Contents

Assignment 2: Building a CNN

1 Task

Building a single CNN network with multiclass classifier heads for 3 different classification tasks on the same data.

2 Objective

Implemented a CNN in Python (using Pytorch) to classify CIFAR100 images. CIFAR100 has 100 classes, with 20 superclasses as described here and further classified into groups as follows:

- Plants/Parts of plants (Superclasses "flowers", "trees", "fruits and vegetables")
- Vehicles (Superclasses "vehicles1" and "vehicles2")
- Invertebrates (Superclasses "non-insect invertebrates" and "insects")
- Aquatic animals (Superclasses "fish" and "aquatic mammals")
- Large animals (Superclasses "large carnivores" and "large omnivores and herbivores")
- Man-made articles (Superclasses "food containers", "household electrical devices", "household furniture", and "large man-made outdoor things")
- People (Superclass "people")
- Normal Terrestrial Animals (Superclass "reptiles", "medium-sized mammals" and "small mammals")
- Outdoor scenes (Superclass "large natural outdoor scenes")

This is the nomenclature we are following for the taxonomy: $CLASSES(100) \rightarrow SUPERCLASSES(20) \rightarrow GROUPS(9)$

3 Dataset

The CIFAR100 dataset is used. CIFAR100 has 100 classes, with 20 superclasses as described here.

4 Network Architecture

- A single model is trained that can determine the class, superclass or (synthesized) groups of any given image as required.
- ResNet is used for (Deep) CNN.
- **Loss function:** Cross Entropy and **Number of epochs:** 10.
- Total trainable and non-trainable parameters are reported.

5 Observations

Randomized 70:30, 80:20 and 90:10 are used and the obtained model performance (Accuracy, Loss, Confusion Matrix) for each split is reported.

5.1 70:30 Split-ratio

The performance metrics for this configuration are summarized in Table 1. Below is the loss and accuracy after each epoch:

Epoch	Training Loss	Val Class Acc	Val Superclass Acc	Val Group Acc
1	4.5274	0.5219	0.5041	0.5392
2	2.3939	0.5963	0.5793	0.6049
3	1.6069	0.6077	0.6003	0.6305
4	1.1175	0.6591	0.6547	0.6703
5	0.7728	0.6673	0.6678	0.6823
6	0.5383	0.6642	0.6594	0.6684
7	0.4260	0.6511	0.6494	0.6640
8	0.3846	0.6499	0.6518	0.6664
9	0.2634	0.6720	0.6700	0.6897
10	0.2327	0.6776	0.6681	0.6873

Table 1: Training and Validation Metrics (70:30 Split)

5.2 80:20 Split-ratio

The performance metrics for this configuration are summarized in Table 2. Below is the loss and accuracy after each epoch:

Epoch	Training Loss	Val Class Acc	Val Superclass Acc	Val Group Acc
1	1.3536	0.8135	0.8189	0.8223
2	0.6525	0.8352	0.8416	0.8507
3	0.3887	0.8314	0.8293	0.8403
4	0.2224	0.8527	0.8556	0.8664
5	0.1468	0.8502	0.8533	0.8603
6	0.1671	0.8286	0.8328	0.8385
7	0.1109	0.8234	0.8270	0.8418
8	0.1312	0.8145	0.8115	0.8150
9	0.2552	0.7814	0.7840	0.7858
10	0.3341	0.7524	0.7511	0.7665

Table 2: Training and Validation Metrics (80:20 Split)

5.3 90:10 Split-ratio

The performance metrics for this configuration are summarized in Table 3. Below is the loss and accuracy after each epoch:

Epoch	Training Loss	Val Class Acc	Val Superclass Acc	Val Group Acc
1	0.7468	0.9130	0.9174	0.9252
2	0.2821	0.9236	0.9244	0.9302
3	0.1581	0.9254	0.9286	0.9302
4	0.0947	0.9334	0.9388	0.9394
5	0.0523	0.9366	0.9432	0.9418
6	0.0392	0.9346	0.9422	0.9436
7	0.0294	0.9370	0.9400	0.9414
8	0.0276	0.9332	0.9382	0.9368
9	0.0472	0.9078	0.9082	0.9072
10	0.4795	0.7560	0.7746	0.8022

Table 3: Training and Validation Metrics (90:10 Split)

5.4 Plots

The following plots provide insights into the performance of the network:

5.4.1 Confusion Matrices for 70:30 Split-ratio

Below are the confusion matrices for each classification level (Class, Superclass, and Group) for the 70:30 split:

