

目 录

图像识别.....	2
1 实验目的.....	2
2 数据集介绍.....	2
3 模型设计.....	2
4 实验结果与分析.....	3
4.1 模型评价指标.....	3
4.2 超参数调整与分析.....	5
5 结论与展望.....	9
5.1 实验结论.....	9
5.2 实验局限性与改进方向.....	9
参考文献.....	10
附录.....	10

图像识别

1 实验目的

本实验的主要目的是探索使用神经网络模型对 Fashion_MNIST 数据集进行图像识别任务的可行性和效果，并通过超参数调整来提高模型的性能。

2 数据集介绍

Fashion_MNIST 数据集是一个包含 10 个类别、共计 70000 张 28x28 像素灰度图像的数据集。每个类别包含 6000 个训练样本和 1000 个测试样本。该数据集被广泛用于图像分类任务的研究和评估。

3 模型设计

该模型包含三个卷积层和两个全连接层。在前向传播过程中，输入的数据先通过三个卷积层和池化层进行特征提取，然后被展平并输入到两个全连接层中，最后输出结果。

```
class NeuralNet(nn.Module):
    def __init__(self, num_classes=10):
        super(NeuralNet, self).__init__()

        #定义神经网络的层
        self.conv1 = nn.Conv2d(1, 32, 3, padding=1)
        self.conv2 = nn.Conv2d(32, 64, 3, padding=1)
        self.conv3 = nn.Conv2d(64, 128, 3, padding=1)
        self.pool = nn.MaxPool2d(2, 2)
        self.fc1 = nn.Linear(128 * 3 * 3, 512)
        self.fc2 = nn.Linear(512, num_classes)

    def forward(self, x):
        # 定义前向传播过程
        x = self.pool(F.relu(self.conv1(x)))
        x = self.pool(F.relu(self.conv2(x)))
        x = self.pool(F.relu(self.conv3(x)))
        x = x.view(-1, 128 * 3 * 3)
        x = F.relu(self.fc1(x))
        x = self.fc2(x)
        return x
```

图 1 定义神经网络模型

```
NeuralNet(
  (conv1): Conv2d(1, 32, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
  (conv2): Conv2d(32, 64, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
  (conv3): Conv2d(64, 128, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
  (pool): MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1, ceil_mode=False)
  (fc1): Linear(in_features=1152, out_features=512, bias=True)
  (fc2): Linear(in_features=512, out_features=10, bias=True)
)
```

图 2 神经网络模型结构

4 实验结果与分析

4.1 模型评价指标

分类报告

在分类问题中，分类报告是一种常用的性能评估方法，通常包含了准确率、召回率、F1 值和支持度等指标。以下是这些指标的详细说明：

准确率 (Accuracy): 准确率表示模型在所有样本中正确分类的比例。它是模型在整个数据集上的表现指标，可以帮助我们了解模型整体的分类能力。准确率的计算公式为：准确率 = $(TP+TN)/(TP+TN+FP+FN)$ ，其中 TP 表示真正例的数量，TN 表示真反例的数量，FP 表示假正例的数量，FN 表示假反例的数量。

召回率 (Recall): 召回率表示模型在所有正样本中正确分类的比例。它是模型在识别某个特定类别时的表现指标，可以帮助我们了解模型对于该类别的分类能力。召回率的计算公式为：召回率 = $TP/(TP+FN)$ ，其中 TP 表示真正例的数量，FN 表示假反例的数量。

精确率 (Precision): 精确率表示模型预测为正样本中实际为正样本的比例。它是模型在预测某个特定类别时的表现指标，可以帮助我们了解模型对于该类别的预测准确程度。精确率的计算公式为：精确率 = $TP/(TP+FP)$ ，其中 TP 表示真正例的数量，FP 表示假正例的数量。

F1 值 (F1 score): F1 值是精确率和召回率的加权平均值，用于综合评估模型的分类能力。它是一个综合指标，同时考虑了精确率和召回率的影响，可以帮助我们找到一个精度和召回率之间的平衡点。F1 值的计算公式为：F1 值 = $2*(精确率*召回率)/(精确率+召回率)$ 。

支持度 (Support): 支持度表示每个类别在数据集中的样本数量。它可以帮助我们了解每个类别的样本数量以及样本分布是否均匀。支持度的计算公式为：支持度 = TP + FN，其中 TP 表示真正例的数量，FN 表示假反例的数量。

这些指标可以提供不同层面的信息，帮助我们了解模型在不同情况下的表现。在解释这些指标时，需要结合具体的场景和任务，以便更好地理解它们的含义和作用。

为了更好地将模型在测试集上的测试结果可视化，我还采用了混淆矩阵来衡量。**混淆矩阵 (Confusion Matrix)** 是一种常用的性能评估方法，用于衡量分类模型在不同类别上的预测准确性。混淆矩阵展示了模型在测试集中对于每一个类别的预测情况，以及真实类别和预测类别之间的关系，包括真正例 (True Positive, TP)、真反例 (True Negative, TN)、假正例 (False Positive, FP) 和假反例 (False Negative, FN) 等四种情况。这几种情况的定义如下：

