Lecture 14

1. RNN的缺陷 (Transformer的motivation)

序列计算抑制了并行化

没有对长期和短期依赖关系进行显式建模

我们想要对层次结构建模

RNN (顺序对齐的状态) 看起来很浪费

2. 卷积神经网络

并行化很简单

利用短期依赖

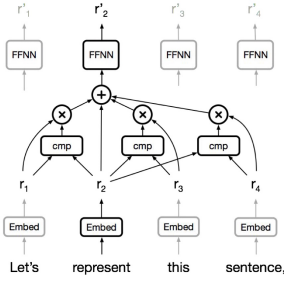
不同位置的交互距离是线性或对数的

长期依赖需要多层

3. Self-Attention

在NMT中, encoder和decoder间的Attention至关重要

因此考虑将attention用于表示



- 任何2个位置间路径长度都是常数级
- 门控/乘法的交互
- 可以并行化 (每层)

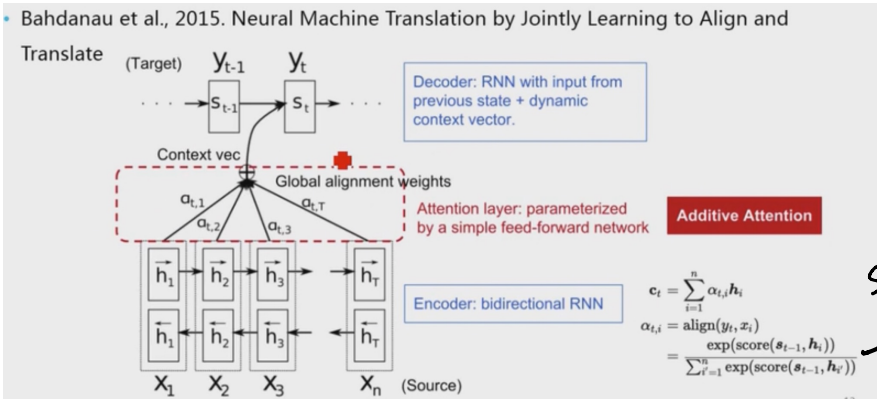
什么是Attention

深度学习中的注意力可被广义理解为表示重要性的权重向量。

为预测或推断一个元素，如句子中的单词或图像中的像素，使用注意力权重来估计其它元素与其相关的强度，并将由注意力权重加权的值的总和作为计算最终目标的特征。

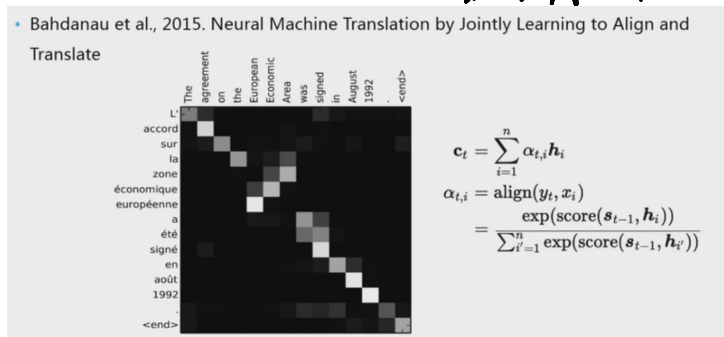
- Step 1：计算其他元素与待预测元素的相关性权重
- Step 2：跟据相关性权重对其他元素进行加权求和

传统的 seq2seq 的问题



softmax函数 → 权重和为1 经过归一化

score：可以具体选择



常见的score函数

Attention

Name	Alignment score function	Citation
Content-base attention	$\text{score}(\boldsymbol{s}_t, \boldsymbol{h}_i) = \text{cosine}[\boldsymbol{s}_t, \boldsymbol{h}_i]$	Graves2014
Additive(*)	$\text{score}(\boldsymbol{s}_t, \boldsymbol{h}_i) = \mathbf{v}_a^\top \tanh(\mathbf{W}_a[\boldsymbol{s}_t; \boldsymbol{h}_i])$	Bahdanau2015
Location-Base	$\alpha_{t,i} = \text{softmax}(\mathbf{W}_a \boldsymbol{s}_t)$ Note: This simplifies the softmax alignment to only depend on the target position.	Luong2015
General	$\text{score}(\boldsymbol{s}_t, \boldsymbol{h}_i) = \boldsymbol{s}_t^\top \mathbf{W}_a \boldsymbol{h}_i$ where $\mathbf{W}_a$ is a trainable weight matrix in the attention layer.	Luong2015
Dot-Product	$\text{score}(\boldsymbol{s}_t, \boldsymbol{h}_i) = \boldsymbol{s}_t^\top \boldsymbol{h}_i$	Luong2015
Scaled Dot-Product(^)	$\text{score}(\boldsymbol{s}_t, \boldsymbol{h}_i) = \frac{\boldsymbol{s}_t^\top \boldsymbol{h}_i}{\sqrt{n}}$ Note: very similar to the dot-product attention except for a scaling factor; where n is the dimension of the source hidden state.	Vaswani2017

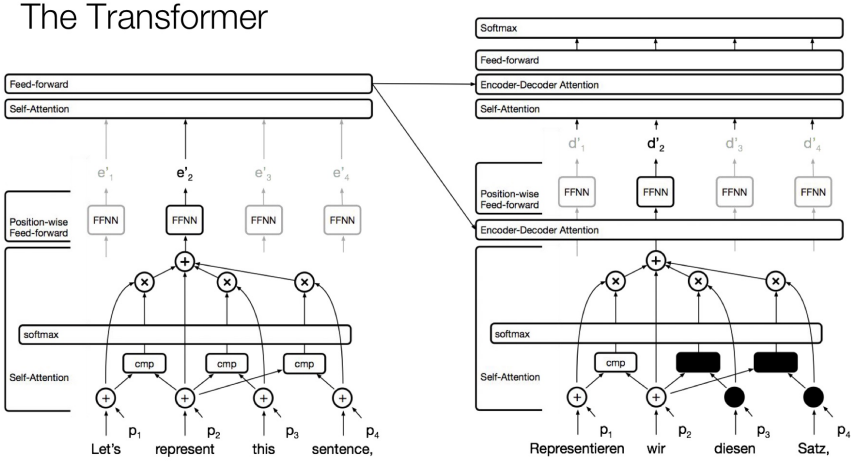
attention的变体的定义

Name	Definition	Citation
Self-Attention(&)	Relating different positions of the same input sequence. Theoretically the self-attention can adopt any score functions above, but just replace the target sequence with the same input sequence.	Cheng2016
Global/Soft	Attending to the entire input state space.	Xu2015
Local/Hard	Attending to the part of input state space; i.e. a patch of the input image.	Xu2015; Luong2015

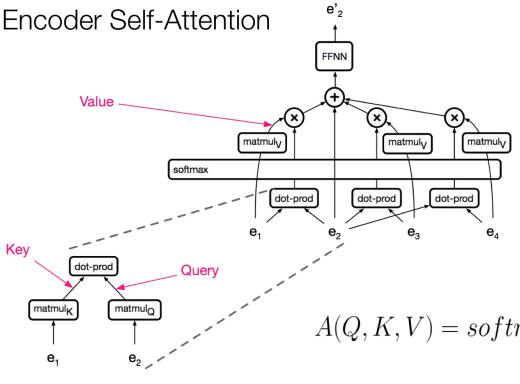
也叫 intra-attention

4. Transformer

The Transformer



Encoder Self-Attention

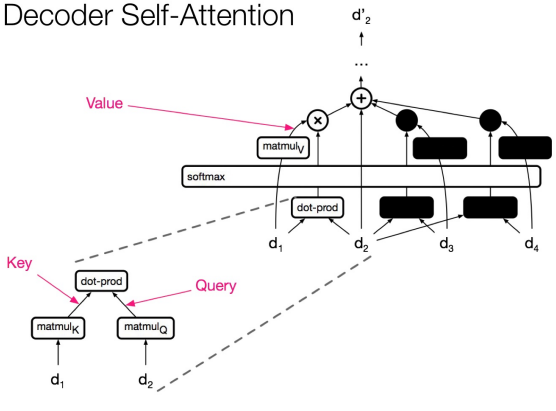


$$A(Q,K,V) = softmax(\frac{QK^T}{\sqrt{d_k}})V$$

Attention的计算复杂度低于RNN、Convolution

k的维度

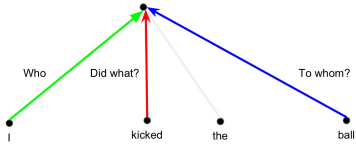
Decoder Self-Attention



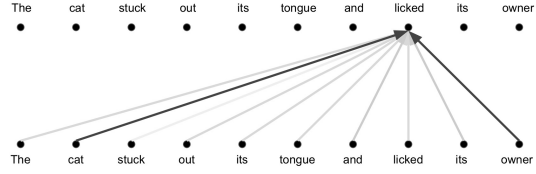
问题：

在CNN的convolution中，可以通过不同的filter，获得不同的信息

Convolutions



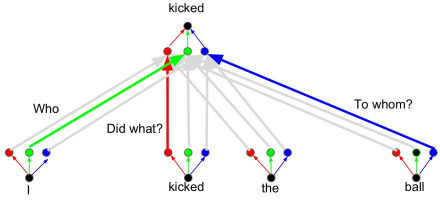
Attention: a weighted average



但Attention仅是加权平均，不能提取多方面信息

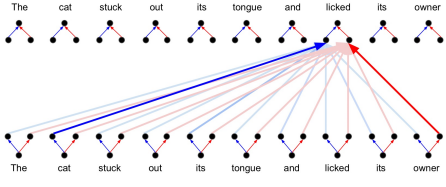
Multihead Attention

通过使用多个Attention head来解决



Multi-head Attention

Parallel attention layers with different linear transformations on input and output.



模型结构

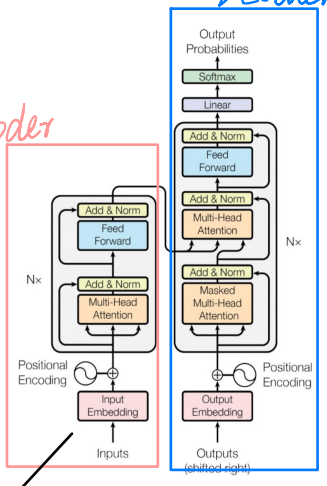


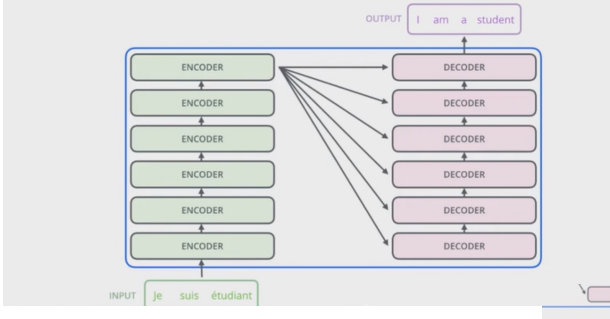
Figure 1: The Transformer - model architecture.

