中国科学院大学

University of Chinese Academy of Sciences

Deep Learning

Application in Natural Language Processing

Xinfeng Zhang（张新峰）

School of Computer Science and Technology

University of Chinese Academy of Sciences

Email: xfzhang@ucas.ac.cn

计算机科学与技术学院

SCHOOL OF COMPUTER SCIENCE AND TECHNOLOGY

提纲

- 语言模型
- 机器翻译
- 机器阅读理解
- 自动摘要
- 图像描述
- 中英文术语对照

中国科学院大学
University of Chinese Academy of Sciences

1 语言模型

语言模型是什么

❑ 例子：

给定拼音串：

“ta shi yan jiu sheng wu de ”

可能的汉字串：

“踏实研究生物的”

“他实验救生物的”

“他使烟酒生物的”

“他是研究生物的”

语言模型是什么

- 语言模型的核心思想是按照特定的训练方式，从语料中提取所蕴含的语言知识，应用于词序列的预测
- 语言模型通常可以分为基于规则的语言模型和统计语言模型
- 统计语言模型处于主流地位，通过对语料库的统计学习，归纳出其中的语言知识，获得词与词之间的连接概率，并以词序列的概率为依据来判断其是否合理

为什么需要语言模型

- 语言是人类最重要的、最有效的一种信息交流的手段，也是人类进行观点、思想及情感交流最便捷、最自然的方式之一，在信息传递中发挥着重要的作用
- 统计语言模型在语音识别、机器翻译、中文分词、问答系统等自然语言处理领域中取得了成功地应用
  - 例如：“厨房里食油用完了”和“厨房里石油用完了”
- 近年来，随着深度学习技术的不断发展，神经网络语言模型已成为目前语言模型领域的主流

语言模型的发展过程

基于文法的语言模型

- 基于文法的语言模型是依据语法规则，由计算机根据这些语法解析文本的含义，其中语法规则来源于语言学家掌握的语言学知识和领域知识
- 弊端：
  - 不能处理大规模真实文本
  - 需要大量专家知识

统计语言模型

- 统计语言模型中，对于特定顺序排列的词序列 $S=\{w_1,w_2,\cdots,w_n\}$，其先验模型有以下计算公式：

$$\begin{aligned} P(s) &= P(w_1 w_2 \cdots w_N) \\ &= P(w_1 w_2 \cdots w_{N-1}) P(w_N \mid w_1 w_2 \cdots w_{N-1}) \\ &= P(w_1) P(w_2 \mid w_1) \cdots P(w_N \mid w_1 w_2 \cdots w_{N-1}) \end{aligned}$$

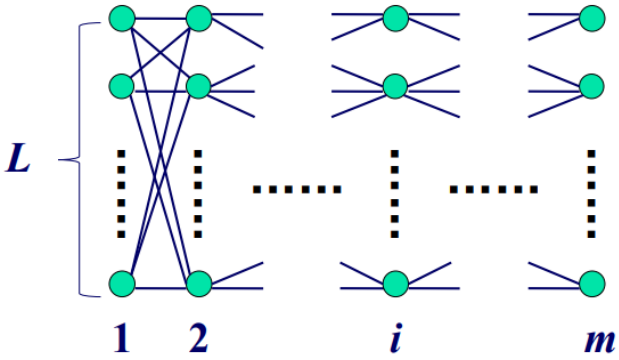
- 回到开头提到的音字转换例子，运用统计语言模型有如下分析过程：

$$\boldsymbol{p}(\boldsymbol{S1}) = \boldsymbol{p}(\text{踏实}|\boldsymbol{B}) \times \boldsymbol{p}(\text{研究}|\text{踏实}) \times \boldsymbol{p}(\text{生物}|\text{踏实研究}) \times \boldsymbol{p}(\text{的}|\text{踏实研究生物})$$

$$\boldsymbol{p}(\boldsymbol{S3}) = \boldsymbol{p}(\text{他}|\boldsymbol{B}) \times \boldsymbol{p}(\text{是}|\text{他}) \times \boldsymbol{p}(\text{研究}|\text{他是}) \times \boldsymbol{p}(\text{生物}|\text{他是研究}) \times \mathbf{p}(\text{的}|\text{他是研究生物})$$

统计语言模型

- 随着历史基元数量的增加，不同的“历史”按指数级增长。对于第 i (i >1) 个统计基元，历史基元的个数为 i-1，如果共有 L 个不同的基元，如词汇表，理论上每一个单词都有可能出现在1到 i-1的每一 个位置上，那么，第i个基元就有 $L^{(i-1)}$ 种不同的历史情况
- 我们必须考虑在所有的$L^{(i-1)}$种不同历史情况下产生第 i 个基元的概率



如果 L=5000, m = 3, 自由参数的数目为 1250 亿！

N-gram语言模型

- **N-gram语言模型不再将所有的上文作为条件，而是只考虑当前词之前的前n-1个单词,计算公式如下：**

$$p(w_i|w_{i-+1},\ldots,w_{i-1}) = \prod_{i=1}^{m} \mathrm{p}\left(w_i \middle| w_{i-n+1}^{i-1}\right) \approx \frac{c(w_{i-n+1}^{i})}{c(w_{i-n+1}^{i-1})}$$

- 其中，$c(w_{i-n+1}^{i-1})$是历史串在给定语料中出现的次数。分子为历史串w_{i-n+1}^{i-1}与当前词w_i同时出现的次数
- n=1，即出现在第i位上的基元w_i独立于历史，成为一元文法，或者uni-gram;
- n=2，2-gram(bi-gram)被称为1阶马尔可夫链；
- n=3，3-gram(tri-gram)被称为2阶马尔可夫链；

N-gram语言模型

❑ 零概率事件

<BOS>John read Moby Dick<EOS>

<BOS>Mary read a different book<EOS>

<BOS>She read a book by Cher<EOS>

❑ 根据2元文法求句子“Cher read a book”的概率过程如下：

$$p(\text{Cher read a book}) = p(\text{Cher}|<\text{BOS}>) \times p(\text{read}|\text{Cher}) \times$$

$$p(\text{a}|\text{read}) \times p(\text{book}|\text{a}) \times p(<\text{EOS}>|\text{book}) = \frac{c(<BOS>cher)}{c(<BOS>)} \times$$

$$\frac{c(cher\ read)}{c(cher)} \times \frac{c(read\ a)}{c(read)} \times \frac{c(a\ book)}{c(a)} \times \frac{c(book<EOS>)}{c(book)} = 0 \times 0 \times \frac{2}{3} \times \frac{1}{2} \times \frac{1}{2}$$

N-gram语言模型

❑ 数据平滑

- 基本思想：调整最大似然估计的概率值，使零概率增值，使非零概率下调，“劫富济贫”，消除零概率，改进模型的整体正确率
- 基本目标：测试样本的语言模型困惑度越小越好
- 基本约束：

$$\sum_{w_i} p(w_i|w_1, w_2, \cdots, w_{i-1}) = 1$$

N-gram语言模型

❑ 数据平滑方法

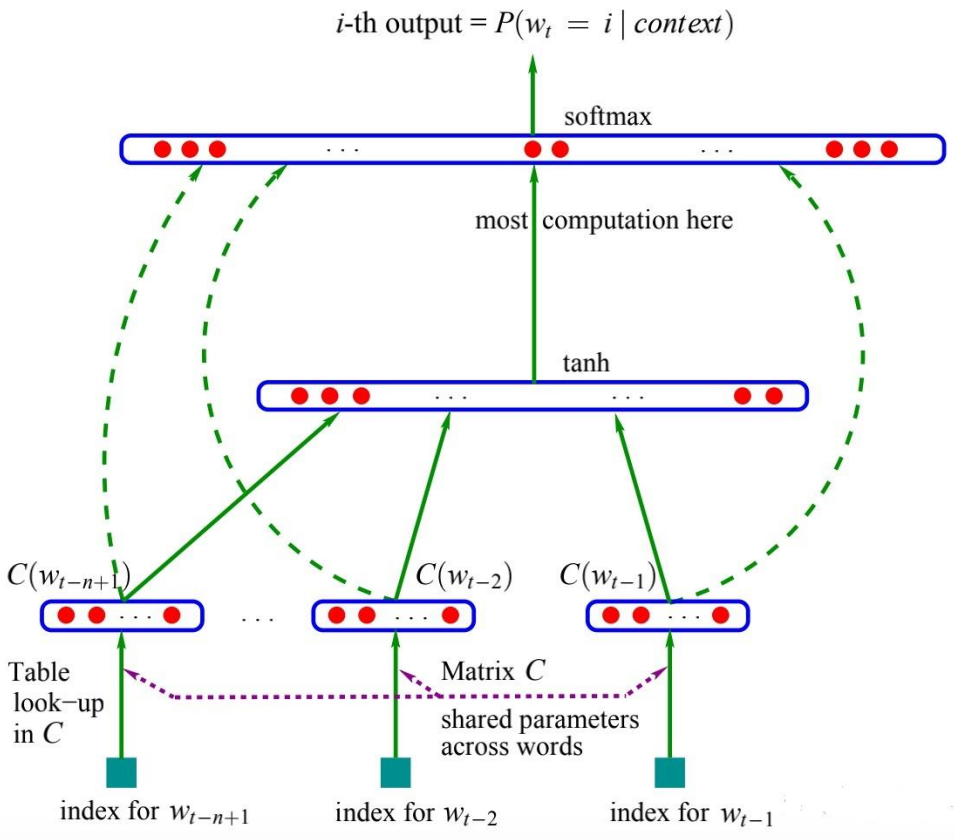
- 加法平滑(additive smoothing)：每种情况的出现次数增加一个固定值
- 折扣平滑(discounting smoothing)：修改实际发生次数，使得不同情况发生概率总和小于1，剩余概率平均分给未出现的情况
- 删除插值(deleted interpolation): 用低阶模型估计高阶模型，并将它们进行加权
- Kneser-Ney平滑：是目前最先进的平滑技术，它是上述技术的综合

神经网络语言模型

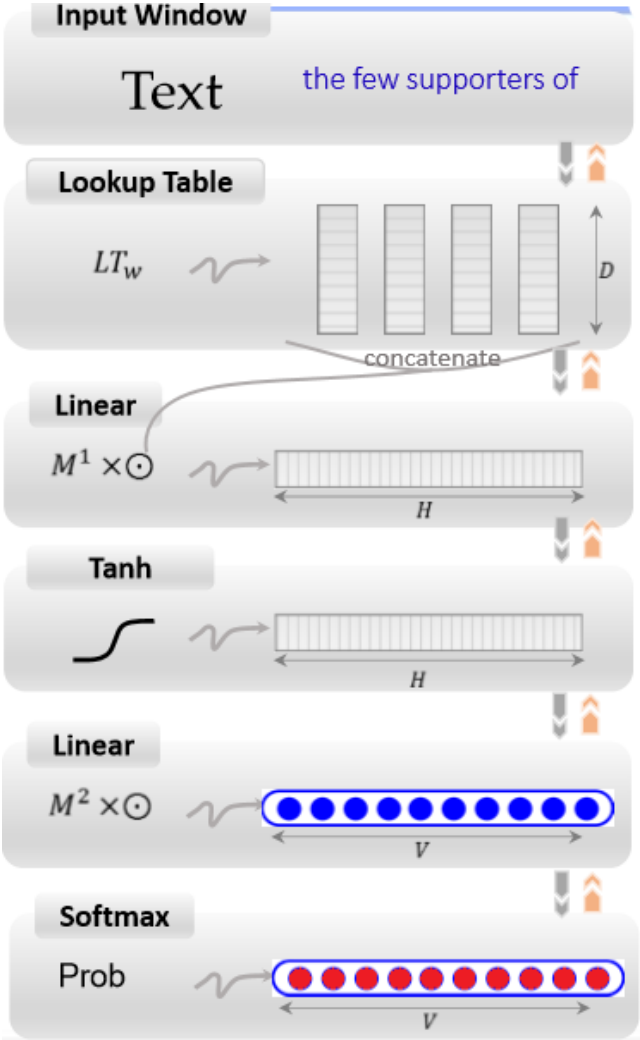
- 前馈神经网络语言模型
- 循环神经网络语言模型
- 长短时记忆神经网络语言模型

前馈神经网络语言模型

- 全连接（Fully Connected Neural Network）神经网络，是最早被引入到语言建模中的神经网络结构
  - 2003年 Bengio 提出



前馈神经网络语言模型



$P(this|the, few, supporters, of)$

将每个词通过词向量矩阵L映射为低维实数向量

of → *(0.23, 0.15, 0.08, 0.31, .., 0.42)*

拼接所有词的向量，形成一个向量

隐藏层:
线性映射+非线性变换

⋮

***Softmax** 输出层:*

$P(this|the, few, supporters, of)$

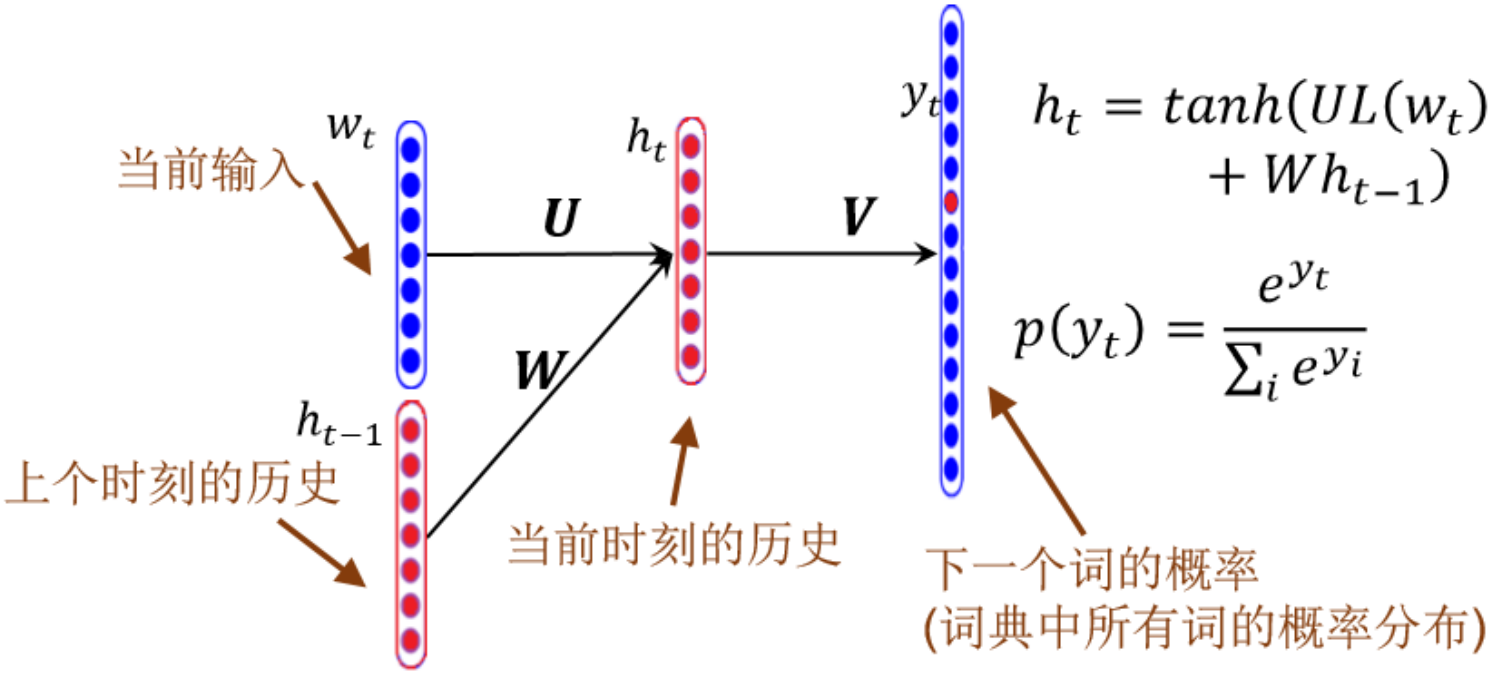
前馈神经网络语言模型

- **缺点：**
  - 参数过多，参数$\theta=\{b,d,W,U,H,C\}$
    - b 输出层偏置
    - d 隐藏层偏置
    - U 隐藏层到输出层的权重
    - W 词特征层到输出层的权重
    - H 隐藏层权重
  - 只能利用前n个词来预测下一个词

