媒体与认知 第三次作业

(V1.0　2020.4.23)

本次作业编程作业内容包括传统循环神经网络和门控循环单元的前向计算与误差沿时间反向传播（BPTT）算法编程实现，以帮助同学们理解相关原理。

【作业内容简介】

一、传统循环神经网络

1. 循环神经网络单元

传统循环神经网络主要有 Elman 网络和 Jordan 网络，二者在反馈回路上有所不同。对于具有一个隐含层和输出层的多层感知机，Elman 网络是将上一时刻隐含层状态作为隐含层的反馈输入。Jordan 网络将上一时刻网络输出层的输出作为隐含层的反馈输入。本次作业基本编程任务为实现 Elman 循环神经网络单元，对应程序为\program\mytorch\rnn_cell.py。

1) 前向计算过程

$$h'_{t,l} = f(W_{ih}^{(l)} h_t^{(l-1)} + b_{ih}^{(l)} + W_{hh}^{(l)} h_{t-1}^{(l)} + b_{hh}^{(l)})$$

$f(\cdot)$为激活函数，循环神经网络的激活函数一般采用双曲正切函数$tanh$或$ReLU$（rectified linear unit，整流线性单元)函数。

输入：

- x (batch size, input size)
  - 当前时刻的输入；
  - 如果是第一层，则为x_t。如果不是第一层，则为当前时刻前一层的隐含状态$h_t^{(l-1)}$，也可记为 $h_{t,l-1}$。
- h (batch size, hidden size)
  - 前一时刻当前层的隐含状态$h_{t-1}^{(l)}$，也可记为$h_{t-1,l}$。

在本次编程作业(rnn_cell.py)中，以 h_prev_l 表示当前时刻(t)前一层（第$l-1$层）隐含状态$h_t^{(l-1)}$，以 h_prev_t 表示前一时刻$(t-1)$当前层（第l层）的隐含状态$h_{t-1}^{(l)}$。

输出：

- h prime: (batch size, hidden size)
  - 当前时刻当前层隐含状态 $h'_{t,l}$

2) 误差沿时间反向传播

涉及如下梯度的计算：

1. $\frac{\partial L}{\partial W_{ih}}$ (self.dW_ih)
2. $\frac{\partial L}{\partial W_{hh}}$ (self.dW_hh)
3. $\frac{\partial L}{\partial b_{ih}}$ (self.db_ih)

4. $\frac{\partial L}{\partial b_{hh}}$ (self.db_hh)
5. *dx* (函数返回值)
6. *dh* (函数返回值)

输入：

- delta: (batch _size, hidden size)
  - 即误差对当前时刻当前隐含状态的梯度 $\frac{\partial L}{\partial h_t^{(l)}}$
  - 利用误差对当前时刻后一层隐含状态的梯度与对后一时刻当前层隐含状态的梯度计算得到$\frac{\partial L}{\partial h_t^{(l)}} = \frac{\partial L}{\partial h_t^{(l+1)}}\frac{\partial h_t^{(l+1)}}{\partial h_t^{(l)}} + \frac{\partial L}{\partial h_{t+1}^{(l)}}\frac{\partial h_{t+1}^{(l)}}{\partial h_t^{(l)}}$。
- h: (batch size, hidden size)
  - 当前时刻当前层隐含状态 $h_t^{(l)}$
- h_prev_l: (batch size, input size)
  - 当前时刻前一层的隐含状态$h_t^{(l-1)}$.
  - 如果是第一层，则为 x_t
- h_prev_t: (batch size, hidden size)
  - 前一时刻当前层的隐含状态$h_{t-1}^{(l)}$

