

第5回データ解析ゼミ

tensorflow2系を用いたCNN の実装

はじめに

Mac book proじゃメモリがたりねえ！！

データセットについて



データセット

メトロポリタン美術館に
ある日本の浮世絵の作者
別にされたデータセット

The Metropolitan
Museum of Art Ukiyoe
Dataset

を使おうとしたけど...

ダメだった

- データが整形されておらず、サイズがまちまち
- 全部で10GBのデータがぶん投げられる



それでどうしたか

- 同じデータを用いたコンペティションを発見
- $224 * 224$ に変換済み

Ukiyoe author
prediction competition
held by ProbSpace

実装

実装(1)

- とりあえず、八谷先生のTensorflow2系でのCNN実装例を参考にSubclassingで実装

構成(1)

- Conv3層(32 -> 64 -> 64)
- 全結合層1層(64, relu)
- 出力層(32, softmax)
- Batch Size: 32
- Epochs: 10

結果(1)

- accuracy: 25.96%
- loss: 2.77

```
Epoch-Batch: 10-1, Time: 1.16s, Loss: 3.1215033531188965, Accuracy: 29.1503849029541
Epoch-Batch: 10-11, Time: 1.39s, Loss: 3.111497163772583, Accuracy: 29.166486740112305
Epoch-Batch: 10-21, Time: 1.38s, Loss: 3.1010477542877197, Accuracy: 29.21631240844726
Epoch-Batch: 10-31, Time: 1.27s, Loss: 3.091341495513916, Accuracy: 29.20159912109375
Epoch-Batch: 10-41, Time: 1.23s, Loss: 3.080164670944214, Accuracy: 29.262107849121094
Epoch-Batch: 10-51, Time: 1.30s, Loss: 3.0700881481170654, Accuracy: 29.29640960693359
Epoch-Batch: 10-61, Time: 1.25s, Loss: 3.060030937194824, Accuracy: 29.366270065307617
Epoch-Batch: 10-71, Time: 1.31s, Loss: 3.0501413345336914, Accuracy: 29.38636779785156
---Test Result---
Epoch: 10, Loss: 2.76566219329834, Accuracy: 25.95879364013672
```

実装(2)

- ぼかしてみることで特徴がわかるんじゃないかな

構成(2)

- 構成(1)と同じ
- ガウシアンフィルタをかける

結果(2)

- accuracy: 29.48%
- loss: 3.72

```
Epoch-Batch: 10-1, Time: 1.44s, Loss: 3.5203919410705566, Accuracy: 35.2886962890625
Epoch-Batch: 10-11, Time: 1.36s, Loss: 3.5050110816955566, Accuracy: 35.31500244140625
Epoch-Batch: 10-21, Time: 1.38s, Loss: 3.488457441329956, Accuracy: 35.3619499206543
Epoch-Batch: 10-31, Time: 1.34s, Loss: 3.474116086959839, Accuracy: 35.44132995605469
Epoch-Batch: 10-41, Time: 1.29s, Loss: 3.4608144760131836, Accuracy: 35.46040344238281
Epoch-Batch: 10-51, Time: 1.37s, Loss: 3.447287082672119, Accuracy: 35.45846176147461
Epoch-Batch: 10-61, Time: 1.27s, Loss: 3.4318783283233643, Accuracy: 35.55375671386719
Epoch-Batch: 10-71, Time: 1.33s, Loss: 3.4172914028167725, Accuracy: 35.63462066650390
---Test Result---
Epoch: 10, Loss: 3.7197721004486084, Accuracy: 29.477020263671875
```

実装(3...k)

- mnistは $28 * 28$ だったけど、今回のは $224 * 224$ だから、Conv層の出力の数を増やすべきでは
- ついでにEpochs毎にランダムなBrightness入れよう

構成(3...k)

- Conv3~6層(色々)
- 他は構成(2)と同じ
- Epochs毎にランダムなBrightness

結果(3...k)

- test実行時にメモリが吹っ飛ぶ
- Epoch1までしか学習できず
- ただ、Accuracyの調子はEpoch1にしては良さそう

```
Epoch-Batch: 1-10, Time: 312.22s, Loss: 703.96240234375, Accuracy: 20.9375
Epoch-Batch: 1-20, Time: 381.05s, Loss: 431.6622314453125, Accuracy: 25.78125
Epoch-Batch: 1-30, Time: 332.54s, Loss: 314.6091613769531, Accuracy: 26.5625
Epoch-Batch: 1-40, Time: 350.42s, Loss: 249.15550231933594, Accuracy: 28.4375019073486
Epoch-Batch: 1-50, Time: 323.96s, Loss: 206.25259399414062, Accuracy: 29.8750019073486
Epoch-Batch: 1-60, Time: 343.19s, Loss: 175.91880798339844, Accuracy: 30.0520839691162
Epoch-Batch: 1-70, Time: 350.23s, Loss: 152.00823974609375, Accuracy: 29.3749980926513
```

原因(3...k)

- 計算量が大きくなりすぎ
- メモリを監視していると
 - 学習時でも90%近く
 - testが走ると100%を越えメモリエラーを吐く
 - テストケースを総データ数の2割確保
 - ミニバッチで実行してなかった

実装(k...)

- テストケースをBatch Sizeと同量に
- ついでに全結合層増やそう
- Conv層は前より控えめにしよう

構成(3...k)

- Conv3層(64 -> 128 -> 256)
- 全結合層3層(256 -> 512 -> 256)
- 他は構成(3...k)と同じ

結果(3...k)

- train_accuracyに対してtest_accuracyが悪い
- lossの値おかしい？

```
Epoch-Batch: 1-10, Time: 312.22s, Loss: 703.96240234375, Accuracy: 20.9375
Epoch-Batch: 1-20, Time: 381.05s, Loss: 431.6622314453125, Accuracy: 25.78125
Epoch-Batch: 1-30, Time: 332.54s, Loss: 314.6091613769531, Accuracy: 26.5625
Epoch-Batch: 1-40, Time: 350.42s, Loss: 249.15550231933594, Accuracy: 28.4375019073486
Epoch-Batch: 1-50, Time: 323.96s, Loss: 206.25259399414062, Accuracy: 29.8750019073486
Epoch-Batch: 1-60, Time: 343.19s, Loss: 175.91880798339844, Accuracy: 30.0520839691162
Epoch-Batch: 1-70, Time: 350.23s, Loss: 152.00823974609375, Accuracy: 29.3749980926513
```