循环神经网络语言模型

- 输入：t-1时刻的历史状态$h_{(t-1)}$，t时刻的输入w_t
  - Word Embeddings $w_t = Ex^{(t)}$

- 输出：t时刻的历史状态h_t，下一个词的概率

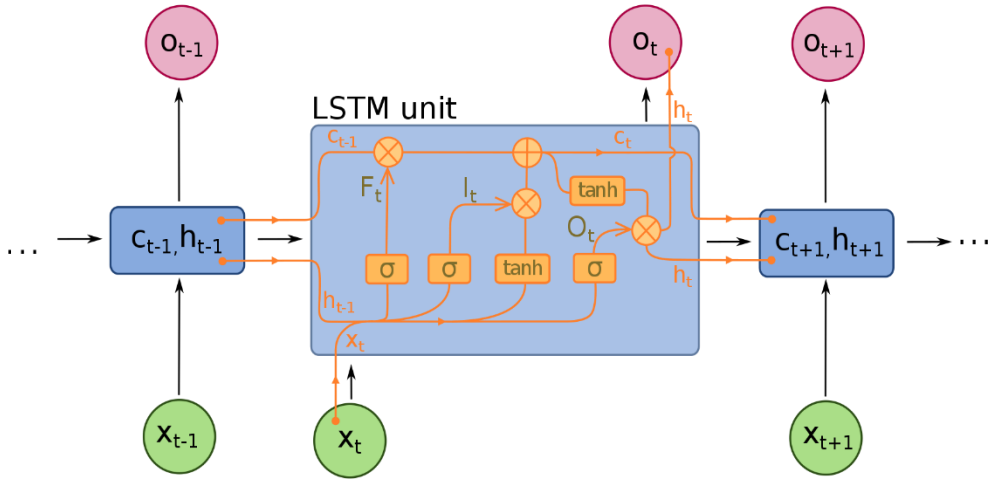


循环神经网络语言模型

- 缺点：理论上可以记录全部的历史信息，但是由于“梯度消失”和“梯度爆炸”现象，在实践中，循环神经网络语言模型并不具备这种处理“长期依赖”的能力
- 解决策略：解决梯度消失和梯度爆炸的方法——有选择的保留和遗忘*t*时刻的信息

LSTM神经网络

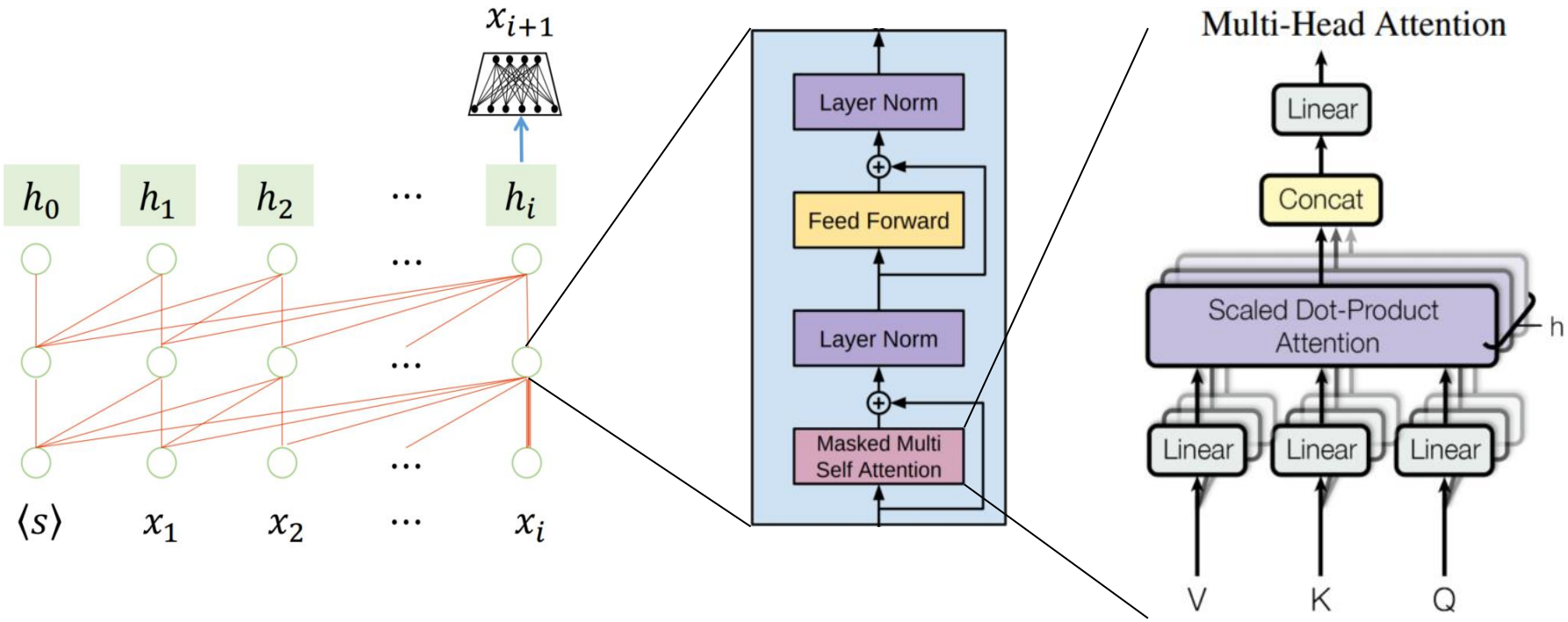
- 长短时记忆神经网络（Long Short Term Memory Networks, LSTM）具备处理长期依赖的能力，目前LSTM已经被广泛应用在语言模型中



$$i_t = \sigma(U_i x_t + W_i s_{t-1} + V_i c_{t-1} + b_i)$$
$$f_t = \sigma(U_f x_t + W_f s_{t-1} + V_f c_{t-1} + b_f)$$
$$g_t = f(U x_t + W s_{t-1} + V c_{t-1} + b)$$
$$c_t = f_t \odot c_{t-1} + i_t \odot g_t$$
$$o_t = \sigma(U_o x_t + W_o s_{t-1} + V_o c_t + b_o)$$
$$s_t = o_t \cdot f(c_t)$$
$$y_t = g(V s_t + M x_t + d)$$

自注意力机制语言模型

- 目前最先进的语言模型结构，它是完全基于注意力机制的语言模型结构



中国科学院大学
University of Chinese Academy of Sciences

2 机器翻译

机器翻译的概念

- **机器翻译 (Machine Translation, MT)是用计算机把一种语言(源语言, source language)翻译成另一种语言(目标语言, target language)的一门技术**

机器翻译的发展史

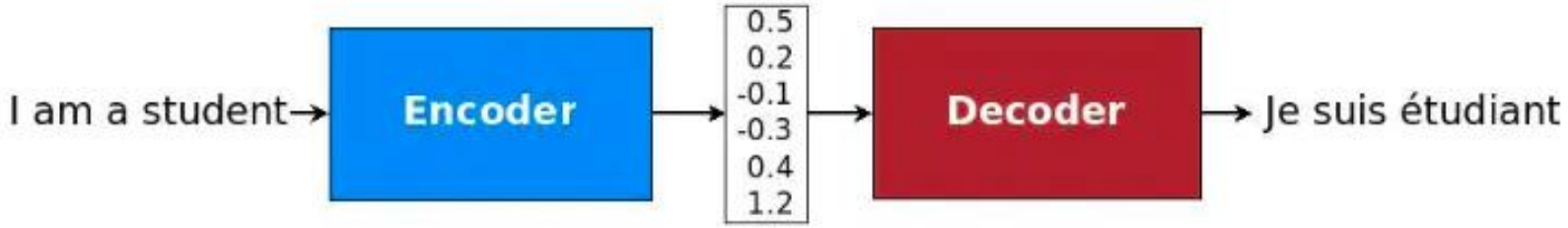
- 1949年，W. Weaver正式提出机器翻译问题
- 1954年，Georgetown大学在IBM协助下，用IBM-701计算机实现了世界上第一个MT系统，实现俄译英翻译
- 1990年，IBM提出统计机器翻译模型，机器翻译研究进入了繁荣时期
- 2014年，神经网络机器翻译被提出，机器翻译研究进入了新的突破时期

Seq2Seq模型

- 在统计翻译模型中，模型的训练步骤分为：预处理，词对齐，短语对齐，抽取短语特征，训练语言模型，学习特征权重等步骤
- Encoder-Decoder模型是使用神经网络进行机器翻译的基本方法，一般也称作Sequence to Sequence (Seq2Seq)模型
- Seq2Seq模型的基本思想则非常简单：使用RNN系列网络读取输入句子，将整个句子信息编码到一个固定维度的向量中；再使用另一个循环神经网络读取这个向量，将其解码为目标语言的一个句子

[1] Sutskever, Ilya, Oriol Vinyals, and Quoc V. Le. "Sequence to sequence learning with neural networks." In Advances in neural information processing systems, pp. 3104-3112. 2014.
[2] Cho, Kyunghyun, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. "Learning phrase representations using RNN encoder-decoder for statistical machine translation." arXiv preprint arXiv:1406.1078 (2014).

Seq2Seq模型


I am a student
Encoder
0.5
0.2
-0.1
-0.3
0.4
1.2
Decoder
Je suis étudiant

Seq2Seq模型

- 解码器部分的结构和语言模型几乎完全相同，输入为单词的词向量，输出为softmax层产生的单词概率，损失函数为log perplexity
- 事实上，解码器可以理解为一个以输入编码为前提的语言模型(Conditional Language Model)
- 编码器部分更为简单，它与解码器一样拥有词向量层和循环神经网络，但是由于在编码阶段并未输出，不需要softmax层

Seq2Seq模型

- 在训练过程中，编码器顺序读入每个单词的词向量，然后将最终的隐藏状态复制到解码器作为初始状态
- 解码器的第一个输入是特殊的<sos>(Start-Of-Sentence)字符，每一步预测的单词是训练数据的目标句子，预测序列的最后一个单词是特殊的<eos>(End-Of-Sentence)字符

Seq2Seq模型

- 在测试过程中，让解码器在没有“正确答案”的情况下，自主生成一个翻译句子，然后采用人工或自动的方法对翻译句子的质量进行测评。让解码器生成句子的过程也称为“解码”(decoding)
- 在解码过程中，使用贪心算法把每一步预测的单词中概率最大的单词选为这一步的输出，并复制到下一步的输入中

注意力机制

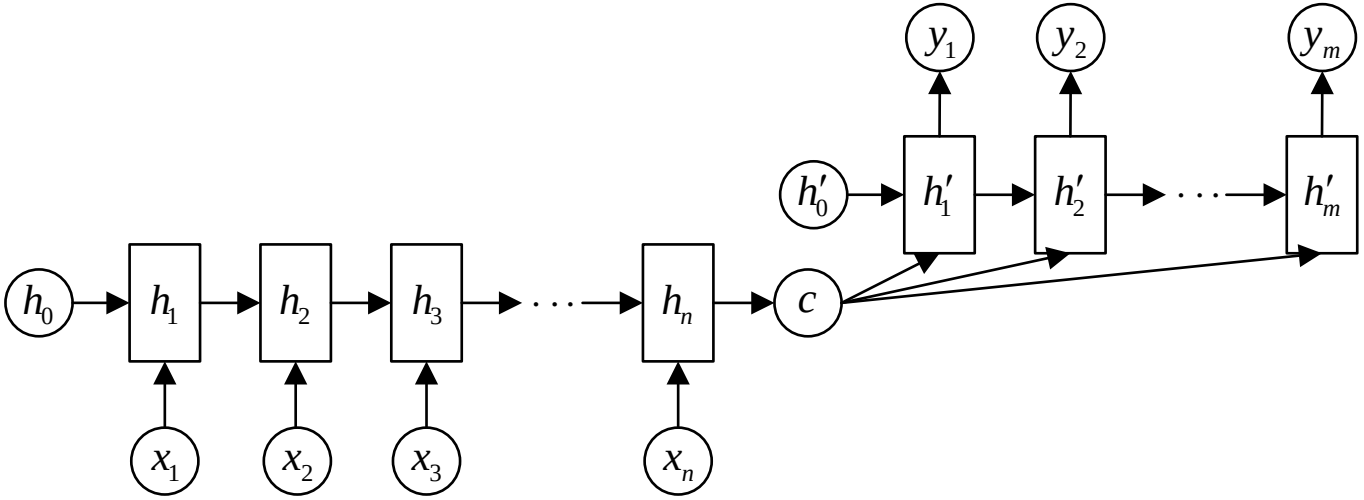
- 在seq2seq模型中，编码器将完整的输入句子压缩到一个维度固定的向量中，然后解码器根据这个向量生成输出句子
- 当输入句子较长时，这个中间向量难以存储足够的信息，成为这个模型的瓶颈

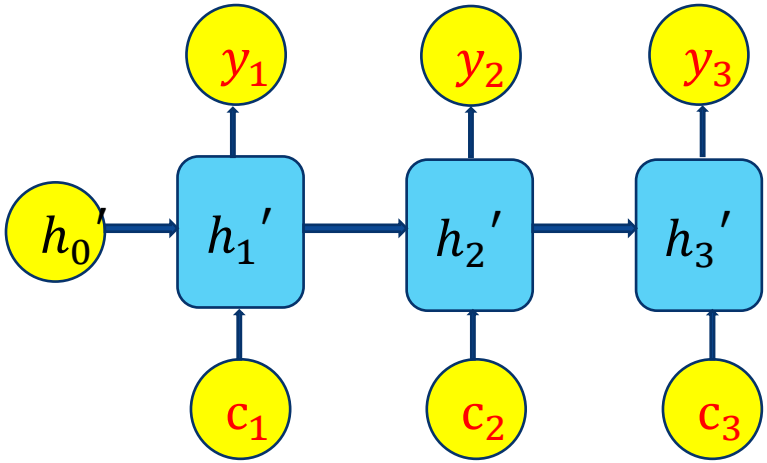
注意力机制

- 注意力机制(Attention)允许解码器随时查阅输入句子中的部分单词或片段，因此不再需要在某个中间向量中存储所有信息
- 它最早提出于论文“Neural Machine Translation by Jointly Learning to Align and Translate”，并在论文“Effective Approaches to Attention-based Neural Machine Translation ”完善

注意力机制

- 在Decoder部分，相比与之前的固定的语义向量c，注意力机制要求在每一步解码时使用不同的c





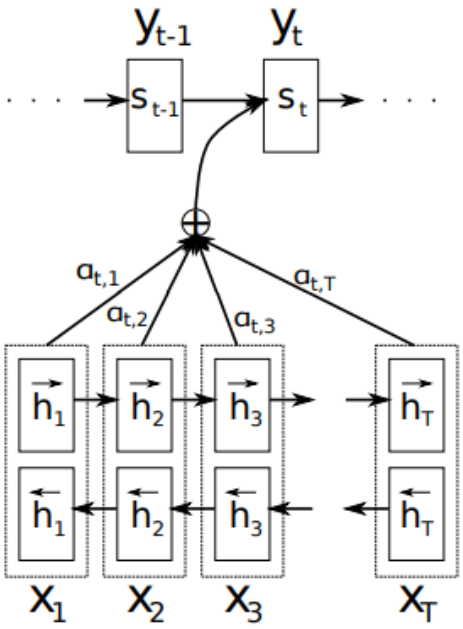
注意力机制

- 神经机器翻译中最原始的注意力机制在Encoder部分使用双向的RNN (bi-directional RNN)
  - 前向RNN正向读取输入序列，并计算前向隐藏层状态
  - 后向RNN反向读取输入序列，并计算反向隐藏层状态
- 对于每个单词x_j，把它对应的前向隐藏状态向量$\overrightarrow{h_j}$和后向隐藏状态向量$\overleftarrow{h_j}$拼接起来表示对x_j的注解：$[\overrightarrow{h_j}; \overleftarrow{h_j}]$

Dzmitry Bahdanau, et al., Neural machine translation by jointly learning to align and translate, 2015

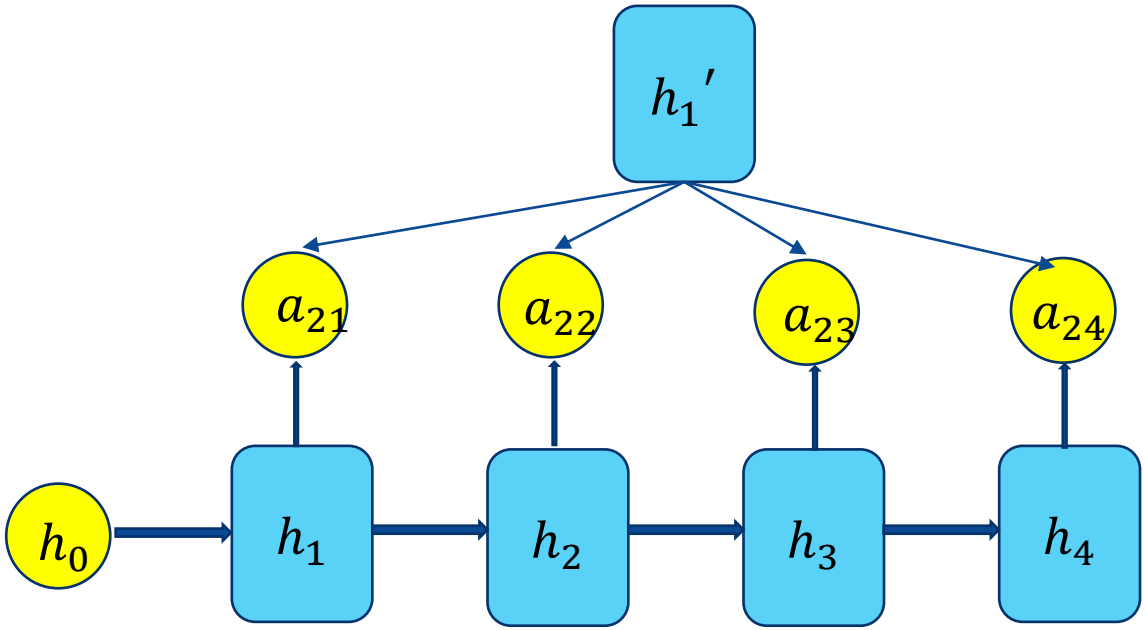
注意力机制

- 使用注意力机制后，每一个c会自动选取与当前所要输出的y最合适的上下文信息
- 具体来说，引入参数a_{ij} 用于衡量Encoder中第j阶段的h_j和Decoder时第i阶段的相关性
- 最终，Decoder中第i阶段输入的上下文信息c_i来自于所有h_j对a_{ij}的加权之和