和大多数 seq2seq 模型一样，transformer 的结构也是由 encoder 和 decoder 组成

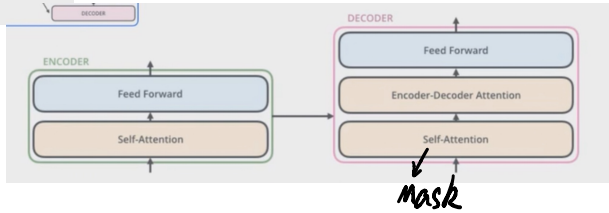
Positional Encoding [sequence length, embedding dim]

Transformer 抛弃了 RNN，而 RNN 最大的优点就是在时间序列上对数据的抽象。将 encoding 后的数据与 embedding 数据求和，加入了相对位置信息

[batch_size, sequence length, embedding dimension]



宏观理解

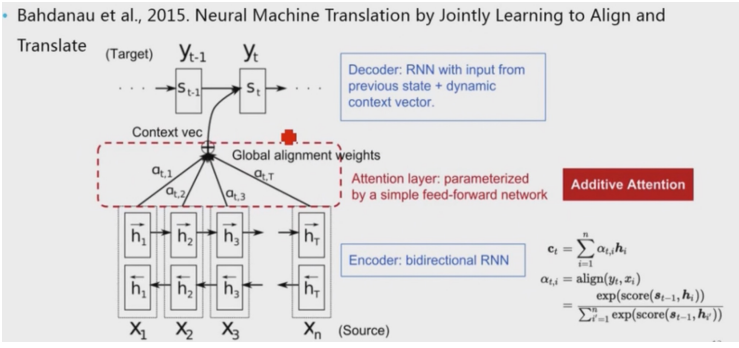


Encoder

由 n=6 个相同的 layer 组成（图中左侧的单元）

每个 layer 由 2 个 sub-layer 组成，分别是 Multi-Head Attention 和 fully connected feed-forward network。其中每个 sub-layer 都加了 residual connection 和 normalization

Attention原理



Attention 可抽象为：

$$\text{Attention}(Q,K,V)=\text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

权重

Scaled Dot-Production Attention

Q：上一时刻隐状态的输出 S_{t-1}

K：可以看成双向LSTM隐状态的拼接，即 X 对应的编码

QK^T：Q与K的相似度

Softmax：归一化

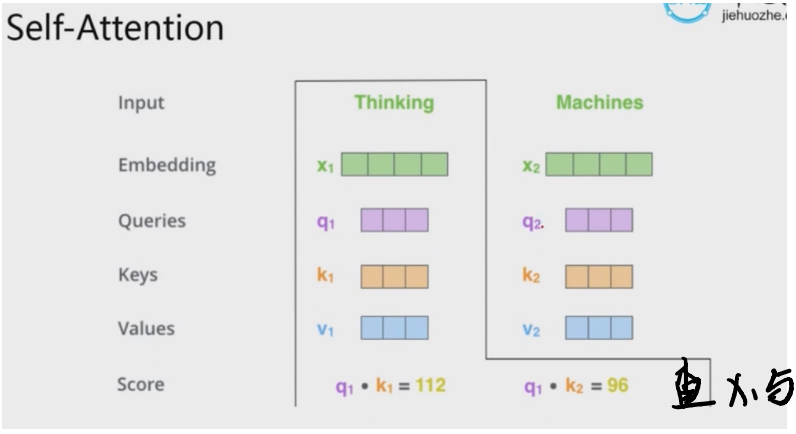
Q、K、V来自同一序列，则是self-Attention

W矩阵随机初始化

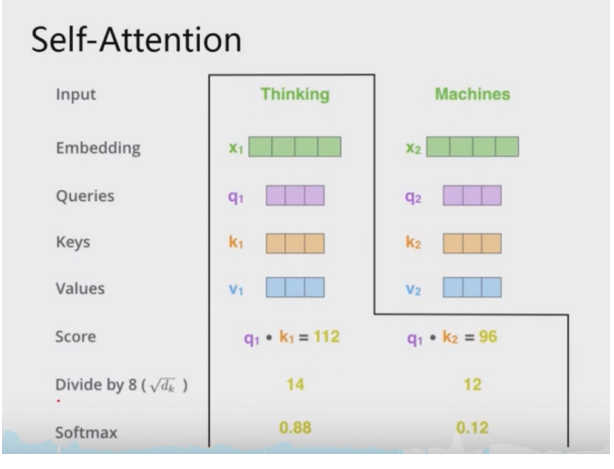
$q_1 = x_1 W^Q$

$k_1 = x_1 W^K$

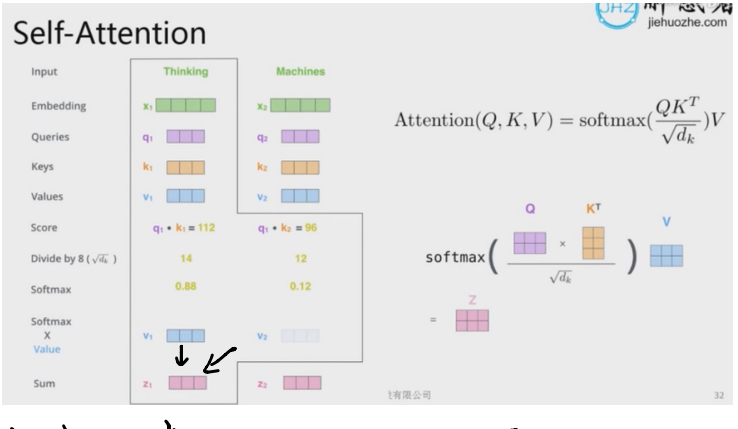
$V_1 = x_1 W^V$



查小与其他词的关系，用Q乘

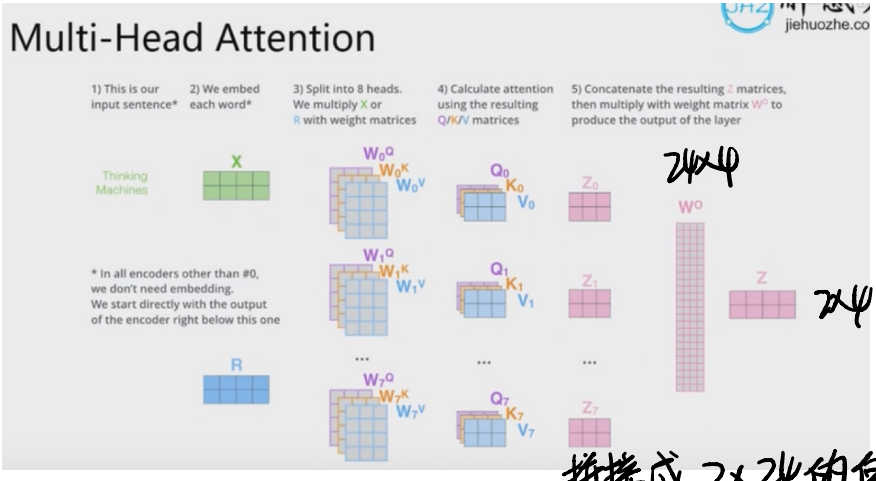


dk：K的维度



在其他情况下，Q、K不同

Multi-Head Attention



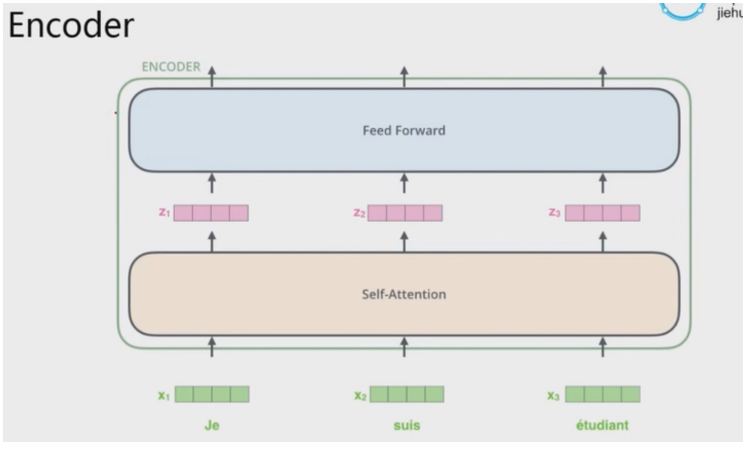
8个

2x4

2x4

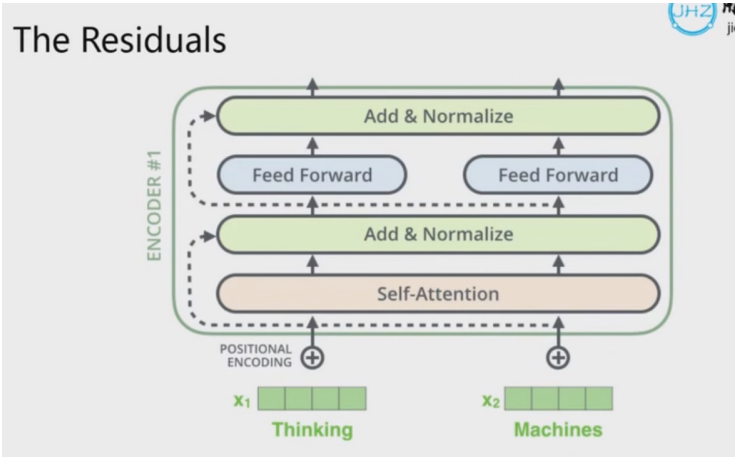
拼接成2x24的向量(横着拼)

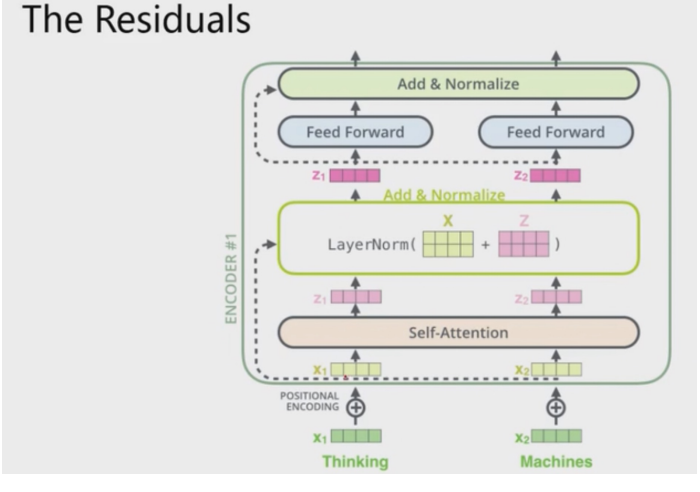
Encoder



· Feed Forward 是个全连接层，主要提供非线性变换

· residual connection：残差连接，避免梯度消失

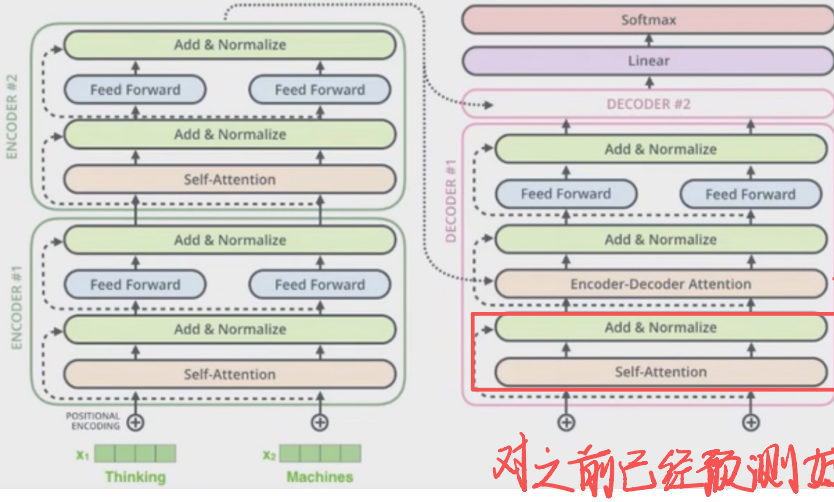




相加

norm：归一为标准正态分布，加速收敛（优化的技巧）

模型结构细节



(Decoder)
Q: Out put
K,V: Encoder

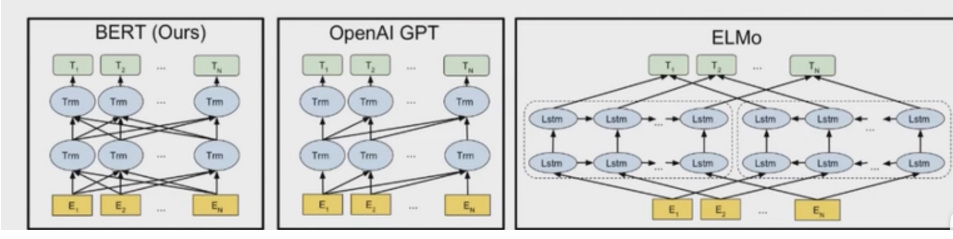
对之前已经预测好的输出
$(y_0, \cdots y_{t-1})$做 self-Attention

