

Day 67 Keras Dataset

Keras embedded dataset 的介紹與應用



出題教練

陳宇春



本日知識點目標

- 了解 Keras 內建的 dataset
- 如何使用 CIFAR10 做類別預測

Keras 自帶數據集與模型(一)

- Keras自帶的數據集
 - CIFAR10小圖像分類
 - CIFAR100小圖像分類
 - IMDB電影評論情緒分類
 - 路透社newswire話題分類
 - 手寫數字的MNIST數據庫
 - 時尚文章的時尚MNIST數據庫
 - 波士頓房屋價格回歸數據集

Keras 自帶數據集與模型 (二)

A. 【關於資料夾】

- 這裡Keras是在Windows環境，使用Anaconda安裝
- Anaconda有兩個主要資料夾需要瞭解：
 - Anaconda 應用程式安裝目錄下的Keras子資料夾，需要搜索找到
 - Anaconda 應用程式存儲Keras模型和資料集檔的檔在 ,用對應的用戶資料夾下的.kears資料夾下，注意有個“.”, 實在找不見可以搜索
- 【資料集】：下載後預設存儲目錄 C:\Users\Administrator\.keras\datasets下的同名檔，注意有個點 “.keras“

B. 執行下載時，要import相應的模組，利用資料集模組提供的函數下載資料，模組檔結構如下圖所示：在Anaconda安裝資料夾下搜索keras即可找到此目錄

Keras 自帶數據集與模型 (三)

在程式中需要下載對應的資料集時，首先導入對應的模組，然後調用.load_data()函數

#從Keras導入相應的模組

```
from keras.datasets import cifar10
```

#從網路即時下載

```
(x_train, y_train), (x_validate, y_validate) = cifar10.load_data()
```

Keras 自帶數據集– CIFAR10

- CIFAR10小圖像分類
- 數據集50,000張32x32彩色訓練圖像，標註超過10個類別，10,000張測試圖像

Usage:

```
from keras.datasets import cifar10  
  
(x_train, y_train), (x_test, y_test) = cifar10.load_data()
```

Keras 自帶數據集– CIFAR100

- CIFAR100小圖像分類
- 數據集50,000張32x32彩色訓練圖像，標註超過100個類別，10,000張測試圖像。

Usage:

```
from keras.datasets import cifar100  
  
(x_train, y_train), (x_test, y_test) = cifar100.load_data(label_mode='fine')
```

Keras 自帶數據集– MNIST database

- 手寫數字的MNIST數據庫
- 數據集包 含10個數字的60,000個28x28灰度圖像，以及10,000個圖像的測試集。

Usage:

```
from keras.datasets import mnist  
(x_train, y_train), (x_test, y_test) = mnsit.load_data()
```

路徑：如果你本地沒有索引文件（at '~/.keras/datasets/' + path），它將被下載到這個位置。

Keras 自帶數據集-時尚文章的時尚MNIST數據庫



- 時尚文章的時尚MNIST數據庫
- 數據集包含10個時尚類別的60,000個28x28灰度圖像，以及10,000個圖像的測試集。
這個數據集可以用作MNIST的直接替換。

Usage:

```
from keras.datasets import fashion_mnsit  
  
(x_train, y_train), (x_test, y_test) = fashion_mnsit.load_data()
```

class labels

Label	Description
0	T-shirt/top
1	Trouser
2	Pullover
3	Dress
4	Coat
5	Sandal
6	Shirt
7	Sneaker
8	Bag
9	Ankle boot

Keras 自帶數據集-波士頓房屋價格回歸

- 數據集取自卡內基梅隆大學維護的StatLib庫。
- 20世紀70年代後期，樣本在波士頓郊區的不同位置包含13個房屋屬性。目標是一個地點房屋的中位值（單位：k \$）。

Usage:

```
from keras.datasets import boston_housing  
  
(x_train, y_train), (x_test, y_test) = boston_housing.load_data()
```