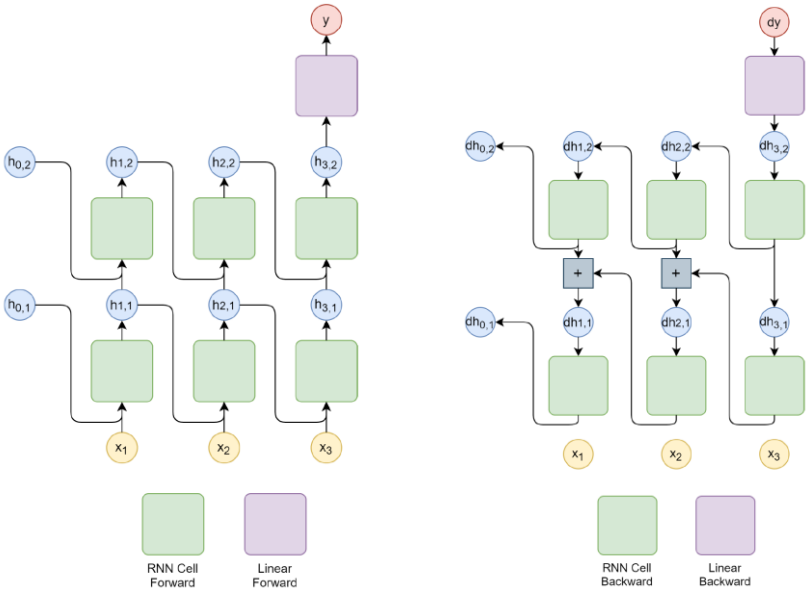
输出：

- dx: (batch_size, input size)
  - 误差经$h_t^{(l)}$传递，对当前时刻前一层隐含状态的梯度$\frac{\partial L}{\partial h_t^{(l)}}\frac{\partial h_t^{(l)}}{\partial h_t^{(l-1)}}$
  - 如果是第一层，则为$\frac{\partial L}{\partial x_t}$
- dh: (batch size, hidden size)
  - 误差经$h_t^{(l)}$传递，对前一时刻当前层隐含状态的梯度$\frac{\partial L}{\partial h_t^{(l)}}\frac{\partial h_t^{(l)}}{\partial h_{t-1}^{(l)}}$

2. 基于循环神经网络的分类任务示意

通常循环神经网络用于序列建模，训练过程采用基于 CTC 的损失函数 ，但循环神经网络也可以用于分类任务。本次作业中的 RNN 音素分类任务定义了如图 1 所示的循环神经网络。循环神经网络第二个隐含层在时刻 T 的输出经过一个线性层，采用 Softmax-Cross Entropy Loss 作为音素分类训练过程的目标函数。

此部分代码只是为理解原理的代码示意，并不是可以实现完整训练过程的代码，比如，缺少更新网络参数等部分代码。在 PyTorch 中，基于梯度下降法更新网络参数是利用 optimizer 类中的 step 方法完成的。



(a) 按时间展开的前向计算过程　　(b)误差沿时间的反向传播过程

图 1. 基于循环神经网络的音素分类

二、门控循环单元的实现

门控循环单元可以有多种具体实现方式，本次编程作业用到的一种门控循环单元的实现方式如图 2 所示。

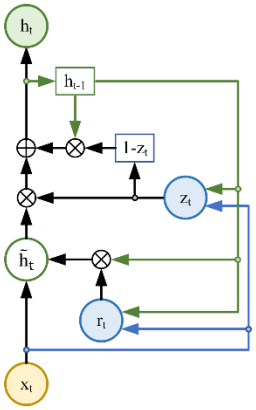


图 2. 本次编程作业中的门控循环单元

1. 前向过程

具体计算公式为：

$$z_t = \sigma(W_{zh}h_{t-1} + W_{zx}x_t)$$
$$r_t = \sigma(W_{rh}h_{t-1} + W_{rx}x_t)$$
$$\tilde{h}_t = tanh(W_h(r_t \otimes h_{t-1}) + W_x x_t)$$
$$h_t = (1 - z_t) \otimes h_{t-1} + z_t \otimes \tilde{h}_t$$

请注意与课件中介绍的标准 GRU 公式相比，省去了偏置量，并且更新门相当于取反。

2. 误差反向传播过程

函数输入为误差对当前时刻当前隐含状态的梯度 delta，需要计算对网络参数的梯度，并返回误差对输入x_t和h_{t-1}的导数。

利用复合函数的梯度计算链式法则，分别计算误差对六个权重矩阵的的偏导数，以及误差经过$h_t^{(l)}$对输入x_t和h_{t-1}的偏导数，共涉及 8 个梯度的计算：

1. $\frac{\partial L}{\partial W_{zx}}$ (self.dWzx)
2. $\frac{\partial L}{\partial W_{zh}}$ (self.dWzh)
3. $\frac{\partial L}{\partial W_{rx}}$ (self.dWrx)
4. $\frac{\partial L}{\partial W_{rh}}$ (self.dWrh)
5. $\frac{\partial L}{\partial W_{x}}$ (self.dWx)
6. $\frac{\partial L}{\partial W_{h}}$ (self.dWh)
7. $\frac{\partial L}{\partial x_t}$ (返回值 dx)
8. $\frac{\partial L}{\partial h_t^{(l)}}\frac{\partial h_t^{(l)}}{\partial h_{t-1}^{(l)}}$ (返回值 dh)