- 真正例 (True Positive, TP)：模型预测为正例，实际为正例的样本数量。
- 真反例 (True Negative, TN)：模型预测为反例，实际为反例的样本数量。
- 假正例 (False Positive, FP)：模型预测为正例，实际为反例的样本数量。
- 假反例 (False Negative, FN)：模型预测为反例，实际为正例的样本数量。

混淆矩阵可以帮助我们了解模型在不同类别上的表现，以及模型存在的误判情况。通过混淆矩阵，我们可以计算出模型在每个类别上的准确率、召回率、精确率和 F1 值等性能指标，以便更好地评估模型的分类能力。

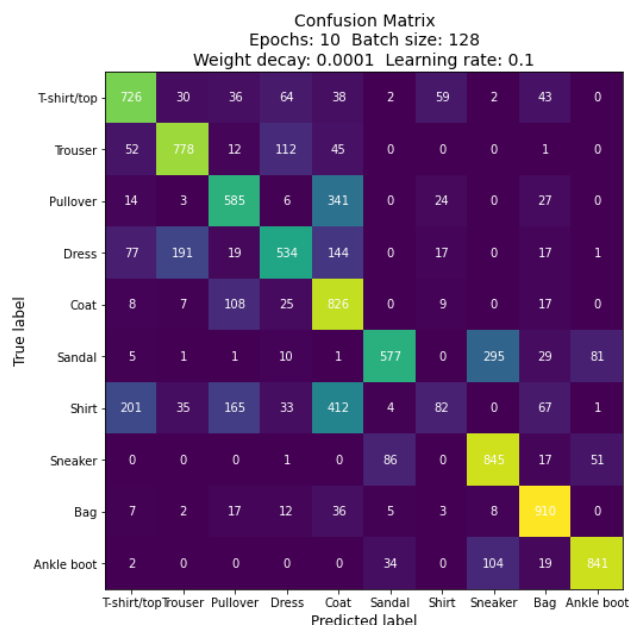


图 3 混淆矩阵

4.2 超参数调整与分析

影响神经网络训练结果的参数主要有以下四个：

num_epochs: 这是指训练的轮数。它的选择应该考虑到模型的复杂性、训练数据集的大小以及计算资源的限制。一般来说，num_epochs 应该足够大以确保模型充分训练，但不要太大以避免过拟合。常见的选择是 10~50 个 epochs。具体的选择可以通过验证集的表现来确定。

batch_size: 这是指每批次的样本数。它的选择应该考虑到计算资源的限制和模型的复杂性。一般来说，batch_size 应该足够大以确保有效的梯度计算，但不要太大以避免内存不足或计算资源的限制。常见的选择是 32~256。具体的选择可以通过训练时间和验证集的表现来确定。

weight_decay: 这是指正则化项的系数，用于控制模型的复杂度。它的选择应该考虑到模型的复杂度和训练数据的噪声程度。一般来说，weight_decay 应该足够大以确保模型的泛化能力，但不要太大以避免欠拟合。常见的选择是 0.001~0.00001。

learning_rate: 这是指优化算法在更新权重时使用的步长。它的选择应该考虑到模型的复杂度和训练数据的特点。一般来说，learning_rate 应该足够小以确保稳定的收敛，但不要太小以避免训练时间过长。常见的选择是 0.1~0.0001。

在以上四个参数的调整上，我的想法是用控制变量的方法，首先设置一个基本的对照组（组 1），然后每次只改动一个参数的值，其他三个参数与对照组相同，训练以后在测试集上对比结果，以此来对比不同参数对模型的影响。

训练模型的软硬件设备参数如下：

操作系统：Windows 10

软件：Visual Studio Code

处理器：Intel(R) Core(TM) i7-10750H CPU @ 2.60GHz

RAM：16.0 GB

GPU：NVIDIA GeForce GTX1650 Ti

各组模型的参数设置如下所示：

表 1 各组模型的参数设置

	num_epochs	batch_size	weight_decay	learning_rate
1(对照组)	10	128	0.0001	0.001
2	10	128	0.0001	0.01
3	10	128	0.0001	0.1
4	10	128	0.001	0.001
5	10	128	0.01	0.001
6	10	64	0.0001	0.001
7	10	256	0.0001	0.001
8	5	128	0.0001	0.001
9	15	128	0.0001	0.001

训练结果如下所示（具体见附录）：

表 2 训练效果对比

	训练时间(s)	平均准确率	F1	平均精确率
1(对照组)	71.3	91%	0.90	91%
2	73.0	87%	0.87	87%
3	75.2	67%	0.65	67%
4	73.9	90%	0.90	90%
5	80.5	85%	0.85	86%
6	98.9	90%	0.90	90%
7	69.4	90%	0.90	91%
8	39.0	90%	0.90	90%
9	117.2	91%	0.90	91%

通过上面设置不同参数下训练出模型的结果可以看出，这几个参数的设置对神经网络的训练和结果有着重要的影响。通过对比训练结果我们可以初步推断它们设置过大或过小可能会产生的影响：

