法律声明

- 本课件包括：演示文稿，示例，代码，题库，视频和声音等，小象学院拥有完全知识产权的权利；只限于善意学习者在本课程使用，不得在课程范围外向任何第三方散播。任何其他人或机构不得盗版、复制、仿造其中的创意，我们将保留一切通过法律手段追究违反者的权利。

- 课程详情请咨询
  - 微信公众号：小象
  - 新浪微博：ChinaHadoop

第五节课：优化对话模型的记忆能力

SEQ2SEQ模型的分层RNN方法和注意力原理

本节内容

- RNN 分层模型
  - 使用分层模型记忆聊天语境 (context)
  - 分层RNN模型代码演示
- Attention mechanism
  - 用于机器翻译的seq2seq模型中的注意力原理
  - 注意力原理应用：文本总结
  - 文本总结代码演示
  - 注意力原理应用：Attention+ConvSeq2seq用于机器翻译

参考文献

RNN 分层模型

- Building end-to-end dialogue systems using generative hierarchical neural network models (2016)
- A hierarchical latent variable encoder-decoder model for generative dialogues (2016)
- Training end-to-end dialogue systems with the Ubuntu dialogue corpus (2017)

参考文献

- Attention mechanism
  - Neural Machine Translation by Jointly Learning to Align and Translate (2014)
  - A neural attention model for abstractive sentence summarization (2015)
  - Abstractive text summarization using sequence-to-sequence RNNs and beyond (2016)
  - Convolutional sequence to sequence learning (2017)

记忆聊天语境

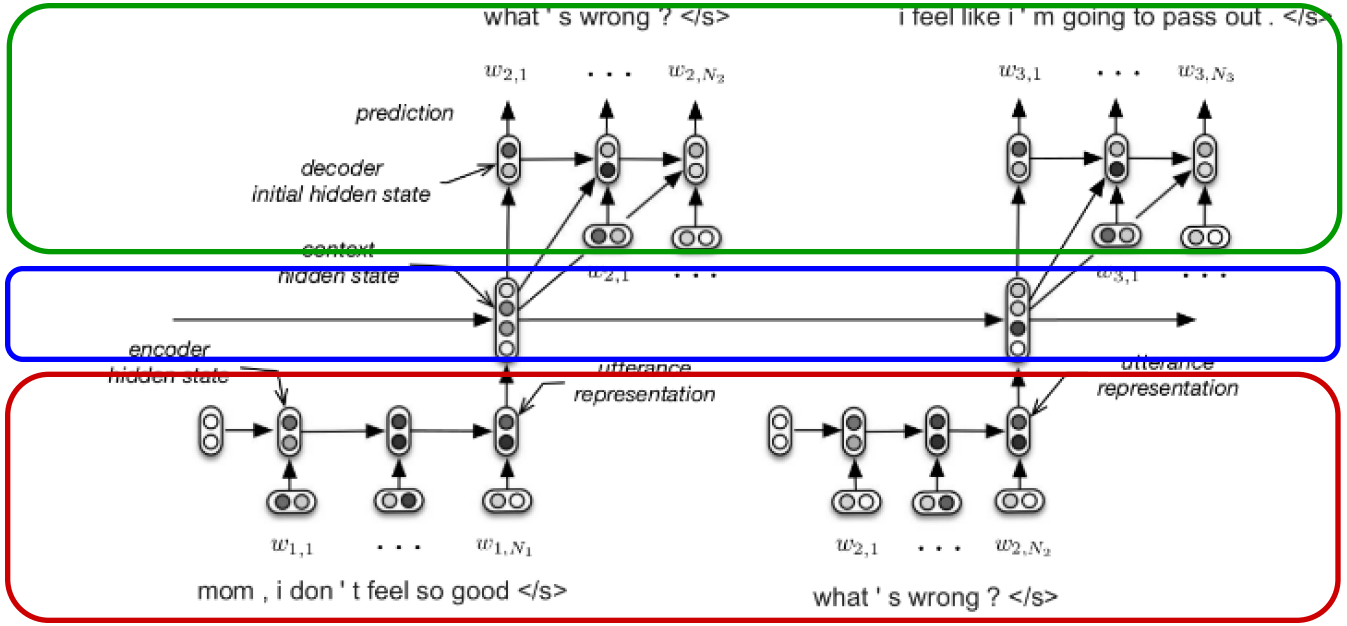
RNN 分层模型

分层RNN结构

- Building end-to-end dialogue systems using generative hierarchical neural network models
  - 提出了一个分层RNN结构用以同时在句子和对话语境层面建模
  - 实验验证在额外数据上做bootstrapping对提高对话模型的表现有明显的帮助

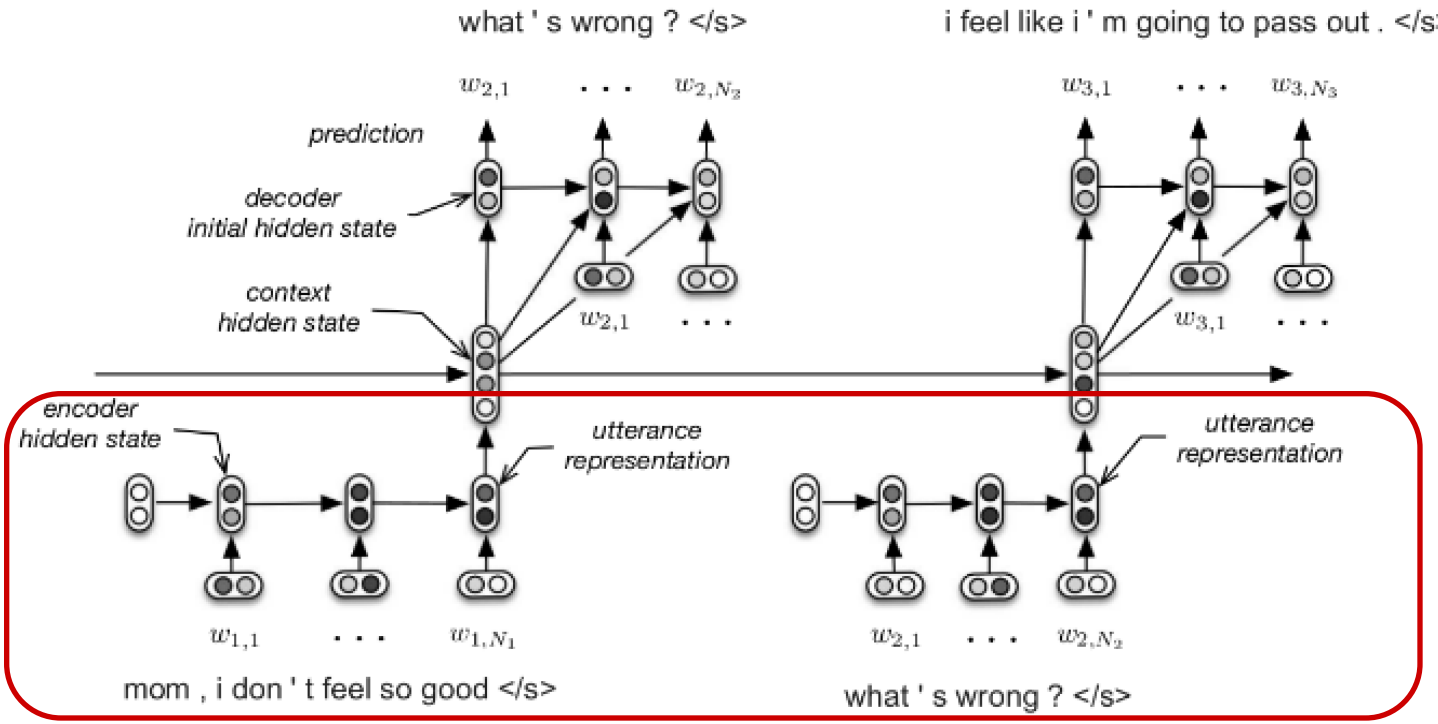
分层RNN结构

- 使用多个RNN模型描述不同层次的信息
  - 编码阶段：*encoder* RNN
  - 编码阶段：*context* RNN
  - 解码阶段：*decoder* RNN