3. 推断

在 hw3 / hw3.py 中，使用上一节中实现的 GRUCell 和线性层组成一个简单神经网络，仅包含一个隐含层，可用于完成一个预测文本中下一个字符的序列建模任务。与 RNN 音素分类器采用最后一层最后时刻的输出不同，该神经网络按时间展开，在每个时刻输出预测值 logits。在本次编程作业中，需要在 init 函数中先初始化 GRU 单元和线性层。然后完成前向计算过程。

- CharacterPredictor 的 forward 方法应同时返回输出层输出及更新的隐含状态。
  - (logits, hnext)

推断函数有如下输入输出：

- 输入
  - net: CharacterPredictor 的实例；
  - inputs (seq len, feature dim): 输入数据序列。
- 输出
  - logits (seq len, num classes): 按序列长度 seq_len 存储的预测值，即每一时刻对应于各类别字符的预测值，取值没有进行归一化。num classes 是线性层预测的字符集合数目。

【作业具体任务】

本次作业具体任务包括：

一、 选择题 （有助于理解本次编程作业）

二、 补全程序代码

三、 自动评判程序实现效果

四、 提交程序及作业报告

本次作业原始成绩 100 分，选择题 10 分，补全程序代码 80 分，作业报告 10 分。

任务一、选择题

1. 查看 PyTorch 中 RNNCell 的说明 (https://pytorch.org/docs/stable/nn.html#rnncell)
对于如下代码：

```
rnncell = nn.RNNCell(10, 20)
input = torch.randn(6, 3, 10)
hx = torch.randn(3, 20)
output = []
for i in range(6):
    hx = rnncell(input[i], hx)
    output.append(hx)
```

若 T 为输入数据序列长度，B 为输入数据批量大小，N 为输入数据 x 的维度，H 为隐含层节点个数或隐含层输出数据维度，如下说法哪个是正确的？

(A) T=3, B=6, N=20, H=10
(B) T=3, B=6, N=10, H=20
(C) T=6, B=3, N=20, H=10
(D) T=6, B=3, N=10, H=20

2. 查看 PyTorch 中 RNN 的说明 (https://pytorch.org/docs/stable/nn.html#rnncell)
对于如下代码：

```
rnn = nn.RNN(10, 20, 2)
input = torch.randn(6, 3, 10)
h0 = torch.randn(2, 3, 20)
output, hn = rnn(input, h0)
```

与第一题中的代码进行比较，如下说法哪些是正确的？（本题为多选题）

(A) nn.RNN 初始化时 batch_first 默认值为 False，其前向过程中输入数据形状为 [6, 3, 10]，即输入数据序列长度 T=6，输入数据批量大小 B=3，输入数据 x 的维度 N=10。
(B) nn.RNN 初始化时 batch_first 默认值为 True，其前向过程中输入数据形状为[6, 3, 10]，即输入数据批量大小 B=6，输入数据序列长度 T=3，输入数据 x 的维度 N=10。
(C) 第一题中 RNNCell 给出了一个循环神经网络隐含层的实现，本题循环神经网络隐含层为 3 层。
(D) 第一题中 RNNCell 给出了一个循环神经网络隐含层的实现，本题循环神经网络隐含层为 2 层。

3. 循环神经网络的前向过程计算为：

$$z_t^{(l)} = W_{ih}^{(l)} h_t^{(l-1)} + b_{ih}^{(l)} + W_{hh}^{(l)} h_{t-1}^{(l)} + b_{hh}^{(l)}$$
$$h_t^{(l)} = f\left(z_t^{(l)}\right)$$

其中，当$l = 1$时，$h_t^{(l-1)} = x_t$。

在误差沿时间反向传播的过程中，如下计算公式哪些是正确的？（本题为多选题）

(A) $\frac{\partial L}{\partial h_t^{(l)}} = \frac{\partial L}{\partial h_t^{(l+1)}} \cdot \frac{\partial h_t^{(l+1)}}{\partial h_t^{(l)}} + \frac{\partial L}{\partial h_{t+1}^{(l)}} \cdot \frac{\partial h_{t+1}^{(l)}}{\partial h_t^{(l)}}$

(B) $\frac{\partial L}{\partial z_t^{(l)}} = \frac{\partial L}{\partial h_t^{(l)}} \cdot \frac{\partial h_t^{(l)}}{\partial z_t^{(l)}}$

