

武汉纺织大学计算机与人工智能学院

深度学习基础

6. 深度学习用于序列处理

吴晓堃

xkun.wu at gmail dot com

2021/05/17

Outline

本章内容

处理文本数据。理解循环神经网络。循环神经网络的高级用法。用卷积神经网络处理序列。

重点：使用预训练的词嵌入、使用 LSTM 层和 GRU 层、使用一维卷积神经网络；

难点：分析不同循环神经网络的适用条件、提高循环神经网络的性能和泛化能力。

3

学习目标

4 / 13

- 理解并掌握处理文本数据的两种主要方法：one-hot 编码、词嵌入；
- 理解并掌握简单循环神经网络（RNN）、LSTM 层和 GRU 层的工作原理；
- 理解并掌握提高循环神经网络的性能和泛化能力的三种高级技巧：循环 dropout 降低过拟合、堆叠循环层提高网络的表示能力、双向循环层提高精度并缓解遗忘问题；
- 理解并掌握一维卷积神经网络的使用方法。

4

文本向量化

标记 (token)

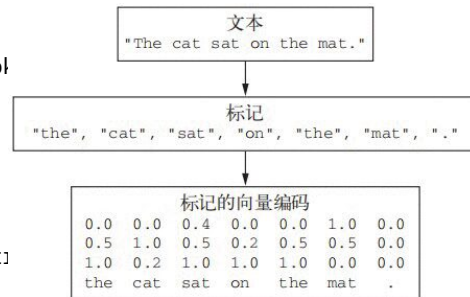
将文本分解而成的单元 (单词、字符或 n-gram)。

```
from keras.preprocessing.text import Tokenizer

samples = ['...', '...', ...]

tokenizer = Tokenizer(num_words=1000)
tokenizer.fit_on_texts(samples)

one_hot_encode = tokenizer.texts_to_matrix(
    samples, mode='binary')
```



5

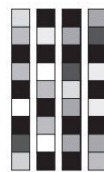
标记的向量编码

两种主要方法

- one-hot 编码：高维稀疏表示，即散列；硬编码得到；
- 标记嵌入：低维密集表示，从数据中学习得到。



one-hot词向量：
- 稀疏
- 高维
- 硬编码



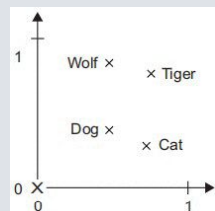
词嵌入：
- 密集
- 低维
- 从数据中学习得到

6

学习词嵌入

几何关系表示语义关系

- 距离：与语义关系远近正相关；
- 方向：同类别在相近方向上聚集。



从数据中学习词嵌入

```
from keras.layers import Embedding
model.add(Embedding(1000, 64, input_length=1))
```

使用预训练的词嵌入

```
model.layers[0].set_weights([embedding_weights])
model.layers[0].trainable = False
```

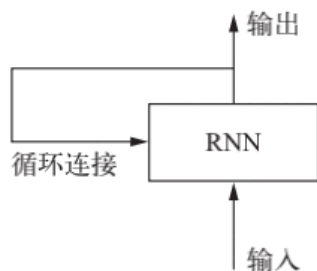
7

循环网络 (RNN)

前馈网络 (feedforward network) 的问题：没有记忆。

具有内部环的网络架构

- 循环结构：遍历所有序列元素；
- 存储状态：包含与已查看内容相关的信息。

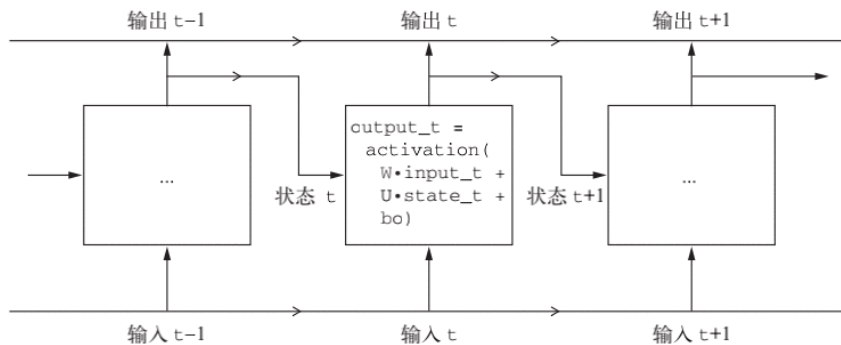


8

RNN 的简单表述：循环展开

时间步函数

```
output_t = np.tanh(np.dot(W, input_t) + np.dot(U, state_t) + b)
```



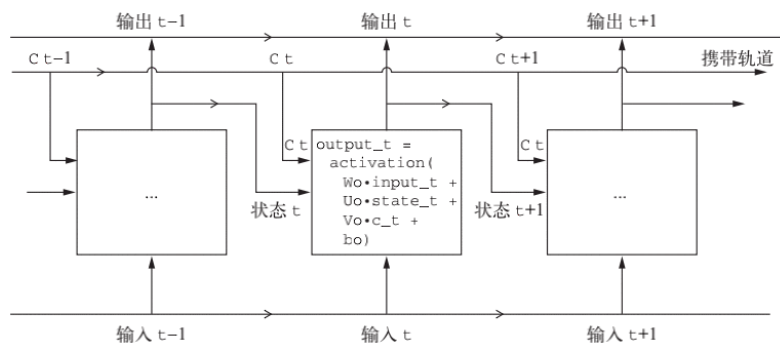
9

长短期记忆 (LSTM)

10 / 13

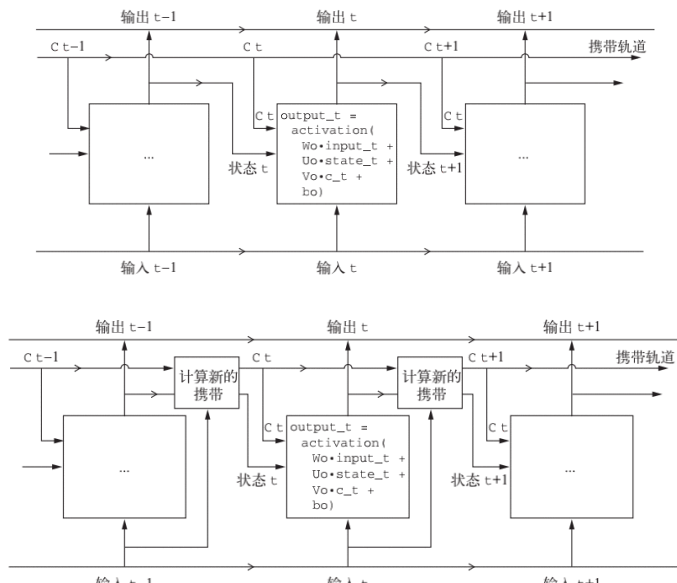
长期记忆

- 解决遗忘问题：深层网络会出现梯度消失 (vanishing gradient) 现象；
- 长期携带信息：跨越多个时间步。



10

LSTM: 计算新的携带信息



11

向量序列的二分类: LSTM 示例

```
from keras import models, layers

model = models.Sequential()
model.add(layers.LSTM(32, return_sequences=True, input_shape=(num_timesteps, num_
model.add(layers.LSTM(32, return_sequences=True))
model.add(layers.LSTM(32))
model.add(layers.Dense(num_classes, activation='sigmoid'))

model.compile(optimizer='rmsprop', loss='binary_crossentropy')
```