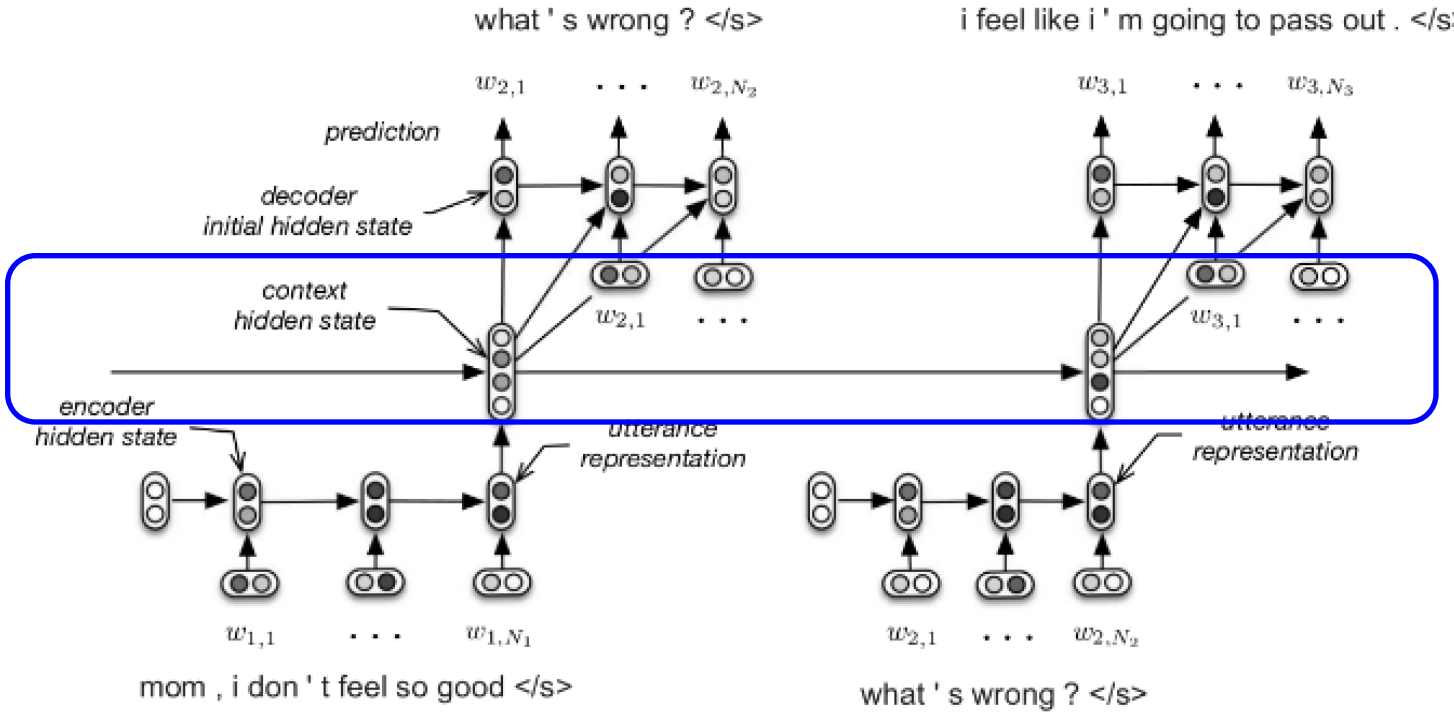
分层RNN结构

- encoder RNN将一句话编码到*utterance vector*
  - 和标准的seq2seq相同



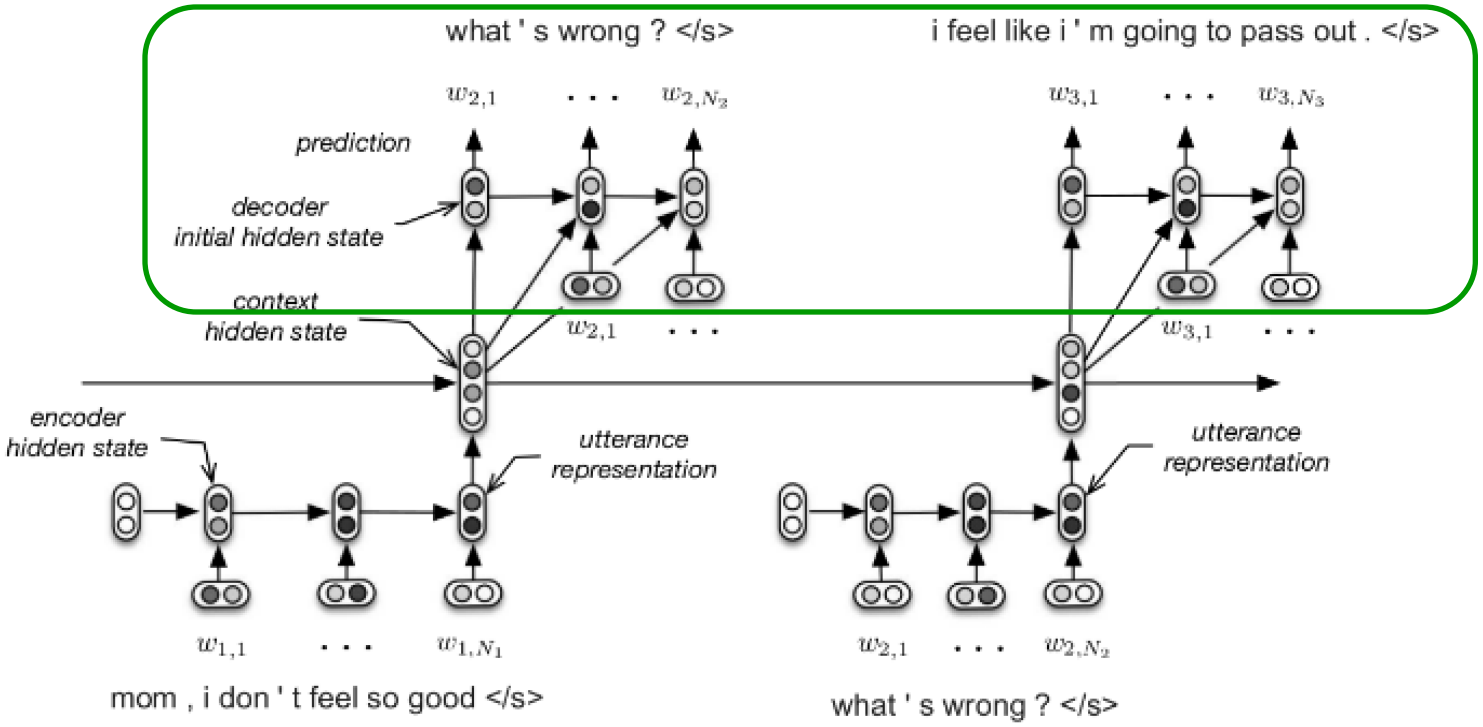
分层RNN结构

- context RNN总结多句话的编码
  - 以encoder RNN的final state作为输入
  - 描述多轮对话的语境

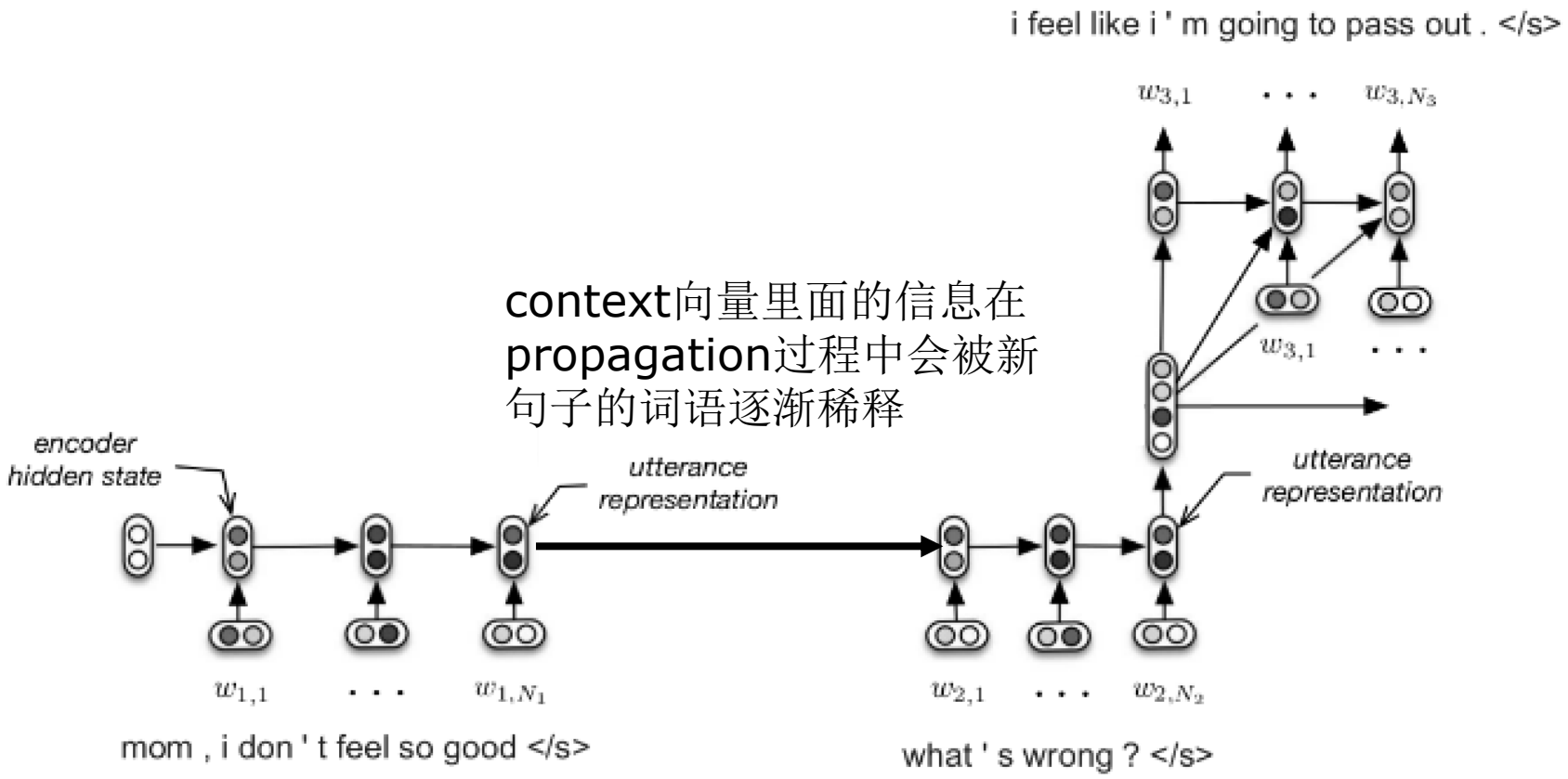


分层RNN结构

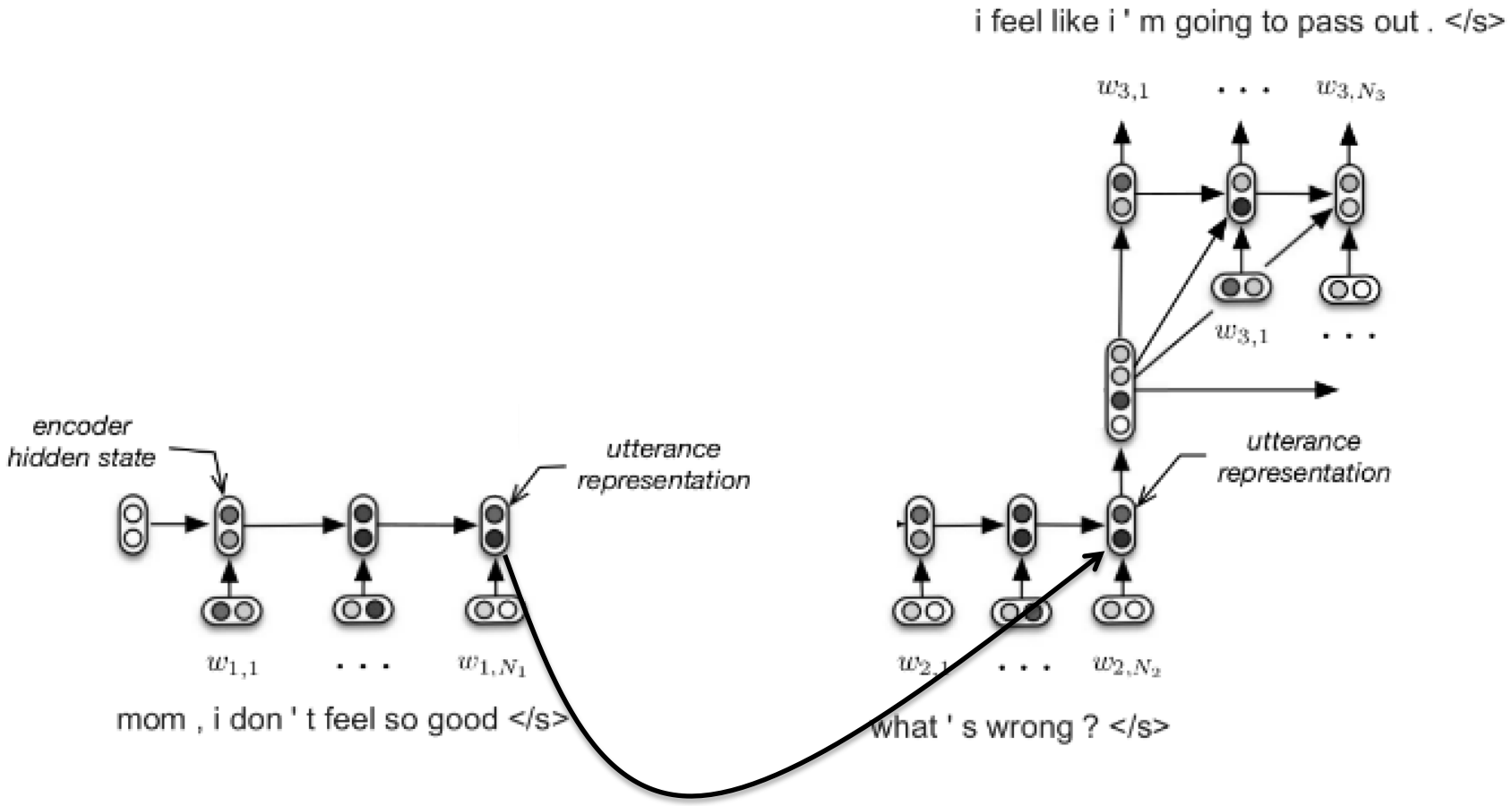
- decoder RNN从query和语境的编码出发产生回复
  - utterance vectors作为initial state输入decoder
  - context vector和单词一起组成decoder的input