注意力机制

❑ 例如，a_{2j}的计算如下图所示：



注意力机制

- 用数学表达式整理注意力机制的工作原理：
  - 上下文向量：

$$c_i = \sum_j h_j a_{ij}$$

  - a_{ij}根据h'_{i-1}与h_j计算：

$$a_{ij} = \frac{\exp(e_{ij})}{\sum_k \exp(e_{ik})} \quad e_{ij} = f\left(h'_{i-1}, h_j\right)$$

  - a_{ij}显然要满足：

$$\sum_j a_{ij} = 1$$

注意力机制

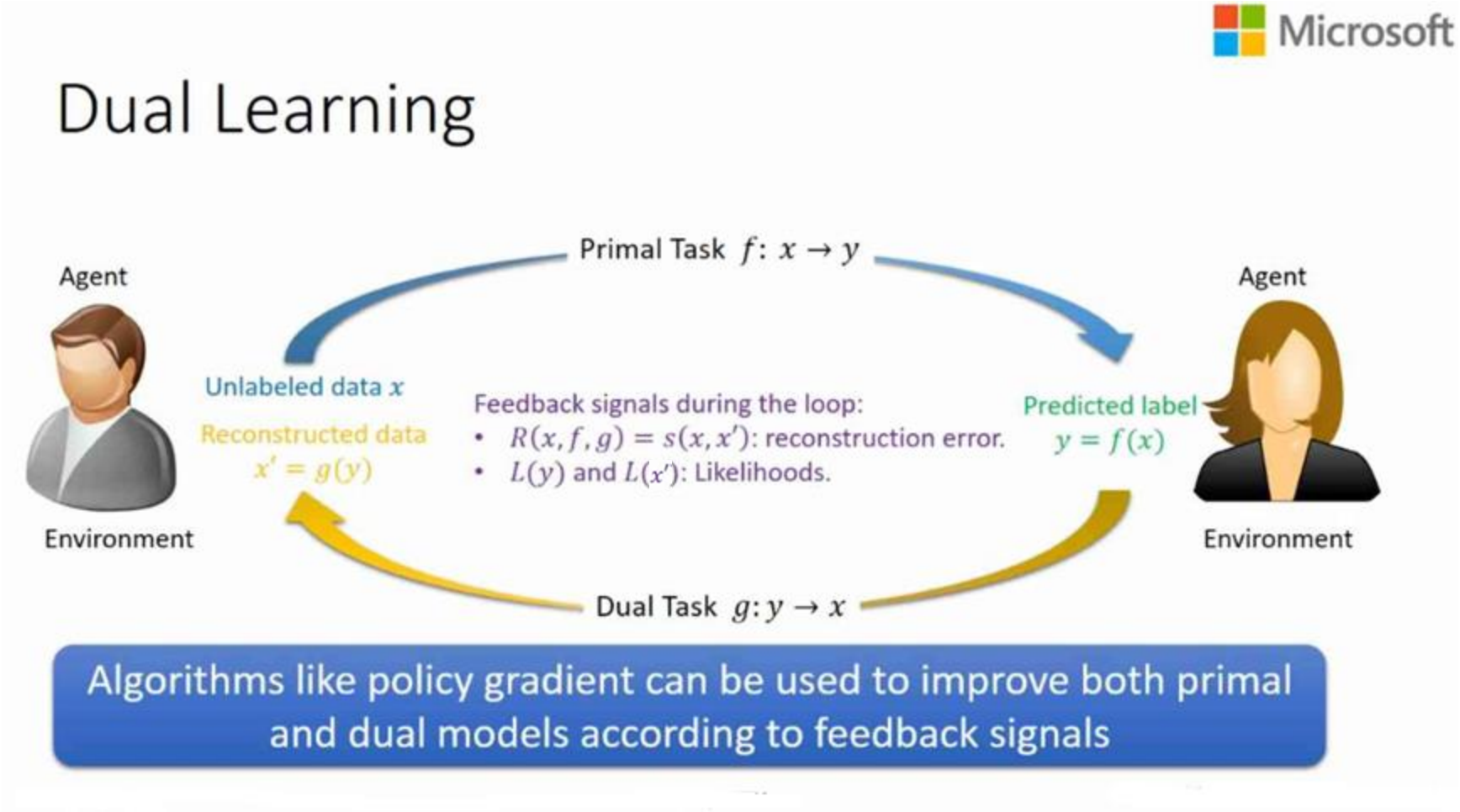
- 注意力系数计算

$$a_{ij} = \frac{\exp(e_{ij})}{\sum_{k=1}^{T_x} \exp(e_{ik})} \qquad e_{ij} = f(h'_{i-1}, h_j)$$

- 计算注意力系数的相似函数（**alignment model**）有以下几种：

$$f(h'_{i-1}, h_j) = \begin{cases} h_j^T \cdot h'_{i-1} \\ h_j^T \cdot W_\alpha \cdot h'_{i-1} \\ W_\alpha \cdot \left[h_j^T, h'^T_{i-1}\right]^T \\ v_\alpha \tanh(U_\alpha h_j + W_\alpha h'_{i-1}) \end{cases}$$

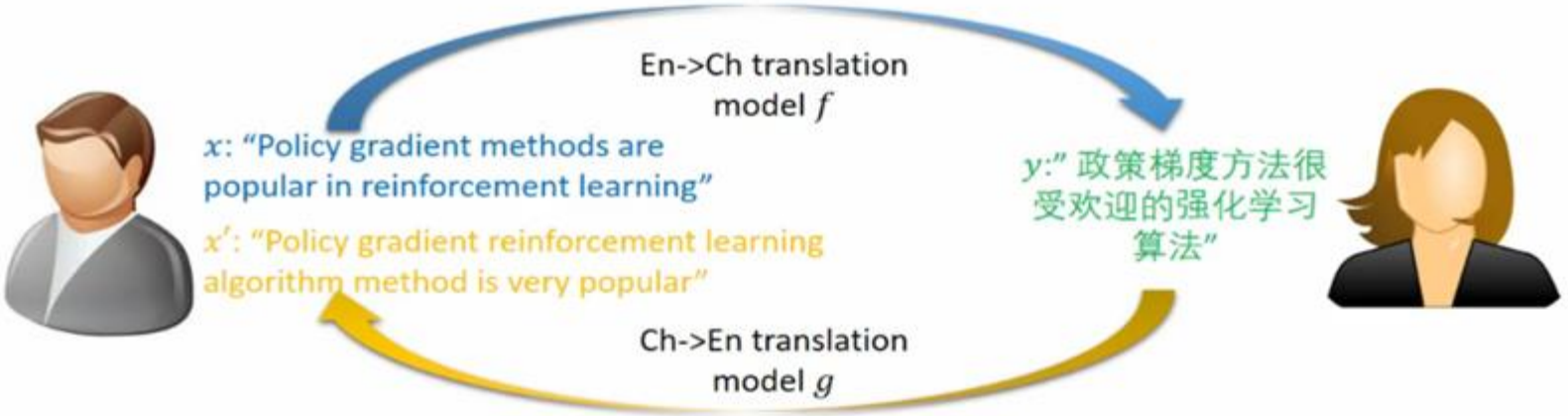
对偶学习


Microsoft
Dual Learning
Primal Task $f: x \to y$
Agent
Agent
Unlabeled data x
Reconstructed data
$x' = g(y)$
Feedback signals during the loop:
• $R(x, f, g) = s(x, x')$: reconstruction error.
• $L(y)$ and $L(x')$: Likelihoods.
Predicted label
$y = f(x)$
Environment
Environment
Dual Task $g: y \to x$
Algorithms like policy gradient can be used to improve both primal and dual models according to feedback signals

对偶学习

Microsoft

Policy Gradient: Iteration t



Feedback signals during the loop:

- $s(x, x') = 0.3$
- $L(y) = 0.2,\ L(x') = 0.1$

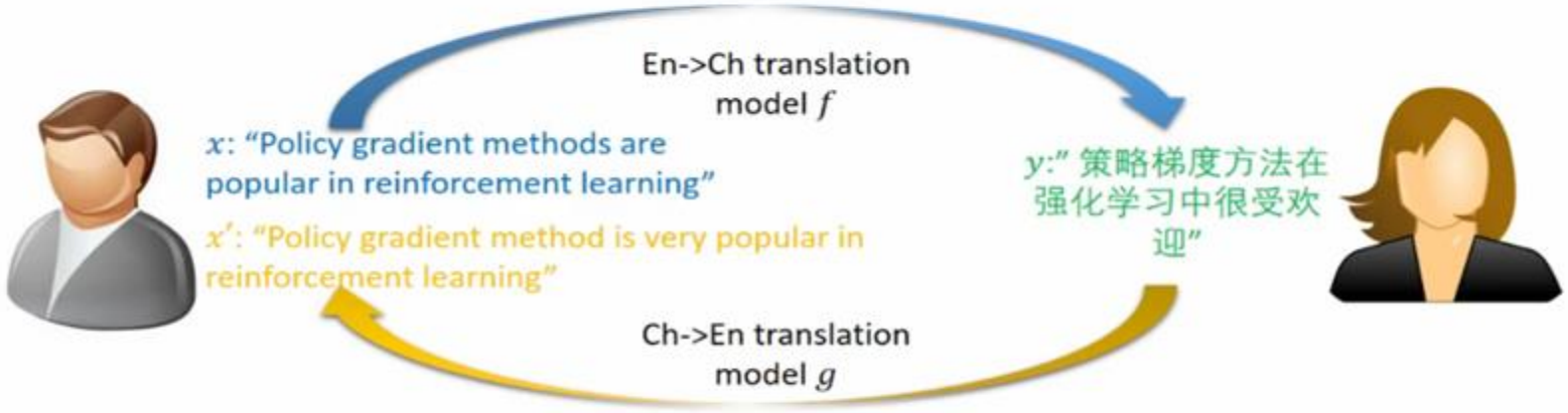
Model update

- $f = f - \alpha\Delta f, g = g - \alpha\Delta g$
- Decrease the probability of $P(x', y|x; f, g)$

对偶学习

Microsoft

Policy Gradient: Iteration $t+1$



Feedback signals during the loop:

- $s(x, x') = 0.8$
- $L(y) = 0.8,\ L(x') = 0.9$

Model update

- $f = f + \alpha\Delta f, g = g + \alpha\Delta g$
- Increase the probability of $P(x', y|x; f, g)$

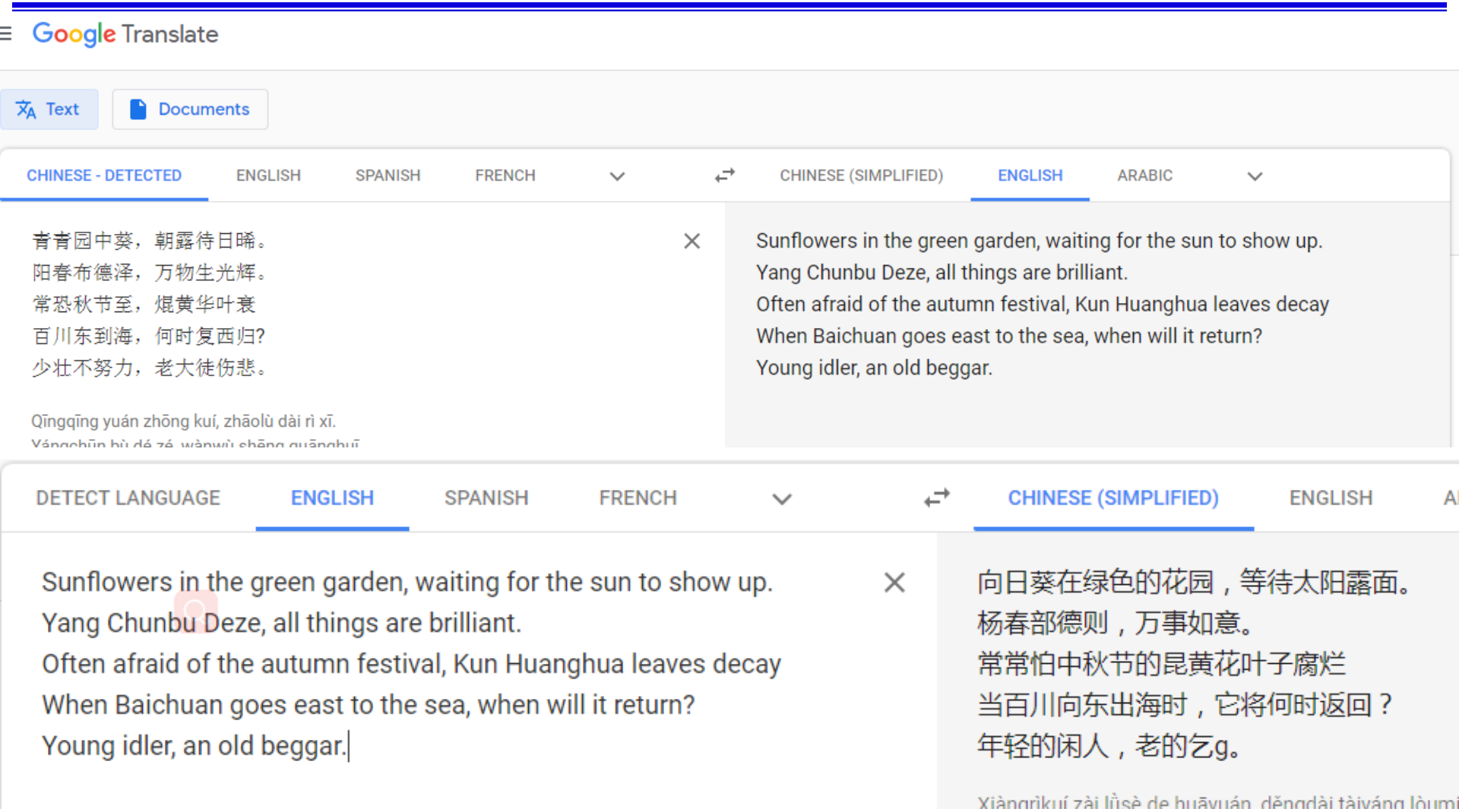
工业界研究机构

- **国外：**
  - Google、Microsoft、IBM、Facebook
- **国内：**
  - 百度、华为、阿里巴巴、腾讯、搜狗、有道

工业界产品


Bai du 翻译
官网
抗疫行动
同传
视频翻译
人工翻译
插件下载
APP下载
Vmatinal
200 语种
检测到中文(简体)
英语
翻 译
人工翻译
助力抗疫
通用领域
生物医药
青青园中葵，朝露待日晞。
阳春布德泽，万物生光辉。
常恐秋节至，焜黄华叶衰
百川东到海，何时复西归?
少壮不努力，老大徒伤悲。
In the green garden, sunflower is waiting for Sun Xi.
In the spring of spring, budze makes all things shining.
It is often feared that the autumn festival will come, and the leaves of Kun Huanghua will decline
When will rivers return to the west when they reach the sea in the east?
Young men don't work hard, old men are sad.
报错
双语对照
英语
中文(简体)
翻 译
人工翻译
通用领域
生物医药
In the green garden, sunflower is waiting for Sun Xi.
In the spring of spring, budze makes all things shining.
It is often feared that the autumn festival will come, and the leaves of Kun Huanghua will decline. When will rivers return to the west when they reach the sea in the east? Young men don't work hard, old men are sad.
在绿色的花园里，向日葵在等孙曦。
春天的春天，百得让万物熠熠生辉。
人们常常担心秋天会来临，坤黄花的叶子会凋谢。当河流到达东方的大海时，它们什么时候会回到西方？年轻人不努力工作，老年人很伤心。
报错
拼音
双语对照

工业界产品


Google Translate
Text
Documents
CHINESE - DETECTED
ENGLISH
SPANISH
FRENCH
CHINESE (SIMPLIFIED)
ENGLISH
ARABIC
青青园中葵，朝露待日晞。
阳春布德泽，万物生光辉。
常恐秋节至，焜黄华叶衰
百川东到海，何时复西归?
少壮不努力，老大徒伤悲。
Qīngqīng yuán zhōng kuí, zhāolù dài rì xī.
Sunflowers in the green garden, waiting for the sun to show up.
Yang Chunbu Deze, all things are brilliant.
Often afraid of the autumn festival, Kun Huanghua leaves decay
When Baichuan goes east to the sea, when will it return?
Young idler, an old beggar.
DETECT LANGUAGE
ENGLISH
SPANISH
FRENCH
CHINESE (SIMPLIFIED)
ENGLISH
Sunflowers in the green garden, waiting for the sun to show up.
Yang Chunbu Deze, all things are brilliant.
Often afraid of the autumn festival, Kun Huanghua leaves decay
When Baichuan goes east to the sea, when will it return?
Young idler, an old beggar.
向日葵在绿色的花园，等待太阳露面。
杨春部德则，万事如意。
常常怕中秋节的昆黄花叶子腐烂
当百川向东出海时，它将何时返回？
年轻的闲人，老的乞g。