Keras Dataset – IMDB電影評論情緒分類

- 來自 IMDB 的 25,000 部電影評論的數據集，標有情緒（正面/負面）。評論已經過預處理，每個評論都被編碼為一系列單詞索引（整數）。
- 單詞由數據集中的整體頻率索引
 - 整數“3”編碼數據中第 3 個最頻繁的單詞。
 - “0”不代表特定單詞，而是用於編碼任何未知單詞

```
from keras.datasets import imdb
```

```
(x_train, y_train), (x_test, y_test) = imdb.load_data(path="imdb.npz", num_words=  
None, skip_top=0, maxlen=None, seed=113, start_char=1, oov_char=2, index_from=3)
```

Keras Dataset – IMDB電影評論情緒分類

- path：如果您沒有本地數據（at '~/.keras/datasets/' + path），它將被下載到此位置。
- num_words：整數或無。最常見的詞彙需要考慮。任何不太頻繁的單詞將oov_char在序列數據中顯示為值。
- skip_top：整數。最常被忽略的詞（它們將 oov_char 在序列數據中顯示為值）。
- maxlen：int。最大序列長度。任何更長的序列都將被截斷。
- 種子：int。用於可重複數據改組的種子。
- start_char：int。序列的開頭將標有此字符。設置為 1，因為 0 通常是填充字符。
- oov_char：int。這是因為切出字 num_words 或 skip_top 限制將這個字符替換。
- index_from：int。使用此索引和更高的索引實際單詞。

Keras Dataset –路透社新聞專題主題分類

- 來自路透社的 11,228 條新聞專線的數據集，標註了 46 個主題。與 IMDB 數據集一樣，每條線都被編碼為一系列字索引

```
from keras.datasets import reuters
```

```
(x_train, y_train), (x_test, y_test) = reuters.load_data(path="reuters npz", num_words=
None, skip_top=0, maxlen=None,
test_split=0.2, seed=113, start_char=1, oov_char=2, index_from=3)
```

前述流程 / python程式 對照

資料準備

```
In [2]: (x_img_train,y_label_train), \  
        (x_img_test, y_label_test)=cifar10.load_data()
```

```
In [3]: print('train:',len(x_img_train))  
        print('test :',len(x_img_test))
```

```
train: 50000  
test : 10000
```

```
In [4]: x_img_train.shape
```

```
Out[4]: (50000, 32, 32, 3)
```

```
In [5]: y_label_train.shape
```

```
Out[5]: (50000, 1)
```

前述流程 / python程式 對照



Image normalize

```
In [13]: x_img_train[0][0][0]
```

```
Out[13]: array([59, 62, 63], dtype=uint8)
```

```
In [14]: x_img_train_normalize = x_img_train.astype('float32') / 255.0  
x_img_test_normalize = x_img_test.astype('float32') / 255.0
```

```
In [15]: x_img_train_normalize[0][0][0]
```

```
Out[15]: array([ 0.23137255,  0.24313726,  0.24705882], dtype=float32)
```

轉換label 為OneHot Encoding

```
In [16]: y_label_train.shape
```

```
Out[16]: (50000, 1)
```

```
In [17]: y_label_train[:5]
```

```
Out[17]: array([[6],  
                [9],  
                [9],  
                [4],  
                [1]], dtype=uint8)
```

```
In [18]: from keras.utils import np_utils  
y_label_train_OneHot = np_utils.to_categorical(y_label_train)  
y_label_test_OneHot = np_utils.to_categorical(y_label_test)
```

```
In [19]: y_label_train_OneHot.shape
```

```
Out[19]: (50000, 10)
```

```
In [20]: y_label_train_OneHot[:5]
```

重點複習：如何使用Keras 自帶數據集做目標學習



- 適用於文本分析與情緒分類
 - IMDB 電影評論情緒分類
 - 路透社新聞專題主題分類
- 適用於 Data/Numerical 學習
 - Boston housing price regression dataset
- 適用於影像分類與識別學習
 - CIFAR10/CIFAR100
 - MNIST/ Fashion-MNIST
- 針對小數據集的深度學習
 - 數據預處理與數據提升

解題時間 It's Your Turn

請跳出PDF至官網Sample Code & 作業
開始解題