分层RNN结构的直观理解

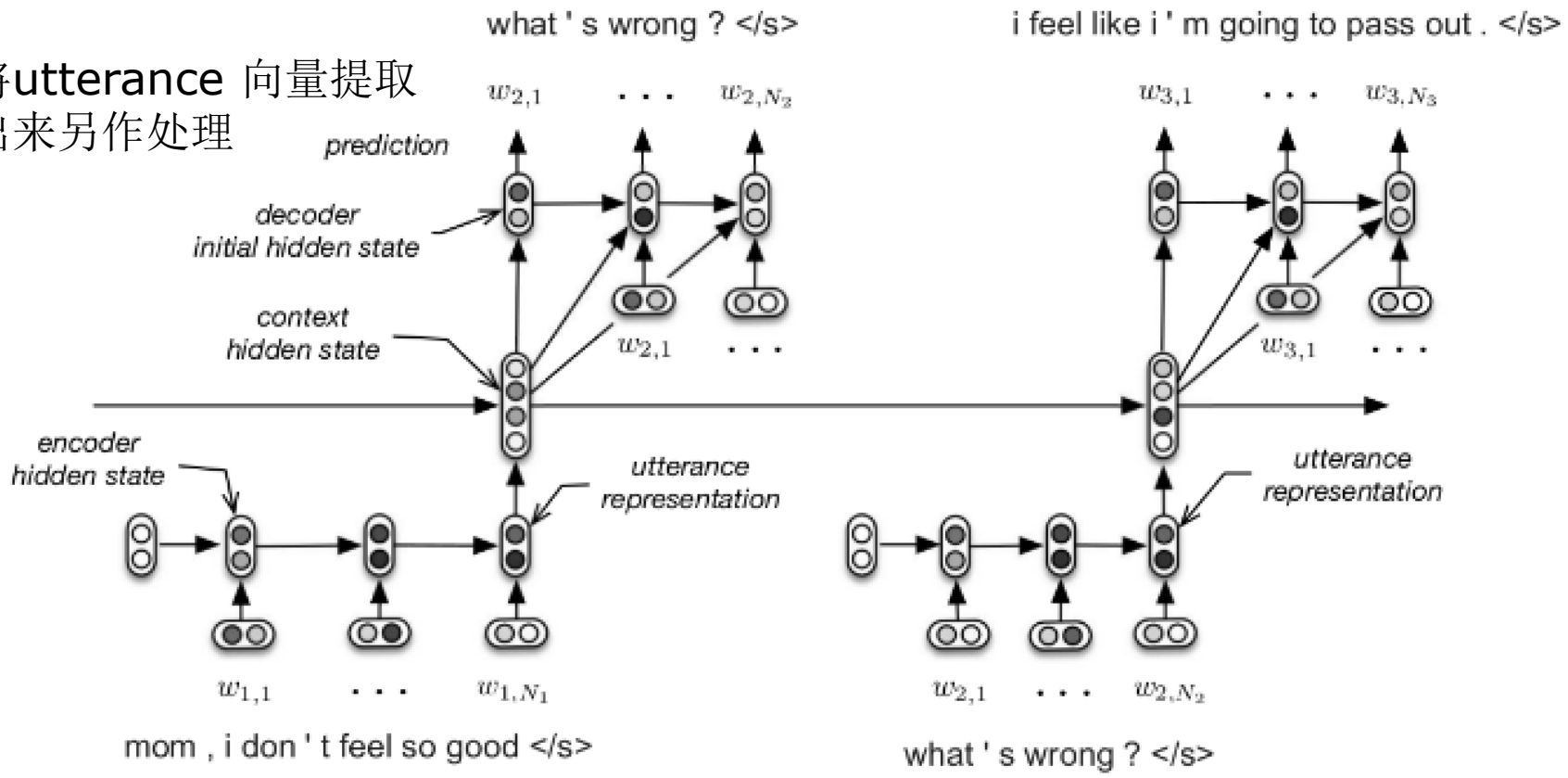


分层RNN结构的直观理解



分层RNN结构的直观理解

将utterance 向量提取出来另作处理



分层RNN结构的直观理解

- context RNN 可以记忆对话双方的语境
  - 在以“交流信息和想法”为重要目的的对话过程中，抓住语境中的话题，关键词，气氛等信息非常重要
  - utterance RNN记忆对话细节
  - context RNN记忆更加全局的语义信息
- 通过引入context RNN，降低了相邻的句子之间的计算步骤 (computational steps between utterances is reduced)，有助于信息的传播 (helps propagate the training signal for first-order optimization methods).

Bootstrapping训练

- 分层RNN相对于传统的单层seq2seq模型的提高并不明显（然而分层RNN模型还是有意义的）
- 而bootstrapping对优化对话效果有更明显的帮助
  - 相对于其他NLP任务，开放领域的聊天模型的训练数据比较稀缺
  - 使用其他NLP任务的数据预训练seq2seq模型，使得模型的参数预先学到一些NLP知识
    - Word embedding部分：Word2vec, Glove初始化
    - 其余参数：在5.5M样本的Question-Answer *SubTle* 数据集上训练对话模型

Bootstrapping训练

Model	Perplexity	Perplexity@U_3	Error-Rate	Error-Rate@U_3
Backoff N-Gram	64.89	65.05	-	-
Modified Kneser-Ney	60.11	54.75	-	-
Absolute Discounting N-Gram	56.98	57.06	-	-
Witten-Bell Discounting N-Gram	53.30	53.34	-	-
RNN	35.63 ± 0.16	35.30 ± 0.22	$66.34\% \pm 0.06$	$66.32\% \pm 0.08$
DCGM-I	36.10 ± 0.17	36.14 ± 0.26	$66.44\% \pm 0.06$	$66.57\% \pm 0.10$
HRED	36.59 ± 0.19	36.26 ± 0.29	$66.32\% \pm 0.06$	$66.32\% \pm 0.11$
HRED + Word2Vec	33.95 ± 0.16	33.62 ± 0.25	$66.06\% \pm 0.06$	$66.05\% \pm 0.09$
RNN + SubTle	27.09 ± 0.13	26.67 ± 0.19	$64.10\% \pm 0.06$	$64.07\% \pm 0.10$
HRED + SubTle	27.14 ± 0.12	26.60 ± 0.19	$64.10\% \pm 0.06$	$64.03\% \pm 0.10$
HRED-Bi. + SubTle	$\mathbf{26.81 \pm 0.11}$	$\mathbf{26.31 \pm 0.19}$	$\mathbf{63.93\% \pm 0.06}$	$\mathbf{63.91\% \pm 0.09}$

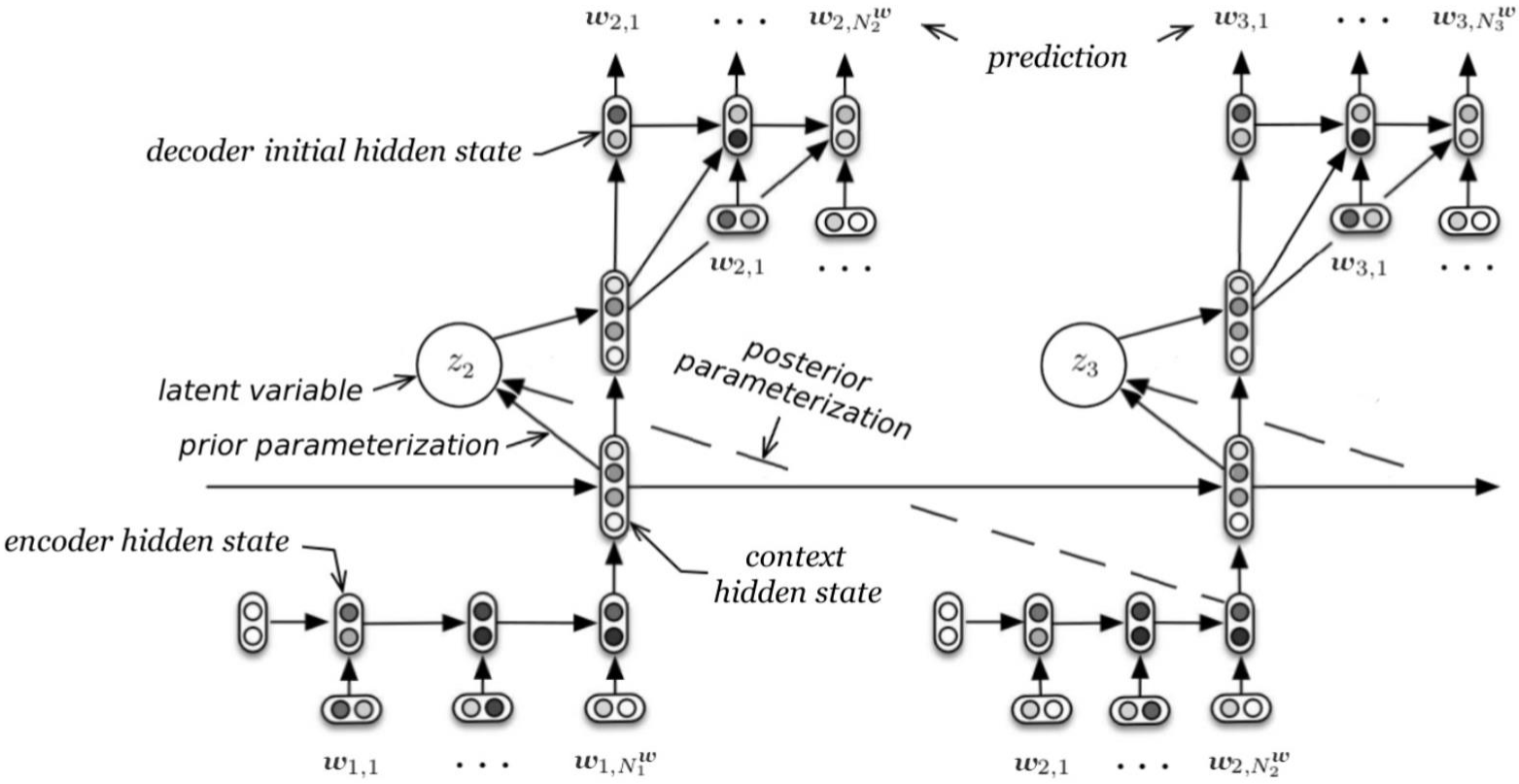
效果演示

Reference (U_1, U_2)	**MAP**	**Target** (U_3)
U_1: yeah , okay . U_2: well , i guess i ' ll be going now .	i ' ll see you tomorrow .	yeah .
U_1: oh . <continued_utterance> oh . U_2: what ' s the matter , honey ?	i don ' t know .	oh .
U_1: it ' s the cheapest . U_2: then it ' s the worst kind ?	no , it ' s not .	they ' re all good , sir .
U_1: <person> ! what are you doing ? U_2: shut up ! c ' mon .	what are you doing here ?	what are you that crazy ?

多(3) 轮对话的效果演示

- (U1, U2): 语境
- U3:回复

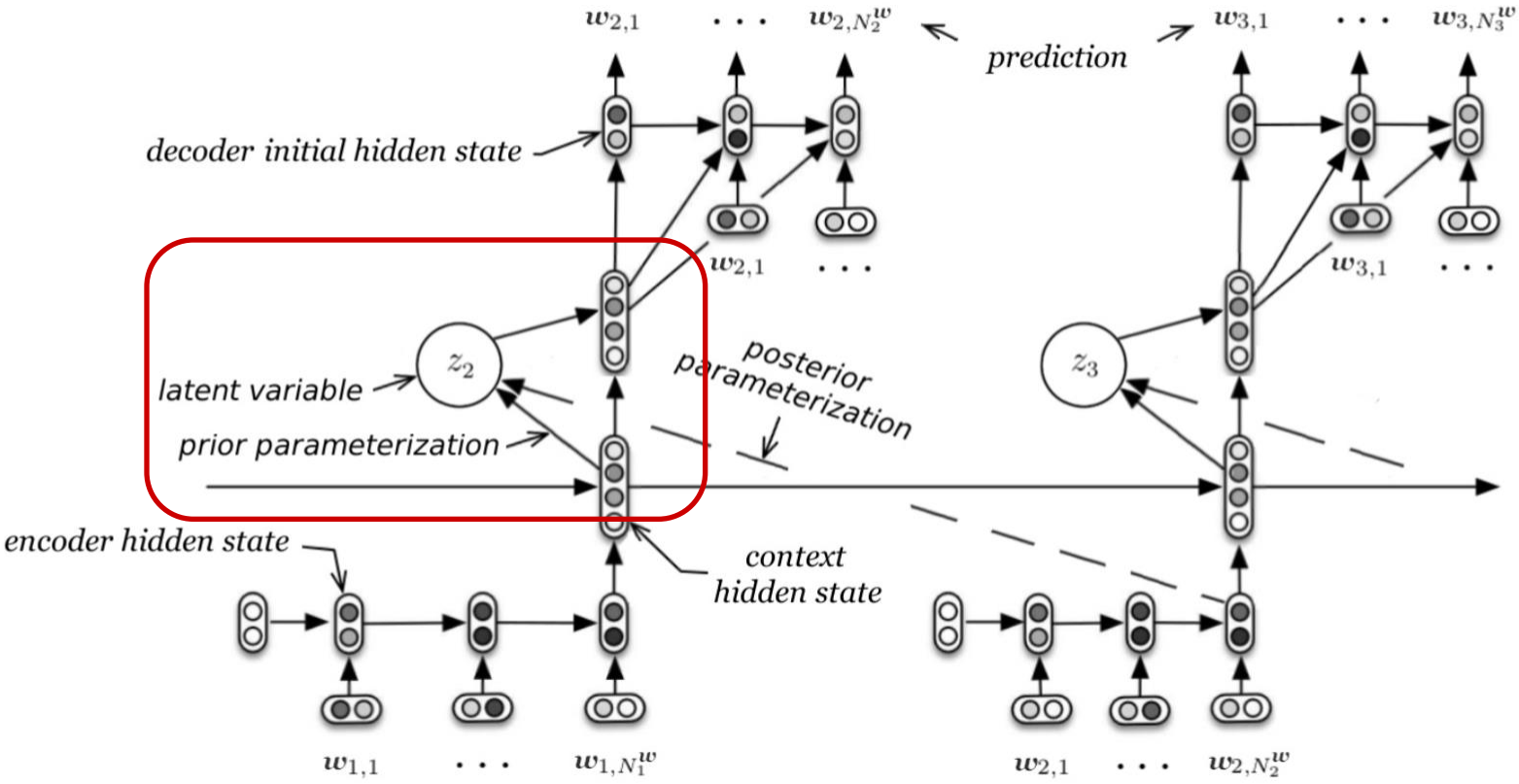
概率版分层RNN模型


$w_{2,1}$ . . . w_{2,N_2^w}
prediction
$w_{3,1}$ . . . w_{3,N_3^w}
decoder initial hidden state
$w_{2,1}$ . . .
$w_{3,1}$ . . .
z_2
z_3
latent variable
posterior parameterization
prior parameterization
encoder hidden state
context hidden state
$w_{1,1}$ . . . w_{1,N_1^w}
$w_{2,1}$ . . . w_{2,N_2^w}