5. BERT

传统 word Embedding 的预训练表示是上下文无关的(如 word2vec)

BERT: Bidirectional Encoder Representations from Transformers

是一种预训练语言表示的方法，上下文相关



BERT 可以应用到各个 nlp 领域中，通用算法

- 从无标签文本中训练，运用上下文语境
- 在实际任务中，只需额外添加一个 output 层，根据自己的任务进行 finetune
- 在很多任务上都达到了顶尖表现 (state-of-art)

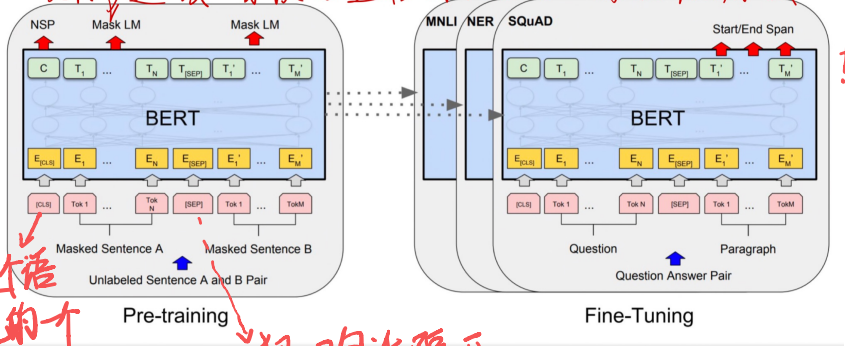
Mask Model

随机隐藏掉一些词，根据上下文预测出 (类似完形填空)

BERT框架包括2步：pre-training　fine-tuning

pre-training：在无标注文本上训练（Google）

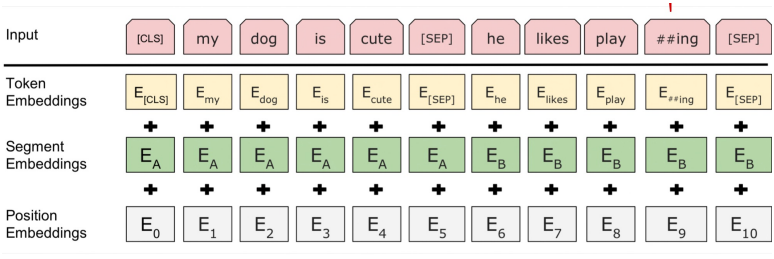
fine-tuning：用pre-training训练好的参数来初始化，用任务相关的有标签文本进行训练



- Train on Wikipedia + BookCorpus
- Train 2 model sizes: 训练了2种size的模型
  - BERT-Base: 12-layer, 768-hidden, 12-head
  - BERT-Large: 24-layer, 1024-hidden, 16-head
- Trained on 4x4 or 8x8 TPU slice for 4 days

layer：不是一层，是一个block

词编码方式



词的编码

标志是第一句话还是第二句

对词的位置进行编码

Token embeddings are word pieces
Learned segmented embedding represents each sentence
Positional embedding is as for other Transformer architectures

60

Pre-traing 训练策略

2种无监督方法：Masked LM 和 Next Sentence Prediction (NSP)

Masked LM

随机选择一些词隐藏起来，然后预测这些隐藏位置的词

15%的隐藏比例最好

mismatch问题：在fine-tuning模型中没有[MASK]这个特殊字符，但pre-traing模型中有。

> Although this allows us to obtain a bidirectional pre-trained model, a downside is that we are creating a mismatch between pre-training and fine-tuning, since the [MASK] token does not appear during fine-tuning. To mitigate this, we do not always replace "masked" words with the actual [MASK] token. The training data generator chooses 15% of the token positions at random for prediction. If the i-th token is chosen, we replace the i-th token with (1) the [MASK] token 80% of the time (2) a random token 10% of the time (3) the unchanged i-th token 10% of the time. Then, T_i will be used to predict the original token with cross entropy loss. We compare variations of this procedure in Appendix C.2.

Next Sentence Prediction

- To learn *relationships* between sentences, predict whether Sentence B is actual sentence that proceeds Sentence A, or a random sentence

预测A的下一句是否是B

Sentence A = The man went to the store.
Sentence B = He bought a gallon of milk.
Label = IsNextSentence

Sentence A = The man went to the store
Sentence B = Penguins are flightless.
Label = NotNextSentence

ELOM. BERT. GPT —李宏毅

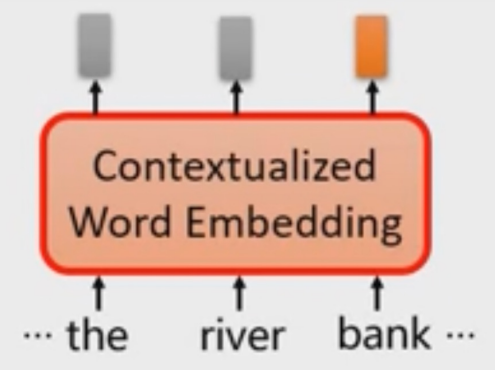
1. 问题:

在传统词向量中，每一个 word type 有一个 embedding。但同一个词有不同的意思（word token），它们使用同一个向量。即没有解决一词多义的问题。

2. Contextualized word embedding

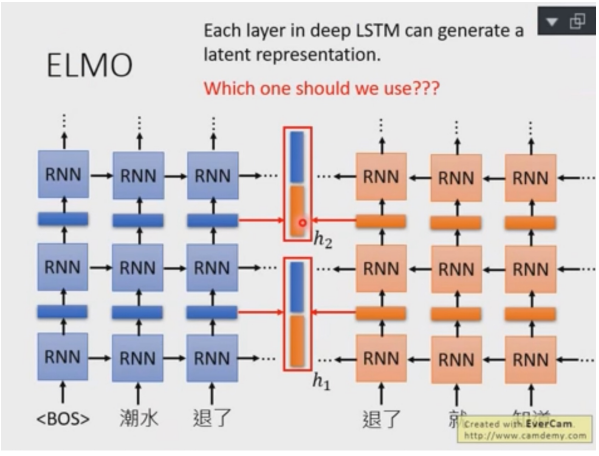
每个 word token 有一个 embedding，即一个词义有一个词向量

如何获得这样的词向量？ 依赖于词的上下文

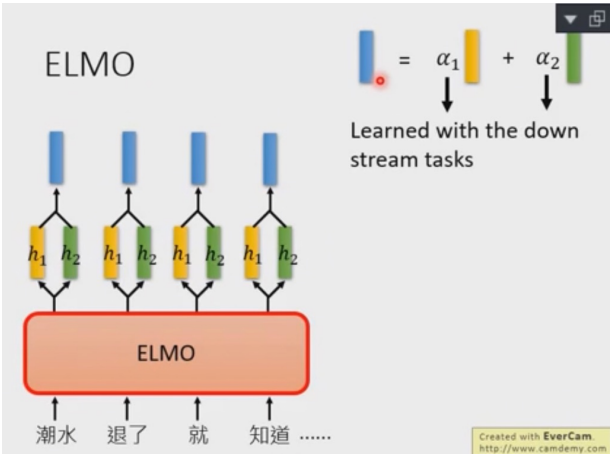


3. ELMo : Embeddings from Language Model

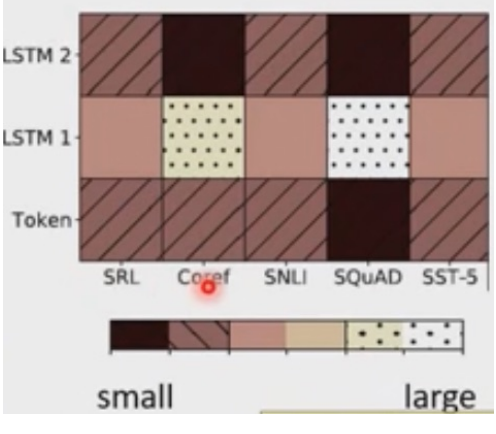
基于 RNN 的语言模型，在大量文本中训练



在深层LSTM中训练，应该取哪一层作为词向量
ELMo：全都利用



假设使用了2层LSTM
第一层给出的embedding是h_1，第二层是h_2
它们的加权和即为最终的词向量
权重α_1，α_2怎么得到？
随机初始化，在具体的下游任务中训练、优化



Token：原来的词向量（不是基于上下文的）

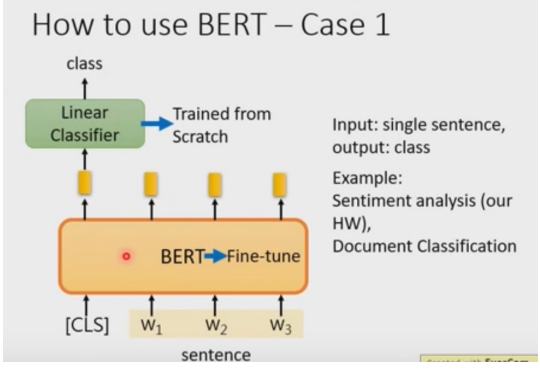
4. BERT Bidirectional Encoder Representations from Transformers

