

# 深度学习

## Lab9-attention mechanism

兰韵诗

**本次Lab有作业，请在5月2日结束之前提交！**

# Lab9

- 1.熟悉用编码器-解码器的框架解决序列逆置任务的流程
- 2.补全rnn\_with\_attn.py文件中的融合注意力机制的RNN模型

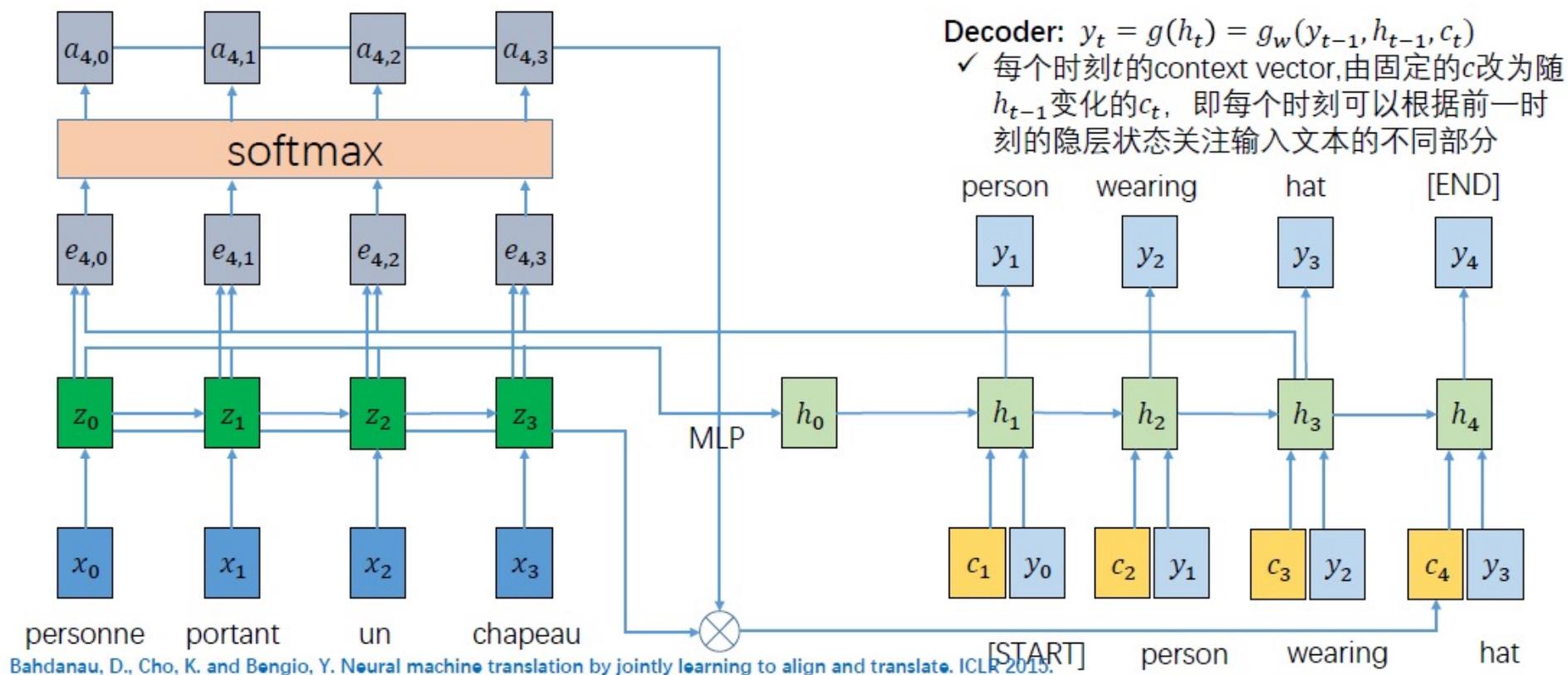
# Attention Mechanism

- 根据提示，补全**融合注意力机制的解码器**代码，实现序列逆置任务
  - 利用设定好的输入完成**要求的RNN with attention模型**
  - 所有预设的网络层都应当用到
  - 不能修改给定的对象属性，不能调用其他工具包，只能在“to do”下面书写代码
  - 提交之后，测试集上的准确率应该提升到一个正确的范围内
  - 可多次提交。即使对自己的代码没有自信也一定要提交，我们会酌情给过程分
  - 本次Lab在截止日期之前将不公开榜单
- **TO DO**：完成《Attention Mechanism》项目。补全rnn\_with\_attn.py文件使exercise\_reverse\_sequence.py文件中的train\_with\_RNN()可以顺利执行。