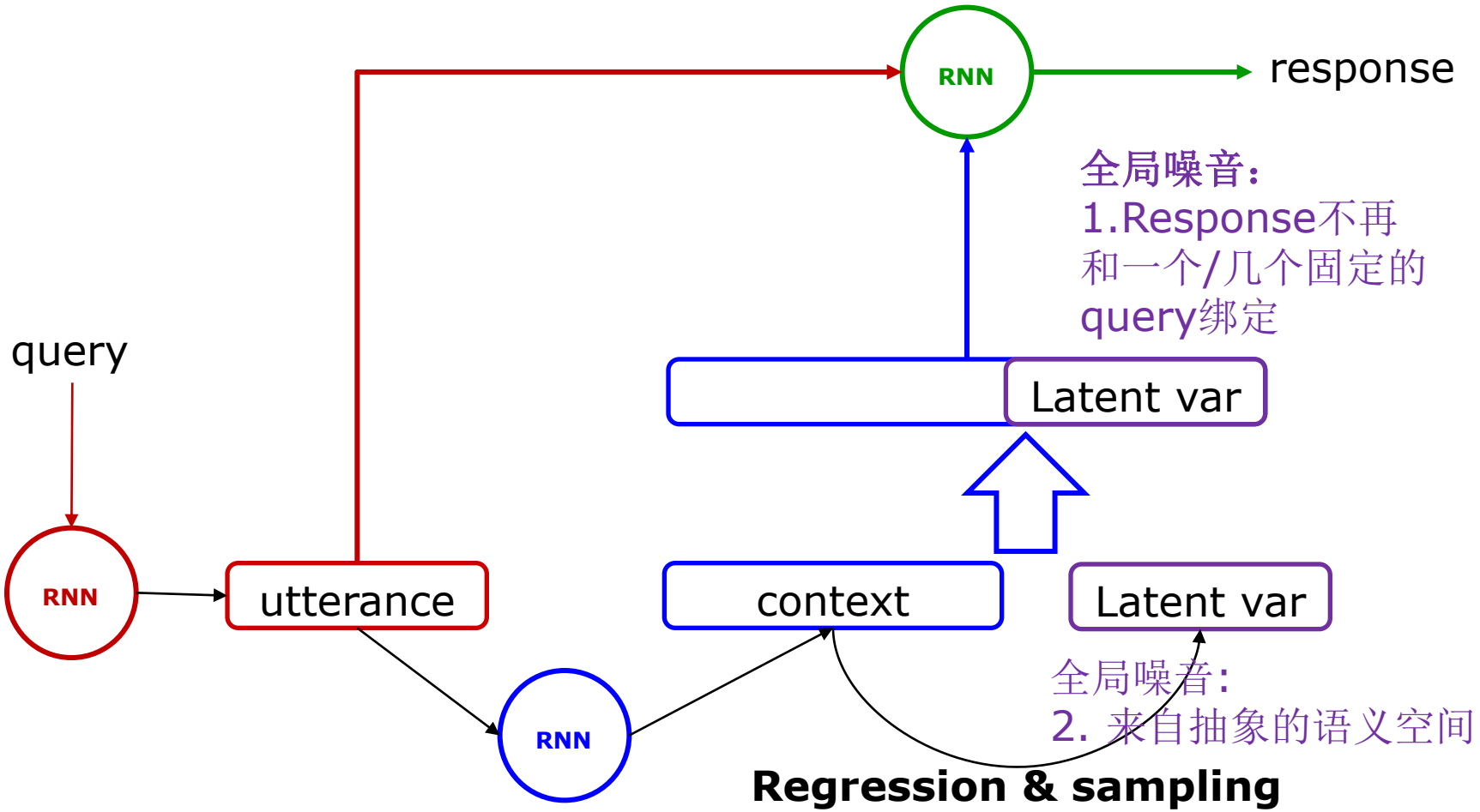
动机与思路

- 短的安全回答
  - 确定性的编码和解码过程
  - 过于着重拟合具体且有限的回复样本而忽略提取抽象语义信息
  - *“The most common form of error was a lack of understanding of the semantics of the responses”*
- 引入全局（语义层面）的随机因素
  - Robustness
  - High level concepts

模型细节


$w_{2,1}$ . . . w_{2,N_2^w}
prediction
$w_{3,1}$ . . . w_{3,N_3^w}
decoder initial hidden state
$w_{2,1}$ . . .
$w_{3,1}$ . . .
z_2
z_3
latent variable
prior parameterization
posterior parameterization
encoder hidden state
context hidden state
$w_{1,1}$ . . . w_{1,N_1^w}
$w_{2,1}$ . . . w_{2,N_2^w}

模型细节


RNN
response
全局噪音：
1.Response不再
和一个/几个固定的
query绑定
query
Latent var
RNN
utterance
context
Latent var
全局噪音：
2. 来自抽象的语义空间
RNN
Regression & sampling

模型效果

Table 2: Twitter examples for the neural network models. The → token indicates a change of turn.

Context	Response
jamaica blood!! <url> → lol. yes [name]!. [name]! → what's poppin in toronto homie	**VHRED:** nothing much. just chillin. what's good with you? **LSTM:** something!!!!!!! **HRED:** something!!!!!!
whhhhhhyyyy is it suddenly cold in tallahassee today?? this is florida! how can i dream of the beach when my fingers are icey cold?? → sittin on deck in sun looking at lake travis in austin! enjoying day b4 birthday	**VHRED:** the sun is shining!!! **LSTM:** i know!!!!!!!!! **HRED:** something!!!!!!!
reaches for [name]! → - approaches him warily - hey there buddy. - swings him into the air - how you doin? → tilts his head and your wariness and reaches for your face [name].	**VHRED:** - looks up at you and smiles - you're so cute. - kisses your cheek - **LSTM:** - smiles - i'm fine thank you. how are you? **HRED:** - grins, looking at you as i look at you -
i think i want an ipad2. but i already have an ibook. want vs. need. → if you have an ibook. then you deserve an ipad!	**VHRED:** that's what i'm thinking. i'm going to buy a new one. **LSTM:** i want one!!!! **HRED:** i don't know what to do with it. i don't know what to do.
ah echt? dan anrtwoord ik volgende keer via twitter als je me smst. → haha doe maar p. goed bezig hoor myrthe d → goed bezig? die snap ik even niet p	**VHRED:** haha, ik heb er geen zin in de something, maar ik heb er geen something meer something **LSTM:** haha something something something **HRED:** something something something something something something something something something
i love katie waissel more than like, anything in life. → i wanna c the show again! i need <number> meet her again! n i want a b-day tweet nxt week doubt that will happen x	**VHRED:** me too. i'm going to have to wait for her to come back. i'm so excited. **LSTM:** i'm sure she will. i'm sure she'll be fine. **HRED:** i'm sure she'll be fine, i'm sure she'll be fine.
my math teacher over here talkn bout nickels nd i'm tryna make big money i need to be in calculus → lol	**VHRED:** i need to go to the gym and get my hair done lol **LSTM:** lol **HRED:** how u been

更长的回复，更多的diversity

代码演示

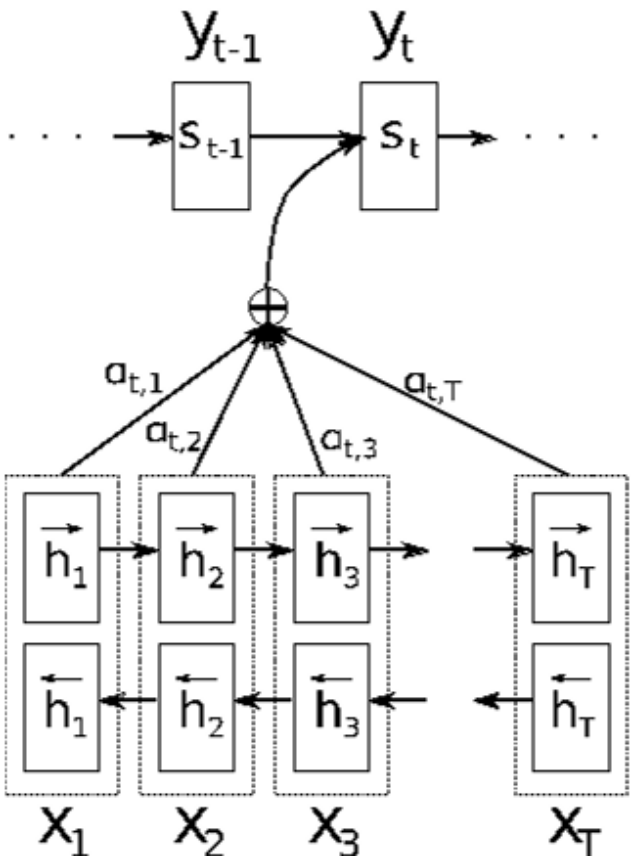
- 使用google/seq2seq实现看似复杂的分层RNN模型

Seq2seq模型中的注意力原理

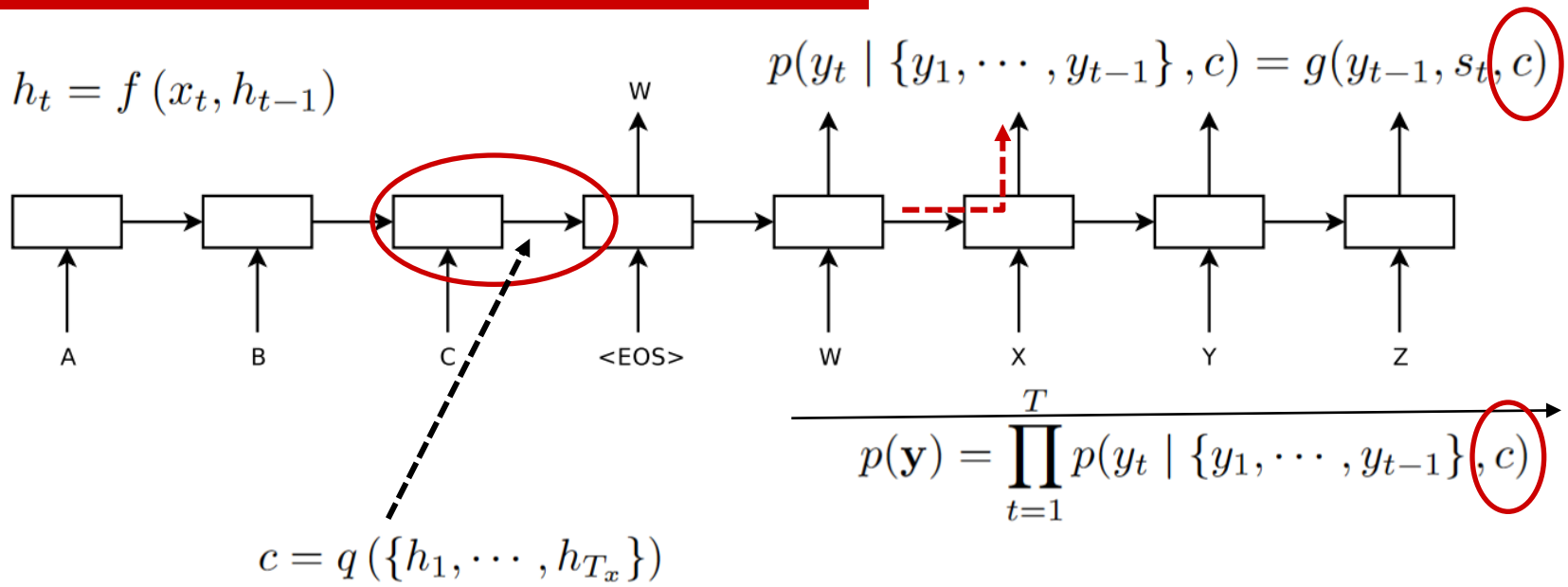
ATTENTION MECHANISM

机器翻译模型中的注意力原理

- Neural Machine Translation by Jointly Learning to Align and Translate (D. Bahdanau 2014)
- tf.contrib.seq2seq.BahdanauAttention:Bhadanau-style (additive) attention