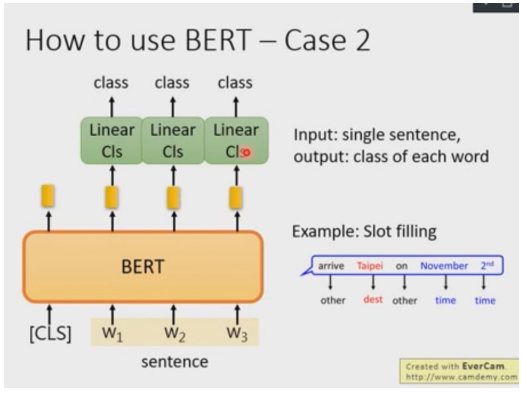
中文按字来表示比较好（词太多，维度太大）

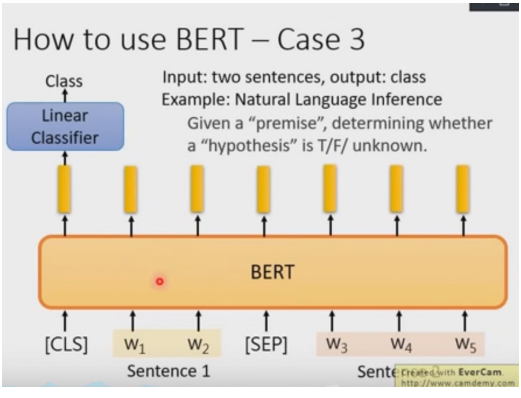
训练方法

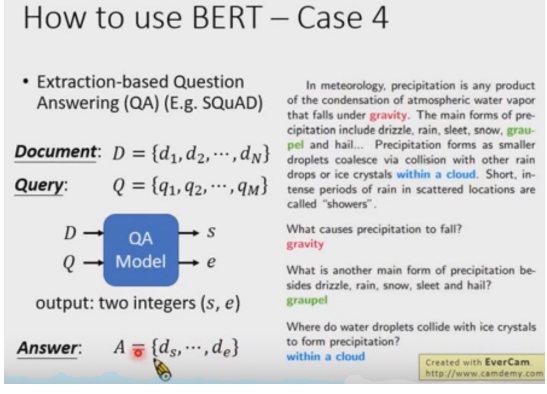
① Masked LM

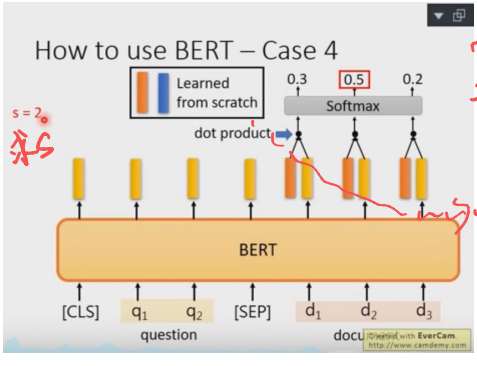
② Next Sentence Prediction





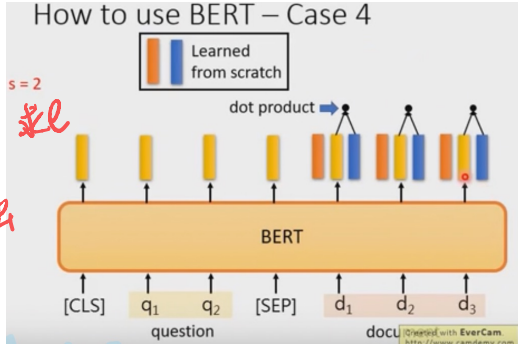




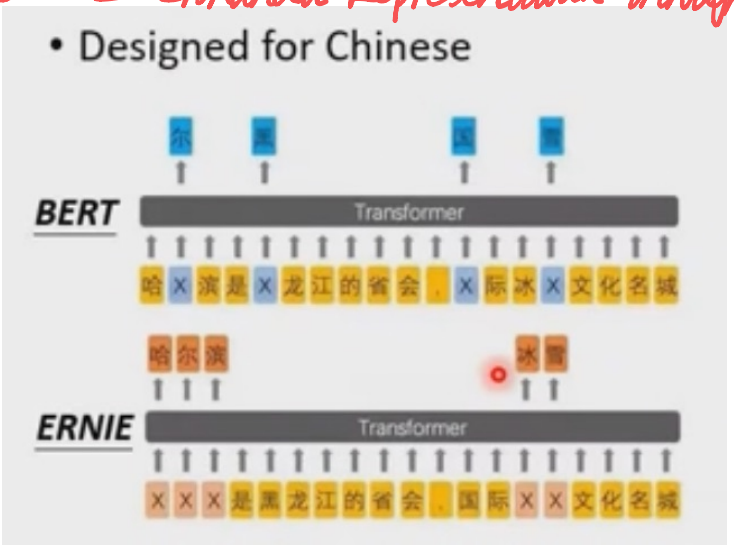


如果e/s，无答案

2个向量训练得出



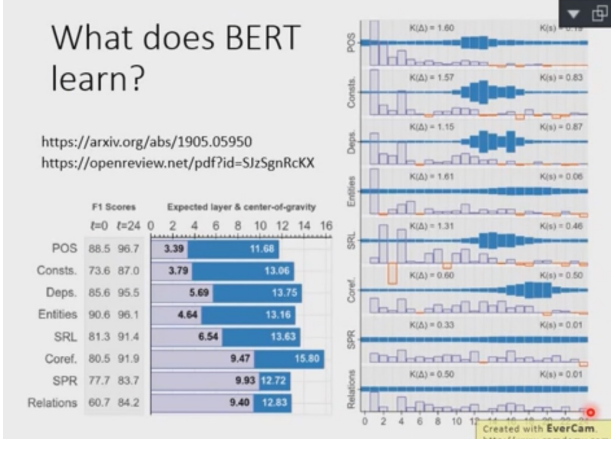
ERNIE: Enhanced Representation through Knowledge Integration



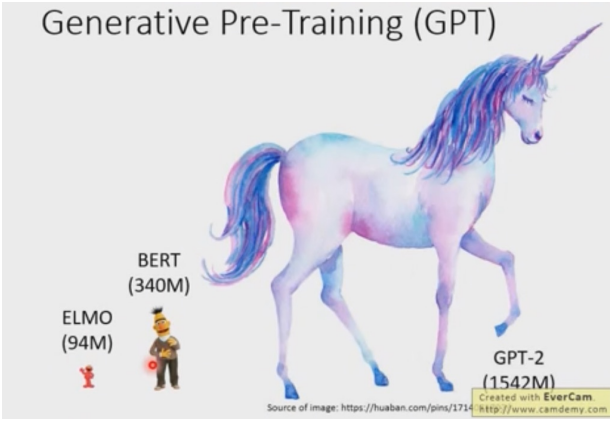
为中文设计

一次 mask 一个词汇，而不是单个字

每一层 bert 学到了什么

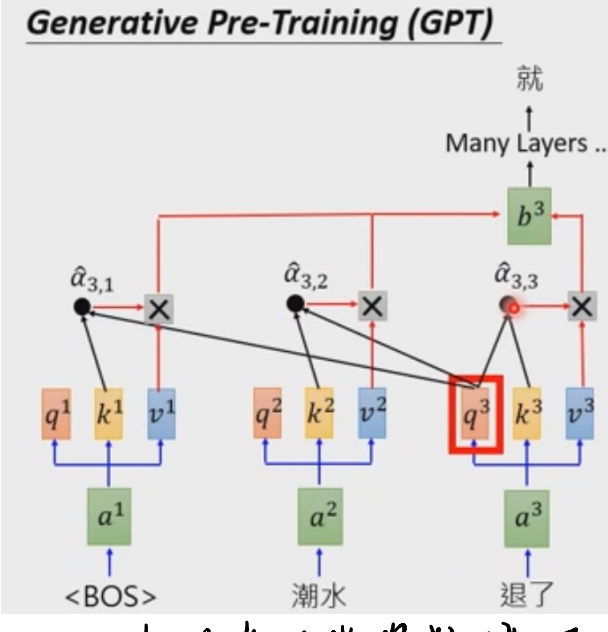


5. Generative Pre-Training (GPT)



模型很大

Transformer Decoder



GPT2在没有训练资料时，可以进行阅读理解、摘要、翻译

lecture 15

1. Recap: LMs and decoding algorithms

Natural Language Generation NLG

指任何生成新文本的任务

它是机器翻译、(提取式)摘要、对话、创造性写作(故事、诗歌)、自由问答(不是提取式)、自动生成字幕等任务的子组件.

- **Language Modeling**: the task of predicting the next word, given the words so far

$$P(y_t|y_1,\dots,y_{t-1})$$

- A system that produces this probability distribution is called a **Language Model**
- If that system is an RNN, it's called a **RNN-LM**

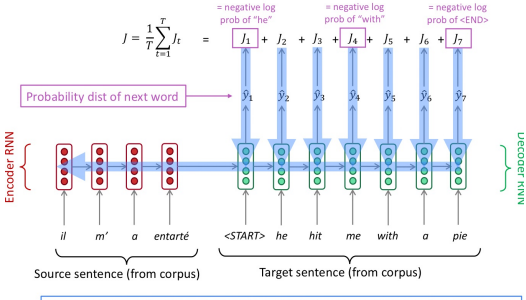
- **Conditional** Language Modeling: the task of predicting the next word, given the words so far, and also some other input x

$$P(y_t|y_1,\dots,y_{t-1},x)$$

- Examples of conditional language modeling tasks:
  - Machine Translation (x=source sentence, y=target sentence)
  - Summarization (x=input text, y=summarized text)
  - Dialogue (x=dialogue history, y=next utterance)
  - …

Recap: training a (conditional) RNN-LM

This example: Neural Machine Translation



During training, we feed the gold (aka reference) target sentence into the decoder, regardless of what the decoder predicts. This training method is called ***Teacher Forcing***.

8

teacher forcing: 每次不使用上一个state的输出作为下一个state的输入,而是直接用训练数据的标准答案(ground truth)对应的上一项作为下一个state的输入

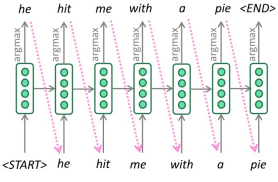
缺点: 因为依赖标签数据,在训练过程中,模型有较好的效果;但在测试时因为不能得到ground truth的支持,模型可能会变得脆弱

Recap: decoding algorithms

- Question: Once you've trained your (conditional) language model, how do you use it to generate text?
- Answer: A *decoding algorithm* is an algorithm you use to generate text from your language model
- We've learnt about two decoding algorithms:
  - Greedy decoding
  - Beam search

Recap: greedy decoding

- A simple algorithm
- On each step, take the most probable word (i.e. argmax)
- Use that as the next word, and feed it as input on the next step
- Keep going until you produce <END> (or reach some max length)



- Due to lack of backtracking, output can be poor
  (e.g. ungrammatical, unnatural, nonsensical)

Recap: beam search decoding

- A search algorithm which aims to find a high-probability sequence (not necessarily the optimal sequence, though) by tracking multiple possible sequences at once.
- Core idea: On each step of decoder, keep track of the *k* most probable partial sequences (which we call *hypotheses*)
  - *k* is the beam size
- After you reach some stopping criterion, choose the sequence with the highest probability (factoring in some adjustment for length)

改变 beam size k 的影响是什么

① 小的 k 值和 greedy decoding 有相似的问题 k=1，可能得到 ungrammatical、unnatural、nonsensical、incorrect 的输出

② 大的 k 值意味着更多的可能性，能减轻上述问题，但计算代价高，还有以下问题。

- 对 NMT，k 太大会降低 BLEU 分数，可能是因为大的 k 产生了更多短句（即使有归一化操作）
- 在开放式任务（如闲聊对话），大的 k 可能产生很多重用的句子