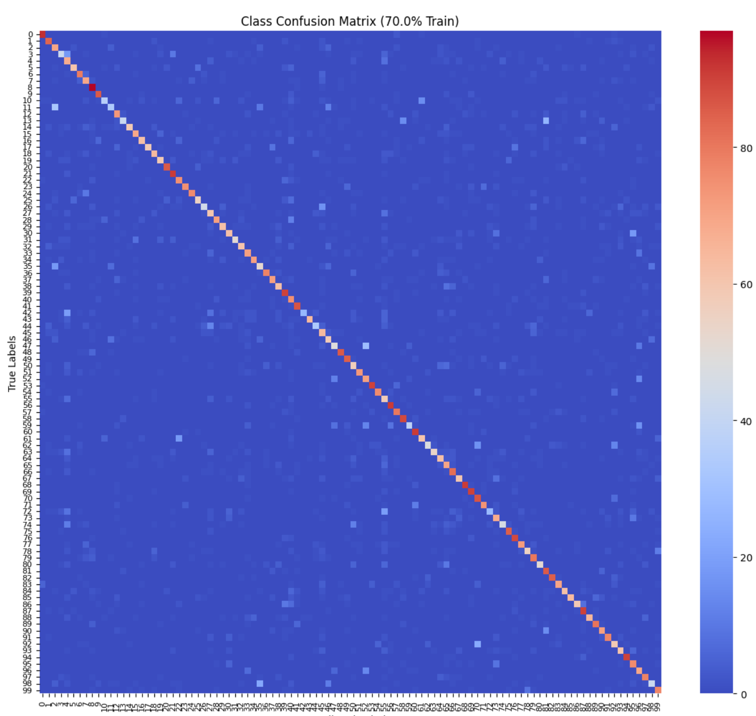

Figure 1: Confusion Matrix for Class (70:30 Split)

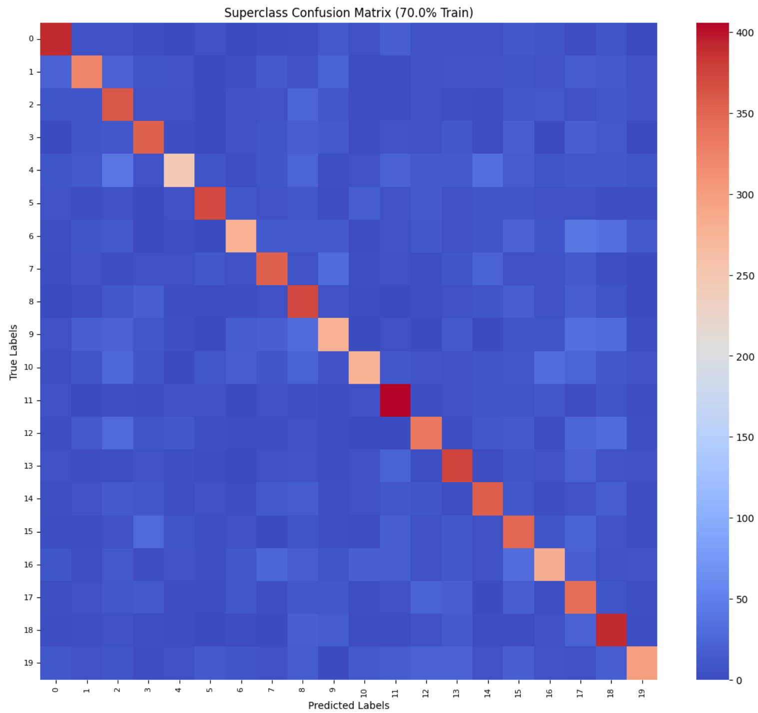

Figure 2: Confusion Matrix for Superclass (70:30 Split)