未来展望

- 神经机器翻译采用编码解码网络，简单有效，已逐渐取代统计机器翻译，成为主流研究范式
- 神经机器翻译仍面临诸多问题
  - 缺乏可解释性
  - 难利用先验知识、语言相关知识
  - 训练、测试复杂度高（需GPU、甚至TPU）
  - 领域、场景迁移性能差

未来展望

❑ 未来展望

- 神经机器翻译的可解释性研究
- 与专家知识、常识知识的融合研究
- 场景、领域的迁移和定制化研究
- 面向资源稀缺语言的机器翻译建模
- 多模态机器翻译（语音和文本的一体化）研究
- 与硬件的一体化研究

中国科学院大学
University of Chinese Academy of Sciences

3 机器阅读理解

机器阅读理解

- 自然语言处理（Natural Language Processing, NLP）是实现人类无障碍人机交互愿景的基石，被誉为“人工智能皇冠的明珠”
- 而机器阅读理解（Machine Read Comprehension, MRC）是近年来 NLP 领域的研究热点之一，被视为“自然语言处理皇冠上的明珠之一”

任务定义

❑ 机器阅读理解类型

- 完型填空
- 多项选择
- 区域预测
- 自由形式

段落

工商协进会报告，12 月消费者信心指数上升到 78.1，明显高于 11 月的 72。另据《华尔街日报》报道，2013 年是 1995 年以来美国股市表现最好的一年。这一年里，投资美国股市的明智做法是追着“傻钱”跑。所谓的“傻钱”策略，其实就是买入并持有美国股票这样的普通组合。这个策略要比对冲基金和其他专业投资者使用的更为复杂的投资方法效果好得多。

问题 1：什么是傻钱策略？

答案：买入并持有美国股票这样的普通组合。

问题 2：12 月的消费者信心指数是多少？

答案：78.1。

问题 3：消费者信心指数由什么机构发布？

答案：工商协进会。

图 1　机器学习阅读理解样例[1]

任务定义

- **完型填空：原文中挖出一个空来，由机器根据对文章上下文的理解去补全。代表数据集有CNN/Daily Mail**
  - https://github.com/abisee/cnn-dailymail
  - 填空型大规模英文机器理解数据集，答案是原文中的某一个词
  - CNN数据集包含美国有线电视新闻网的新闻文章和相关问题。大约有90k文章和380k问题
  - Dailymail数据集包含每日新闻的文章和相关问题。大约有197k文章和879k问题

任务定义

- **多项选择：每篇文章对应多个问题，每个问题有多个候选答案，机器需要在这些候选答案中找到最合适的那个。代表数据集有RACE**
  - http://www.cs.cmu.edu/~glai1/data/race/
  - 数据集为中国中学生英语阅读理解题目，给定一篇文章和 5 道 4 选 1 的题目，包括了 28000+ passages 和 100,000 问题

任务定义

- **区域预测：也称为抽取式问答（Extractive QA），即给定文章和问题，机器需要在文章中找到答案对应的区域（span），给出开始位置和结束位置。代表数据集有 SQuAD（Stanford Question Answering Dataset）**
  - https://rajpurkar.github.io/SQuAD-explorer/
  - 机器理解领域的ImageNet，数据来源于对维基百科文章的问题的整理，其中SQuAD初期版本在500多篇文章中有超过100,000个问题答案对，答案不是原文的一个词，而是原文的一个连续文本片段

任务定义

- **自由形式：不限定问题所处的段落，即一个问题可能是需要理解多个段落甚至多篇文章。代表数据集有DuReader（百度），MS MARCO（微软）**
  - DuReader: https://github.com/baidu/DuReader,所有的问题、原文都来源于实际数据（百度搜索引擎数据和百度知道问答社区），答案是由人类回答的
  - MS MARCO: https://microsoft.github.io/msmarco/,英文阅读理解数据集之一, 该数据集来源于bing, 也同样是雇佣人工完成编辑审核的

MRC数据集

数据集名称	数据来源	类型	文档数目	问题数目	评价指标
CNN/Daily Mail	CNN,Daily Mail	完型填空	300k	1.4M	准确率
Children's Book Test	Children's Book	完型填空	108	688k	准确率
MCTest	Fictional Stories	多项选择	500	2k	准确率
RACE	English Exams	多项选择	28k	97k	准确率
SQuAD	Wikipedia	区域预测	536	100k	F1 EM
SQuAD2.0	Wikipedia	区域预测	505	150k	F1 EM
DuReader	user logs(Baidu)	自由形式	1M	200k	BLEU,ROUGE-L
MARCO	user logs(Bing)	自由形式	3.2M	1M	准确率，BLEU,ROUGE-L

机器阅读理解

- **神经网络机器阅读理解在2015年迎来了繁荣时期：**
  - 2015年，Google DeepMind发表了“Teaching Machines to Read and Comprehend”，给出了用双向LSTM构建阅读器的基本模型
  - 2016，Stanford University发表了“SQuAD: 100,000+ Questions for Machine Comprehension of Text”，基于该机器阅读理解数据集，大量有创造性的模型涌现出来

MRC数据集

- SQuAD：Stanford Question Answering Dataset
- SQuAD1.1 paper：SQuAD:100000+ Question for Machine Comprehension of Test
- SQuAD2.0 paper: Know What You Don't Know: Unanswerable Questions for SQuAD
- Website：https://rajpurkar.github.io/SQuAD-explorer/

Dataset	Question + Answer	Documents
SQuAD 1.1	100，000	500+
SQuAD 2.0	100,000+50,000 unanswerable questions	500+

SQuAD竞赛排名

SQuAD1.1 Leaderboard

Here are the ExactMatch (EM) and F1 scores evaluated on the test set of SQuAD v1.1

Rank	Model	EM	F1
	Human Performance *Stanford University* (Rajpurkar et al. '16)	82.304	91.221
1 Apr 10, 2020	LUKE (single model) *Studio Ousia & NAIST & RIKEN AIP*	**90.202**	**95.379**
2 May 21, 2019	XLNet (single model) *Google Brain & CMU*	89.898	95.080
3 Dec 11, 2019	XLNET-123++ (single model) *MST/EOI* http://tia.today	89.856	94.903
3 Aug 11, 2019	XLNET-123 (single model) *MST/EOI*	89.646	94.930
4 Sep 25, 2019	BERTSP (single model) *NEUKG* http://www.techkg.cn/	88.912	94.584
4 Jul 21, 2019	SpanBERT (single model) *FAIR & UW*	88.839	94.635
5 Jul 03, 2019	BERT+WWM+MT (single model) *Xiaoi Research*	88.650	94.393

SQuAD2.0 tests the ability of a system to not only answer reading comprehension questions, but also abstain when presented with a question that cannot be answered based on the provided paragraph.

Rank	Model	EM	F1
	Human Performance *Stanford University* (Rajpurkar & Jia et al. '18)	86.831	89.452
1 Feb 21, 2021	FPNet (ensemble) *Ant Service Intelligence Team*	**90.871**	**93.183**
2 May 16, 2021	IE-NetV2 (ensemble) *RICOH_SRCB_DML*	90.860	93.100
3 Feb 24, 2021	IE-Net (ensemble) *RICOH_SRCB_DML*	90.758	93.044
4 Apr 06, 2020	SA-Net on Albert (ensemble) *QIANXIN*	90.724	93.011
5 May 05, 2020	SA-Net-V2 (ensemble) *QIANXIN*	90.679	92.948
5 Apr 05, 2020	Retro-Reader (ensemble) *Shanghai Jiao Tong University* http://arxiv.org/abs/2001.09694	90.578	92.978
5 Feb 05, 2021	FPNet (ensemble) *YuYang*	90.600	92.899
6 Apr 18, 2021	TransNets + SFVerifier + SFEnsembler (ensemble) *Senseforth AI Research*	90.487	92.894

MRC数据集

- **MS MARCO：Microsoft MAchine Reading Comprehension**
- **Paper：MS MARCO: A Human Generated MAchine Reading COmprehension Dataset**
- **Website: http://www.msmarco.org/**

  - 1,010,916 Real Bing User Queries
  - 182,669 Natural Language Answers
  - 10 Passages Per Query

MS MARCO竞赛排名

Question Answering Task: RETIRED(03/01/2018-10/30/2020) Leaderboard

Rank	Model	Submission Date	Rouge-L	Bleu-1
1	**Multi-doc Enriched BERT** Ming Yan of Alibaba Damo NLP	June 20th, 2019	0.540	0.565
2	**Human Performance**	April 23th, 2018	0.539	0.485
3	**BERT Encoded T-Net** Y. Zhang, C. Wang, X.L. Chen	August 5th, 2019	0.526	0.539
4	**Selector+Combine-Content-Generator QA Model** Shengjie Qian of Caiyun xiaoyi AI and BUPT	March 19th, 2019	0.525	0.544
5	**LM+Generator** Alibaba Damo NLP	November 25th,2019	0.522	0.516
6	**Masque Q&A Style** NTT Media Intelligence Laboratories [Nishida et al. '19]	January 3rd, 2019	0.522	0.437
7	**Deep Cascade QA** Ming Yan of Alibaba Damo NLP [Yan et al. '18]	December 12th, 2018	0.520	0.546
8	**Unnamed** anonymous	December 9th,2019	0.518	0.507
9	**PALM** Alibaba Damo NLP	December 9th,2019	0.518	0.507

MRC数据集

- **DuReader：百度阅读理解数据集**
- **Website:**
  http://ai.baidu.com/broad/introduction?dataset=dureader

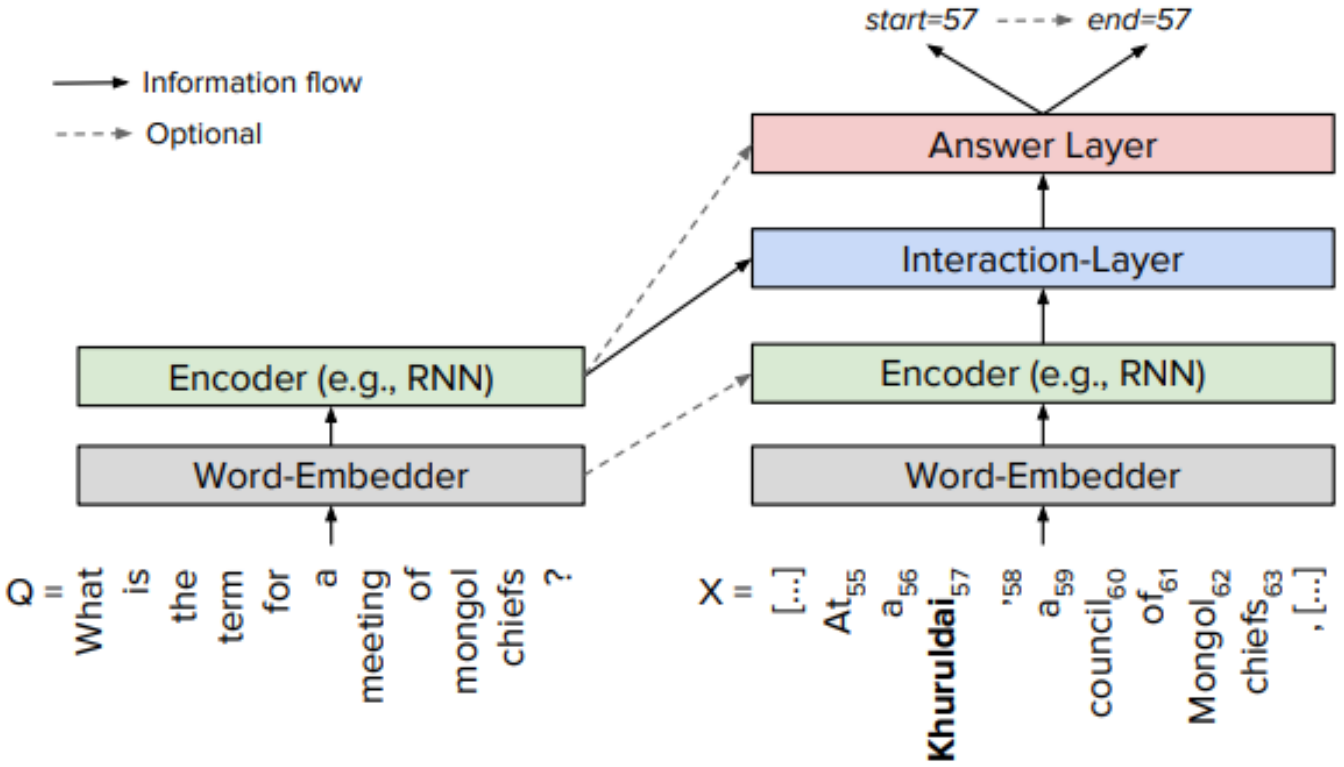
Data Statistics

-	question	document	answer
amount	301574	1431429	665723
avg len	26(char)	1793(char)	299(char)

DuReader 竞赛排名

Rank	Model	ROUGE-L	BLEU-4	Submit Time
1	AliReader renaissance	63.48	61.54	2018-10-23
2	[MRC2018]NI-Reader(ensemble) Naturali Naturali.io	63.38	59.23	2018-04-28
3	mrc_try_mingyan_single mrc_try	62.2	59.72	2018-09-17
4	[MRC2018]Z-Reader（ensemble） ZWYC北京大学 Dlib研究组	60.99	55.93	2018-04-26
5	test_20180731	58.74	54.15	2018-07-31
6	[MRC2018]BIDAF+S+predict (single) NEUKG-NReader 东北大学	58.08	52.49	2018-04-30
7	[MRC2018]myreader(single) iioiio 上海理工大学	57.55	50.87	2018-04-30
8	[MRC2018]D-Reader-L2+cls（ensemble） Delta text analysis 台達電子	56.57	48.03	2018-04-30

问答型MRC框架



答案层

交互层

编码层

向量化层

Dirk Weissenborn et al. Making Neural QA as Simple as Possible but not Simpler. arXiv:1703.04816v3 [cs.CL] 8 Jun 2017

问答型MRC框架

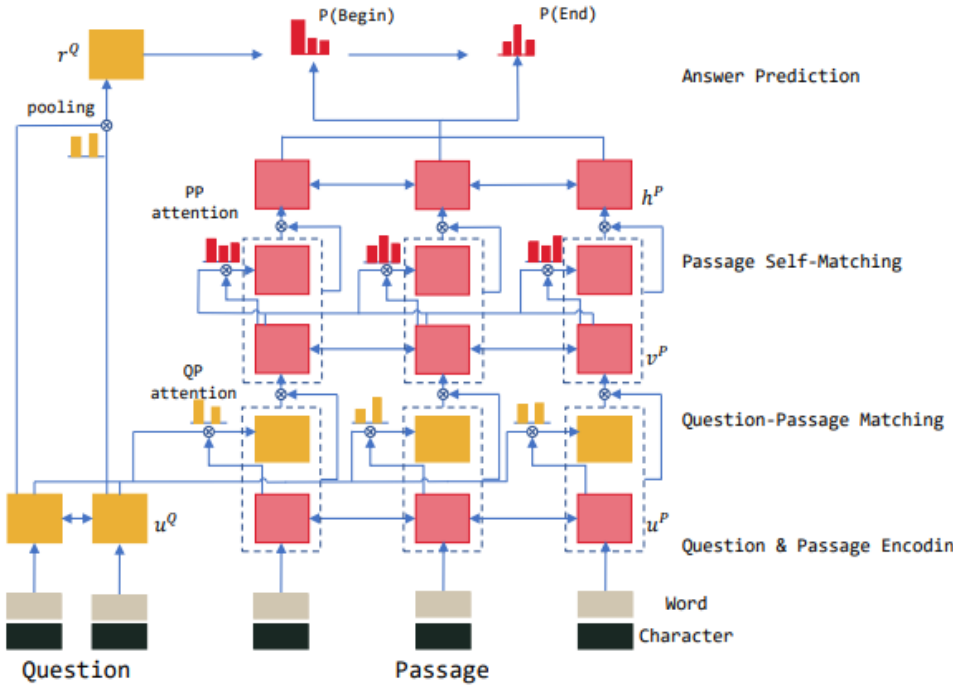
- 向量化层：分别将原文和问题中的tokens 映射为向量表示（Word2Vec，Glove）
- 编码层：主要使用循环神经网络(Recurrent Neural Network，RNN)来对原文和问题进行编码，这样编码后每个 token 的向量表示就蕴含了上下文的语义信息
- 交互层：主要负责分析问题和原文之间的交互关系，并输出编码了问题语义信息的原文表示，即 query-aware 的原文表示
- 答案层：基于 query-aware 的原文表示来预测答案范围(答案起始位置和终止位置)

Microsoft R-net

- **向量化层：使用GloVe词向量 和 Char embedding 两种方法以丰富输入特征**
- **编码层：采用循环神经网络进行编码**
- **交互层：双交互层结构**
  - 第一层基于门限的注意力循环神经网络（gated-attention based recurrent network）匹配question和passage，获取问题的相关段落表示（question-aware passage representation）；
  - 第二层基于自匹配注意力机制的循环神经网络（self-matching attention network）将passage和它自己匹配，从而实现整个段落的高效编码
- **答案层：基于指针网络（pointer-network）定位答案所在位置**