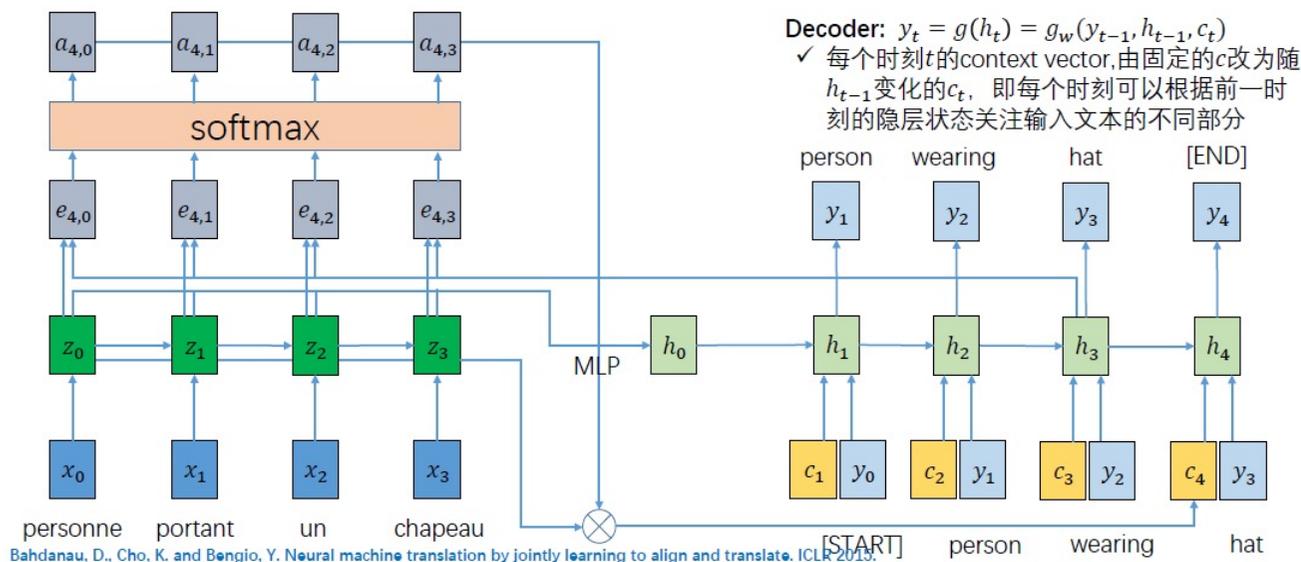
# 序列逆置任务

- 输入一个序列要求模型输出序列的逆置
  - 如：输入“ABCDEFGH”，输出“HGFEDCBA”

# 模型结构



# 模型结构



Input:  $x = x_1, x_2, \dots, x_T$

Output:  $y = y_1, y_2, \dots, y_T$

Encoder:  $z_t = RNN(x_t, z_{t-1})$

$z_t$  为编码器的隐藏状态

$$h_0 = \frac{1}{n} \sum_{i=1}^T \{MLP_1([z_1, z_1, \dots, z_t])\}$$

Decoder:

$h_t = RNN(h_{t-1}, [c_t \oplus y_{t-1}])$

$h_t$  为解码器的隐藏状态

$c_t = \text{attn}(Z, h_{t-1})$

其中  $Z = [z_1, z_2, \dots, z_T]$

$$\alpha_i = \frac{\exp(z_i^T h_{t-1})}{\sum_{j=1}^T \exp(z_j^T h_{t-1})}$$

$$c_t = \sum_{i=1}^T \alpha_i z_i$$

Output:  $\hat{y}_t = MLP_2(h_t)$

$\hat{y}_t$  为预测的字符

# Evaluation脚本

```
def compute_acc(pred_file):  
    with open('data/test_X.txt') as f:  
        gold = f.readlines()  
    gold = [sent.strip() for sent in gold]  
    gold = [''.join([o for o in reversed(e_idx)]) for e_idx in gold]  
  
    with open(pred_file) as f:  
        pred = f.readlines()  
    pred = [sent.strip() for sent in pred]  
    correct_case = [i for i, _ in enumerate(gold) if gold[i] == pred[i]]  
  
    acc = len(correct_case)*1./len(gold)  
    print('The predicted accuracy is %s' %acc)
```