

Seq2Seq

本次上机作业包括两个py文件：seq2seq以及seq2seq_attention。两个文件各自都需要进行填空

本次上机作业是使用序列到序列模型实现 **将输入的大写字母序列逆置**

具体而言，对于Seq2seq模型，encoder输入一个大写字母序列 'ABCDEFGHJIJ'，我们希望在decoder处生成它的逆序序列 'JIHGFEDCBA'

seq2seq_attention则是在seq2seq模型的基础上添加了attention机制(这里采用双线性attention)

需要做的：

在看懂代码的基础上，填空补全代码，使之可以运行，将测试阶段运行结果(测试阶段输出的例子,选择一部分)注释在代码底部，**将补全后的并注释了运行结果的py文件上传**

注释长这样就行：

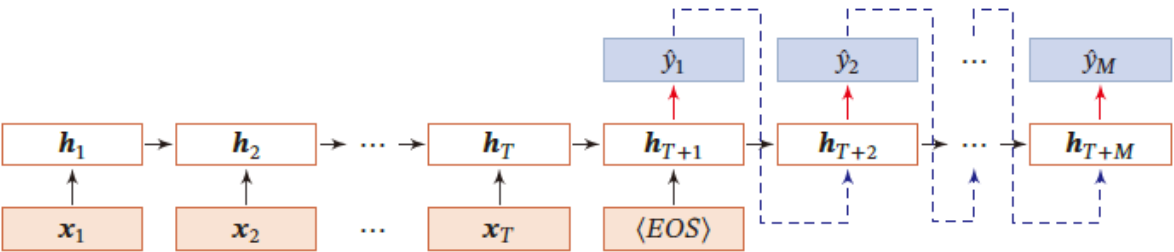
```
# Input: ['JJDDNZGZZF'], Should get: FZZGZNDDJJ, Model generate: FZZGNDDJJJ |The result is False
# Input: ['ZUPEIDWNVG'], Should get: GVNWDIEPUZ, Model generate: GVNWDIEPUZ |The result is True
# Input: ['EKWKBYLDV'], Should get: VDLYRBKWKE, Model generate: VDLYRBKWKE |The result is True
```

建议：

- 1. 先补全seq2seq.py再补全seq2seq_attention.py
- 2. 在补全代码之前，可以先看一看《神经网络与机器学习》-15.6.1 基于循环神经网络的序列到序列模型 与 15.6.2 基于注意力的序列到序列模型

序列到序列模型图示（《神经网络与深度学习》--6.3.3章）：

图6.5给出了异步的序列到序列模式示例，其中 $\langle EOS \rangle$ 表示输入序列的结束
虚线表示将上一个时刻的输出作为下一个时刻的输入。



双线性attention机制(《神经网络与深度学习》--8.2章)

双线性模型

$$s(\mathbf{x}, \mathbf{q}) = \mathbf{x}^\top \mathbf{W} \mathbf{q},$$

基于注意力的序列到序列模型:

15.6.2 基于注意力的序列到序列模型

为了获取更丰富的输入序列信息,我们可以在每一步中通过**注意力机制**来从输入序列中选取有用的信息.

在解码过程的第 t 步时,先用上一步的隐状态 $\mathbf{h}_{t-1}^{\text{dec}}$ 作为查询向量,利用注意力机制从所有输入序列的隐状态 $\mathbf{H}^{\text{enc}} = [\mathbf{h}_1^{\text{enc}}, \dots, \mathbf{h}_S^{\text{enc}}]$ 中选择相关信息

$$\mathbf{c}_t = \text{att}(\mathbf{H}^{\text{enc}}, \mathbf{h}_{t-1}^{\text{dec}}) = \sum_{i=1}^S \alpha_i \mathbf{h}_i^{\text{enc}} \quad (15.104)$$

$$= \sum_{i=1}^S \text{softmax}\left(s(\mathbf{h}_i^{\text{enc}}, \mathbf{h}_{t-1}^{\text{dec}})\right) \mathbf{h}_i^{\text{enc}} \quad (15.105)$$

其中 $s(\cdot)$ 为注意力打分函数.

然后,将从输入序列中选择的信息 \mathbf{c}_t 也作为解码器 $f_{\text{dec}}(\cdot)$ 在第 t 步时的输入,得到第 t 步的隐状态

$$\mathbf{h}_t^{\text{dec}} = f_{\text{dec}}(\mathbf{h}_{t-1}^{\text{dec}}, [\mathbf{e}_{y_{t-1}}; \mathbf{c}_t], \theta_{\text{dec}}). \quad (15.106)$$

最后,将 $\mathbf{h}_t^{\text{dec}}$ 输入到分类器 $g(\cdot)$ 中来预测词表中每个词出现的概率.