Effect of beam size in chitchat dialogue

I mostly eat a fresh and raw diet, so I save on groceries

Human chit-chat partner

Beam size	Model response
1	*I love to eat healthy and eat healthy*
2	*That is a good thing to have*
3	*I am a nurse so I do not eat raw food*
4	*I am a nurse so I am a nurse*
5	*Do you have any hobbies?*
6	*What do you do for a living?*
7	*What do you do for a living?*
8	*What do you do for a living?*

Low beam size: More on-topic but nonsensical; bad English

High beam size: Converges to safe, "correct" response, but it's generic and less relevant

新的decoding方法 Sampling-based decoding

Pure sampling：在每个步骤t中，从概率分布P_t中随机抽样以获取下一个单词

Top-n sampling：在每个步骤t中，在前n个最可能的单词中进行随机采样。n=1是贪婪搜索，n=V是纯采样。

增加n：多样化/有风险的输出

减少n：通用/安全的输出

这2种方法都比 beam search 更有效率，不用跟踪多个假设

Softmax temperature

t↑ P_t分布更均匀⇒输出更多样化

train的时候不需要

Softmax temperature

- Recall: On timestep t, the LM computes a prob dist P_t by applying the softmax function to a vector of scores $s \in \mathbb{R}^{|V|}$

$$P_t(w) = \frac{\exp(s_w)}{\sum_{w' \in V} \exp(s_{w'})}$$

- You can apply a *temperature hyperparameter* τ to the softmax:

$$P_t(w) = \frac{\exp(s_w/\tau)}{\sum_{w' \in V} \exp(s_{w'}/\tau)}$$

- Raise the temperature τ: P_t becomes more uniform
  - Thus more diverse output (probability is spread around vocab)
- Lower the temperature τ: P_t becomes more spiky
  - Thus less diverse output (probability is concentrated on top words)

Note: softmax temperature is not a decoding algorithm!
It's a technique you can apply at test time, in conjunction with a decoding algorithm (such as beam search or sampling)

总结：

greedy decoding：简单，低质量输出

beam search：搜索高概率输出。比 greedy 提供更高质量的输出，但如果 beam size 太大，可能返回高概率但不合适的输出

Sampling method：获得更多的多样性、随机性，适合开放式/创意式任务。Top-n 允许控制多样性

Softmax temperature：控制多样性的另一种技巧，可用在任何的解码算法

2. NLG tasks and neural approaches to them
NLG任务和适用的神经网络方法

Summarization 摘要

定义：给定一个输入文本x，输出一个摘要y（比x短，包含了x的主要信息）

single-document：为一个文档x写一个摘要y

multi-document：为多个文档$x_1, \cdots, x_n$写一个摘要y，$x_1, \cdots x_n$通常有重叠的内容

single-document summarization任务的数据集

Gigaword：新闻的第一、二句话

LCSTS（中国微博）：paragraph → sentence summary

NYT，CNN/DailyMail：news artical → (multi) sentence summary

Wikihow：full how-to artical → sentence summary

Sentence simplification

与摘要相关但不同，以更简洁的方式重写源文本

Simple Wikipedia

Newsela

Summarization 的 2 种策略

① Extractive summarization 抽取式摘要

选择部分（通常是句子）原始文本来形成摘要

更简单，无需解释，有限制的

② Abstractive summarization 抽象式摘要

使用自然语言生成技术生成新的文本

更困难，更多变

Pre-neural summarization

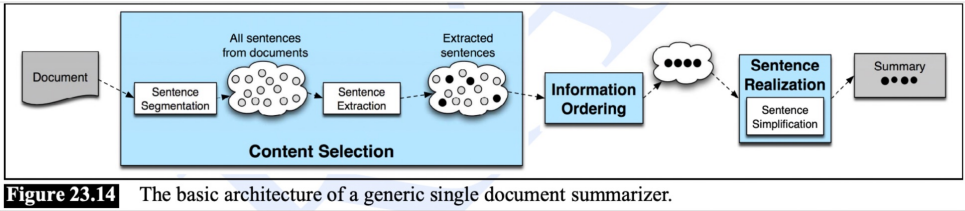


Figure 23.14 The basic architecture of a generic single document summarizer.

几乎都是抽取式的

Content selection：选择要包括哪些句子

information ordering：选择这些句子的顺序

sentence realization：编辑并输出句子（简化、修复连贯性问题）

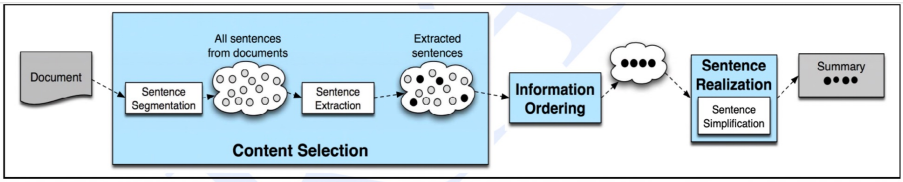


Figure 23.14 The basic architecture of a generic single document summarizer.

content selection 算法：

句子得分函数可以基于：

- 主题关键词，通过计算tf-idf等
- 一些特征，如这句话出现在文档的哪个位置

Graph-based algorithm 将文档视为句子(结点)的集合，每对句子间存在边

- 边的权重与句子相似性成比例
- 通过图算法识别哪些句子是中心句(最重要的句子)

Summarization evaluation: ROUGE

ROUGE (Recall-Oriented Understudy for Gisting Evaluation)

$$\text{ROUGE-N} = \frac{\sum_{S \in \{ReferemceSummaries\}} \sum_{gram_n \in S} Count_{match}(gram_n)}{\sum_{S \in \{ReferenceSummaries\}} \sum_{gram_n \in S} Count(gram_n)} \quad (1)$$

与BLEU类似，基于n-gram

差异：① 没有简短惩罚

② ROUGE基于recall，BLEU基于Precision

对于机器翻译，准确率更重要，通过添加简洁惩罚来修正翻译过短；但对摘要来说，召回率更重要，因为需要抓住重要信息

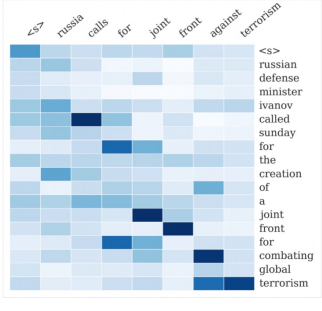
但通常使用F1(结合了准确率和召回率)

- BLEU is reported as a single number, which is combination of the precisions for n=1,2,3,4 n-grams
- ROUGE scores are reported separately for each n-gram

- The most commonly-reported ROUGE scores are:
  - ROUGE-1:* unigram overlap
  - ROUGE-2: bigram overlap
  - ROUGE-L: longest common subsequence overlap

Neural Summarization (2015 - Present)

- 2015: Rush et al. publish the first seq2seq summarization paper
- Single-document abstractive summarization is a translation task!
- Thus we can apply standard seq2seq + attention NMT methods



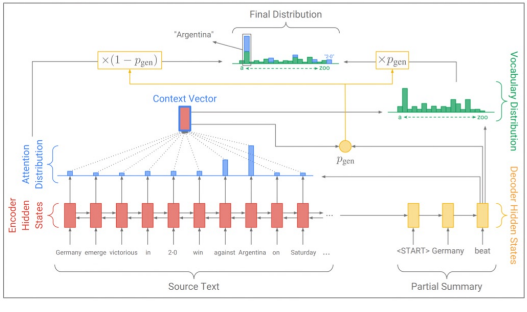
- Since 2015, there have been lots more developments!
  - Making it easier to copy
    - But also preventing too much copying!
  - Hierarchical / multi-level attention
  - More global / high-level content selection
  - Using Reinforcement Learning to directly maximize ROUGE, or other discrete goals (e.g. length)
  - Resurrecting pre-neural ideas (e.g. graph algorithms for content selection) and working them into neural systems
  - …

Neural summarization: copy mechanisms 复制机制

Seq2Seq + attention systems 擅长生成流畅的输出，但不擅长正确地复制一些细节（如罕见字）

copy mechanisms 使用注意力机制，使 seq2seq 系统能容易地从输入复制单词到输出。允许复制和文本生成给了我们一个混合了抽取/抽象式的方法

Neural summarization: copy mechanisms



One example of how to do a copying mechanism:

On each decoder step, calculate p_{gen}, the probability of *generating* the next word (rather than copying it). The final distribution is a mixture of the generation (aka "vocabulary") distribution, and the copying (i.e. attention) distribution:

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

copy mechanisms 的问题

① 复制过多。大多数长短语，有时是整个句子

② 不善于对整体内容进行选择，尤其是输入文档很长的情况下

Better content selection

- Recall: pre-neural summarization had separate stages for **content selection** and **surface realization** (i.e. text generation)

- In a standard seq2seq+attention summarization system, these two stages are mixed in together
  - On each step of the decoder (i.e. surface realization), we do word-level content selection (attention)
  - This is bad: no *global* content selection strategy

- One solution: bottom-up summarization

Bottom-up Summarization

Content selection 阶段：使用神经网络序列标记模型 neural sequence-tagging model 来标记哪些词包括，哪些不包括

Bottom-up attention 阶段：seq2seq + attention 系统，不处理不包括的词（使用 mask）

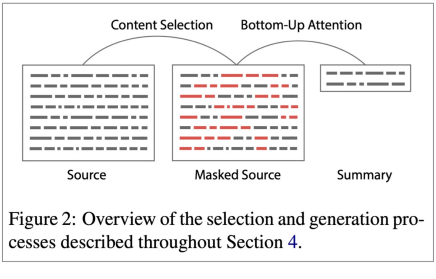


Figure 2: Overview of the selection and generation processes described throughout Section 4.

Simple but effective!

- Better overall content selection strategy
- Less copying of long sequences (i.e. more abstractive output)

简单有效

更好的整体内容选择

对于长句子的复制更少

Neural summarization: 增强学习

Neural summarization via Reinforcement Learning

- In 2017 Paulus et al published a "deep reinforced" summarization model
- Main idea: Use Reinforcement Learning (RL) to directly optimize ROUGE-L
  - By contrast, standard maximum likelihood (ML) training can't directly optimize ROUGE-L because it's a non-differentiable function 不可微
- Interesting finding:
  - Using RL instead of ML achieved higher ROUGE scores, but lower human judgment scores

Model	ROUGE-1	ROUGE-2	ROUGE-L
ML, no intra-attention	44.26	27.43	40.41
ML, with intra-attention	43.86	27.10	40.11
RL, no intra-attention	**47.22**	30.51	**43.27**
ML+RL, no intra-attention	47.03	**30.72**	43.10

Model	Readability	Relevance
ML	6.76	7.14
RL	4.18	6.32
ML+RL	**7.04**	**7.45**

"We observed that our models with the highest ROUGE scores also generated barely-readable summaries."

Overall, a hybrid approach does best!

