

令和7年度 深層学習

MNIST の学習における
全結合層と畳み込み層の違いが
学習時間と精度に与える影響の調査

令和7年5月14日

大阪国際工科専門職大学
工科学部
情報工学科
AI 開発コース

OK240100 藤村勇仁

目次

1	概要	1
2	畳み込み層の有無による学習時間と精度の比較	1
2.1	実行結果	1
2.2	考察	2
3	全結合層のユニット数を増やすことでモデルのサイズと精度の関係を調べる	2
3.1	実行結果	3
3.2	考察	3

1 概要

本課題では、MNIST の学習において畳み込み層の有無が学習時間と精度にどのような影響を与えるかを調べる。また、精度を維持したままモデルのサイズを小さくする方法を調査する。具体的には、全結合層のみのモデルと畳み込み層を含むモデルを比較し、学習時間と精度の関係を調べる。さらに、全結合層のみのモデルにおいて、精度を向上させるために全結合層のユニット数を増やすことでモデルのサイズと精度の関係を調べる。

2 畳み込み層の有無による学習時間と精度の比較

MNIST データセットを用いて、畳み込み層を含むモデルと全結合層のみのモデルを比較する。畳み込み層を含むモデルは、LMS にアップロードされている mnist.py を途中結果の評価である、`if batch_idx % 16 == 0:` 内を削除して使用した。全結合層のみのモデルは、畳み込み層を含むモデルの計測に使用したコードの MyModel クラスを以下のように変更した。

```
1 class MyModel(nn.Module):
2     def __init__(self):
3         super(MyModel, self).__init__()
4         self.flatten = nn.Flatten()
5         self.fc1 = nn.Linear(1 * 28 * 28, 128) # 全結合層
6         self.fc2 = nn.Linear(128, 10) # 全結合層
7
8     def forward(self, xx):
9         xx = self.flatten(xx) # 1x28x28->784
10        xx = self.fc1(xx) # 784->128
11        xx = nn.ReLU()(xx)
12        xx = self.fc2(xx)
13        return xx
```

2.1 実行結果

畳み込み層を含むモデルの実行結果は以下ようになった。

```
1 =====
2 Layer (type:depth-idx) Output Shape Param #
3 =====
4 MyModel [1, 10] --
5 |---Conv2d: 1-1 [1, 4, 26, 26] 40
6 |---Conv2d: 1-2 [1, 4, 24, 24] 148
7 |---Flatten: 1-3 [1, 2304] --
8 |---Linear: 1-4 [1, 128] 295,040
9 |---Linear: 1-5 [1, 10] 1,290
10 =====
11 Total params: 296,518
12 Trainable params: 296,518
13 Non-trainable params: 0
14 Total mult-adds (Units.MEGABYTES): 0.41
15 =====
16 Input size (MB): 0.00
17 Forward/backward pass size (MB): 0.04
18 Params size (MB): 1.19
19 Estimated Total Size (MB): 1.23
```

```

20 =====
21 Epoch: 1, Loss: 0.113473, Time: 6.84 sec
22 Accuracy on the test dataset: 96.03%

```

全結合層のみのモデルの実行結果は以下のようになった。

```

1 =====
2 Layer (type:depth-idx) Output Shape Param #
3 =====
4 MyModel [1, 10] --
5 |---Flatten: 1-1 [1, 784] --
6 |---Linear: 1-2 [1, 128] 100,480
7 |---Linear: 1-3 [1, 10] 1,290
8 =====
9 Total params: 101,770
10 Trainable params: 101,770
11 Non-trainable params: 0
12 Total mult-adds (Units.MEGABYTES): 0.10
13 =====
14 Input size (MB): 0.00
15 Forward/backward pass size (MB): 0.00
16 Params size (MB): 0.41
17 Estimated Total Size (MB): 0.41
18 =====
19 Epoch: 1, Loss: 0.153714, Time: 5.87 sec
20 Accuracy on the test dataset: 95.38%

```

2.2 考察

実行結果より、畳み込み層を含むモデルのパラメータ数は 296,518 であり、全結合層のみのモデルのパラメータ数は 101,770 であることがわかる。これにより、全結合層のみのモデルは畳み込み層を含むモデルに比べてパラメータ数が約 3 倍少ないことがわかる。さらに、学習時間は畳み込み層を含むモデルが約 6.84 秒、全結合層のみのモデルが約 5.87 秒であり、全結合層のみのモデルの方が約 1.2 倍早いことがわかる。また、精度は畳み込み層を含むモデルが 96.03%、全結合層のみのモデルが 95.38% であり、畳み込み層を含むモデルの方が約 0.65% 高いことがわかる。以上の結果から、全結合層のみのモデルは畳み込み層を含むモデルに比べてパラメータ数が約 3 倍少なく、学習時間が約 1.2 倍早く、精度が約 0.65% 低いことがわかる。ただし、この実行結果は誤差が大きく、学習時間や精度は実行環境や初期値に依存すると考えられるため、あくまで参考程度に留めておく必要がある。