(C) $\frac{\partial L}{\partial W_{ih}^{(l)}} = \frac{\partial L}{\partial z_t^{(l)}} \cdot h_t^{(l-1)}$, $\frac{\partial L}{\partial W_{hh}^{(l)}} = \frac{\partial L}{\partial z_t^{(l)}} \cdot h_{t-1}^{(l)}$

(D) $\frac{\partial L}{\partial W_{ih}^{(l)}} = \sum_{t=1}^{T} \frac{\partial L}{\partial z_t^{(l)}} \cdot h_t^{(l-1)}$, $\frac{\partial L}{\partial W_{hh}^{(l)}} = \sum_{t=1}^{T} \frac{\partial L}{\partial z_t^{(l)}} \cdot h_{t-1}^{(l)}$

4. 对于循环神经网络的编程实现，需要注意数据的表示方法，比如，x_t和h_t的维度为$[B, N]$，其中 B 为 Batch Size, N 为数据各自的维度，对于x_t，N 为 input_size，对于h_t，N 为 hidden_size。对于一个 batch，x_t和h_t为**行向量**。循环神经网络的前向计算过程原为：

$$h_t = \tanh(W_{ih}x_t + b_{ih} + W_{hh}h_{t-1} + b_{hh})$$

采用x_t和h_t行向量表示后，如下计算公式哪个是正确的？

(A) $h_t = \tanh(W_{hh}x_t + b_{ih} + W_{ih}h_{t-1} + b_{hh})$

(B) $h_t = \tanh(x_t W_{hh}^T + b_{ih} + h_{t-1}W_{ih}^T + b_{hh})$

(C) $h_t = \tanh(W_{ih}x_t + b_{ih} + W_{hh}h_{t-1} + b_{hh})$

(D) $h_t = \tanh(x_t W_{ih}^T + b_{ih} + h_{t-1}W_{hh}^T + b_{hh})$

5. 对于本次编程作业所采用的门控循环单元为：

$$z_t = \sigma(W_{zh}h_{t-1} + W_{zx}x_t)$$
$$r_t = \sigma(W_{rh}h_{t-1} + W_{rx}x_t)$$
$$\tilde{h}_t = tanh(W_h(r_t \otimes h_{t-1}) + W_x x_t)$$
$$h_t = (1 - z_t) \otimes h_{t-1} + z_t \otimes \tilde{h}_t$$

其中，$\otimes$表示按元素相乘，用于门控机制的实现。

在误差沿时间反向传播的过程中，如下计算公式哪些是正确的？（本题为多选题）

(A) $\frac{\partial L}{\partial z_t} = \frac{\partial L}{\partial h_t} \otimes (\tilde{h}_t - h_{t-1})$

(B) $\frac{\partial L}{\partial \tilde{h}_t} = \frac{\partial L}{\partial h_t} \cdot z_t$

(C) $\frac{\partial L}{\partial W_h} = \frac{\partial L}{\partial \tilde{h}_t} \cdot tanh' \cdot r_t \otimes h_{t-1}$

(D) $\frac{\partial L}{\partial h_{t-1}} = \frac{\partial L}{\partial h_t} \cdot \frac{\partial h_t}{\partial h_{t-1}} + \frac{\partial L}{\partial \tilde{h}_t} \cdot \frac{\partial \tilde{h}_t}{\partial h_{t-1}} + \frac{\partial L}{\partial r_t} \cdot \frac{\partial r_t}{\partial h_{t-1}} + \frac{\partial L}{\partial z_t} \cdot \frac{\partial z_t}{\partial h_{t-1}}$

(E) $\frac{\partial L}{\partial x_t} = \frac{\partial L}{\partial \tilde{h}_t} \cdot \frac{\partial \tilde{h}_t}{\partial x_t} + \frac{\partial L}{\partial r_t} \cdot \frac{\partial r_t}{\partial x_t} + \frac{\partial L}{\partial z_t} \cdot \frac{\partial z_t}{\partial x_t}$

任务二、补全程序代码

程序清单如下：

目录	说明	注意事项
\program\hw3	rnn_classifier.py 利用 RNN 进行音素分类的示意代码 hw3.py 利用 GRU RNN 进行文本字符预测	rnn_classifier.py 和 hw3.py 需要补全代码
\program\mytorch	rnn_cell.py 循环神经网络单元 gru_cell.py 门控循环单元	rnn_cell.py 和 gru_cell.py 需要补全代码
	activation.py 激活函数 linear.py 线性层 loss.py 目标函数	不需要修改
\program\autograder\hw1_autograde	runner.py 自动评判编程作业的主控程序	自动评判程序 runner.py 利用存储的标准结果以及 PyTorch 对应程序，与同学们完成的模型得到的结果进行比较，给出各步骤通过与否的判断，此环节分值为 80 分。 请注意需要配置好 PyTorch 运行环境。