Natural Language Computing Group, Microsoft Research Asia. R-NET: MACHINE READING COMPREHENSION WITH SELF-MATCHING NETWORKS. In Proceedings of ACL 2017

Microsoft Rnet



Performance on SQuAD

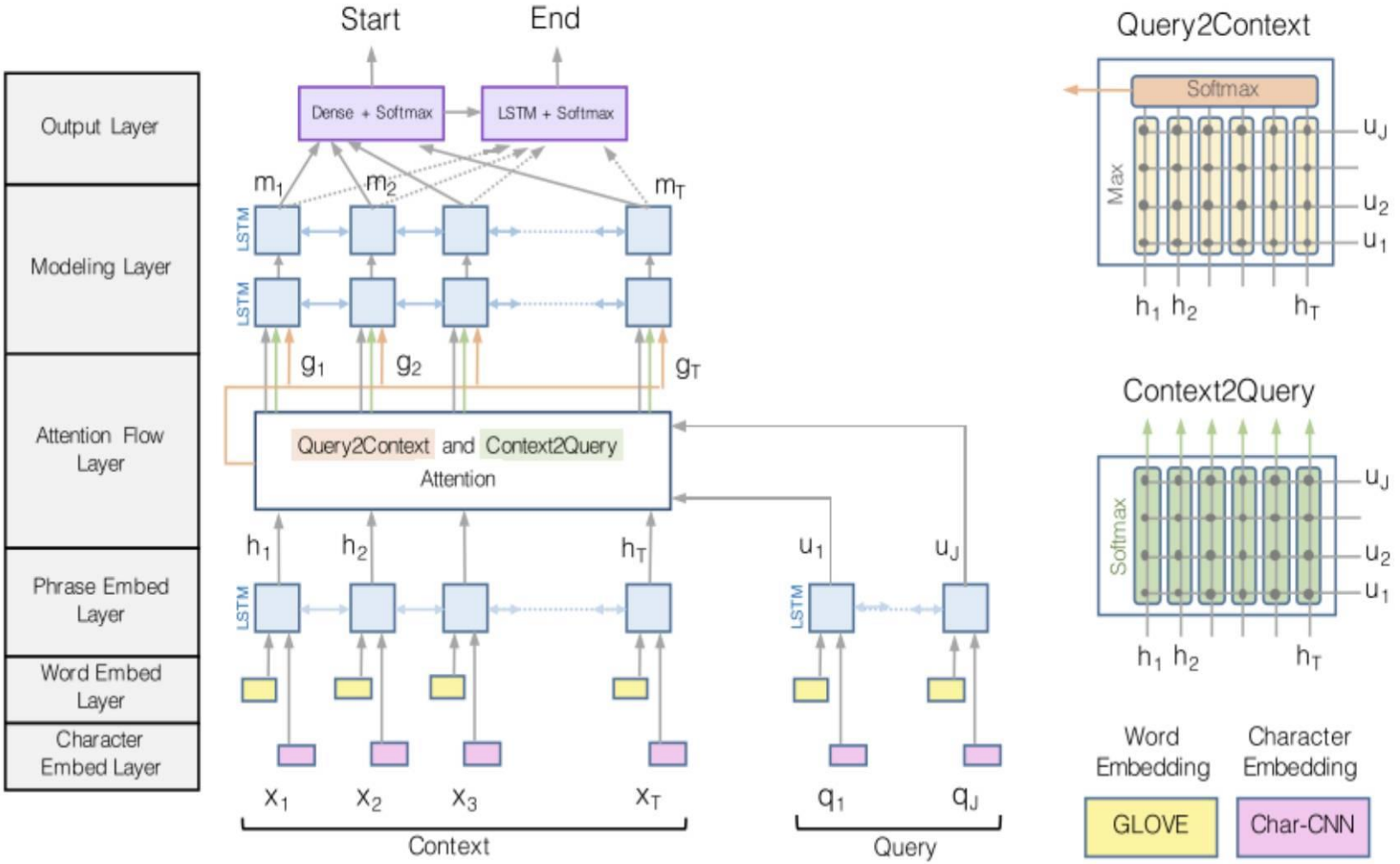
	Dev Set	Test Set
Single model	**EM / F1**	**EM / F1**
LR Baseline (Rajpurkar et al., 2016)	40.0 / 51.0	40.4 / 51.0
Dynamic Chunk Reader (Yu et al., 2016)	62.5 / 71.2	62.5 / 71.0
Attentive CNN context with LSTM (NLPR, CASIA)	- / -	63.3 / 73.5
Match-LSTM with Ans-Ptr (Wang & Jiang, 2016b)	64.1 / 73.9	64.7 / 73.7
Dynamic Coattention Networks (Xiong et al., 2016)	65.4 / 75.6	66.2 / 75.9
Iterative Coattention Network (Fudan University)	- / -	67.5 / 76.8
FastQA (Weissenborn et al., 2017)	- / -	68.4 / 77.1
BiDAF (Seo et al., 2016)	68.0 / 77.3	68.0 / 77.3
T-gating (Peking University)	- / -	68.1 / 77.6
RaSoR (Lee et al., 2016)	- / -	69.6 / 77.7
SEDT+BiDAF (Liu et al., 2017)	- / -	68.5 / 78.0
Multi-Perspective Matching (Wang et al., 2016)	- / -	70.4 / 78.8
FastQAExt (Weissenborn et al., 2017)	- / -	70.8 / 78.9
Mnemonic Reader (NUDT & Fudan University)	- / -	69.9 / 79.2
Document Reader (Chen et al., 2017)	- / -	70.7 / 79.4
ReasoNet (Shen et al., 2016)	- / -	70.6 / 79.4
Ruminating Reader (Gong & Bowman, 2017)	- / -	70.6 / 79.5
jNet (Zhang et al., 2017)	- / -	70.6 / 79.8
Interactive AoA Reader (Joint Laboratory of HIT and iFLYTEK Research)	- / -	71.2 / 79.9
R-NET (Wang et al., 2017)	**71.1 / 79.5**	**71.3 / 79.7**
R-NET (March 2017)	**72.3 / 80.6**	**72.3 / 80.7**
Ensemble model		
Fine-Grained Gating (Yang et al., 2016)	62.4 / 73.4	62.5 / 73.3
Match-LSTM with Ans-Ptr (Wang & Jiang, 2016b)	67.6 / 76.8	67.9 / 77.0
QFASE (NUS)	- / -	71.9 / 80.0
Dynamic Coattention Networks (Xiong et al., 2016)	70.3 / 79.4	71.6 / 80.4
T-gating (Peking University)	- / -	72.8 / 81.0
Multi-Perspective Matching (Wang et al., 2016)	- / -	73.8 / 81.3
jNet (Zhang et al., 2017)	- / -	73.0 / 81.5
BiDAF (Seo et al., 2016)	- / -	73.7 / 81.5
SEDT+BiDAF (Liu et al., 2017)	- / -	73.7 / 81.5
Mnemonic Reader (NUDT & Fudan University)	- / -	73.7 / 81.7
ReasoNet (Shen et al., 2016)	- / -	75.0 / 82.6
R-NET (Wang et al., 2017)	**75.6 / 82.8**	**75.9 / 82.9**
R-NET (March 2017)	**76.7 / 83.7**	**76.9 / 84.0**
Human Performance (Rajpurkar et al., 2016)	- / -	82.3 / 91.2

BiDAF

- **向量化层**：混合了词级 Embedding 和字符级 Embedding，词级 embedding 使用预训练的词向量进行初始化，而字符级 embedding 使用 CNN 进一步编码
- **编码层**：两种 Embedding 共同经过 2 层 Highway Network 作为 Encode 层输入
- **交互层**：Interaction 层中引入了双向注意力机制，即首先计算一个原文和问题的 Alignment matrix，然后基于该矩阵计算 Query2Context 和 Context2Query 两种注意力，并基于注意力计算 query-aware 的原文表示，接着使用双向 LSTM 进行语义信息的聚合
- **答案层**：使用 Boundary Model 来预测答案开始和结束位置

Minjoon Seo, Aniruddha Kembhavi, Ali Farhadi, and Hananneh Hajishirzi. Bidirectional attention flow for machine comprehension. In Proceedings of ICLR 2017.

BiDAF


Output Layer
Modeling Layer
Attention Flow Layer
Phrase Embed Layer
Word Embed Layer
Character Embed Layer
Start
End
Dense + Softmax
LSTM + Softmax
LSTM
Query2Context and Context2Query Attention
Context
Query
Query2Context
Softmax
Max
Context2Query
Word Embedding
Character Embedding
GLOVE
Char-CNN

Alibaba SLQA

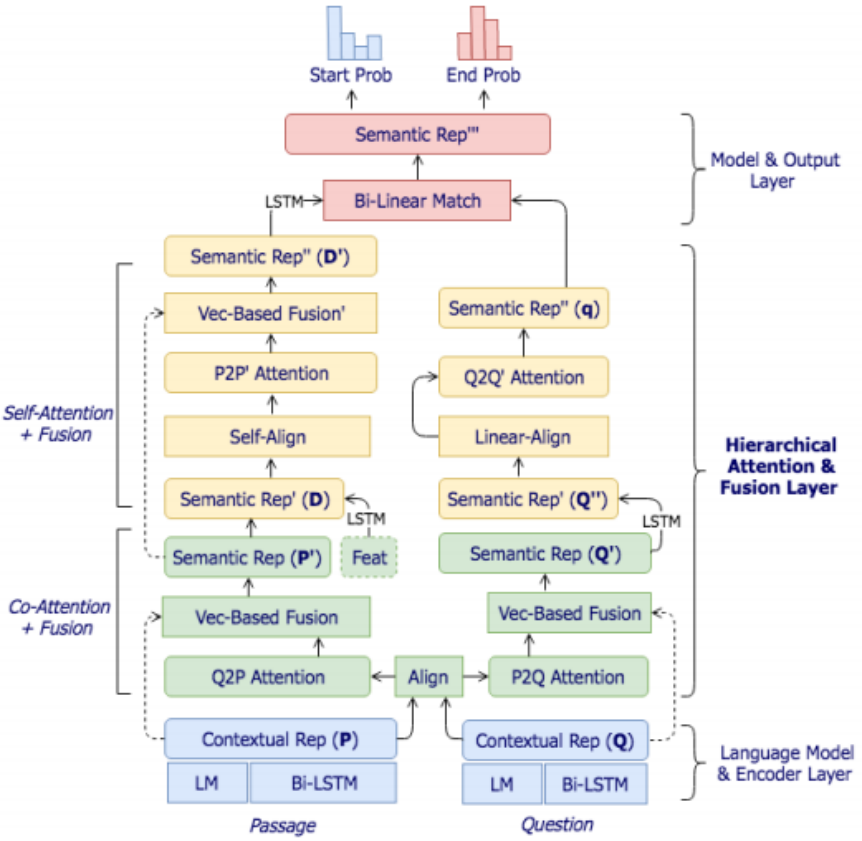


Table 1: The performance of our SLQA model and competing approaches on SQuAD.

	Dev Set	Test Set
Single model	**EM / F1**	**EM / F1**
LR Baseline (Rajpurkar et al., 2016)	40.0 / 51.0	40.4 / 51.0
Match-LSTM (Wang and Jiang, 2016)	64.1 / 73.9	64.7 / 73.7
DrQA (Chen et al., 2017a)	- / -	70.7 / 79.4
DCN+ (Xiong et al., 2017)	74.5 / 83.1	75.1 / 83.1
Interactive AoA Reader+ (Cui et al., 2016)	- / -	75.8 / 83.8
FusionNet (Huang et al., 2017)	- / -	76.0 / 83.9
SAN (Liu et al., 2017b)	76.2 / 84.0	76.8 / 84.4
AttentionReader+ (unpublished)	- / -	77.3 / 84.9
BiDAF + Self Attention + ELMo (Peters et al., 2018)	- / -	78.6 / 85.8
r-net+ (Wang et al., 2017)	- / -	79.9 / 86.5
SLQA+	**80.0 / 87.0**	**80.4 / 87.0**
Ensemble model		
FusionNet (Huang et al., 2017)	- / -	78.8 / 85.9
DCN+ (Xiong et al., 2017)	- / -	78.9 / 86.0
Interactive AoA Reader+ (Cui et al., 2016)	- / -	79.0 / 86.4
SAN (Liu et al., 2017b)	78.6 / 85.9	79.6 / 86.5
BiDAF + Self Attention + ELMo (Peters et al., 2018)	- / -	81.0 / 87.4
AttentionReader+ (unpublished)	- / -	81.8 / 88.2
r-net+ (Wang et al., 2017)	- / -	82.6 / 88.5
SLQA+	**82.0 / 88.4**	**82.4 / 88.6**
Human Performance	80.3 / 90.5	82.3 / 91.2

Wei Wang, Ming Yan, Chen Wu. Multi-Granularity Hierarchical Attention Fusion Networks for Reading Comprehension and Question Answering. Proceedings of ACL 2018

U-net

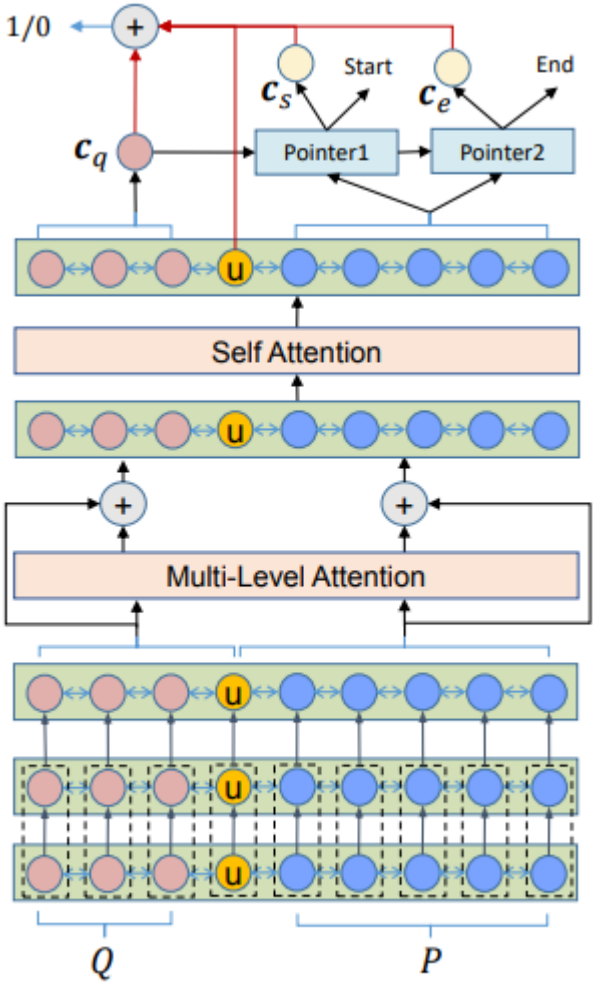


Figure 1: Architecture of the U-Net.