训练轮数（num_epochs）过小会导致模型无法充分学习数据集的模式，从而导致欠拟合。训练轮数过多则可能导致模型过度拟合训练数据，无法泛化到测试数据。因此，需要根据数据集的大小和复杂程度来设置适当的训练轮数。

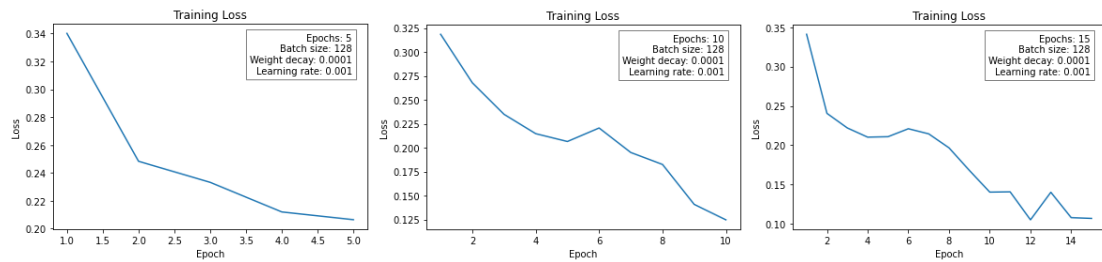


图 4 不同 epochs 下 loss 函数的变化情况

批处理大小 (batch_size) 过小会使训练过程更加嘈杂, 因为每个小批次的梯度更新具有更高的方差。批处理大小过大会占用过多的内存, 可能导致模型无法适应内存限制或训练速度变慢。因此, 需要根据可用内存和模型复杂度来选择适当的批处理大小。

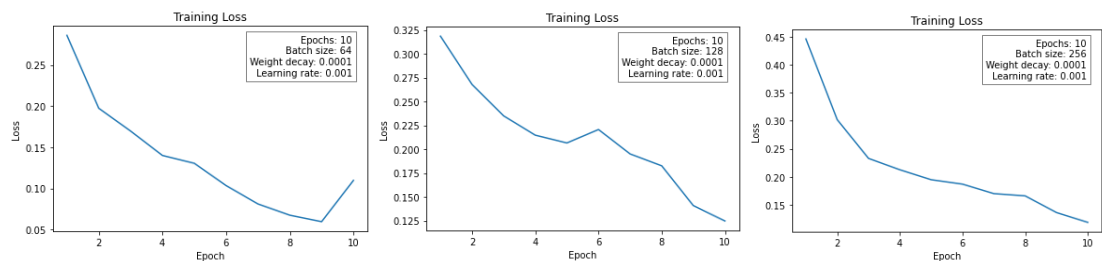


图 5 不同 batch size 下 loss 函数的变化情况

权重衰减 (weight_decay) 可以减少模型的过度拟合, 但如果设置过大, 可能会导致模型欠拟合, 因为它会强制将权重降低到接近零的较小值。因此, 需要根据模型复杂度和数据集噪声水平来选择合适的权重衰减系数。

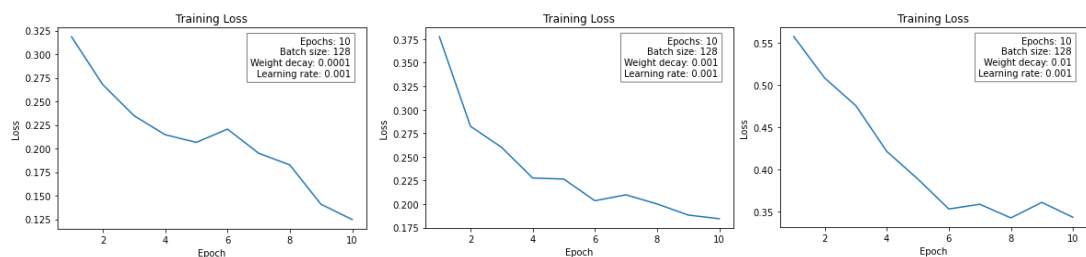


图 6 不同 weight decay 下 loss 函数的变化情况

学习率 (learning_rate) 控制模型参数更新的速度。学习率过小会导致模型收敛缓慢, 需要更多的训练轮数才能达到最佳性能, 而学习率过大可能会使模型在训练过程中发生震荡, 导致难以收敛。因此, 需要根据模型的复杂度、数据集的大小和梯度更新的频率来选择适当的学习率。

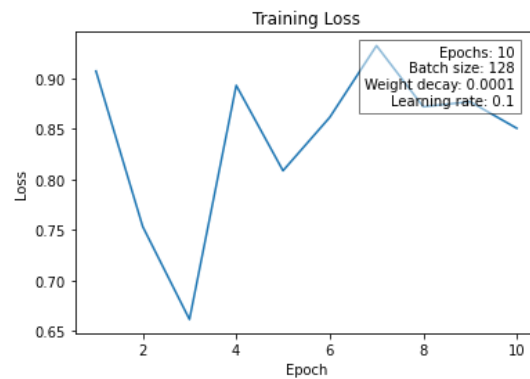


图 7 学习率过大导致震荡

总之，这些参数的设置需要根据具体问题和数据集来进行调整，并且最好进行实验和交叉验证来确定最佳参数设置。通常，可以通过尝试不同的参数设置，观察模型的训练和验证损失或准确率，以及模型在测试数据上的性能来确定最佳参数设置。