A Deep Reinforced Model for Abstractive Summarization, Paulus et al, 2017 https://arxiv.org/pdf/1705.04304.pdf

Blog post: https://www.salesforce.com/products/einstein/ai-research/tl-dr-reinforced-model-abstractive-summarization/

Dialogue 对话

有许多类型：

① Task-oriented dialogue：

Assistive: 客户服务、提供建议，帮助用户完成一些任务

Co-operative: 通过对话，两人共同解决一个任务

Adversarial: 通过对话，两个在任务中竞争

② Social dialogue:

Chit-chat：闲聊

Theraphy / mental well being 心理治疗类对话

Pre- and post-neural dialogue

由于开放式NLG的难度，Pre-neural对话系统经常使用预定义的模板，或从语料库中检索一个适当的反应

从2015年开始，许多人将seq2seq方法应用到对话任务中

- Some early seq2seq dialogue papers include:
- *A Neural Conversational Model*, Vinyals et al, 2015
  https://arxiv.org/pdf/1506.05869.pdf
- *Neural Responding Machine for Short-Text Conversation*, Shang et al, 2015
  https://www.aclweb.org/anthology/P15-1152

基于seq2seq的dialogue

简单地将标准seq2seq + attention的方法应用在对话(闲聊)任务上有严重缺陷。

① 普遍的/无聊的回应

② 与上下文无关的回应

③ 重复

④ 缺乏上下文(不记得对话历史)

⑤ 缺乏一致的角色人格

Irrelevant response problem 与上下文无关

问题：seq2seq常产生一些和用户说的话无关的回应

可能的原因：

① 那个回答是通用的，如 I don't know

② 将话题转到不相关的事情上

解决方案：最大化 Maximum Mutual Information (MMI)

Input S 和 response T

$$\log \frac{p(S,T)}{p(S)p(T)}$$

$$\hat{T} = \arg\max_{T} \{ \log p(T|S) - \log p(T) \}$$

Genericness / boring response problem 通用/无聊

- Easy test-time fixes:
  - Directly upweight rare words during beam search
  - Use a sampling decoding algorithm rather than beam search
- Conditioning fixes:
  - Condition the decoder on some additional content (e.g. sample some content words and attend to them)
  - Train a retrieve-and-refine model rather than a generate-from-scratch model
    - i.e. sample an utterance from your corpus of human-written utterances, and edit it to fit the current scenario.
    - This usually produces much more diverse / human-like / interesting utterances!

检索和优化

改编人类的话以适应当前语境

Repetition problem 重复

简单的解决方案：

直接在 beam search 中禁止重复 n-grams，通常很有效

复杂的解决方案：

① 在 seq2seq 中训练一个 coverage mechanism，这是一个防止注意力机制多次注意相同单词的机制

② 定义一个阻止重复的训练目标

Lack of consistent persona problem 缺乏一致的角色人格

- In 2016, Li et al proposed a seq2seq dialogue model that learns to encode both conversation partners' personas as embeddings
  - The generated utterances are conditioned on the embeddings
- More recently, there is now a chitchat dataset called PersonaChat, which includes personas (collections of 5 sentences describing personal traits) for every conversation.
  - This provides a light type of *grounding*, allowing researchers to build persona-conditional dialogue agents

将人格作为向量

生成的话语以该向量为条件

Storytelling

- 给定图像生成故事情节
- 给定一个简短的写作提示生成一个故事
- 故事续写

- Neural storytelling is taking off!
  - The first Storytelling Workshop was held in 2018
  - It held a competition (generate a story to accompany a sequence of 5 images)

Getting a story from an image

不是直接的有监督的问题，没有配对的数据可以学习

问题：如何解决缺乏并行数据的问题

回答：使用一个通用的句子编码空间 sentence-encoding space

skip-thought 向量是一种通用的句子嵌入方法，想法类似于我如何通过预测周围的词来学习一个词的词向量

使用 COCO（图片标题数据集），学习从图像到其标题的 skip-thought 编码的映射

使用目标语料库训练 RNN-LM，将 skip-thought 向量解码为原文

Generating a story from a writing prompt

- In 2018, Fan et al released a new story generation dataset collected from Reddit's WritingPrompts subreddit.
- Each story has an associated brief writing prompt.

2018年，一个新的故事生成数据集

> **Prompt:** The Mage, the Warrior, and the Priest
>
> **Story:** A light breeze swept the ground, and carried with it still the distant scents of dust and time-worn stone. The Warrior led the way, heaving her mass of armour and muscle over the uneven terrain. She soon crested the last of the low embankments, which still bore the unmistakable fingerprints of haste and fear. She lifted herself up onto the top the rise, and looked out at the scene before her. [...]

Fan et al also proposed a complex seq2seq prompt-to-story model:

一个复杂的 seq2seq 模型

- It's convolutional-based
  - This makes it faster than RNN-based seq2seq

卷积

- Gated multi-head multi-scale self-attention
  - The self-attention is important for capturing long-range context
  - The gates allow the attention mechanism to be more selective
  - The different attention heads attend at different scales – this means there are different attention mechanisms dedicated to retrieving fine-grained information and coarse-grained information
- Model fusion:
  - Pretrain one seq2seq model, then train a *second* seq2seq model that has access to the hidden states of the first
  - The idea is that the first seq2seq model learns general LM and the second learns to condition on the prompt

3. NLG Evaluation

基于词重叠的指标 word overlap based metrics

BLEU, ROUGE, METEOR, F1 等

① 不适合机器翻译

② 对摘要任务更差，因为比 MT 更开放，与抽象摘要相比，提取式摘要更适合 ROUGE

③ 对对话任务更糟，类似的还有故事生成

Word overlap metrics are not good for dialogue

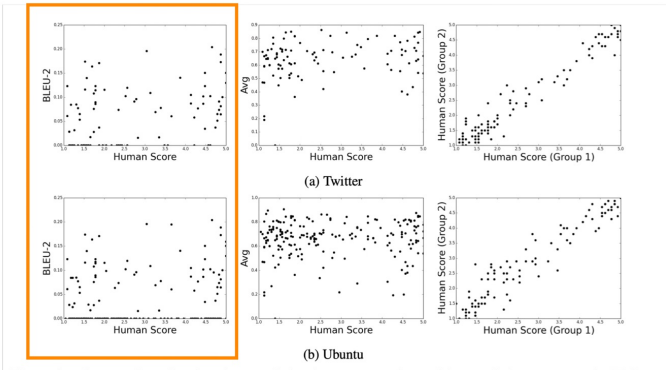


Figure 1: Scatter plots showing the correlation between metrics and human judgements on the Twitter corpus (a) and Ubuntu Dialogue Corpus (b). The plots represent BLEU-2 (left), embedding average (center), and correlation between two randomly selected halves of human respondents (right).

Automatic evaluation metrics for NLG

- What about perplexity?
  - Captures how powerful your LM is, but doesn't tell you anything about generation (e.g. if your decoding algorithm is bad, perplexity is unaffected)

只关注LM，不关注文本生成表现

- Word embedding based metrics?
  - Main idea: compare the similarity of the word embeddings (or average of word embeddings), not just the overlap of the words themselves. Captures semantics in a more flexible way.
  - Unfortunately, still doesn't correlate well with human judgments for open-ended tasks like dialogue.

用词向量相似度而不只是词重叠，更灵活

但仍然和人类的判断关联得不是很好

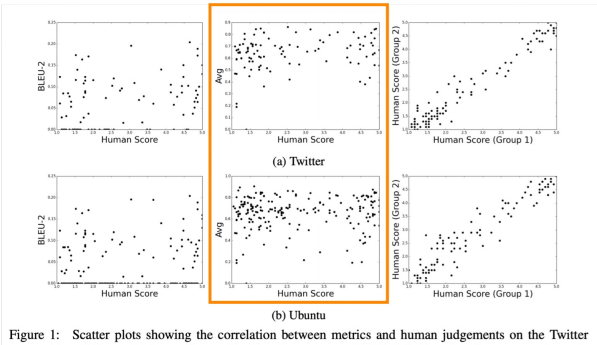


Figure 1: Scatter plots showing the correlation between metrics and human judgements on the Twitter corpus (a) and Ubuntu Dialogue Corpus (b). The plots represent BLEU-2 (left), embedding average (center), and correlation between two randomly selected halves of human respondents (right).

我们没有一个 automatic evaluation metrics 充分捕捉整体的质量

但可以定义更多的标准来捕捉文本的特定方面

- But we can define more focused automatic metrics to capture particular aspects of generated text:
  - Fluency (compute probability w.r.t. well-trained LM) 流畅度
  - Correct style (prob w.r.t. LM trained on target corpus) 正确的风格
  - Diversity (rare word usage, uniqueness of n-grams) 多样性
  - Relevance to input (semantic similarity measures) 与输入的相关性
  - Simple things like length and repetition 长度、重复
  - Task-specific metrics e.g. compression rate for summarization 特殊的标准
- Though these don't measure overall quality, they can help us track some important qualities that we care about.

72

Human evaluation

人类的判断被认为是黄金标准

但速度慢、代价高

进行有效的人类评估很困难

人类：

- 不一致的
- 可能是不合逻辑的
- 失去注意力
- 误解你的问题
- 不能总是解释为什么他们会这样做

Possible new avenues for NLG eval

- Corpus-level metrics
  - Should an eval metric be applied to each example in the test set independently, or a function of the whole corpus?
  - e.g. if a dialogue model always gives the same generic answer to every example in the test set, it should be penalized
- Eval metrics that measure the diversity-safety tradeoff
- Human eval for free
  - Gamification: make the task (e.g. talking to a chatbot) fun, so humans provide supervision and implicit evaluation for free
- Adversarial discriminator as an evaluation metric
  - Test whether the NLG system can fool a discriminator which is trained to distinguish human text from artificially generated text

4. Thoughts on NLG research, current trends, and the future

当前趋势

- Incorporating discrete latent variables into NLG
  - May help with modeling structure in tasks that really need it, like storytelling, task-oriented dialogue, etc
- Alternatives to strict left-to-right generation
  - Parallel generation, iterative refinement, top-down generation for longer pieces of text
- Alternative to maximum likelihood training with teacher forcing
  - More holistic sentence-level (rather than word-level) objectives

Neural NLG community is rapidly maturing

将NMT的成功方法迁移到NLG中

更多有创造性的NLG方法出现

- During the early years of NLP + Deep Learning, community was mostly transferring successful NMT methods to NLG tasks.
- Now, increasingly more inventive NLG techniques emerging, specific to *non-NMT* generation settings.
- Increasingly more (neural) NLG workshops and competitions, especially focusing on open-ended NLG:
  - NeuralGen workshop
  - Storytelling workshop
  - Alexa challenge
  - ConvAI2 NeurIPS challenge
- These are particularly useful to organize the community, increase reproducibility, standardize eval, etc.
- The biggest roadblock for progress is eval

8 things I've learnt from working in NLG

1. The more open-ended the task, the harder everything becomes.
   - Constraints are sometimes welcome!
2. Aiming for a specific improvement can be more manageable than aiming to improve overall generation quality.
3. If you're using a LM for NLG: improving the LM (i.e. perplexity) will most likely improve generation quality.
   - ...but it's not the only way to improve generation quality.
4. Look at your output, a lot
5. You need an automatic metric, even if it's imperfect.
   - You probably need *several* automatic metrics.
6. If you do human eval, make the questions as focused as possible.
7. Reproducibility is a huge problem in today's NLP + Deep Learning, and a huger problem in NLG.
   - Please, publicly release all your generated output along with your paper!
8. Working in NLG can be very frustrating. But also very funny...