Table 2: Evaluation results on the SQuAD 2.0 (extracted on Sep 9, 2018)

Model	Dev		Test	
	EM	F1	EM	F1
End-to-end Model				
BNA* (Rajpurkar, Jia, and Liang 2018)	59.8	62.6	59.2	62.1
DocQA (Rajpurkar, Jia, and Liang 2018)	65.1	67.6	63.4	66.3
FusionNet++ (Huang et al. 2017)	-	-	66.6	69.6
SAN (Liu et al. 2017)	-	-	68.6	71.4
VS^3-Net	-	-	68.4	71.3
U-Net	70.3	74.0	**69.2**	**72.6**
Ensemble Model				
FusionNet++ (ensemble)	-	-	70.3	72.6
SAN (ensemble)	-	-	71.3	73.7
U-Net (ensemble)	-	-	**71.5**	**75.0**
Pipeline Model				
RMR+ELMo+Verifier (Hu et al. 2018)	72.3	74.8	71.7	74.2
Human	86.3	89.0	86.9	89.5

Fu Sun, Linyang Li, Xipeng Qiu, Yang Liu. U-Net: Machine Reading Comprehension with Unanswerable Questions. https://arxiv.org/abs/1810.06638, 2018

其他MRC模型

- DCN(Dynamic Co-attention Networks)使用了协注意力机制；
- FastQA使用一种轻量级的结构，没有交互层；
- ReasoNet在推理阶段，利用强化学习动态确定是否继续推理；
- Mnemonic Reader同样使用强化学习，并使用re-attention机制避免注意力冗余或注意力不足

预训练模型

- 人类进行阅读理解任务时，会使用许多并未出现在上下文中的知识，通常称为“先验知识”，这是以上模型所无法解决的问题
- 因此，**Pre-training**模型在NLP任务上取得了更为优异的表现，其基本思路为：
  - 在大规模语料库上进行pre-training，学习语法、句法、语言逻辑、先验知识等；
  - 针对不同的任务进行fine-tuning

预训练模型

- 现有的预训练模型主要有 **ELMO，GPT，BERT，GPT2.0/3.0等**
- **BERT和GPT2.0摒弃了传统的LSTM，使用Transformer作为特征提取器，在多项NLP任务上取得了最好的表现效果**

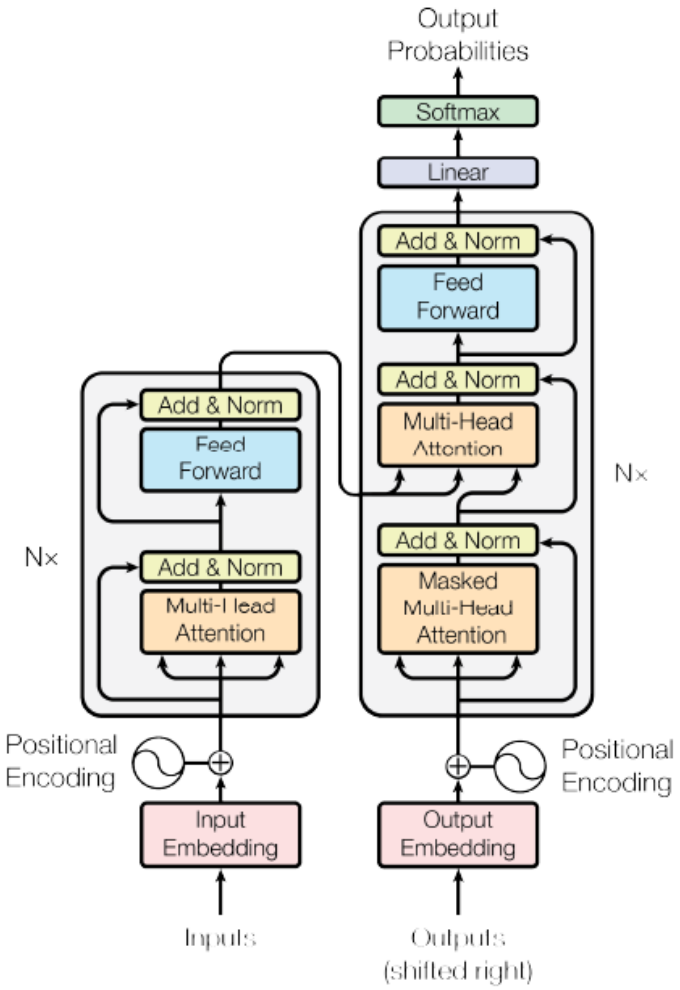


Figure 1: The Transformer - model architecture.

Google BERT模型

- Google AI 2018年提出的(Bidirectional Encoder Representations from Transformers, BERT）模型在机器阅读理解顶级水平测试SQuAD1.1中表现出惊人的成绩：全部两个衡量指标上全面超越人类。在SQuAD2.0上也排名第一
- 并且还在其他10种不同NLP任务测试中创出最佳成绩，包括将GLUE基准推至80.4%（绝对改进7.6%），MultiNLI准确度达到86.7%（绝对改进率5.6%）等

Google BERT模型

- **Website**
  - https://github.com/google-research/bert（Tensorflow）
- **BERT PyTorch实现**
  - https://github.com/huggingface/pytorch-pretrained-BERT

Google BERT模型

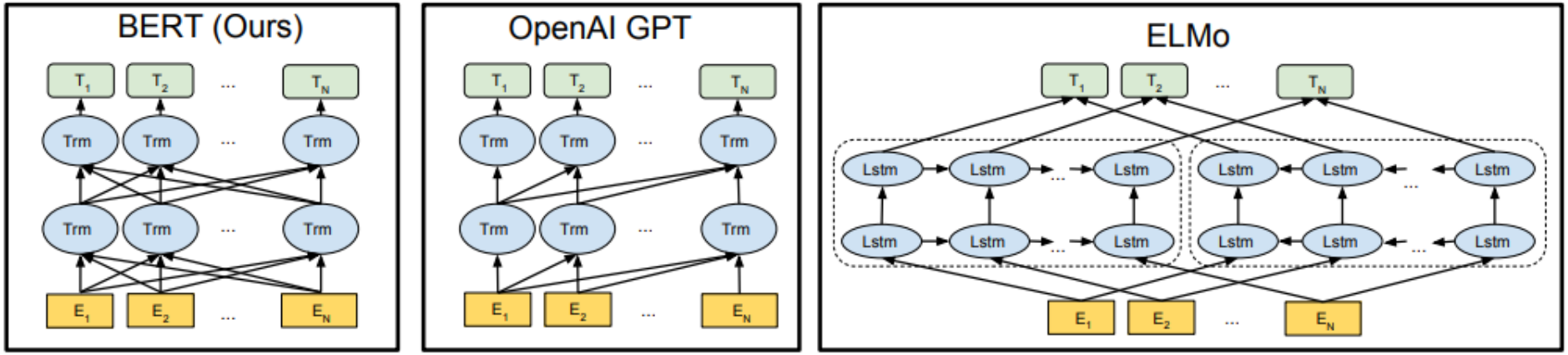


Figure 3: Differences in pre-training model architectures. BERT uses a bidirectional Transformer. OpenAI GPT uses a left-to-right Transformer. ELMo uses the concatenation of independently trained left-to-right and right-to-left LSTMs to generate features for downstream tasks. Among the three, only BERT representations are jointly conditioned on both left and right context in all layers. In addition to the architecture differences, BERT and OpenAI GPT are fine-tuning approaches, while ELMo is a feature-based approach.

Google BERT模型

❑ 输入层（向量化层）

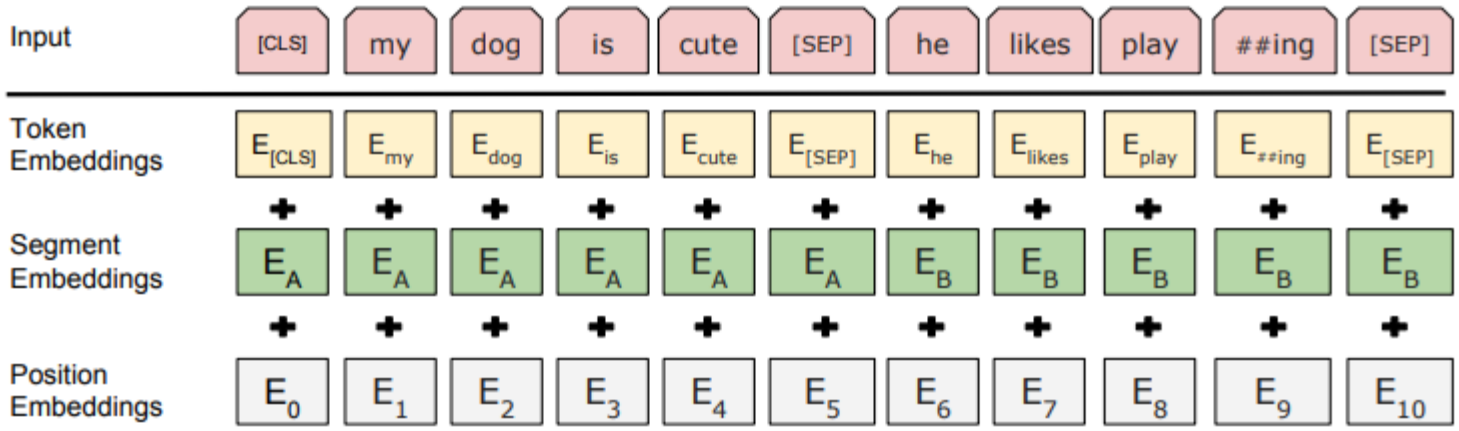
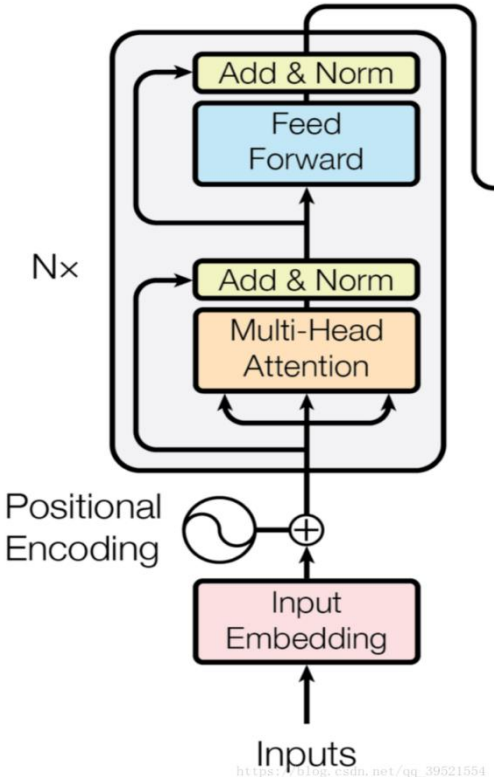


Figure 2: BERT input representation. The input embeddings are the sum of the token embeddings, the segmentation embeddings and the position embeddings.

Google BERT模型

❑ 编码层

- 采用双向的Transformer模型
- N表示层数（即Transformer blocks），H表示隐藏层大小，A表示self-attention heads的数量。在所有情况下，将feed-forward/filter 的大小设置为4H

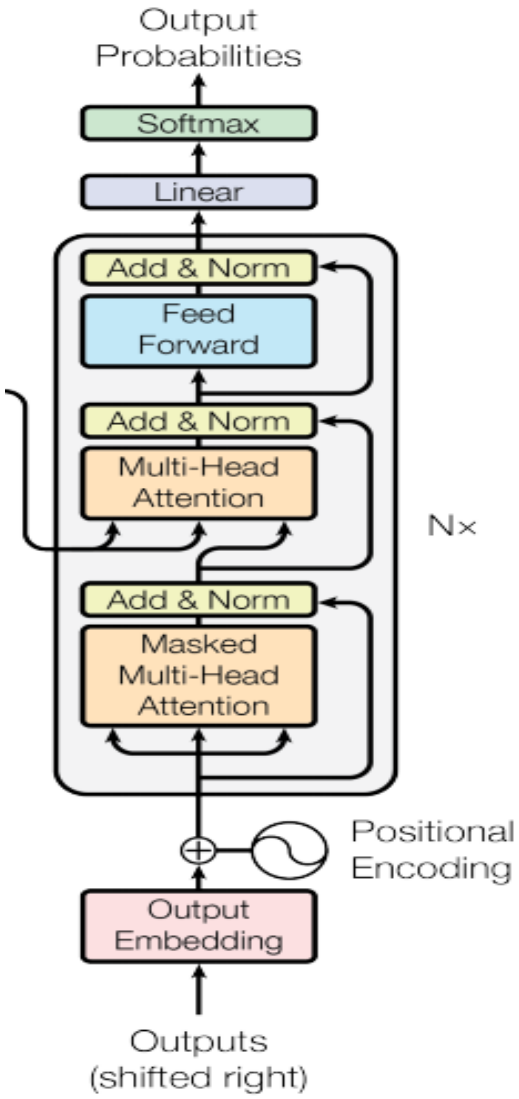


❑ 两种模型：

- BERT Base: N=12, H=768, A=12, Total Parameters=110M
- BERT Large: N=24, H=1024, A=16, Total Parameters=340M

Google BERT模型

- 解码层



Google BERT模型

❑ Mask语言模型

- 借鉴完型填空的思想
- 随机mask输入中的一些单词，然后在预训练中根据两个方向上的上下文对它们进行预测

Google BERT模型

- 具体作法
  - 随机Mask一个句子中15%的单词
  - 80%的时间真的用[MASK]取代被选中的词：如 my dog is hairy -> my dog is [MASK]
  - 10%的时间用一个随机词取代它：my dog is hairy -> my dog is apple
  - 10%的时间保持不变: my dog is hairy -> my dog is hairy
- 下一个句子预测任务的训练
  - 将有50,000个训练例子或句子对作为训练数据。
  - 对于50%的对来说，第二个句子实际上是第一个句子的下一个句子
  - 对于剩下的50%，第二句是语料库中的一个随机句子

Google BERT模型

- **BERT训练语料**
  - 英文开源语料BooksCorpus（8亿单词）
  - 英文维基百科数据（25亿单词）
- **BERT模型参数**
  - BERT Base有1亿多参数（与OpenAI GPT持平）
  - BERT Large有3亿多参数（当时NLP中最大的语言模型）

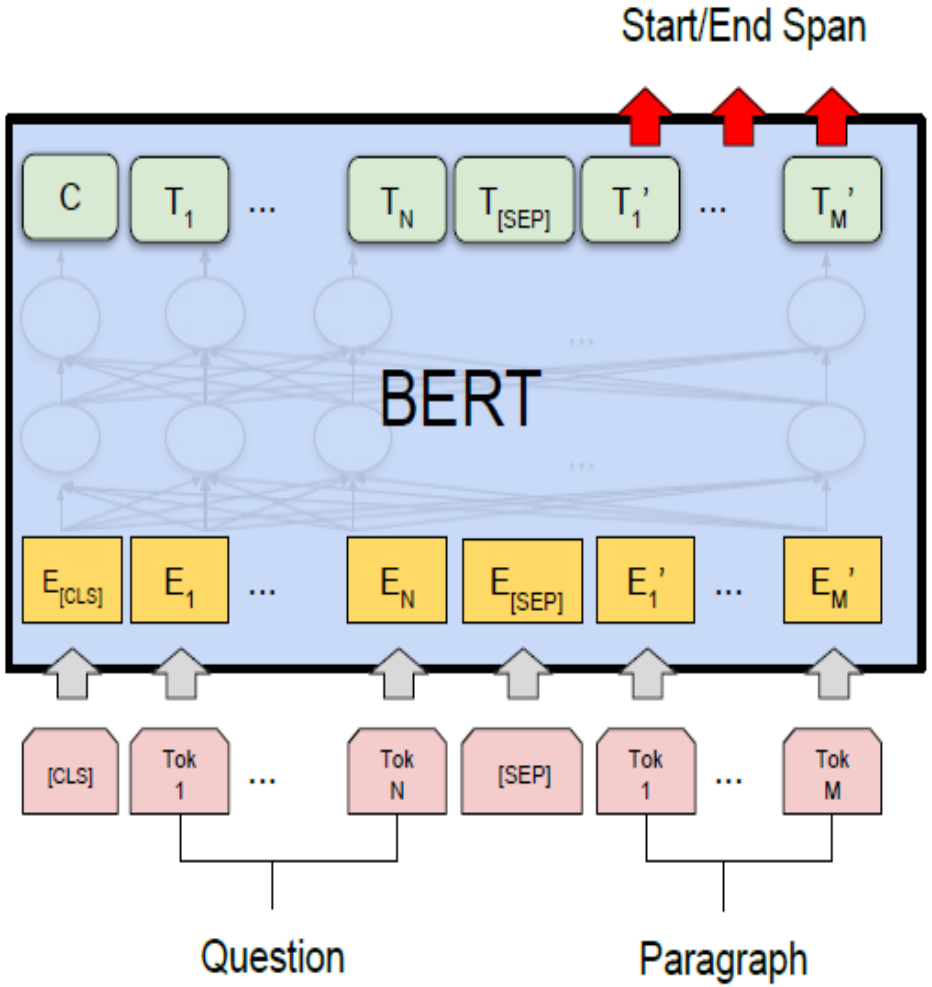
Google BERT模型

❑ 训练时间

- BERT Base: 4个TPU集群，4天时间
- BERT Large: 16个TPU集群，4天时间
- OpenAI训练GPT用了将近1个月的时间，而如果用同等的硬件条件来训练BERT估计需要1年的时间

Google BERT模型

❑ SQuAD1.1BERT模型结构



Google BERT模型

❑ 实验结果

– SQuAD1.1上的结果

System	Dev		Test	
	EM	F1	EM	F1
Top Leaderboard Systems (Dec 10th, 2018)				
Human	-	-	82.3	91.2
#1 Ensemble - nlnet	-	-	86.0	91.7
#2 Ensemble - QANet	-	-	84.5	90.5
Published				
BiDAF+ELMo (Single)	-	85.6	-	85.8
R.M. Reader (Ensemble)	81.2	87.9	82.3	88.5
Ours				
$\text{BERT}_{\text{BASE}}$ (Single)	80.8	88.5	-	-
$\text{BERT}_{\text{LARGE}}$ (Single)	84.1	90.9	-	-
$\text{BERT}_{\text{LARGE}}$ (Ensemble)	85.8	91.8	-	-
$\text{BERT}_{\text{LARGE}}$ (Sgl.+TriviaQA)	**84.2**	**91.1**	**85.1**	**91.8**
$\text{BERT}_{\text{LARGE}}$ (Ens.+TriviaQA)	**86.2**	**92.2**	**87.4**	**93.2**

4 自动摘要

背景

- 随着近几年文本信息的爆发式增长，人们每天能接触到海量的文本信息，如新闻、博客、聊天、报告、论文、微博等。
- 从大量文本信息中提取重要的内容，已成为我们的一个迫切需求，而自动文本摘要（automatic text summarization）则提供了一个高效的解决方案

自动摘要概述

- 定义：自动摘要是利用计算机按照某类应用自动地将文本（或文本集合）转换生成简短摘要的一种信息压缩技术
- 要求：一份合格的摘要，需要包含足够的信息量、较低的冗余度和较高的可读性