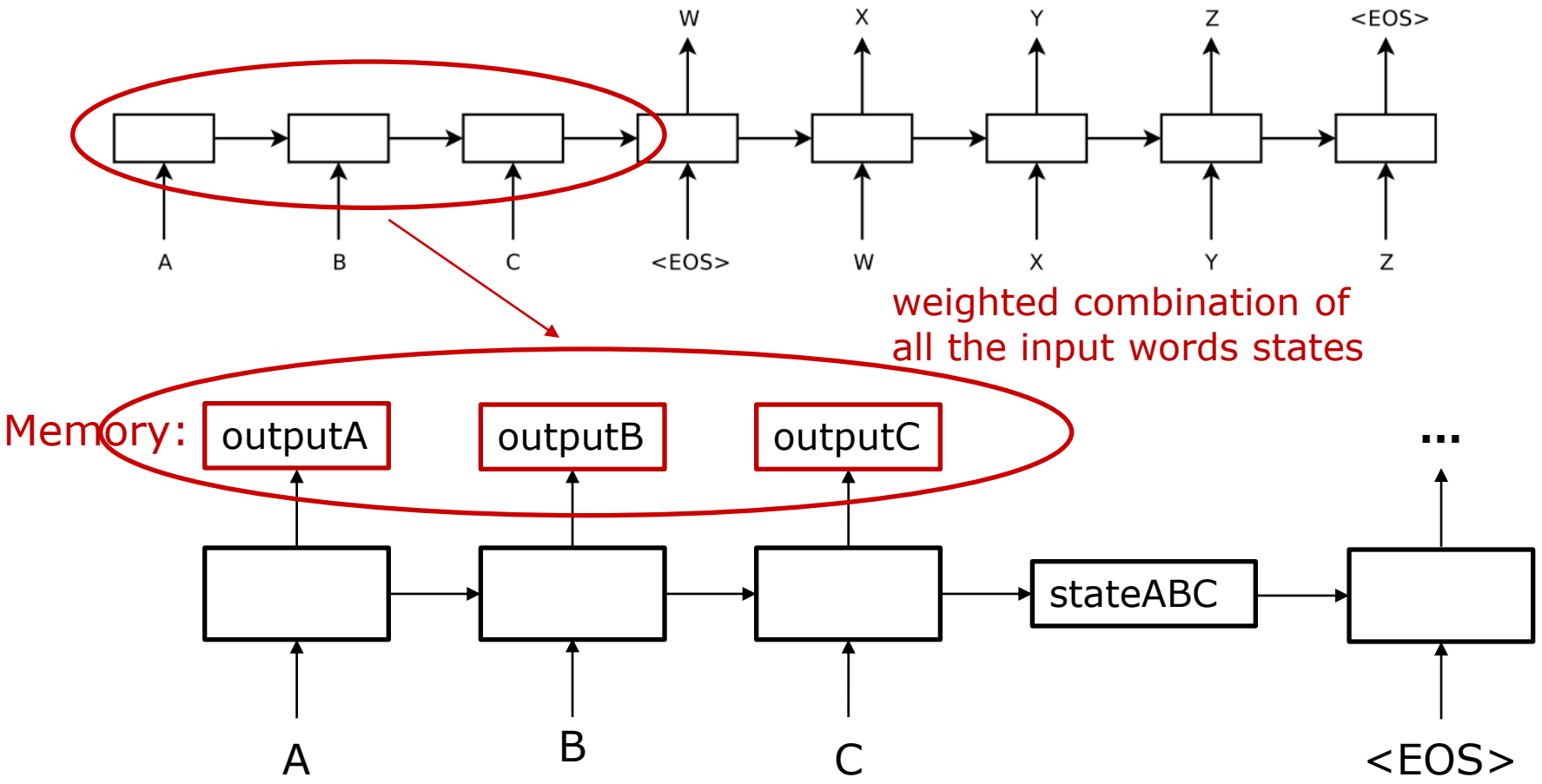


动机

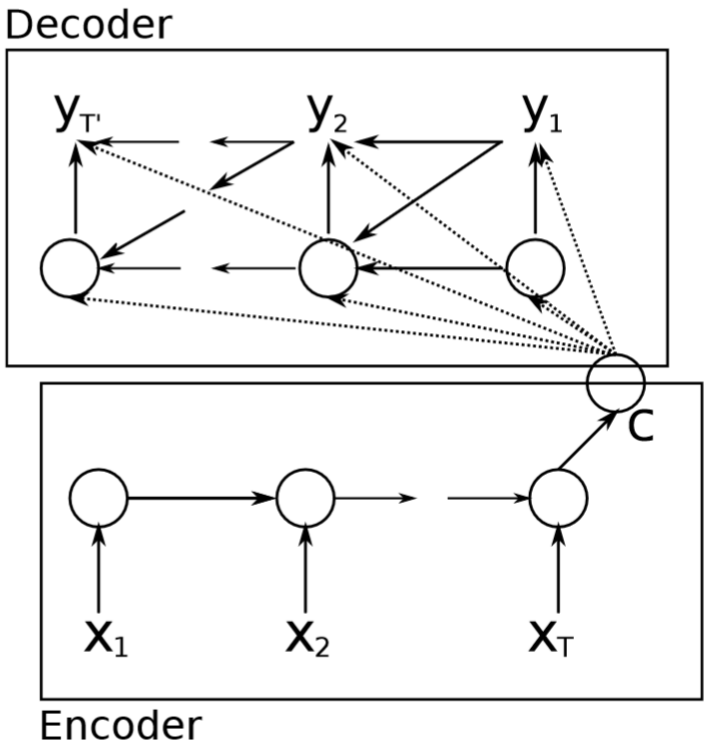


- 简单的seq2seq模型中，所有的语义信息保存在一个RNN cell的状态向量里面
- 当输入的句子比较长的时候，会丢失一些细节信息

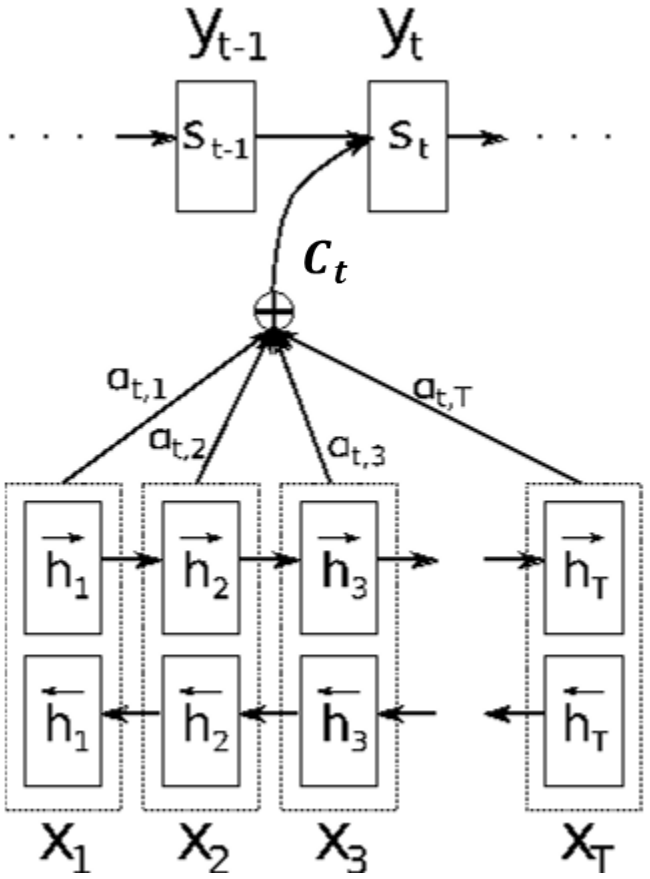
动机


W
X
Y
Z
<EOS>
A
B
C
<EOS>
W
X
Y
Z
weighted combination of
all the input words states
Memory:
outputA
outputB
outputC
...
stateABC
A
B
C
<EOS>

context 和 weighted context



RNN Encoder–Decoder
固定的context vector



RNN Encoder–Decoder
随时调整的 context vector

Attention mechanism

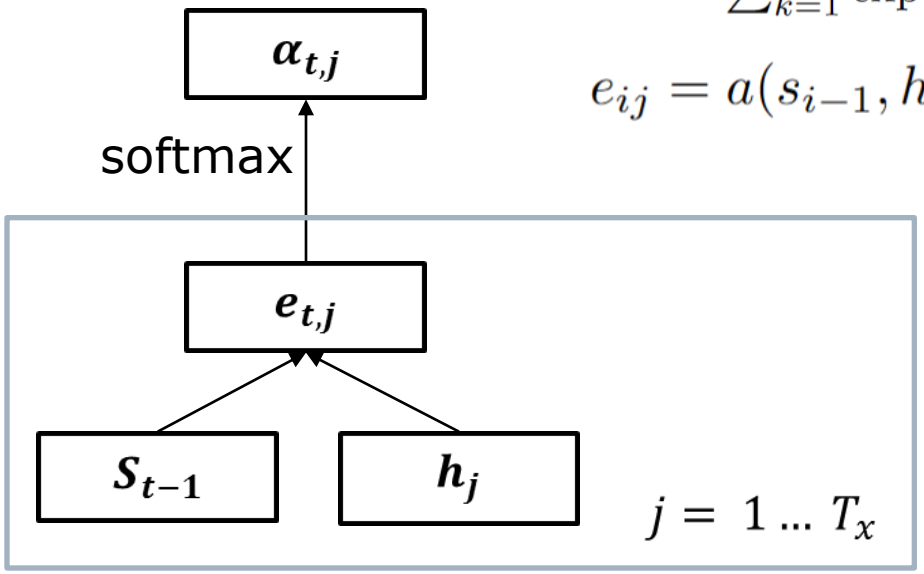
$$p(y_i|y_1,\ldots,y_{i-1},\mathbf{x}) = g(y_{i-1}, s_i, c_i)$$