根据以上的分析和对比，可以基本得出能使模型效果较好的一组参数为：

`num_epochs = 15`

`batch_size = 64`

`weight_decay = 0.0001`

`learning_rate = 0.001`

设置该参数训练模型得到的结果如下：

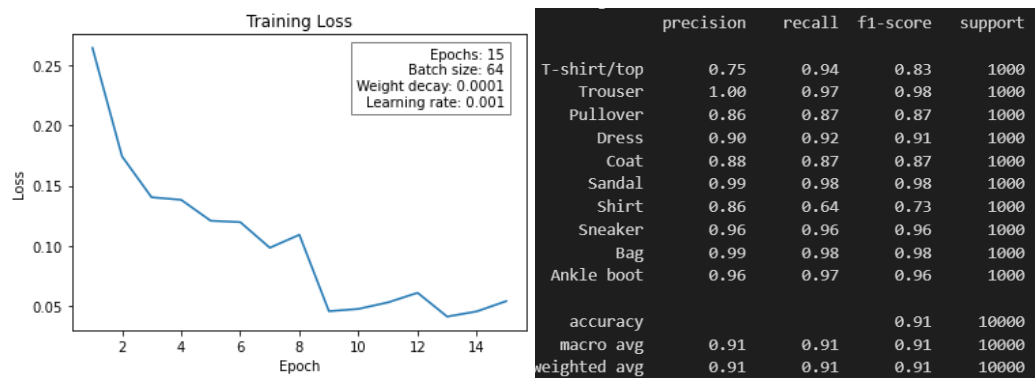


图 8 改进模型后的训练过程及在其在测试集上的测试结果

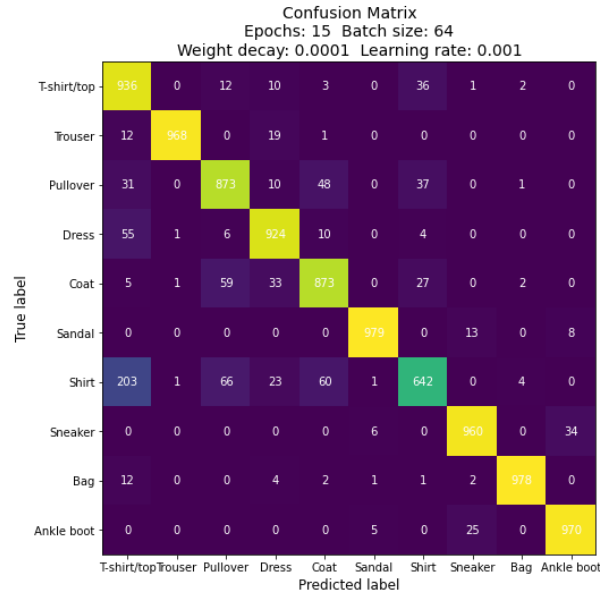


图 9 改进模型测试结果的混淆矩阵

5 结论与展望

5.1 实验结论

本实验通过使用卷积神经网络对 Fashion_MNIST 数据集进行图像识别，得到了较好的分类准确率。通过超参数调整，了解了四个参数对训练神经网络模型的影响，并依据此改进了模型参数，进一步提高了模型的性能，达到了 91.03% 的测试集准确率。

5.2 实验局限性与改进方向

本实验虽然取得了较好的结果，但仍存在一些局限性和改进方向：

1. **数据增强：**我们没有对数据进行增强操作，如旋转、平移、缩放等。通过数据增强，可以增加数据量，并且可以提升模型的泛化性能。
2. **更复杂的模型：**我们在本实验中采用了较简单的卷积神经网络结构。可以尝试使用更复杂的模型，如 ResNet、Inception 等，进一步提高模型的性能。
3. **模型融合：**可以采用模型融合的方法，将多个模型的预测结果进行加权平均或投票，进一步提高模型的准确率和泛化性能。

总之，本实验为图像识别任务提供了一个基于 Fashion_MNIST 数据集的实验平台，并展示了使用卷积神经网络对该数据集进行图像识别的可行性和效果。未来可以在此基础上进行进一步的研究和探索，以提高模型的性能和实用性。