自动摘要的分类

❑ 抽取式摘要

- 定义：抽取式摘要就是从原文中抽取一些句子组成摘要，这本质上是一种排序问题，通过一定的算法对原文中的句子进行重要性评分，抽取高分句子，去除冗余得到摘要
- 优点：抽取式方法主要考虑句子的重要性，直接从原文中抽取已有的句子组成摘要，方法简单易实现，因此应用比较广泛
- 缺点：缺乏语义信息，不符合摘要生成的本质

抽取式摘要

❑ Text rank排序算法步骤

- 第一步，去除原文中的一些停用词，度量原文中每个句子的相似度，计算每一句相对于另一句的相似度得分，迭代传播，直至误差小于某一个范围。
- 第二步，对关键句子进行排序，根据摘要长度选择一定数量的句子组成摘要

文本摘要的分类

❑ 生成式摘要

- 定义：改写或者重新组织原文生成摘要。具体地，生成式方法首先根据输入文本获得对原文本的语义理解，然后使用任意的单词或者其他表示来生成文本摘要
- 随着深度学习技术和序列到序列模型的发展，生成式文本摘要成为一种主流的摘要生成方法
- 优点：对原文有更全面的把握，更符合摘要的本质。
- 缺点：句子的可读性、流畅度等不如抽取式方法

生成式摘要

❑ 基本框架

- 生成式文本摘要生成的框架是序列到序列模型（seq2seq），也就是编解码(encoder-decoder)结构。
- 其中，编码器和解码器分别由循环神经网络构成，普遍的做法是使用双向循环神经网络构成编码器，将原文进行编码，使用单向循环神经网络构成解码器，负责从编码器所生成的向量中提取语义信息，生成文本摘要

生成式摘要

❑ 关键技术

- 注意力机制（Attention mechanism）
- 指针机制 （Pointer mechanism）
- 覆盖机制 （Coverage mechanism）

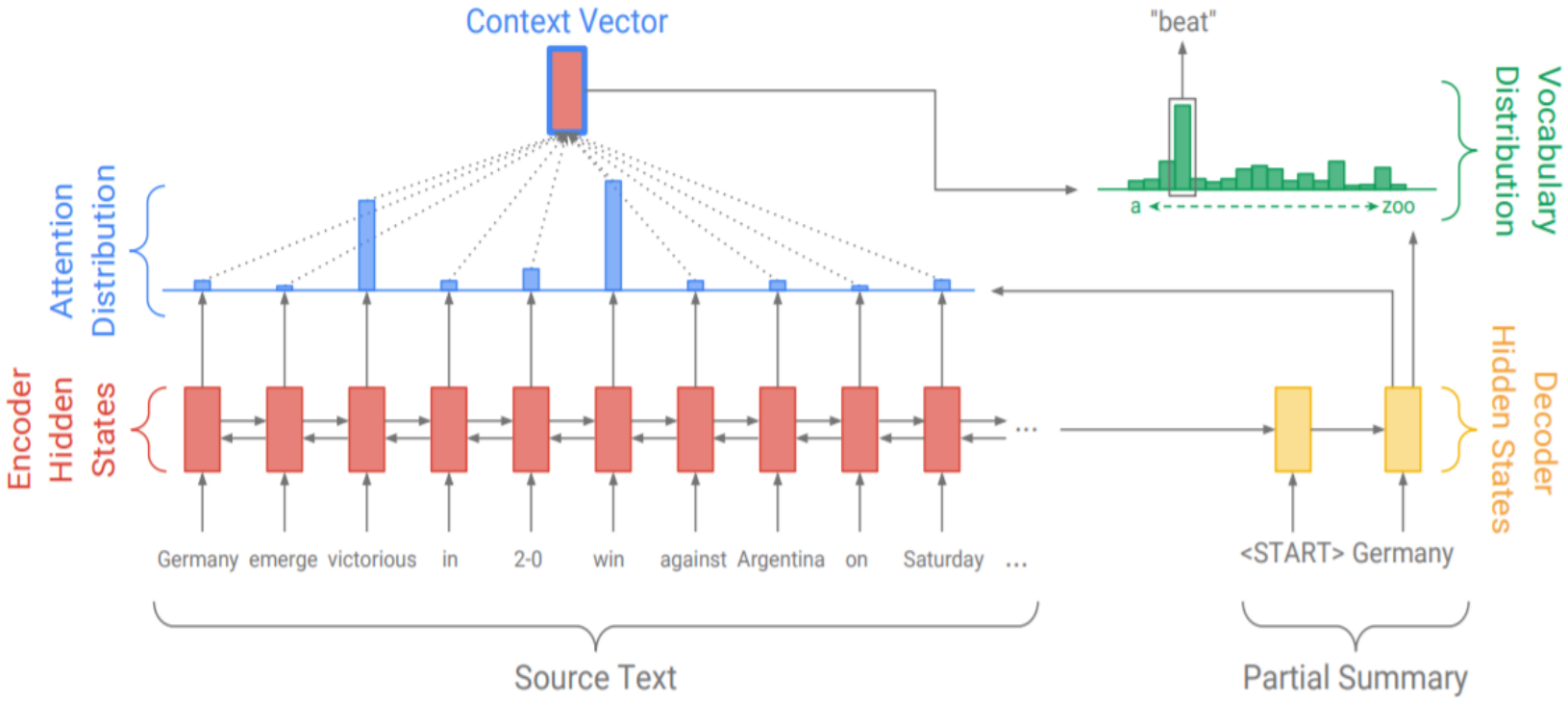
生成式摘要的关键技术

❑ 注意力机制

- 由于“长距离依赖”问题的存在，RNN到最后一个时间步输入单词时，已经丢失了相当一部分信息
- 此时编码生成的语义向量C同样也丢失了大量信息，就可能导致生成摘要准确性不足
- 与机器翻译相同，为了解决这个问题，在摘要生成的任务中同样使用了注意力机制

生成式摘要的关键技术

❑ 基于注意力机制的生成式文本摘要模型图



生成式摘要的关键技术

❑ 指针机制

– 序列到序列模型在应用于摘要生成时还存在两个主要的问题：
  - 难以准确复述原文的事实细节、无法处理原文中的未登录词(OOV)；
  - 生成的摘要中存在重复的片段

– 针对这两个问题，2017年see等人提出了pointer mechanism 以及coverage mechanism，近两年的生成式摘要模型大多以该为基础模型

生成式摘要的关键技术

❑ 指针机制（pointer mechanism）

- pointer-generator network是seq2seq模型和pointer network的混合模型
- 一方面通过seq2seq模型保持抽象生成的能力，另一方面通过pointer mechanism 直接从原文中取词，提高摘要的准确度和缓解OOV问题

See, Abigail, Peter J. Liu, and Christopher D. Manning. "Get to the point: Summarization with pointer-generator networks." arXiv preprint arXiv:1704.04368 (2017).

生成式摘要的关键技术

❑ 指针机制（pointer mechanism）

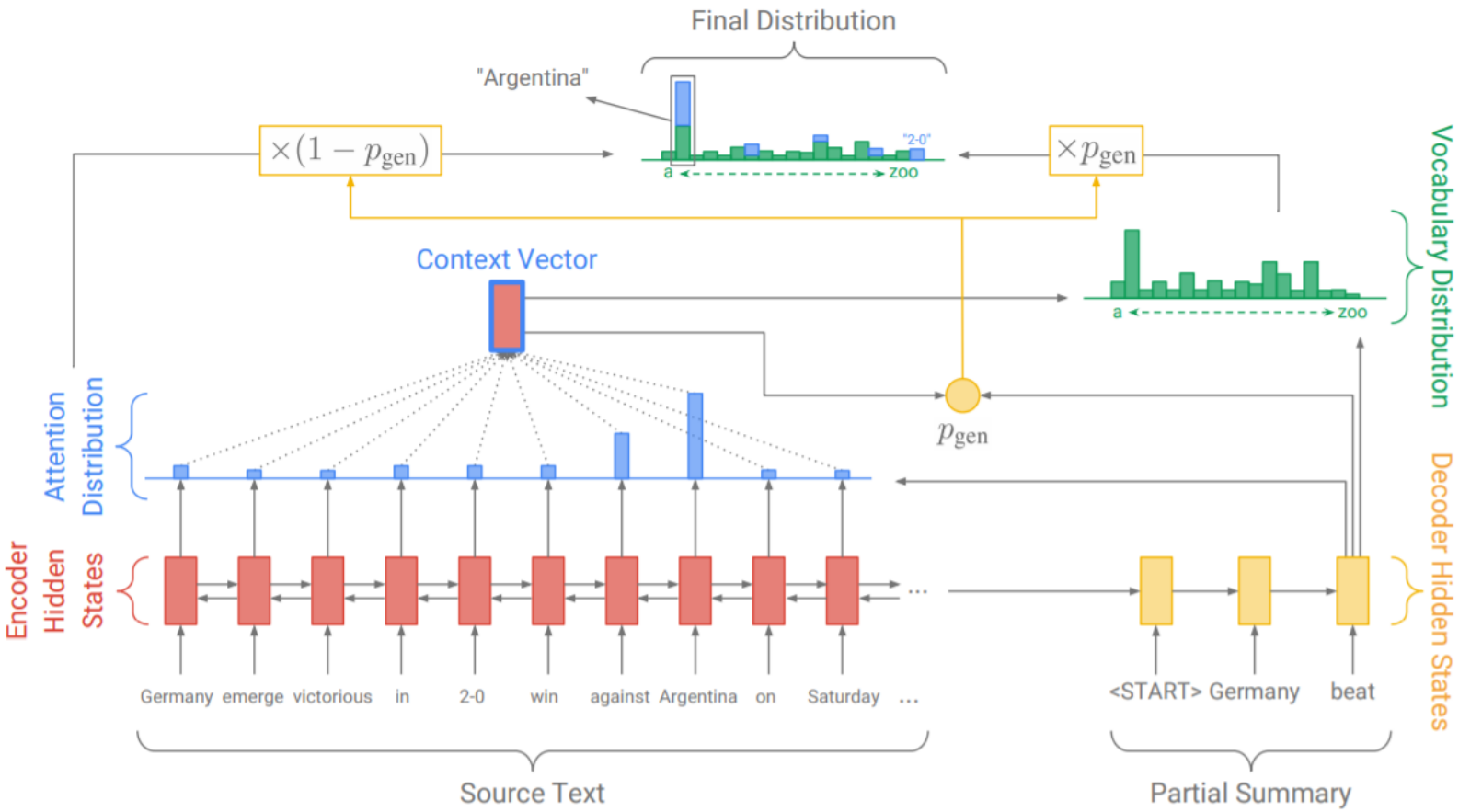
– 在预测的每一步，通过动态计算一个生成概率p_{gen}把二者软性地结合起来。这样，在每一步单词的概率分布计算如下：

$$P(w) = p_{\text{gen}} P_{\text{vocab}}(w) + (1 - p_{\text{gen}}) \sum_{i:w_i=w} a_i^t$$

– 其中，p_{gen}表示使用序列到序列模型生成该单词的概率。除此之外，会有(1- p_{gen})的概率，在每次摘要生成过程中，把原文动态地加入到词表中去，并且在每一步的预测过程中，相比于单纯的seq2seq模型，选原文中出现的词作为摘要的概率要更大一些

生成式摘要的关键技术

❑ 指针机制



生成式摘要的关键技术

❑ 覆盖机制(Coverage mechanism)

– 文本生成问题通常面临着重复问题，see等人将用于机器翻译的coverage机制应用到摘要生成问题上，取得了有效的结果。coverage机制就是在预测的过程中，维护一个coverage向量

$$c^t = \sum_{t'=0}^{t-1} a^{t'}$$

– Coverage向量表示过去每一步预测中attention分布的累积和，记录着模型已经关注过原文的哪些词并且让这个coverage向量影响当前步的attention计算。这样能有效避免模型持续关注到某些特定的词上

$$e_i^t = v^T \tanh(W_h h_i + W_s s_t + w_c c_i^t + b_{\text{attn}})$$

评价

- **目前文本摘要生成领域应用最广泛的自动评价指标是ROUGE(Recall-Oriented Understudy for Gisting Evaluation)。ROUGE是Lin提出的一个指标集合，包括一些衍生的指标，最常用的有ROUGE-n，ROUGE-L**
  - ROUGE-n：该指标旨在通过比较生成的摘要和参考摘要的n-grams（连续的n个词）评价摘要的质量。常用的有ROUGE-1，ROUGE-2，ROUGE-3
  - ROUGE-L：不同于ROUGE-n，该指标基于最长公共子序列（LCS）评价摘要。如果生成的摘要和参考摘要的LCS越长，那么认为生成的摘要质量越高

常用数据集

- 文本生成摘要技术最常用的公开数据集是**CNN/DailyMail**
- 该数据集是2016年IBM Watson公开的，用于多句文本摘要任务，为此后大量的相关工作提供了数据保障。
- 论文：**Abstractive text summarization using sequence-to-sequence rnns and beyond**

5 图像描述

概述

- 自动为图像生成对图像内容的自然语言描述被称为图像描述任务(Image captioning)，可将其划分为两个子任务：
  - 理解图像，正确获取图像相关信息
  - 基于对图像的理解生成语言描述
- 同时连接了计算机视觉(CV)和自然语言处理(NLP)两个领域。
- 要求：生成的语言描述应可信，需保证描述中涉及到的图像中的目标、属性信息、语义信息及位置关系信息等的正确性

概述

❑ 子任务1：对图像的理解（图像的特征提取）

- 传统方法：大多利用人工设计的特征提取算子提取图像的底层视觉特征（如几何、纹理、颜色等特征），并组合构成图像高维整体特征。
- 特征提取算法设计过于依赖经验和运气；
- “语义鸿沟”问题(semantic gap)的存在，底层视觉特征无法对高维语义特征进行有效准确的表达。
- 深度学习方法：卷积神经网络

概述

❑ 子任务2：生成图像的描述

- 传统方法：基于检索(Retrieval-Based)和基于模板(Template-Based)的方法生成图像的描述
- 缺陷：基于检索的方法提取出来的描述可能并不完全符合图像；基于模板的方法生成的图像描述可能显得过于生硬，缺乏多样性
- 深度学习方法：循环神经网络（LSTM、GRU）等

传统方法

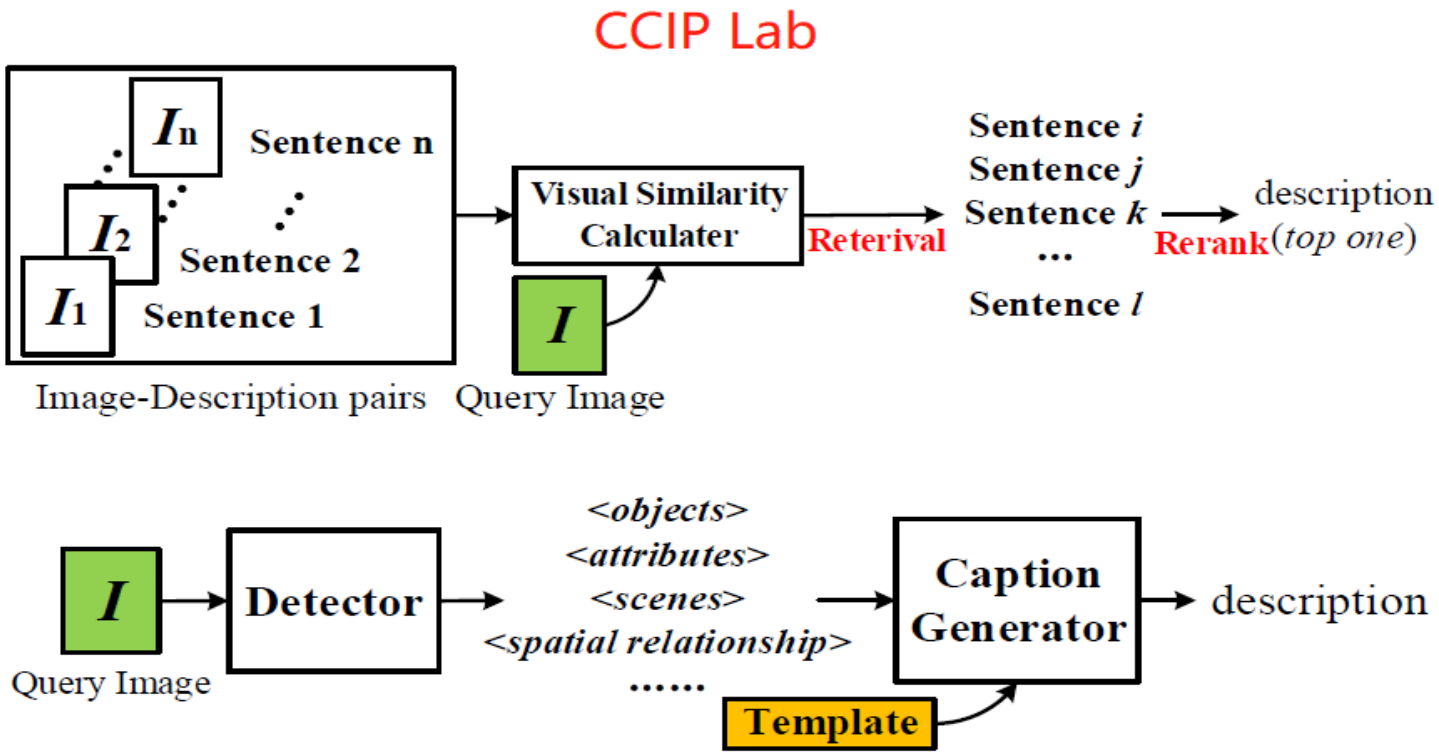


Figure 1: The fundamental process of traditional methods: Reterival-Based Method (top). Template-Based Method (bottom).