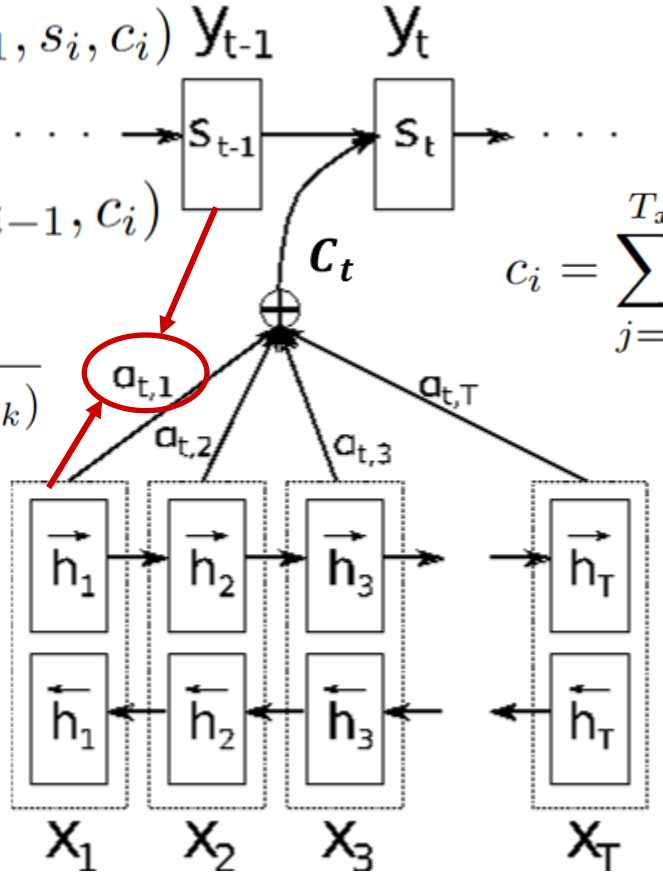
$$s_i = f(s_{i-1}, y_{i-1}, c_i)$$

$$\alpha_{ij} = \frac{\exp(e_{ij})}{\sum_{k=1}^{T_x}\exp(e_{ik})}$$

$$e_{ij} = a(s_{i-1}, h_j)$$

$$c_i = \sum_{j=1}^{T_x}\alpha_{ij}h_j$$

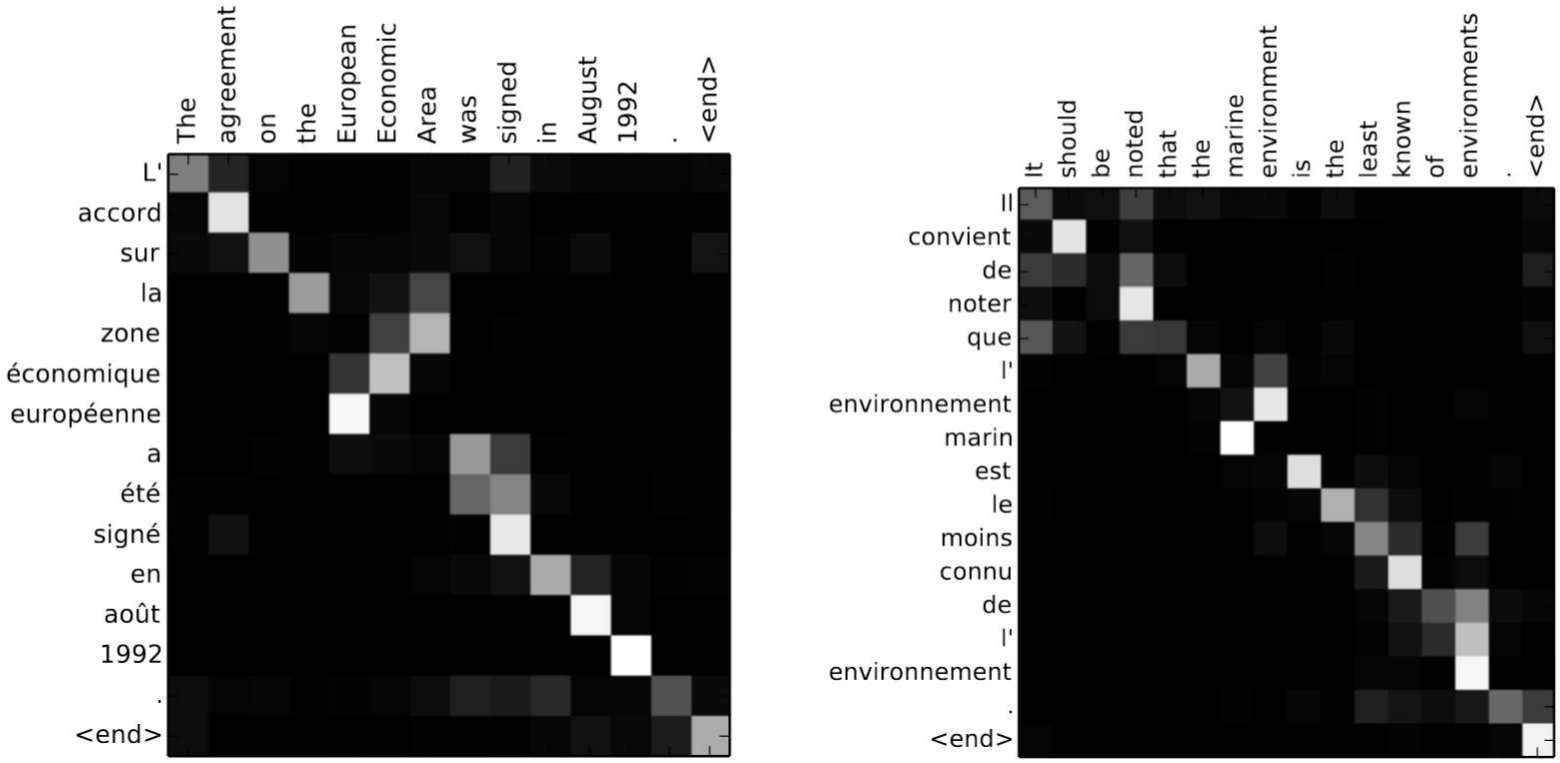




RNN Encoder–Decoder

随时调整的 context vector

Attention mechanism: example

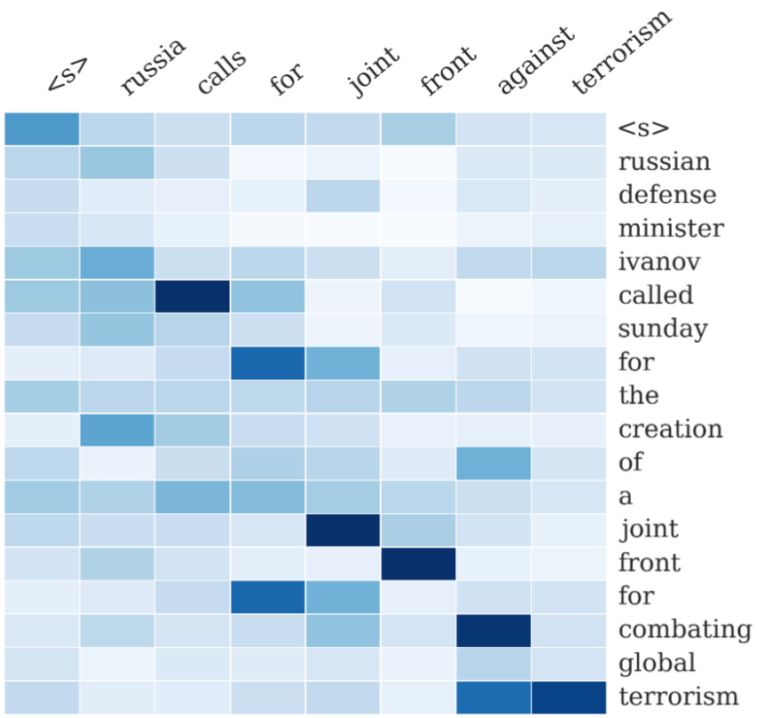


Attention Mechanism: example

- 动图演示
  - https://github.com/google/seq2seq
  - https://distill.pub/2016/augmented-rnns/#attentional-interfaces

Attention原理应用：文本总结

A neural attention model for abstractive sentence summarization (2015)

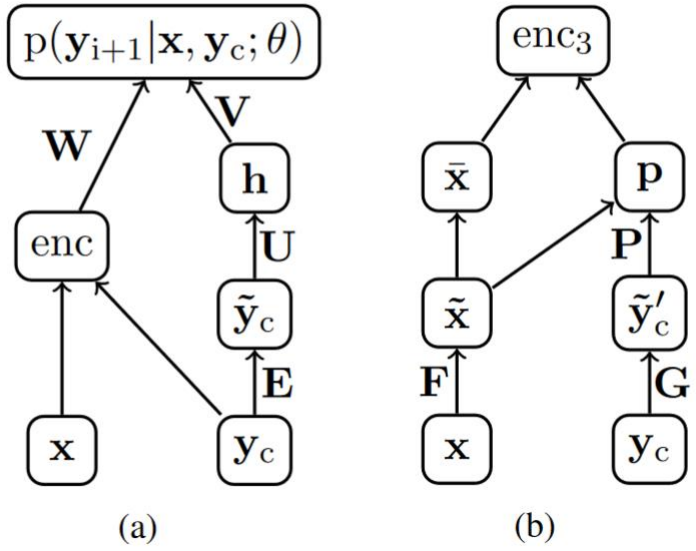


e.g.

- Russian defense minister Ivanov called Sunday for the creation of A joint front for combating global terrorism
- Russia calls for joint front against terrorism

Attention原理应用：文本总结

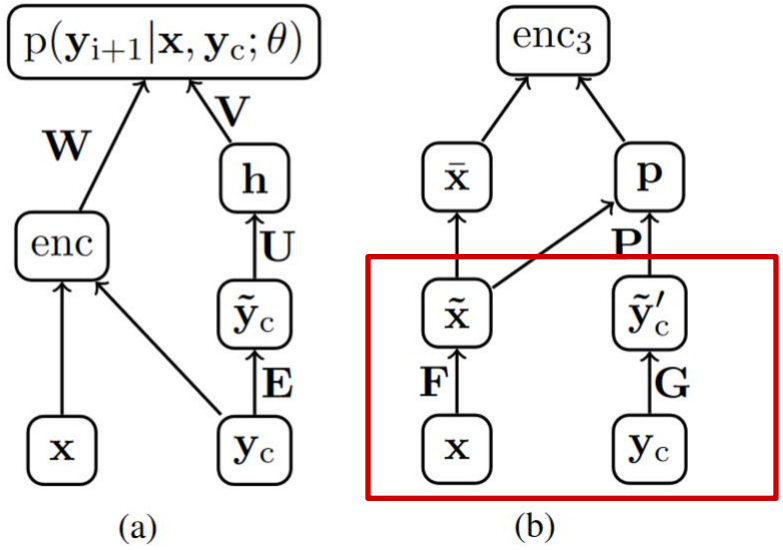
一个完全使用attention方法处理不定长度的sequence，不使用RNN模型



$$
\begin{aligned}
\mathrm{enc}_3(\mathbf{x}, \mathbf{y}_c) &= \mathbf{p}^\top \bar{\mathbf{x}}, \\
\mathbf{p} &\propto \exp(\tilde{\mathbf{x}} \mathbf{P} \tilde{\mathbf{y}}'_c), \\
\tilde{\mathbf{x}} &= [\mathbf{F}\mathbf{x}_1, \ldots, \mathbf{F}\mathbf{x}_M], \\
\tilde{\mathbf{y}}'_c &= [\mathbf{G}\mathbf{y}_{i-C+1}, \ldots, \mathbf{G}\mathbf{y}_i], \\
\forall i \quad \bar{\mathbf{x}}_i &= \sum_{q=i-Q}^{i+Q} \tilde{\mathbf{x}}_i / Q.
\end{aligned}
$$

$$
\log p(\mathbf{y}|\mathbf{x};\theta) \approx \sum_{i=0}^{N-1} \log p(\mathbf{y}_{i+1}|\mathbf{x}, \mathbf{y}_c;\theta)
$$

Attention原理应用：文本总结

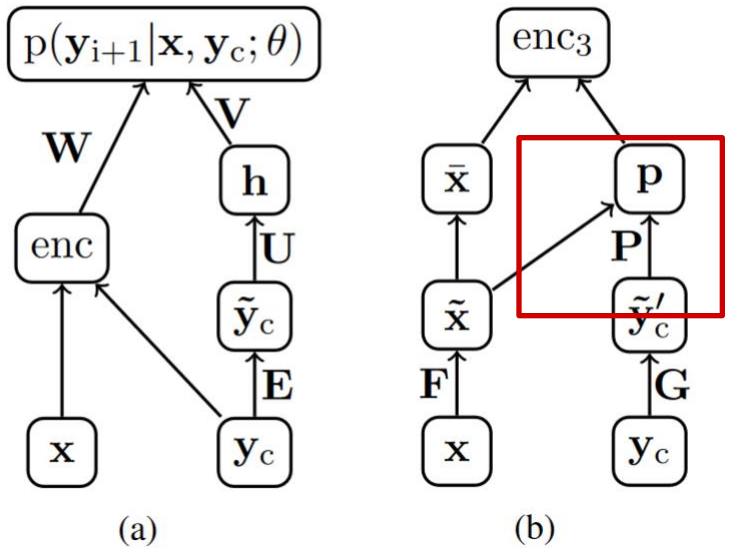


$$\log p(\mathbf{y}|\mathbf{x};\theta) \approx \sum_{i=0}^{N-1} \log p(\mathbf{y}_{i+1}|\mathbf{x},\mathbf{y}_c;\theta)$$

$$\begin{aligned}
\text{enc}_3(\mathbf{x},\mathbf{y}_c) &= \mathbf{p}^\top \bar{\mathbf{x}}, \\
\mathbf{p} &\propto \exp(\tilde{\mathbf{x}}\mathbf{P}\tilde{\mathbf{y}}'_c), \\
\tilde{\mathbf{x}} &= [\mathbf{F}\mathbf{x}_1,\ldots,\mathbf{F}\mathbf{x}_M], \\
\tilde{\mathbf{y}}'_c &= [\mathbf{G}\mathbf{y}_{i-C+1},\ldots,\mathbf{G}\mathbf{y}_i], \\
\forall i \quad \bar{\mathbf{x}}_i &= \sum_{q=i-Q}^{i+Q} \tilde{\mathbf{x}}_i/Q.
\end{aligned}$$