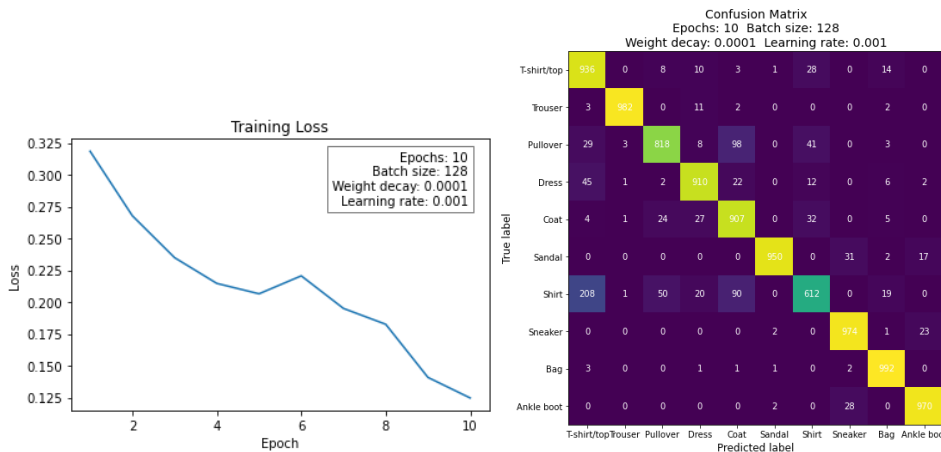
参考文献

[1] 老师发的代码

附录

第一组结果如下：

✓ 1m 11.3s

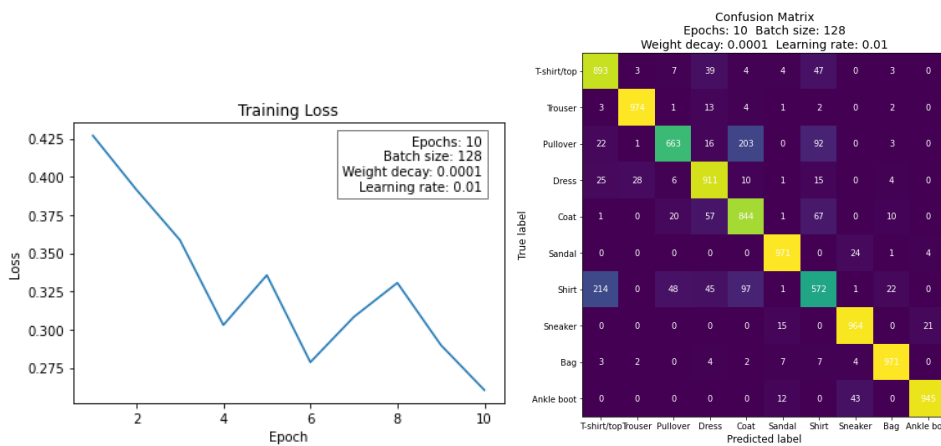


Epochs: 10
Batch size: 128
Weight decay: 0.0001
Learning rate: 0.001

	precision	recall	f1-score	support
T-shirt/top	0.76	0.94	0.84	1000
Trouser	0.99	0.98	0.99	1000
Pullover	0.91	0.82	0.86	1000
Dress	0.92	0.91	0.92	1000
Coat	0.81	0.91	0.85	1000
Sandal	0.99	0.95	0.97	1000
Shirt	0.84	0.61	0.71	1000
Sneaker	0.94	0.97	0.96	1000
Bag	0.95	0.99	0.97	1000
Ankle boot	0.96	0.97	0.96	1000
accuracy			0.91	10000
macro avg	0.91	0.91	0.90	10000
weighted avg	0.91	0.91	0.90	10000

第二组结果如下：

✓ 1m 13.0s

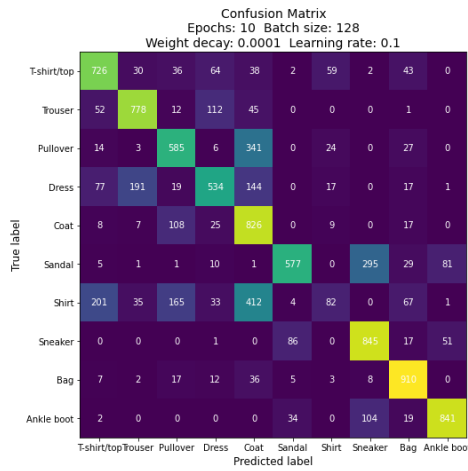
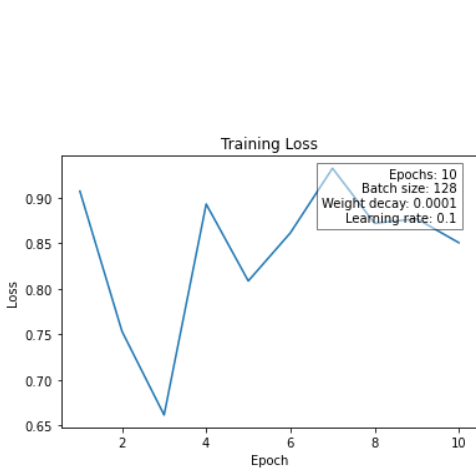


Epochs: 10
Batch size: 128
Weight decay: 0.0001
Learning rate: 0.01

	precision	recall	f1-score	support
T-shirt/top	0.77	0.89	0.83	1000
Trouser	0.97	0.97	0.97	1000
Pullover	0.89	0.66	0.76	1000
Dress	0.84	0.91	0.87	1000
Coat	0.73	0.84	0.78	1000
Sandal	0.96	0.97	0.96	1000
Shirt	0.71	0.57	0.63	1000
Sneaker	0.93	0.96	0.95	1000
Bag	0.96	0.97	0.96	1000
Ankle boot	0.97	0.94	0.96	1000
accuracy			0.87	10000
macro avg	0.87	0.87	0.87	10000
weighted avg	0.87	0.87	0.87	10000