另外， \program\autograder\hw3_autograder 中的 \data 子目录为存储的自动评判测试标准结果。

请在程序“# ToDo:”及“???”提示处补全代码；补全代码后，可将原有的“raise NotImplemented”语句删除或注释掉。需要补全代码的清单如下：

步骤 (Step)	内容	程序	初始程序中“# ToDo: ”所在行数	说明
I	RNN 单元： 前向计算 误差反向传播	\mytorch\rnn_cell.py	61 96	循环神经网络原理可参见： https://tangshusen.me/Dive-into-DL-PyTorch/#/ 6. 循环神经网络
	基于 RNN 的音素分类	\hw3\rnn_classifier.py	15 77 123	RNN 音素分类任务示意代码，RNN 最后一层最后时刻的输出用于分类
II	GRU	\hw3\ gru_cell.py	58 91	门控循环单元原理可参见： https://tangshusen.me/Dive-into-DL-PyTorch/#/ 6.7.1 门控循环单元
	基于 GRU 的循环神经网络	\hw3\hw3.py	15 34 60	基于 GRU 的循环神经网络的语言模型示意，预测文本中下一个字符

任务三、自动评判程序实现效果

Windows 平台	在 Windows 搜索框中搜索并打开 Anaconda Prompt。若需安装软件，可用鼠标右键点击“Anaconda Prompt”应用图标，选“以管理员身份运行”。 用 cd 命令进入作业所附程序解压生成的子目录 program。若要换盘符，需用 “<盘符>:”命令，比如从 c 盘换至 d 盘，键入 d: 和回车即可。 运行 python .\autograder\hw2_autograder\runner.py
Linux 平台 Mac 平台	在命令行终端中，用 cd 命令进入作业所附程序解压生成的子目录 program。 运行 python ./autograder/hw2_autograder/runner.py

若程序运行无误，将输出如图 3 所示的信息。

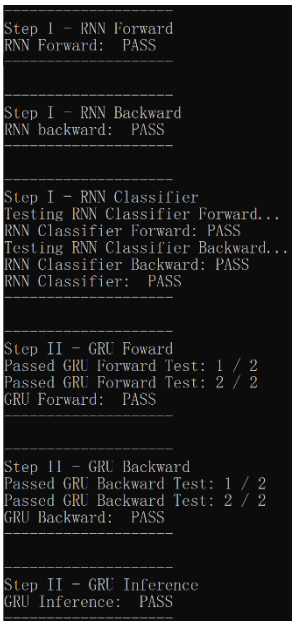


图 3. 自动评判程序预期输出信息

任务四、提交作业

在 Windows 平台上利用 WinZip 或 WinRAR 等软件将 \hw3 和 \mytorch 子目录打包在一个文件中，在 Linux 平台和 Mac 平台上可在终端中运行 sh create_tarball.sh 命令打包修改后的程序，将程序打包文件和作业报告一同提交到网络学堂。 作业报告中包括选择题答案，任务三运行结果，本次作业遇到的问题及解决方法，对本次作业的意见及建议。

本次作业责任助教为闫睿劼（Email: yrj17@mails.tsinghua.edu.cn）。

【致谢】本次编程作业原始资料由 CMU 11-785 Deep Learning 课程 http://deeplearning.cs.cmu.edu/ 主讲教师 Bhiksha Raj 教授提供。程序源代码由 6 字班彭逸凡同学协助整理，闫睿劼助教测试。

【关于 Python 编程】

1. Python 中*号运算符用于函数参数传递的打包和拆解说明：
https://www.geeksforgeeks.org/packing-and-unpacking-arguments-in-python/
用法示例：hw3\rnn_classifier.py(41): rnn_cell.init_weights(*rnn_weights[i])

2. Python 数组切片索引相关内容请参见：
https://github.com/lijin-THU/notes-python/blob/master/02-python-essentials/02.05-indexing-and-slicing.ipynb
负索引值即从后向前开始计数，例如，索引 -1 表示倒数第 1 个元素。用法示例：
\hw3\rnn_classifier.py(97): logits = self.output_layer(self.hiddens[-1][-1])

【关于作业迟交的说明】 请同学们争取按时提交作业，迟交会酌情扣分。