1. 转化为词向量
- 完整的输入句子
- 滑动窗内的输出句子

Attention原理应用：文本总结



$$\log p(\mathbf{y}|\mathbf{x};\theta) \approx \sum_{i=0}^{N-1} \log p(\mathbf{y}_{i+1}|\mathbf{x},\mathbf{y}_c;\theta)$$

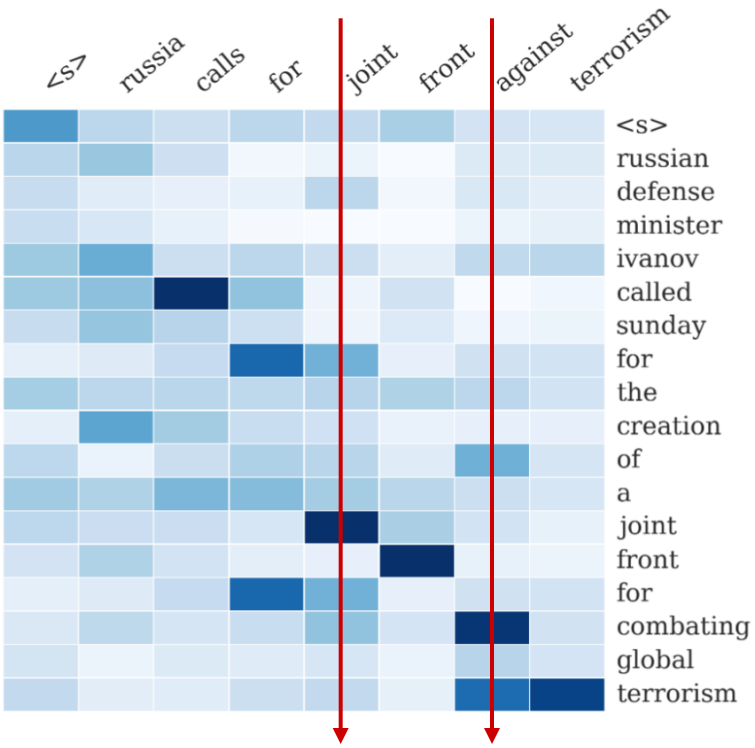
$$\begin{aligned}
\mathrm{enc}_3(\mathbf{x},\mathbf{y}_c) &= \mathbf{p}^\top \bar{\mathbf{x}}, \\
\mathbf{p} &\propto \exp(\tilde{\mathbf{x}}\mathbf{P}\tilde{\mathbf{y}}'_c), \\
\tilde{\mathbf{x}} &= [\mathbf{F}\mathbf{x}_1,\ldots,\mathbf{F}\mathbf{x}_M], \\
\tilde{\mathbf{y}}'_c &= [\mathbf{G}\mathbf{y}_{i-C+1},\ldots,\mathbf{G}\mathbf{y}_i], \\
\forall i \quad \bar{\mathbf{x}}_i &= \sum_{q=i-Q}^{i+Q} \tilde{\mathbf{x}}_i/Q.
\end{aligned}$$

Shape: [M, 1]

2. 计算输入和输出句子中的单词的相关度，即attention值

- P_i表示第i个输入单词和滑动窗里面的内容的attention权重

Attention原理应用：文本总结

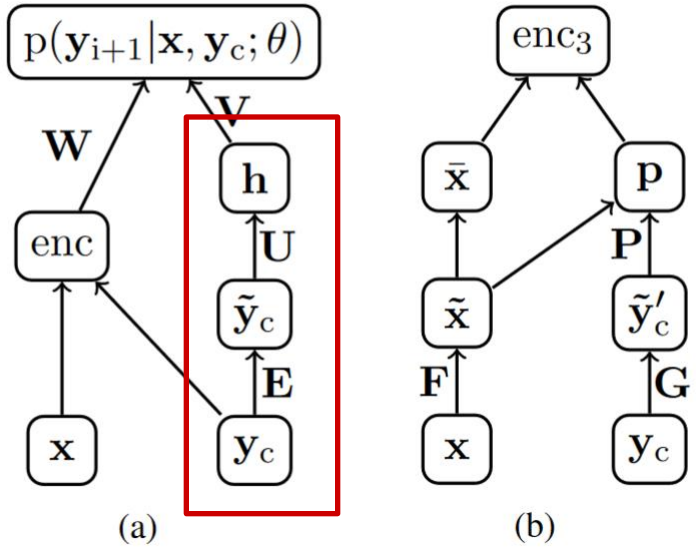


$$\begin{aligned}
\mathrm{enc}_3(\mathbf{x}, \mathbf{y}_c) &= \mathbf{p}^\top \bar{\mathbf{x}}, \qquad \text{Shape: [M, 1]} \\
\mathbf{p} &\propto \exp(\tilde{\mathbf{x}} \mathbf{P} \tilde{\mathbf{y}}'_c), \\
\tilde{\mathbf{x}} &= [\mathbf{F}\mathbf{x}_1, \ldots, \mathbf{F}\mathbf{x}_M], \\
\tilde{\mathbf{y}}'_c &= [\mathbf{G}\mathbf{y}_{i-C+1}, \ldots, \mathbf{G}\mathbf{y}_i], \\
\forall i \quad \bar{\mathbf{x}}_i &= \sum_{q=i-Q}^{i+Q} \tilde{\mathbf{x}}_i / Q.
\end{aligned}$$

2. 计算输入和输出句子中的单词的相关度，即attention值

- P_i表示第i个输入单词和滑动窗里面的内容的attention权重
- 如左图所示

Attention原理应用：文本总结



$$\log p(\mathbf{y}|\mathbf{x};\theta) \approx \sum_{i=0}^{N-1} \log p(\mathbf{y}_{i+1}|\mathbf{x},\mathbf{y}_c;\theta)$$

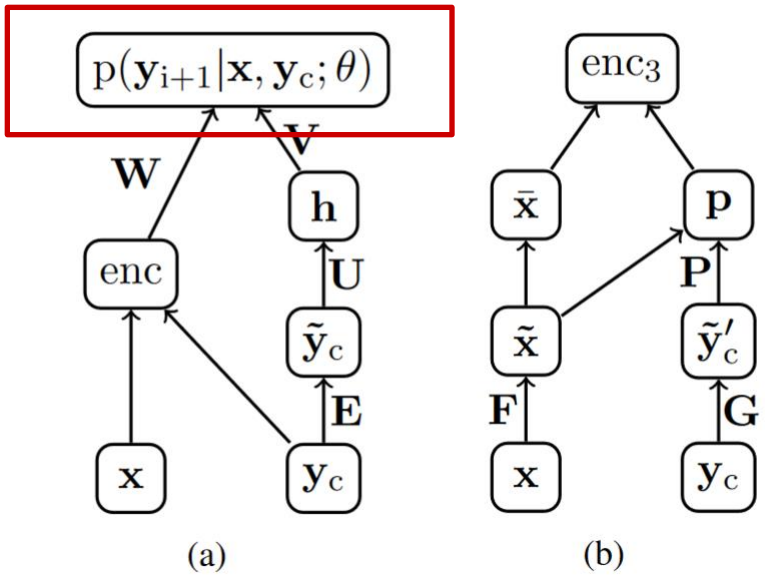
$$p(\mathbf{y}_{i+1}|\mathbf{y}_c,\mathbf{x};\theta) \propto \exp(\mathbf{V}\mathbf{h} + \mathbf{W}\mathrm{enc}(\mathbf{x},\mathbf{y}_c)),$$

$$\tilde{\mathbf{y}}_c = [\mathbf{E}\mathbf{y}_{i-C+1},\ldots,\mathbf{E}\mathbf{y}_i],$$

$$\mathbf{h} = \tanh(\mathbf{U}\tilde{\mathbf{y}}_c).$$

4. 简单的单层神经网络得到当前输出的部分句子的总结

Attention原理应用：文本总结

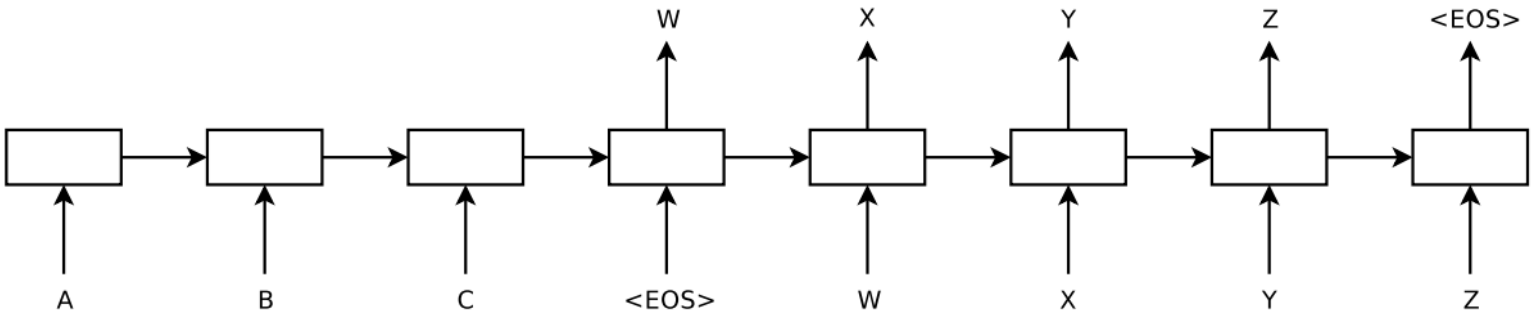


$$\log p(\mathbf{y}|\mathbf{x};\theta) \approx \sum_{i=0}^{N-1} \log p(\mathbf{y}_{i+1}|\mathbf{x},\mathbf{y}_c;\theta)$$

$$\begin{aligned} p(\mathbf{y}_{i+1}|\mathbf{y}_c,\mathbf{x};\theta) &\propto \exp(\mathbf{V}\mathbf{h} + \mathbf{W}\mathrm{enc}(\mathbf{x},\mathbf{y}_c)), \\ \tilde{\mathbf{y}}_c &= [\mathbf{E}\mathbf{y}_{i-C+1},\ldots,\mathbf{E}\mathbf{y}_i], \\ \mathbf{h} &= \tanh(\mathbf{U}\tilde{\mathbf{y}}_c). \end{aligned}$$