第三组结果如下：

✓ 1m 15.2s

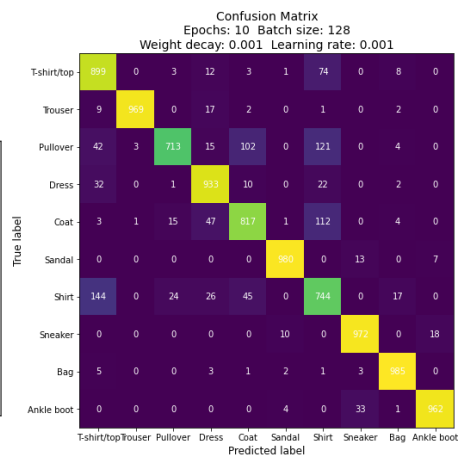
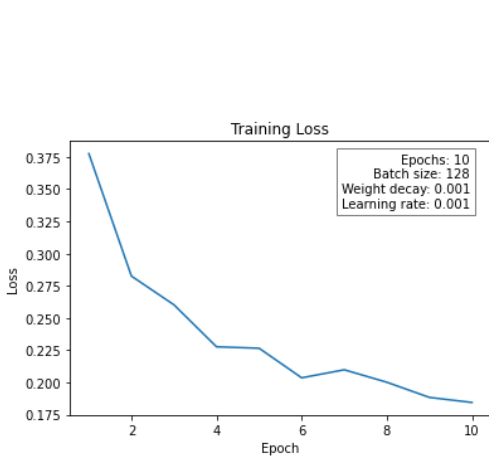


Epochs: 10
Batch size: 128
Weight decay: 0.0001
Learning rate: 0.1

	precision	recall	f1-score	support
T-shirt/top	0.66	0.73	0.69	1000
Trouser	0.74	0.78	0.76	1000
Pullover	0.62	0.58	0.60	1000
Dress	0.67	0.53	0.59	1000
Coat	0.45	0.83	0.58	1000
Sandal	0.81	0.58	0.68	1000
Shirt	0.42	0.08	0.14	1000
Sneaker	0.67	0.84	0.75	1000
Bag	0.79	0.91	0.85	1000
Ankle boot	0.86	0.84	0.85	1000
accuracy			0.67	10000
macro avg	0.67	0.67	0.65	10000
weighted avg	0.67	0.67	0.65	10000

第四组结果如下：

✓ 1m 13.9s

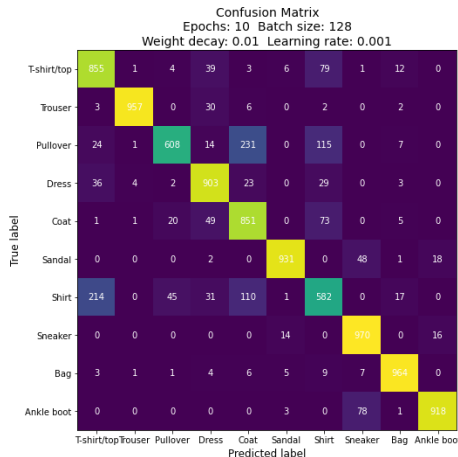
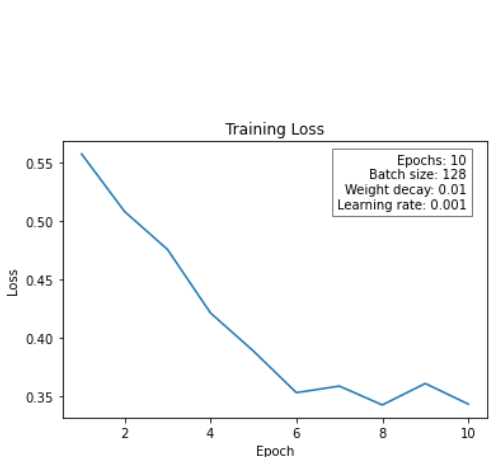


Epochs: 10
Batch size: 128
Weight decay: 0.001
Learning rate: 0.001

	precision	recall	f1-score	support
T-shirt/top	0.79	0.90	0.84	1000
Trouser	1.00	0.97	0.98	1000
Pullover	0.94	0.71	0.81	1000
Dress	0.89	0.93	0.91	1000
Coat	0.83	0.82	0.83	1000
Sandal	0.98	0.98	0.98	1000
Shirt	0.69	0.74	0.72	1000
Sneaker	0.95	0.97	0.96	1000
Bag	0.96	0.98	0.97	1000
Ankle boot	0.97	0.96	0.97	1000
accuracy			0.90	10000
macro avg	0.90	0.90	0.90	10000
weighted avg	0.90	0.90	0.90	10000