3 全結合層のユニット数を増やすことでモデルのサイズと精度の関係を調べる

全結合層のみのモデルにおいて、全結合層のパラメータ数を増やすことでモデルのサイズと精度の関係を調べる。具体的には、1 つ目の全結合層の出力数を 64, 128, 192, 256, 320, 384, 448, 512, 576, 640, 704 に変更し、それぞれのモデルの学習時間と精度を比較する。

3.1 実行結果

各出力数に対するモデルのパラメータ数とモデルのサイズと学習時間と精度は以下のようになった。

1	出力数: 64, パラメータ数: 50,890, サイズ: 0.21MB, 学習時間: 6.05 sec, 精度: 94.02%
2	出力数: 128, パラメータ数: 101,770, サイズ: 0.41MB, 学習時間: 6.05 sec, 精度: 95.32%
3	出力数: 192, パラメータ数: 152,650, サイズ: 0.62MB, 学習時間: 6.05 sec, 精度: 95.86%
4	出力数: 256, パラメータ数: 203,530, サイズ: 0.82MB, 学習時間: 6.19 sec, 精度: 96.10%
5	出力数: 320, パラメータ数: 254,410, サイズ: 1.02MB, 学習時間: 6.33 sec, 精度: 95.91%
6	出力数: 384, パラメータ数: 305,290, サイズ: 1.23MB, 学習時間: 6.04 sec, 精度: 96.48%
7	出力数: 448, パラメータ数: 356,170, サイズ: 1.43MB, 学習時間: 5.93 sec, 精度: 96.53%
8	出力数: 512, パラメータ数: 407,050, サイズ: 1.64MB, 学習時間: 6.19 sec, 精度: 96.62%
9	出力数: 576, パラメータ数: 457,930, サイズ: 1.84MB, 学習時間: 6.04 sec, 精度: 96.70%
10	出力数: 640, パラメータ数: 508,810, サイズ: 2.04MB, 学習時間: 6.21 sec, 精度: 96.89%
11	出力数: 704, パラメータ数: 559,690, サイズ: 2.25MB, 学習時間: 6.05 sec, 精度: 96.36%

3.2 考察

実行結果より、サイズと精度の関係は1のようになった。

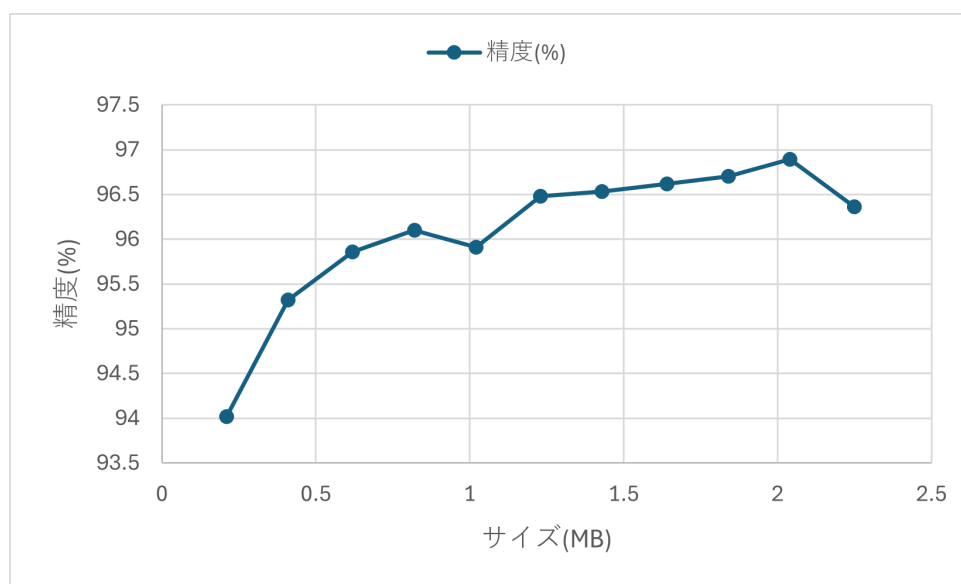


図1 サイズと精度の関係

この図より、サイズが大きくなるにつれて精度が向上することがわかる。ただし、サイズが大きくなるにつれて精度の向上は緩やかになっている。

また、畳み込み層を含むモデルのパラメータ数 296,518、サイズ 1.23MB、精度 96.03% と、それにサイズが近い全結合層のみのモデルの出力数 384 のモデルのパラメータ数 305,290、サイズ 1.23MB、精度 96.48% を比較すると、全結合層のみのモデルの出力数を調節したほうが畳み込み層を含むモデルよりも精度が高いことがわかる。

以上の結果から、今回のような単純なモデルにおいては、全結合層の出力数を増やす方が、畳み込み層を追加するよりも精度が高くなることがわかる。

今回は全結合層の出力数を調節するにとどめたが、全結合層の数を増やしたり、畳み込み層の数を増やしたりすることで、より精度を向上させることができると考えられる。