传统方法----基于检索的方法

- 思路：给定待检索图像*I*，从一系列给定的文本描述库（即数据集：图像+描述）中检索出与该图像*I*最为匹配的一系列图像所对应的描述，并从中再选取最合适的描述作为待检索图像*I*的描述
- 缺陷：生成的描述质量好坏很大程度上受制于给定的文本描述库，文本描述库由人为建立，因此可保证语句流畅和语法正确；但是文本描述库需要足够大以保证描述内容及语义的准确性，但实际上这种方法往往不能覆盖足够丰富的场景，生成的描述可能并不能正确适应新图像出现的目标和场景

传统方法----基于检索的方法

❑ 参考阅读文献

– [Ordonez et al., 2011] V. Ordonez, G. Kulkarni, and T. L. Berg. Im2text: Describing images using 1 million captioned photographs. In NIPS, pages 1143–1151, 2011.

– [Hodosh et al., 2015] M. Hodosh, P. Young, and J. Hockenmaier. Framing image description as a ranking task: Data, models and evaluation metrics (extended abstract). In IJCAI, pages 4188–4192, 2015.

传统方法----基于模板的方法

- 思路：给定待检索图像*I*，先从图像中检索一些实体目标、属性或语义目标，然后利用指定的语法规则将检索得到的信息进行组合或者将检索到的相关信息填入到预定义的语句模板的空白中，从而得到待检索图像*I*的描述
- 缺陷：无法生成可变长度的图像描述，限制了不同图像描述之间的多样性，描述显得呆板不自然；另一方面性能也受制于图像目标检测的结果，因此生成的描述可能会遗漏图像细节

传统方法----基于模板的方法

❑ 参考阅读文献

– [Li et al., 2011] S. Li, G. Kulkarni, T. L. Berg, A. C. Berg, and Y. Choi. Composing simple image descriptions using web-scale n-grams. In CoNLL, pages 220–228, 2011.

– [Kulkarni et al., 2011] G. Kulkarni, V. Premraj, S. Dhar, S. Li, Y. Choi, A. C. Berg, and T. L. Berg. Baby talk: Understanding and generating simple image descriptions. In CVPR, pages 1601–1608, 2011

深度学习方法

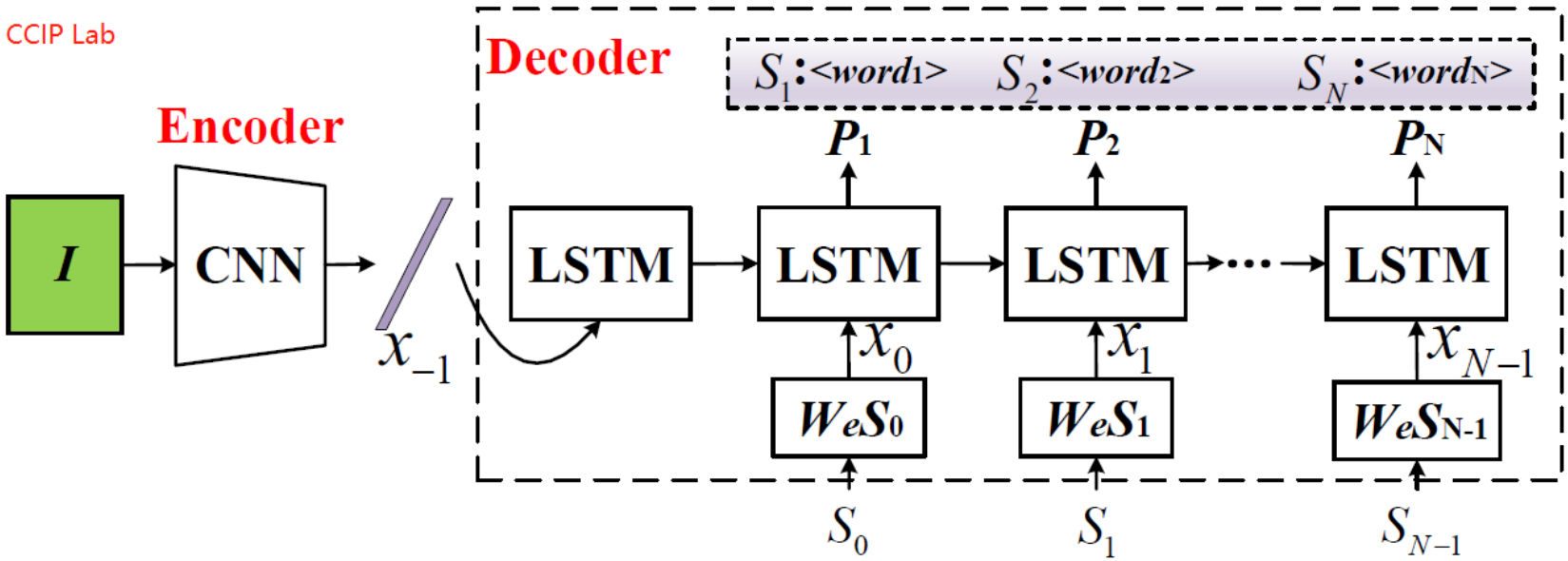
- 受机器翻译Encoder-Decoder模型结构启发，深度学习方法普遍使用CNN作为Encoder对图像进行编码，使用RNN作为Decoder对编码信息进行解码为语言描述，把图像描述任务视为一个从图像语言到自然语言的“翻译”任务

深度学习方法

- 图像描述深度学习方法开篇经典论文
  - [Vinyals et al., 2015] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan. Show and tell: A neural image caption generator. In CVPR, pages 3156–3164, 2015.
  - [Xu et al., 2015] K. Xu, J. Ba, R. Kiros, K. Cho, A. C. Courville, R. Salakhutdinov, R. S. Zemel, and Y. Bengio. Show, attend and tell: Neural image caption generation with visual attention. In ICML, pages 2048–2057, 2015.

深度学习方法---show and tell

- **Show and Tell应该是最早将机器翻译中Encoder-Decoder结构应用于图像描述任务中的一项工作。模型结构如图所示**



其中，S_0为描述开始标记<START>，S_i为第i步生成的描述单词（在词汇表中的编号），S_N为描述开始标记<END>，x_i为第i步生成的描述单词的词嵌入表示（通过词嵌入矩阵W_e和S_i获取）

深度学习方法---show and tell

❑ Show and Tell模型的流程

- 采用卷积神经网络（具体为GoogLeNet Inception V3）作为Encoder部分，将图像编码为固定长度的向量，作为图像特征映射x_{-1}；
- 将图像特征映射x_{-1}送入作为Decoder部分的LSTM，逐步生成图像描述

$$x_{-1} = \text{Encoder}(I) \tag{1}$$

$$x_t = W_e S_t, \qquad t \in \{0, \ldots, N-1\} \tag{2}$$

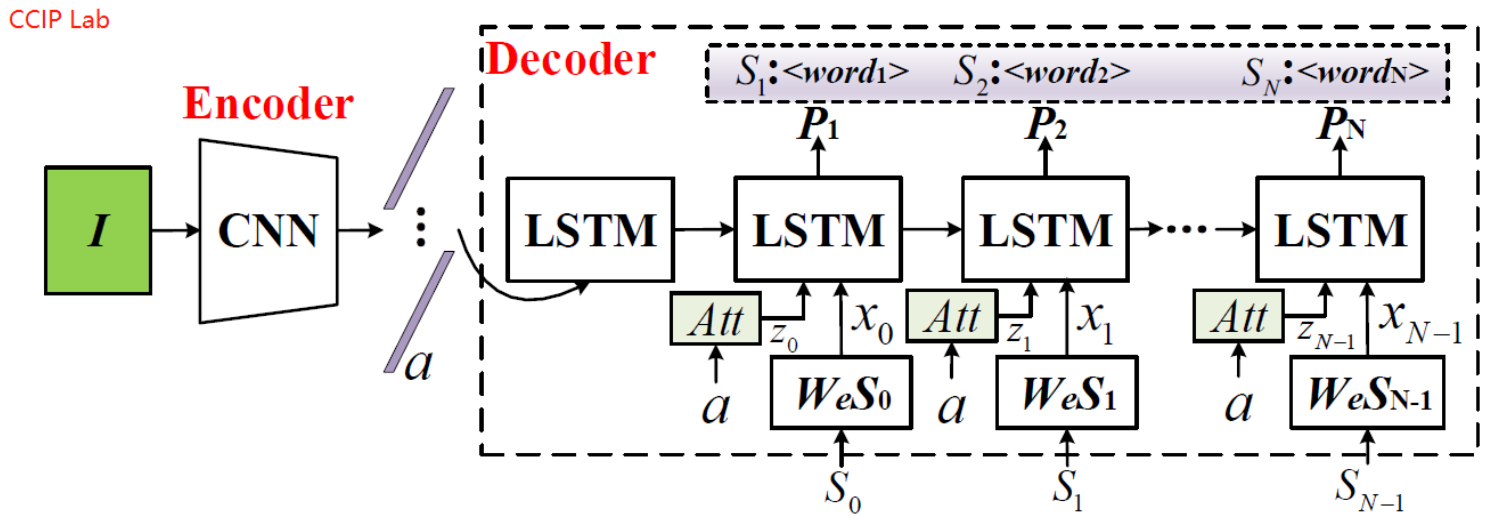
$$p_{t+1} = \text{Decoder}(x_t), \qquad t \in \{0, \ldots, N-1\} \tag{3}$$

深度学习方法---show and tell

- 注意：图像特征映射x_{-1}仅在最开始作为LSTM的输入（LSTM的初始状态可设为全零），可视为对LSTM进行初始化计算第一步的状态，而后LSTM的输入均为描述单词的词嵌入向量及上一步的LSTM状态输出
- 论文的解释是，在每一步都输入图像特征映射没有得到效果提升，反而容易导致过拟合，但是在此之后一些论文的工作中，又在每个时间步输入了图像特征映射

深度学习方法---show, attend and tell

- Show, attend and Tell是对show and tell的一个扩展。在基本的Encoder-Decoder结构上引入了视觉注意力机制（attention mechanism），可以在Decoder生成图像描述过程中动态关注图像的显著区域。模型结构如图所示



深度学习方法---show, attend and tell

❑ Show, attend and Tell模型的流程

– 采用卷积神经网络（具体为VGGNet）作为Encoder部分，将图像编码为L个K维的向量，每个向量对应图像的一部分区域（实际上就是CNN的中间层响应激活输出，假设其维度为[14，14，512]，则L=14*14，K=512）

$$a = \{a_1, ..., a_L\}, a_i \in \mathbb{R}^K \qquad (4)$$

深度学习方法---show, attend and tell

❑ Show, attend and Tell模型的流程

– 在每一步基于图像特征向量a计算该步上下文向量$z_t = \sum_{i=1}^{L} \alpha_{ti} a_i$（Soft注意力机制），送入作为Decoder部分的LSTM，逐步生成图像描述

– 其中$\alpha_{ti} \in \mathbb{R}^L$为第$t$步注意力概率向量，且满足$\sum_{i=1}^{L} \alpha_{ti} = 1$。可通过简单的MLP和Softmax激活函数进行计算

$$\alpha_{ti} \propto \exp\{f_{\text{att}}(a_i, m_{t-1})\}$$

深度学习方法---show, attend and tell

❑ Show, attend and Tell模型

$$a = \text{Encoder}(I) \tag{6}$$

$$z_t = \sum_{i=1}^{L} \alpha_{ti} a_i, \qquad \alpha_{ti} \in \mathbb{R}, a_i \in \mathbb{R}^K \tag{7}$$

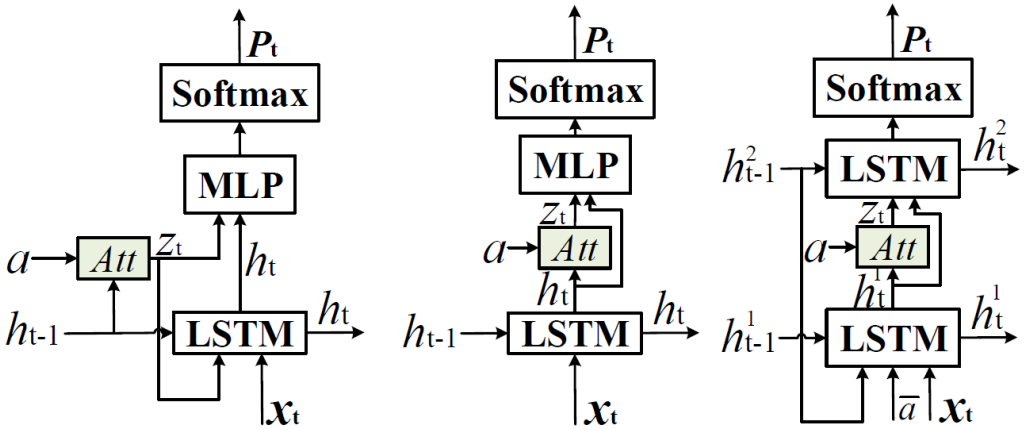
$$x_t = W_e S_t, \qquad t \in \{0, ..., N-1\} \tag{8}$$

$$p_{t+1} = \text{Decoder}(x_t, z_t), \qquad t \in \{0, ..., N-1\} \tag{9}$$

注意：在Show, attend and tell模型中，LSTM的输入包含上一步生成描述单词的词嵌入向量、上一步的LSTM状态输出以及基于注意力机制计算的上下文特征向量。此外，在论文中，还介绍了Hard注意力，但在之后的论文中，Soft注意力使用较多。

深度学习方法---show, attend and tell

- **Show, attend and Tell注意力层结构如下图（左）所示。注意力层利用LSTM上一步的状态输出h_{t-1}计算上下文向量z_t，并作为第t步LSTM的输入**
  - 也有部分论文进行了改进，他们认为第t步的单词输出就应该与LSTM第t步的状态输出更相关，因此计算的上下文向量z_t直接用于计算单词概率（中）。还有使用多层LSTM用于单词概率生成（右）



数据集

- **Flickr8K：**
  - 共8092张图像，有University of Illinois研究人员从Flickr.com收集，通过Amazon Mechanical Turk提供的众包服务获取对应的图像描述，每张图像包含5个不同描述，对图像中人物、目标、场景和活动进行了准确描述。描述平均长度为11.8个单词
  - https://forms.illinois.edu/sec/1713398
- **Flickr30K：**
  - 对Flickr8K的扩展，包含31783张图像，158915条描述，其余同上。
  - http://shannon.cs.illinois.edu/DenotationGraph/

数据集

❑ MS COCO

- MS COCO可用于目标检测、实例分割和图像描述等任务。2014年发布部分包含82783张训练集图像、40504张验证集图像和40775张测试集图像，但是测试集图像描述注释非公开可用，因此大多会对训练集和验证集进行二次划分，而不使用其测试集。http://cocodataset.org/
- GitHub开源工具包：https://github.com/tylin/coco-caption

❑ Visual Genome

- 大规模数据集，包含超过108K张图像和更多图像属性及目标之间的交互关系信息，对于引入了语义关系、空间关系等的图像描述任务，可采用VG进行预训练。http://visualgenome.org/

评价指标

❑ BLEU-{1,2,3,4}

– 起初用于机器翻译质量评估，核心思想在于“待检测语句越接近参考语句，则越好”。通过对比待检测语句和参考语句在n-gram层面的相似度进行评估，不考虑语法正确性、同义词和相近表达，仅在较短语句下比较可信

❑ METEOR

– 常用于机器翻译评估，首先对待检测语句与参考语句进行对齐（单词精准匹配、snowball stemmer词干匹配、基于WordNet数据集近义词匹配等），然后基于对齐结果计算相似度得分，解决了BLEU存在的一些缺陷

评价指标

❑ CIDEr

– 针对图像描述任务提出，将每个语句视为一篇文档，表示为tf-idf向量形式，计算待检测语句和参考语句之间的余弦相似度进行评估。

❑ SPICE

– 针对图像描述任务提出，基于图的语义表示对描述中的目标、属性和关系进行编码，比之前的基于n-gram的度量方法能更准确的比较模型之间的优劣

6 中英文术语对照

中英文术语对照

- 语言模型：Language Model
- 词嵌入：Word Embedding
- 数据平滑：Data Smoothing
- 机器翻译：Machine Translation, MT
- 源语言：Source language
- 目标语言：Target language
- 序列到序列模型： Sequence to Sequence, Seq2Seq

中英文术语对照

- 机器阅读理解: Machine Read Comprehension, MRC
- 预训练模型：Pre-Training model
- 指针网络：Pointer Network
- 自动文本摘要: Automatic Text Summarization
- 注意力机制：Attention mechanism
- 指针机制：Pointer mechanism
- 覆盖机制：Coverage mechanism

中英文术语对照

- 图像描述：Image Captioning
- 特征提取：Feature Extraction
- 语义鸿沟：Semantic Gap
- 基于检索的方法：Retrieval-based method
- 基于模板的方法：Template-based method

谢谢！