第五组结果如下：

✓ 1m 20.5s

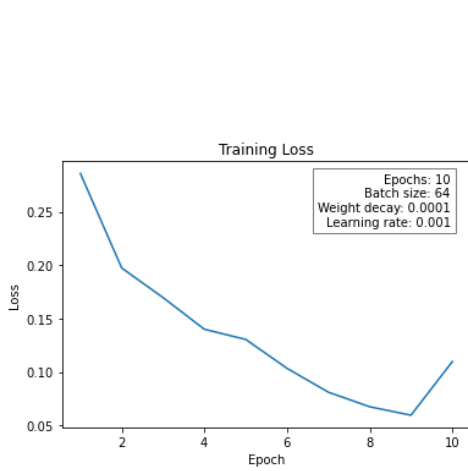


Epochs: 10
Batch size: 128
Weight decay: 0.01
Learning rate: 0.001

	precision	recall	f1-score	support
T-shirt/top	0.75	0.85	0.80	1000
Trouser	0.99	0.96	0.97	1000
Pullover	0.89	0.61	0.72	1000
Dress	0.84	0.90	0.87	1000
Coat	0.69	0.85	0.76	1000
Sandal	0.97	0.93	0.95	1000
Shirt	0.65	0.58	0.62	1000
Sneaker	0.88	0.97	0.92	1000
Bag	0.95	0.96	0.96	1000
Ankle boot	0.96	0.92	0.94	1000
accuracy			0.85	10000
macro avg	0.86	0.85	0.85	10000
weighted avg	0.86	0.85	0.85	10000

第六组结果如下：

✓ 1m 38.9s



Confusion Matrix
Epochs: 10 Batch size: 64
Weight decay: 0.0001 Learning rate: 0.001

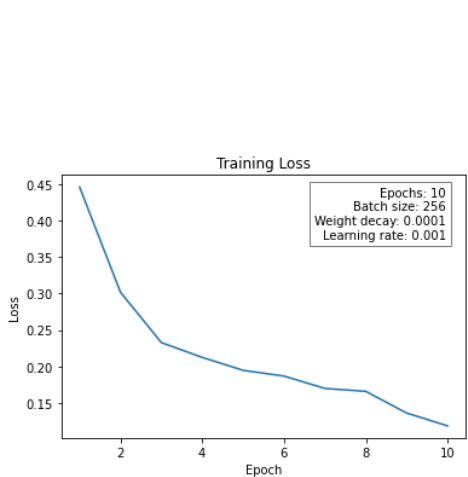
True label \ Predicted label	T-shirt/top	Trouser	Pullover	Dress	Coat	Sandal	Shirt	Sneaker	Bag	Ankle boot
T-shirt/top	885	1	9	9	2	4	89	0	1	0
Trouser	6	884	0	9	1	0	0	0	0	0
Pullover	28	0	755	7	96	1	113	0	0	0
Dress	41	0	2	896	32	2	23	0	4	0
Coat	3	1	32	25	825	2	108	0	4	0
Sandal	0	0	0	0	0	978	0	15	0	7
Shirt	126	3	34	20	56	0	749	0	12	0
Sneaker	0	0	0	0	0	5	0	978	0	17
Bag	4	1	0	4	1	4	3	0	883	0
Ankle boot	0	0	0	0	0	6	0	33	0	961

Epochs: 10
Batch size: 64
Weight decay: 0.0001
Learning rate: 0.001

	precision	recall	f1-score	support
T-shirt/top	0.81	0.89	0.85	1000
Trouser	0.99	0.98	0.99	1000
Pullover	0.91	0.76	0.82	1000
Dress	0.92	0.90	0.91	1000
Coat	0.81	0.82	0.82	1000
Sandal	0.98	0.98	0.98	1000
Shirt	0.69	0.75	0.72	1000
Sneaker	0.95	0.98	0.97	1000
Bag	0.98	0.98	0.98	1000
Ankle boot	0.98	0.96	0.97	1000
accuracy			0.90	10000
macro avg	0.90	0.90	0.90	10000
weighted avg	0.90	0.90	0.90	10000

第七组结果如下：

✓ 1m 9.4s



Confusion Matrix
Epochs: 10 Batch size: 256
Weight decay: 0.0001 Learning rate: 0.001

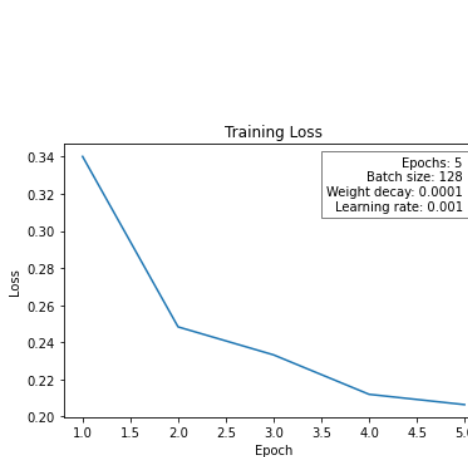
True label \ Predicted label	T-shirt/top	Trouser	Pullover	Dress	Coat	Sandal	Shirt	Sneaker	Bag	Ankle boot
T-shirt/top	938	0	19	9	2	1	21	0	10	0
Trouser	1	988	0	9	0	0	0	0	2	0
Pullover	20	2	928	6	26	0	16	0	2	0
Dress	45	7	12	903	20	0	8	0	4	1
Coat	3	1	104	27	826	0	30	0	9	0
Sandal	0	0	0	0	0	959	0	27	1	13
Shirt	210	4	114	21	60	0	574	0	17	0
Sneaker	0	0	0	0	0	3	0	953	1	43
Bag	2	1	1	2	1	1	0	1	991	0
Ankle boot	0	0	0	0	0	4	0	17	1	976

