自動駕駛實務

Project

Traffic Sign Classifier

姓名 : 謝秉宏

學號 : E24094164

目錄

1. 資料集探索
2. 模型訓練環境
3. 模型架構
4. Optimizer選擇
5. 資料前處理
6. 測試和評估
7. 模型優化
8. 結論
9. 資料集探索
   1. 資料集介紹

本次專案所使用的資料集是German Traffic Sign Dataset(德國交通號誌資料集)，包含德國交通號誌的圖片和分類，其用途是用於訓練和測試交通標誌識別系統。這個資料集是由德國卡爾斯魯厄理工學院的計算機視覺小組建立的，旨在提供一個標準的基準來評估交通標誌識別算法的性能。該資料集包含超過50,000張來自不同角度、照明條件和天氣條件下的交通標誌圖像，如下圖所示，分為43個類別，每個類別代表一種不同的交通標誌。

一張含有 文字, 螢幕擷取畫面, 交通標誌, 字型 的圖片

自動產生的描述

* 1. 資料集內容

該資料集的圖片大小為32 pixels \* 32 pixels，並且將資料集分割成訓練集、驗證集和測試集。以下將資料集中每一個分類的數量以直方圖顯示出來:

* + 1. 訓練集 (34799筆資料)

一張含有 文字, 螢幕擷取畫面, 圖表, 平行 的圖片

自動產生的描述

* + 1. 驗證集 (4410筆資料)

一張含有 文字, 螢幕擷取畫面, 圖表, 平行 的圖片

自動產生的描述

* + 1. 測試集 (12630筆資料)

一張含有 文字, 螢幕擷取畫面, 圖表, 平行 的圖片

自動產生的描述

從訓練集、驗證集和測試集的各類別資料直方圖可以看出來每一個資料集的類別比例基本相同，但是從直方圖可以看出資料的分佈並不是很平均，可能會導致訓練上的一些問題。

1. 模型訓練環境

|  |  |
| --- | --- |
| **Package** | **Version** |
| python | 3.7.1 |
| tensorflow | 2.5.0 |
| keras | 2.4.3 |
| numpy | 1.21.5 |
| pandas | 1.3.5 |
| scikit-learn | 1.0.2 |
| seaborn | 0.12.2 |

1. 模型架構

本次專案使用了LeNet-5的卷積神經網路(CNN)架構，將其修改為適合本次專案的需求。模型架構如下:

一張含有 文字, 螢幕擷取畫面, 字型, 收據 的圖片

自動產生的描述

|  |  |
| --- | --- |
| Total params | 64,811 |
| Trainable params | 64,811 |
| Non-trainable params | 0 |

1. Optimizer選擇

在開始訓練模型時，先測試要選哪一種optimizer，模型訓練的速度快效果不錯，在tensorflow 2.5中提供了SGD、Adam、RMSprop、Adadelta等等optimizer。我選用了其中三種來比較，分別是SGD、Adam和RMSprop。

我直接使用原始資料，不做任何資料前處理和資料增強的手法，learning rate設為0.0001，訓練50個epoch，其中SGD額外設置參數momentum=0.9，加快初始的訓練速度，觀察其loss和accuracy的變化，結果如下所示:

|  |  |  |
| --- | --- | --- |
| **Optimizer** | **Loss Curve** | **Accuracy Curve** |
| SGD |  |  |
| Adam |  |  |
| RMSprop |  |  |

同時使用scikit-learn的classification\_report功能，計算分類測試集的precision、recall和f1-score結果，在下表呈現:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **Avg Precision** | | **Avg Recall** | | **Avg F1-score** | |
| **macro** | **weighted** | **macro** | **weighted** | **macro** | **weighted** |
| SGD | 0.01 | 0.03 | 0.03 | 0.06 | 0.01 | 0.03 |
| Adam | 0.80 | 0.85 | 0.78 | 0.85 | 0.79 | 0.85 |
| RMSprop | **0.89** | **0.91** | **0.86** | **0.91** | **0.87** | **0.90** |

從上面的loss curve和accuracy curve結果可以很明顯的看出來，即使加上momentum參數來提升模型收斂速度，SGD的收斂速度仍然遠低於其他兩種optimizer。而使用Adam和RMSprop作為optimizer來訓練模型，在50個epoch後，模型大致上都有收斂，只是RMSprop的速度更快，且單用原始資料做訓練，不經過資料前處理和資料增強，F1-score就可以達到0.90，略優於使用Adam的結果。因此選用RMSprop作為之後訓練使用的optimizer。

1. 資料前處理
2. 測試和評估
3. 模型優化
4. 結論