Lecture 20

Deep Learning for NLP 5 years ago

- No Seq2Seq
- No Attention
- No large-scale QA/reading comprehension datasets
- No TensorFlow or Pytorch
- …

Future of Deep Learning + NLP

利用无标签数据

无监督机器翻译

提高预训练和GPT-2的规模

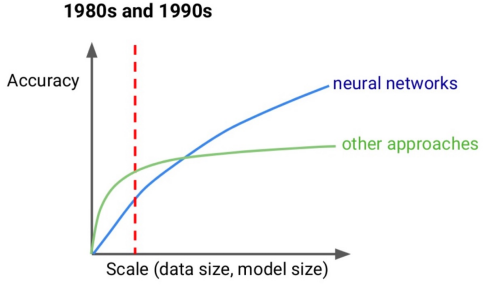
NLP技术的风险，社会影响

未来的研究方向

- **Harnessing Unlabeled Data**
  - Back-translation and unsupervised machine translation
  - Scaling up pre-training and GPT-2
- **What's next?**
  - Risks and social impact of NLP technology
  - Future directions of research

Why has deep learning been so successful recently?

比其他的方法，能更好地利用海量数据



NLP Datasets

- Even for English, most tasks have 100K or less labeled examples.
- And there is even less data available for other languages.
  - There are thousands of languages, hundreds with > 1 million native speakers
  - <10% of people speak English as their first language
- Increasingly popular solution: use **unlabeled** data.

其他语言的数据较少

越来越流行的解决方案：使用无标签数据

1. 在MT任务中使用无标签数据

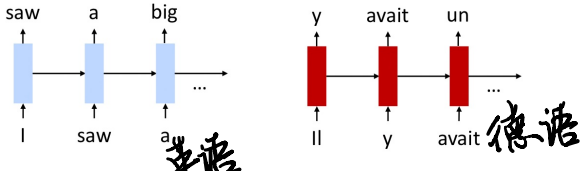
获得翻译需要人类的专业知识（限制了数据的规模和领域）

没有标记的文本数据很多（非双语）

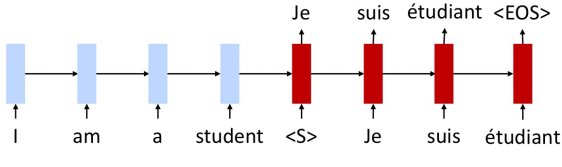
Pre-training

Pre-Training

1. Separately Train Encoder and Decoder as Language Models



2. Then Train Jointly on Bilingual Data

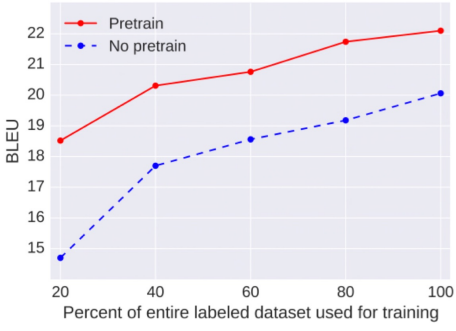


作为LM，分别训练Encoder和Decoder

在双语数据上共同训练

表现提升

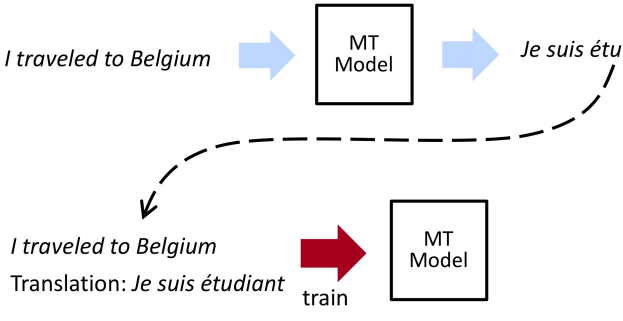
- English -> German Results: 2+ BLEU point improvement



Self Training

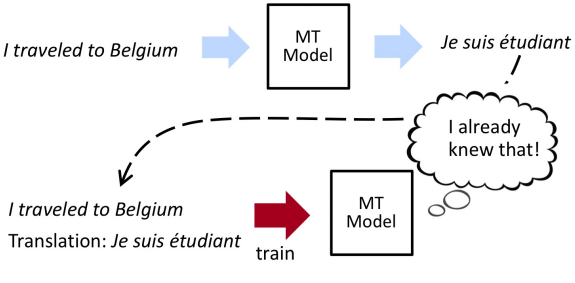
Pre-training 的问题：在 pre-training 中两种语言没有交互

Self-training：标记未标记的数据以获得有噪声的训练样本



不完美的 noisy 但仍作为标签

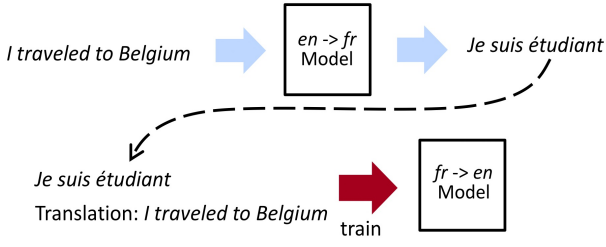
- Circular?



因为这个问题，在实际训练中，不常用这种方法

Back-Translation

- Have two machine translation models going in opposite directions (*en -> fr*) and (*fr -> en*)



有2种方向相反的机器翻译模型

不再是循环

模型看不到"坏翻译"，只有"好输入"

Large-Scale Back-Translation

- 4.5M English-German sentence pairs and 226M monolingual sentences

Citation	Model	BLEU
Shazeer et al., 2017	Best Pre-Transformer Result	26.0
Vaswani et al., 2017	Transformer	28.4
Shaw et al, 2018	Transformer + Improved Positional Embeddings	29.1
Edunov et al., 2018	**Transformer + Back-Translation**	**35.0**

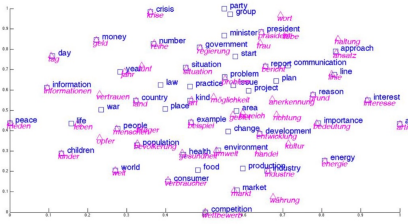
如果没有双语数据：

当我们只有无标签的句子时，我们进行一种比完全翻译更简单的任务：

做单词翻译，而不是句子翻译

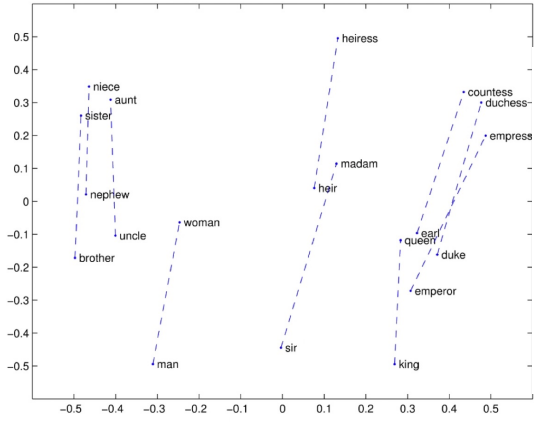
Unsupervised word translation

- *Cross-lingual word embeddings*
  - Shared embedding space for both languages
  - Keep the normal nice properties of word embeddings
  - But also want words close to their translations
- Want to learn from monolingual corpora

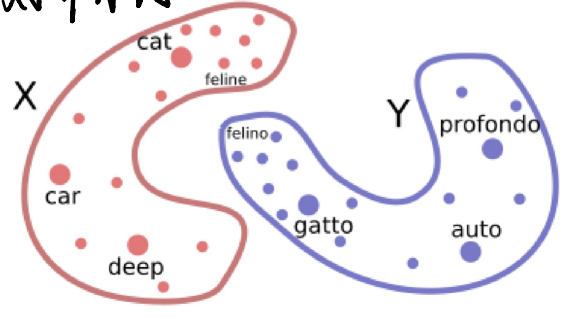


跨语言文字嵌入
共享同一向量空间
保持词向量良好属性
希望单词与它们的翻译邻近
↘从单词语料库中学习

词向量有多种结构
猜想：语言间结构应相似

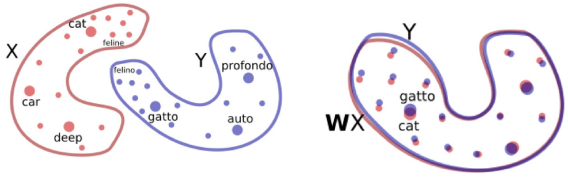


学习不同语言的词嵌入间的对齐方式



- First run word2vec on monolingual corpora, getting words embeddings *X* and *Y*
- Learn an (orthogonal) matrix *W* such that *WX* ~ *Y*

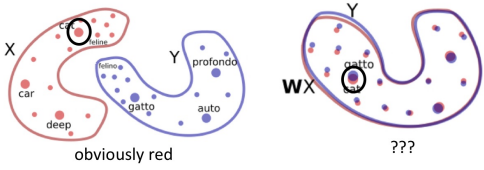
正交



学习W的方法：

- Learn *W* with *adversarial training*. 对抗训练
- Discriminator: predict if an embedding is from *Y* or it is a transformed embedding *Wx* originally from *X*. 预测一个嵌入是来自Y还是Wx（由X转换来）
- Train *W* so the Discriminator gets "confused" 训练目标

Discriminator predicts: is the circled point red or blue?



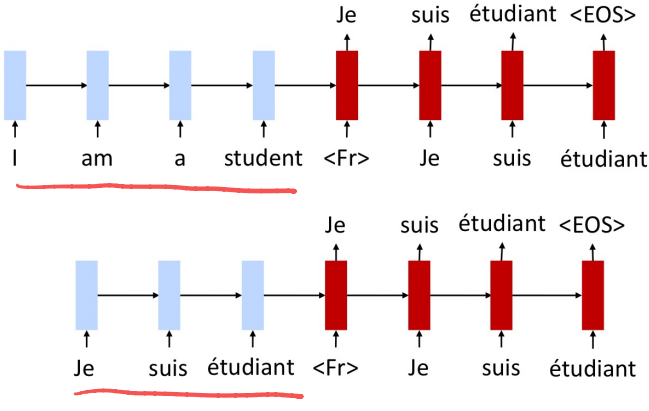
- Other tricks can be used to further improve performance, see Word Translation without Parallel Data

Unsupervised Machine Translation

无监督翻译整个句子

Unsupervised Machine Translation

- Model: **same** encoder-decoder used for both languages
  - Initialize with cross-lingual word embeddings



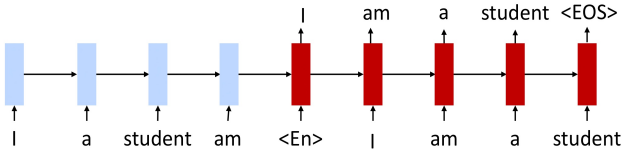
encoder可以处理任何输入

decoder：特殊标记<Fr>告诉模型应生成什么语言的输出

- Training objective 1: de-noising autoencoder

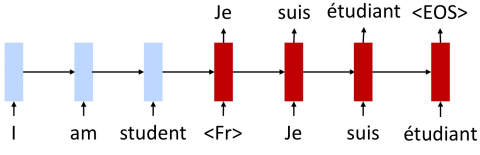
去噪自编码器.