Epochs: 10
Batch size: 256
Weight decay: 0.0001
Learning rate: 0.001

	precision	recall	f1-score	support
T-shirt/top	0.77	0.94	0.85	1000
Trouser	0.99	0.99	0.99	1000
Pullover	0.79	0.93	0.85	1000
Dress	0.92	0.90	0.91	1000
Coat	0.88	0.83	0.85	1000
Sandal	0.99	0.96	0.97	1000
Shirt	0.88	0.57	0.70	1000
Sneaker	0.95	0.95	0.95	1000
Bag	0.95	0.99	0.97	1000
Ankle boot	0.94	0.98	0.96	1000
accuracy			0.90	10000
macro avg	0.91	0.90	0.90	10000
weighted avg	0.91	0.90	0.90	10000

第八组结果如下：

✓ 39.0s



Confusion Matrix
Epochs: 5 Batch size: 128
Weight decay: 0.0001 Learning rate: 0.001

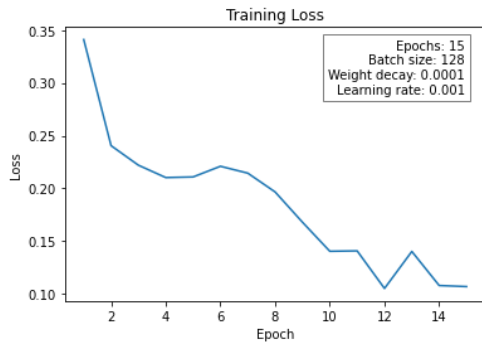
True label \ Predicted label	T-shirt/top	Trouser	Pullover	Dress	Coat	Sandal	Shirt	Sneaker	Bag	Ankle boot
T-shirt/top	893	0	5	8	2	2	86	0	4	0
Trouser	4	972	0	20	1	0	1	0	2	0
Pullover	28	1	749	11	86	0	122	0	3	0
Dress	41	0	2	919	11	0	27	0	0	0
Coat	2	1	15	52	822	1	102	0	5	0
Sandal	0	0	0	0	0	943	0	44	0	13
Shirt	136	1	22	21	49	0	765	0	6	0
Sneaker	0	0	0	0	0	2	0	963	0	15
Bag	3	0	0	3	1	1	1	6	985	0
Ankle boot	1	0	0	0	0	3	0	32	0	964

Epochs: 5
Batch size: 128
Weight decay: 0.0001
Learning rate: 0.001

	precision	recall	f1-score	support
T-shirt/top	0.81	0.89	0.85	1000
Trouser	1.00	0.97	0.98	1000
Pullover	0.94	0.75	0.84	1000
Dress	0.89	0.92	0.90	1000
Coat	0.85	0.82	0.83	1000
Sandal	0.99	0.94	0.97	1000
Shirt	0.69	0.77	0.73	1000
Sneaker	0.92	0.98	0.95	1000
Bag	0.98	0.98	0.98	1000
Ankle boot	0.97	0.96	0.97	1000
accuracy			0.90	10000
macro avg	0.90	0.90	0.90	10000
weighted avg	0.90	0.90	0.90	10000

第九组结果如下：

✓ 1m 57.2s



Confusion Matrix
Epochs: 15 Batch size: 128
Weight decay: 0.0001 Learning rate: 0.001

True label \ Predicted label	T-shirt/top	Trouser	Pullover	Dress	Coat	Sandal	Shirt	Sneaker	Bag	Ankle boot
T-shirt/top	909	0	22	20	5	1	35	0	8	0
Trouser	1	967	2	20	9	0	0	0	1	0
Pullover	26	1	910	10	35	0	15	0	3	0
Dress	26	3	9	916	39	0	5	0	2	0
Coat	4	1	74	17	890	0	14	0	0	0
Sandal	0	0	0	0	0	980	0	14	0	6
Shirt	192	0	93	25	115	0	572	0	13	0
Sneaker	0	0	0	0	0	6	0	986	0	14
Bag	5	0	0	2	4	1	0	2	966	0
Ankle boot	0	0	0	0	0	3	0	31	1	965

Epochs: 15
Batch size: 128
Weight decay: 0.0001
Learning rate: 0.001

	precision	recall	f1-score	support
T-shirt/top	0.79	0.91	0.84	1000
Trouser	0.99	0.97	0.98	1000
Pullover	0.82	0.91	0.86	1000
Dress	0.91	0.92	0.91	1000
Coat	0.81	0.89	0.85	1000
Sandal	0.99	0.98	0.98	1000
Shirt	0.89	0.57	0.70	1000
Sneaker	0.95	0.98	0.97	1000
Bag	0.97	0.99	0.98	1000
Ankle boot	0.98	0.96	0.97	1000
accuracy			0.91	10000
macro avg	0.91	0.91	0.90	10000
weighted avg	0.91	0.91	0.90	10000