5. 简单的单层神经网络得到当前输出的部分句子的总结

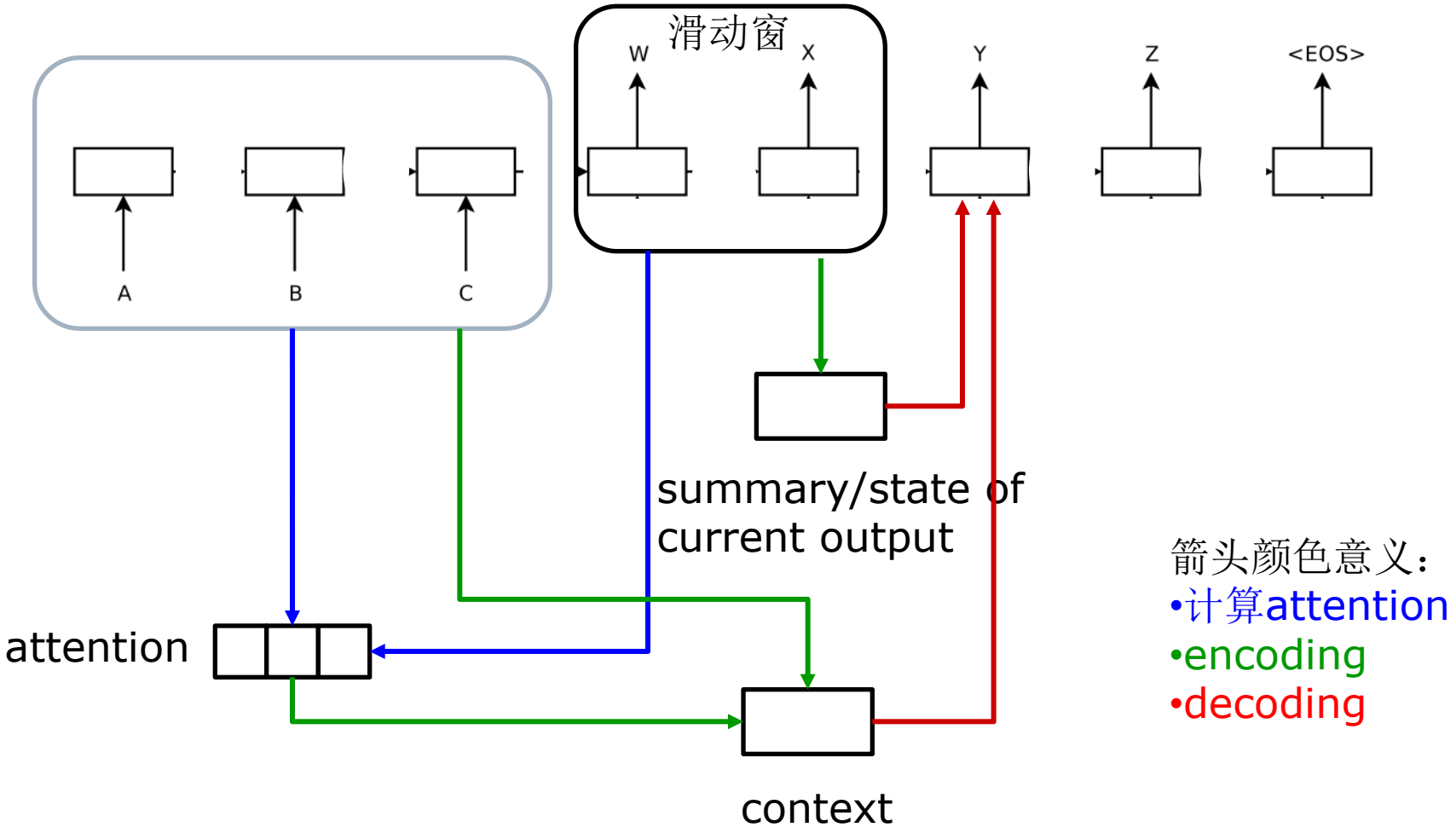
Attention原理应用：文本总结



我们试着在seq2seq这个图的基础上修改得到上面的模型。。。

- 去掉时间方向的关联
- 定位滑动窗
- 计算attention

Attention原理应用：文本总结

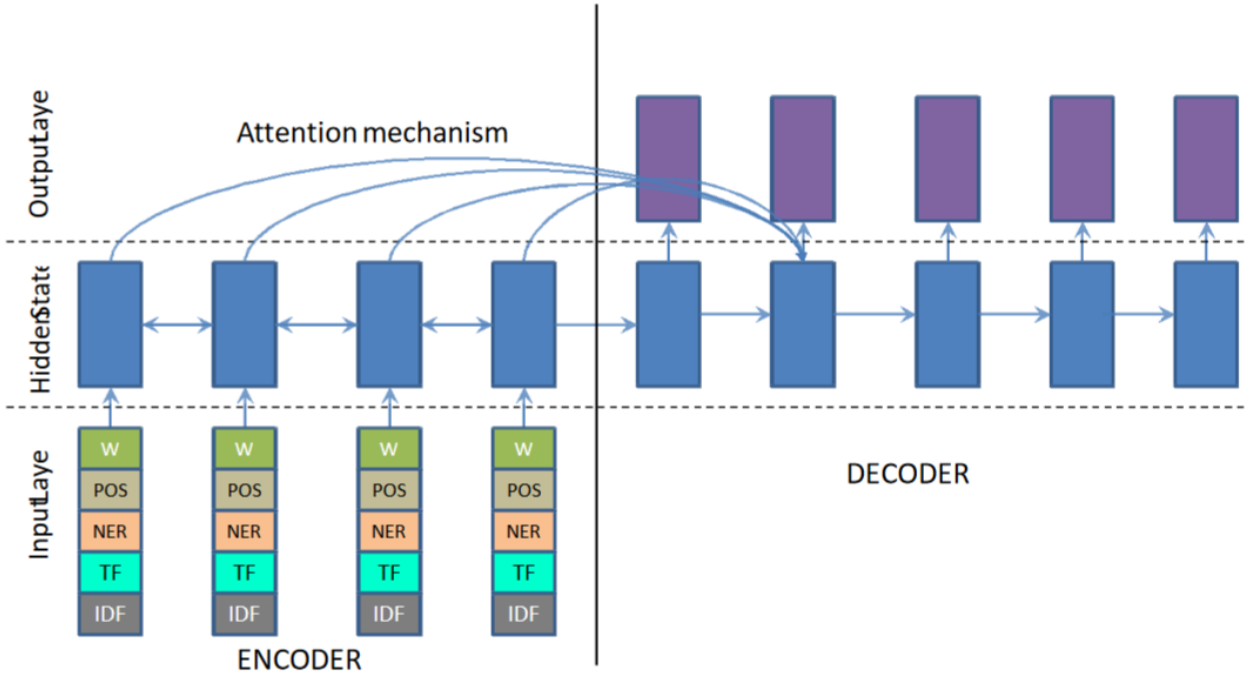


Attention原理应用：文本总结

- Abstractive text summarization using sequence-to-sequence RNNs and beyond (2016)
- 在seq2seq模型的基础上，提出三个工程方法提高文本总结的质量
  - 丰富的输入特征(Feature rich encoder)
  - 使用一个copy-gate处理稀有词汇问题(Switching generator-pointer model)
  - 使用分层RNN和分层attention实现包含多个句子的文本的总结(Hierarchical attention model)

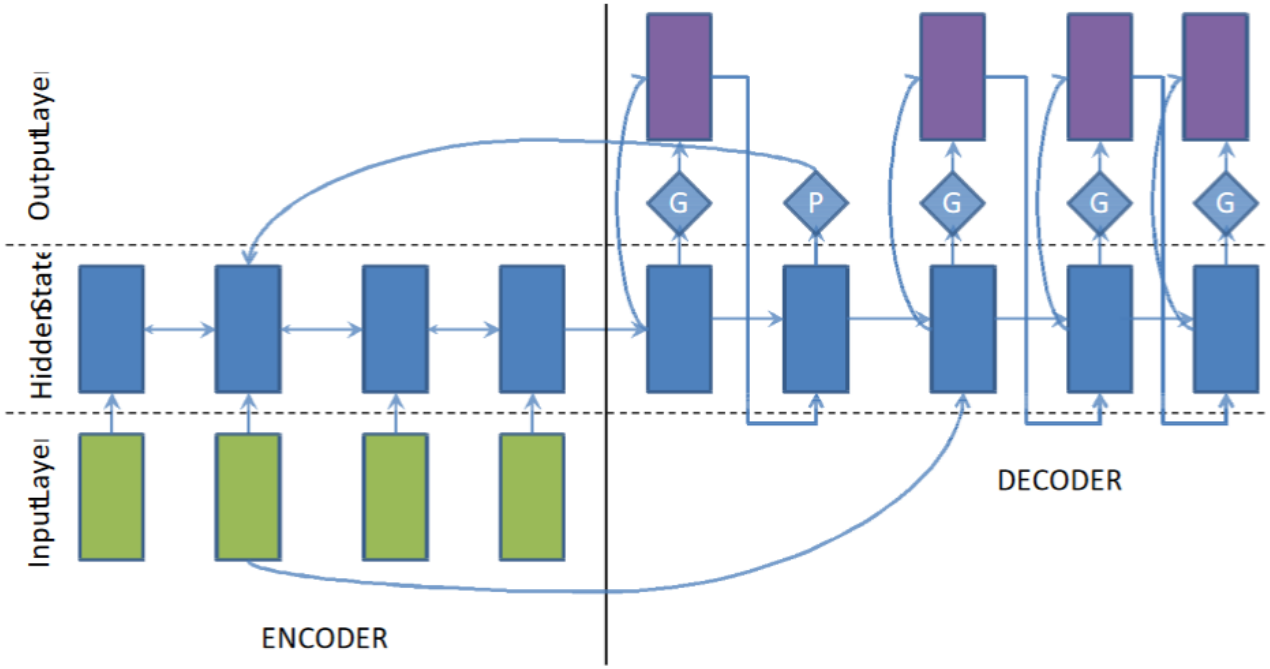
Attention原理应用：文本总结

- 丰富的输入特征(Feature rich encoder)



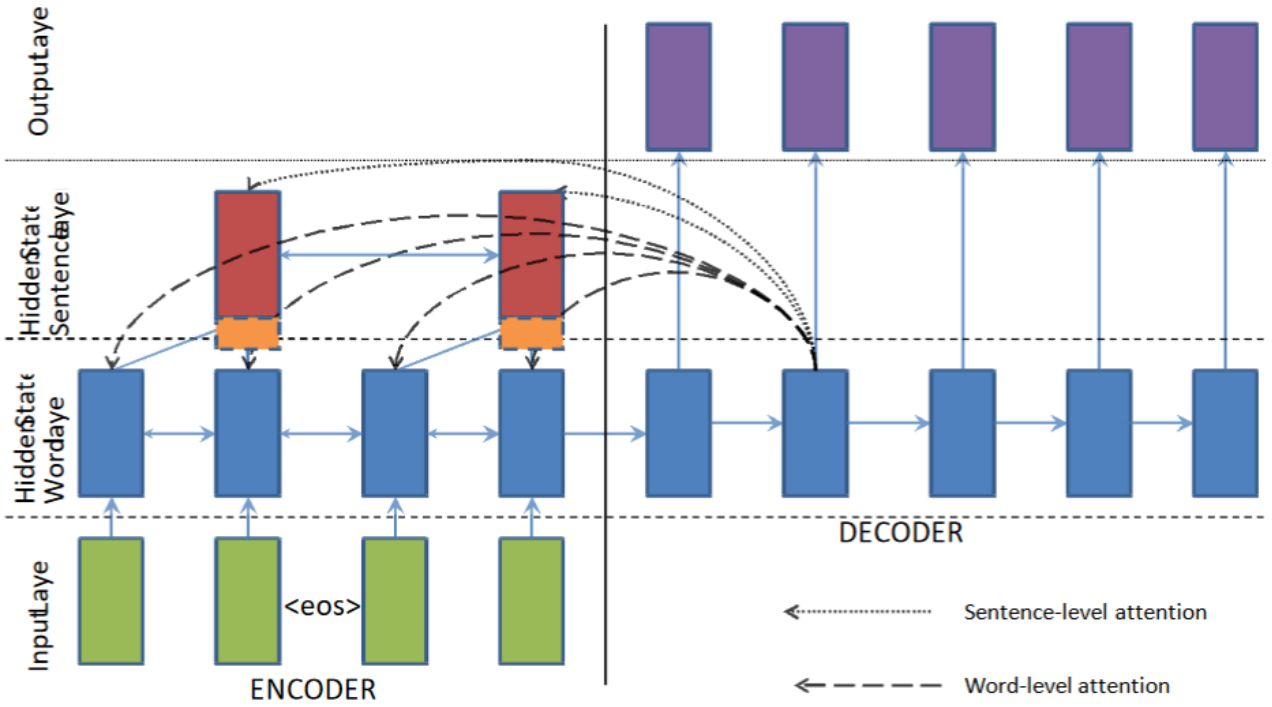
Attention原理应用：文本总结

- 使用一个copy-gate处理稀有词汇问题(Switching generator-pointer model)



Attention原理应用：文本总结

- 使用分层RNN和分层attention实现包含多个句子的文本的总结(Hierarchical attention model)



代码演示

- Attention seq2seq 模型用于文本（长句）总结

Attention原理应用：记忆网络

- 下周

疑问

- 问题答疑： http://www.xxwenda.com/
  - 可邀请老师或者其他人回答问题

联系我们

小象学院：互联网新技术在线教育领航者

- 微信公众号：大数据分析挖掘
- 新浪微博：ChinaHadoop