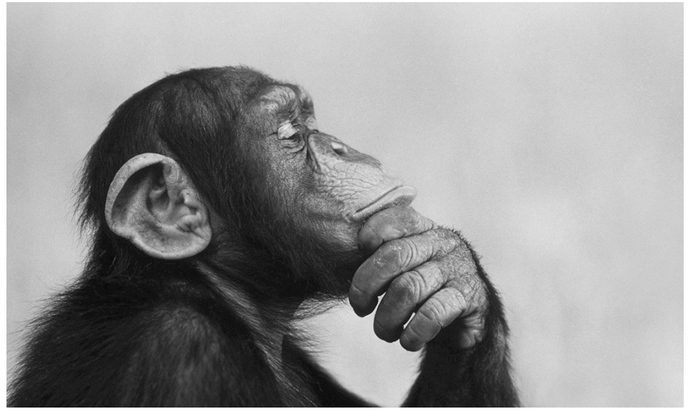
a man sitting in the grass with a toy plane

Look (computer vision) + **Talk** (natural language processing)
= **True** AI ?

Look (computer vision) + **Talk** (natural language processing) does mean **Think**

+

≠



END