输入一个打乱顺序的句子
输出其原来顺序的句子



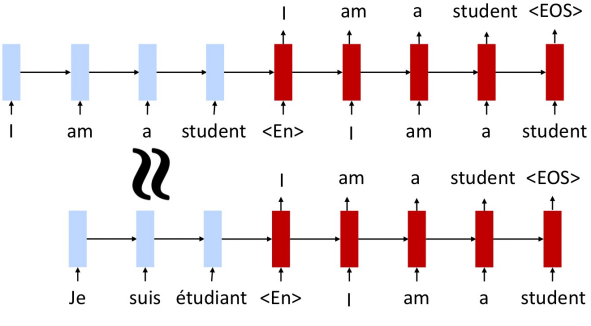
由于这是一个没有注意力机制的模型，编码器将整个源句子转换为单个向量。自编码器的作用是保证来自encoder的向量包含和这个句子有关的，能使我们恢复原来句子的所有信息

- Training objective 2: back translation
  - First translate *fr -> en*
  - Then use as a "supervised" example to train *en -> fr*



Why Does This Work?

- Cross lingual embeddings and shared encoder gives the model a starting point

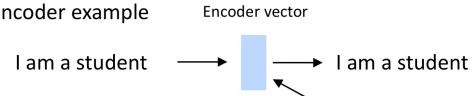


Why Does This Work?

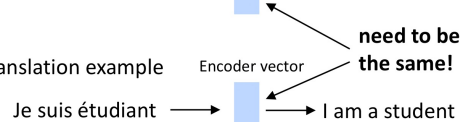
语言无关的表示

- Objectives encourage language-agnostic representation

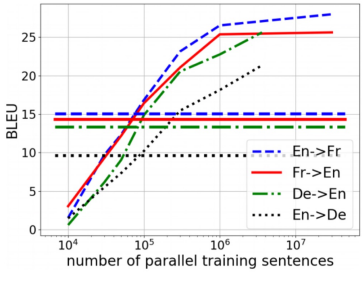
Auto-encoder example



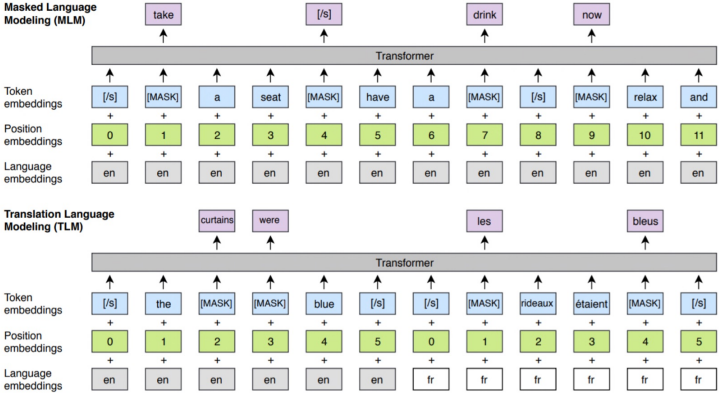
Back-translation example



- Horizontal lines are unsupervised models, the rest are supervised



Cross-Lingual BERT



Unsupervised MT Results

Model	En-Fr	En-De	En-Ro
UNMT	25.1	17.2	21.2
UNMT + Pre-Training	33.4	26.4	**33.3**
Current supervised State-of-the-art	**45.6**	**34.2**	29.9

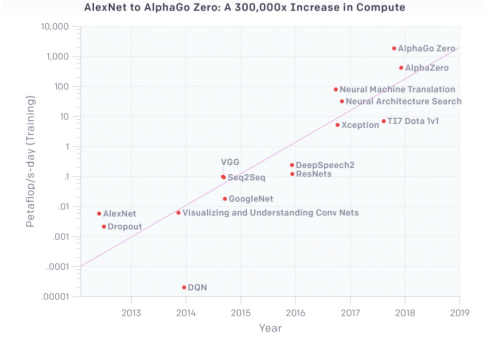
2. Huge Models and GPT-2

Training Huge Models

Model	# Parameters
Medium-sized LSTM	10M
ELMo	90M
GPT	110M
BERT-Large	320M
GPT-2	1.5B

比蜜蜂大脑有更多突触

This is a General Trend in ML



模型规模↑是机器学习的一个普遍趋势.

训练巨大的模型，需要好的硬件，模型和数据的并行化

GPT-2

只是一个非常大的 Transformer LM

40GB的训练文本 一投入了相当多的努力去确保数据质量

So What Can GPT-2 Do?

- Obviously, language modeling (but very well)!
- Gets state-of-the-art perplexities on datasets it's not even trained on!

	LAMBADA (PPL)	LAMBADA (ACC)	CBT-CN (ACC)	CBT-NE (ACC)	WikiText2 (PPL)	PTB (PPL)	enwik8 (BPB)	text8 (BPC)	WikiText103 (PPL)	1BW (PPL)
SOTA	99.8	59.23	85.7	82.3	39.14	46.54	0.99	1.08	18.3	**21.8**
117M	**35.13**	45.99	**87.65**	**83.4**	**29.41**	65.85	1.16	1.17	37.50	75.20
345M	**15.60**	55.48	**92.35**	**87.1**	**22.76**	47.33	1.01	**1.06**	26.37	55.72
762M	**10.87**	**60.12**	**93.45**	**88.0**	**19.93**	**40.31**	**0.97**	**1.02**	22.05	44.575
1542M	**8.63**	**63.24**	**93.30**	**89.05**	**18.34**	**35.76**	**0.93**	**0.98**	**17.48**	42.16

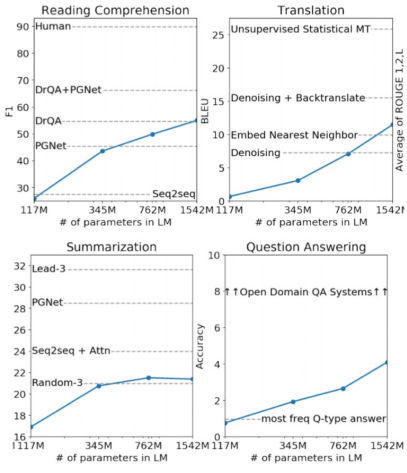
So What Can GPT-2 Do?

- **Zero-Shot Learning**: no supervised training data!
  - Ask LM to generate from a prompt

在没有接受过训练的情况下产生输出

- **Reading Comprehension:** <context> <question> A: 阅读理解
- **Summarization:** <article> TL;DR: 摘要
- **Translation:**
  <English sentence1> = <French sentence1>
  <English sentence 2> = <French sentence 2>
  .....
  <Source sentence> = 翻译
- **Question Answering:** <question> A: 问答.

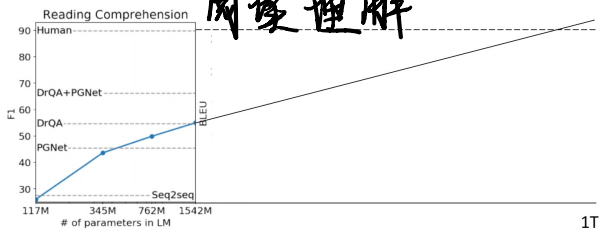
GPT-2 Results



What happens as models get even bigger?

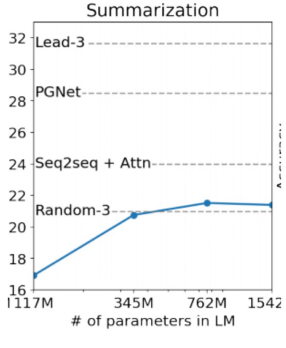
在一些任务中，模型越大，表现越好

- For several tasks performance seems to increase with log(model size)



- But trend isn't clear

但趋势不清晰



GPT-2 Reaction

Some arguments for release:

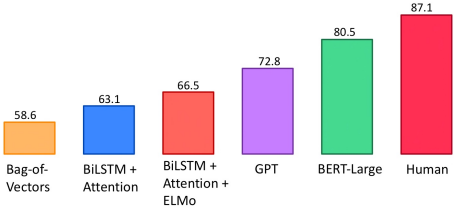
- This model isn't much different from existing work
- Not long until these models are easy to train
  - And we're already at this point for images/speech
- Photoshop
- Researchers should study this model to learn defenses
- Dangerous PR Hype
- Reproducibility is crucial for science
- …

Some arguments against:

- Danger of fake reviews, news comments, etc.
  - Already done by companies and governments
- Precedent
  - Event if this model isn't dangerous, later ones will be even better
- Smaller model is being released
- ….

3. BERT解决了什么，我们之后要做什么

GLUE Benchmark Results



更困难的自然语言理解

- Reading comprehension… 阅读理解
  - On longer documents or multiple documents 长文档，多个文档
  - That requires multi-hop reasoning 跳跃推理（从多个地方结合，获得答案）
  - Situated in a dialogue
  - 定位问答
- Key problem with many existing reading comprehension datasets: *People writing the questions see the context* 现存的问题 按原文编问题
  - Not realistic 不真实
  - Encourages easy questions 容易出现简单问题

HotPotQA

- Designed to require *multi-hop reasoning*
- Questions are over *multiple documents*

Paragraph A, Return to Olympus:
[1] *Return to Olympus is the only album by the alternative rock band Malfunkshun.* [2] *It was released after the band had broken up and after lead singer Andrew Wood (later of Mother Love Bone) had died of a drug overdose in 1990.* [3] Stone Gossard, of Pearl Jam, had compiled the songs and released the album on his label, Loosegroove Records.

Paragraph B, Mother Love Bone:
[4] *Mother Love Bone was an American rock band that formed in Seattle, Washington in 1987.* [5] The band was active from 1987 to 1990. [6] *Frontman Andrew Wood's personality and compositions helped to catapult the group to the top of the burgeoning late 1980s/early 1990s Seattle music scene.* [7] *Wood died only days before the scheduled release of the band's debut album, "Apple", thus ending the group's hopes of success.* [8] The album was finally released a few months later.

Q: What was the former band of the member of Mother Love Bone who died just before the release of "Apple"?
A: Malfunkshun
Supporting facts: 1, 2, 4, 6, 7

Figure 1: An example of the multi-hop questions ir HOTPOTQA. We also highlight the supporting facts ir *blue italics*, which are also part of the dataset.

Zang et al., 2018

需要综合多处信息进行推理

问题涉及多个文档

Multi-Task Learning 多任务学习

让一个模行执行许多任务

在 BERT 的基础上，多任务学习产生了改进

Low-Resource Settings

不需要很多计算能力的模型

低资源语言

低数据环境（小样本学习）

Interpreting / understanding model

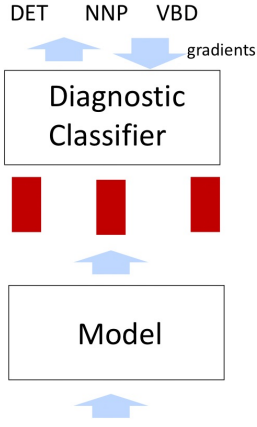
对模型的预测进行解释

理解模型，例如知道 BERT 为什么工作得好

这对某些应用很重要，如 healthcare

Diagnostic/Probing Classifiers

- Popular technique to see what linguistic information models "know"
- Diagnostic classifier takes representations produced by a model (e.g., BERT) as input and do some task
- Only the diagnositic classifier is trained



NLP in Industry

- NLP is rapidly growing in industry as well. Two particularly big areas:
- Dialogue
  - Chatbots
  - Customer service

- Healthcare
  - Understanding health records
  - Understanding biomedical literature

Conclusion

- Rapid progress in the last 5 years due to deep learning.
- Even more rapid progress in the last year due to larger models, better usage of unlabeled data
  - Exciting time to be working on NLP!
- NLP is reaching the point of having big social impact, making issues like bias and security increasingly important.