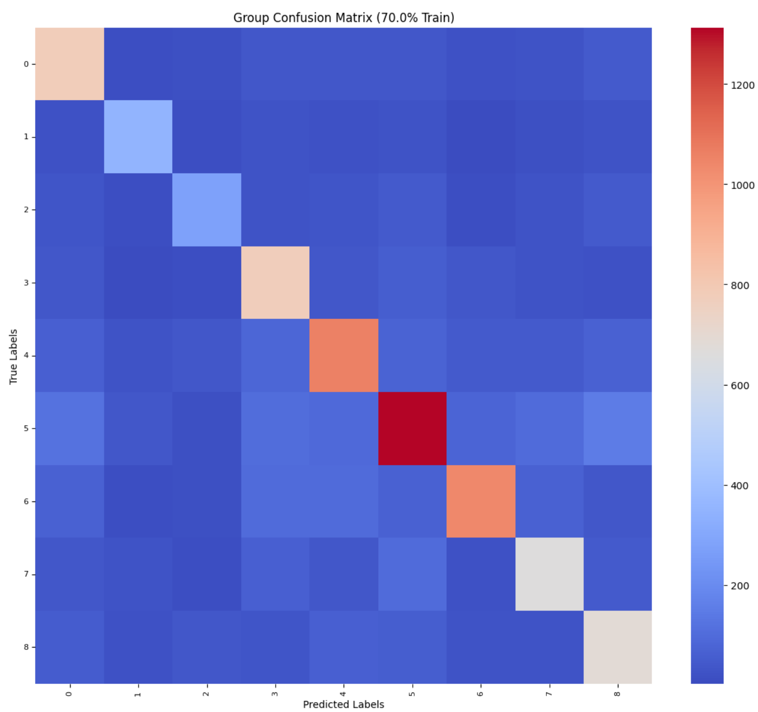

Figure 3: Confusion Matrix for Group (70:30 Split)

5.4.2 Confusion Matrices for 80:20 Split-ratio

Below are the confusion matrices for each classification level (Class, Superclass, and Group) for the 80:20 split:

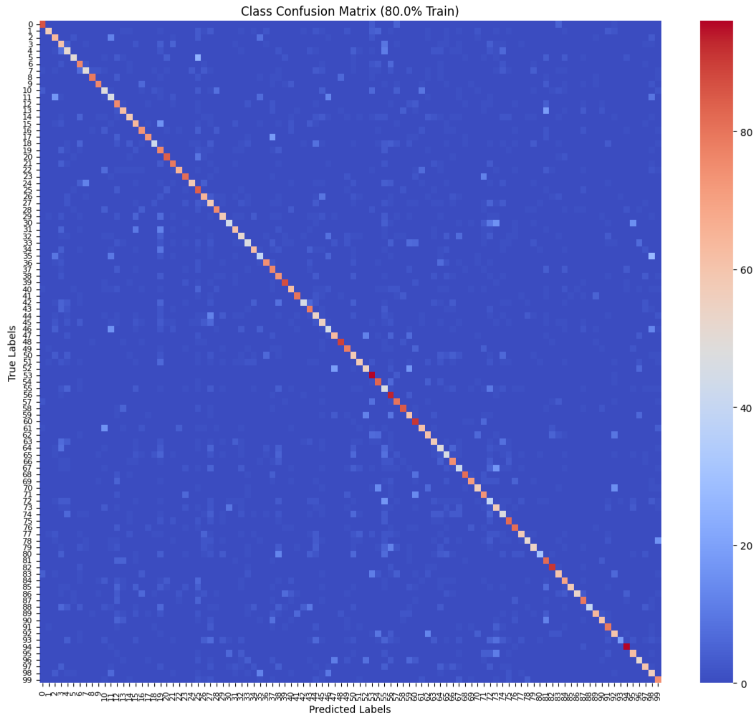

Figure 4: Confusion Matrix for Class (80:20 Split)

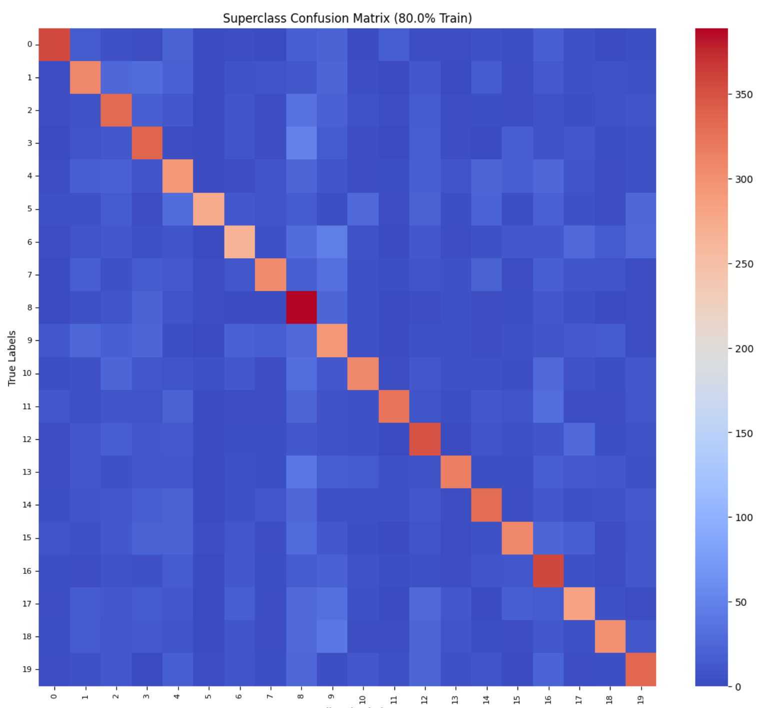

Figure 5: Confusion Matrix for Superclass (80:20 Split)

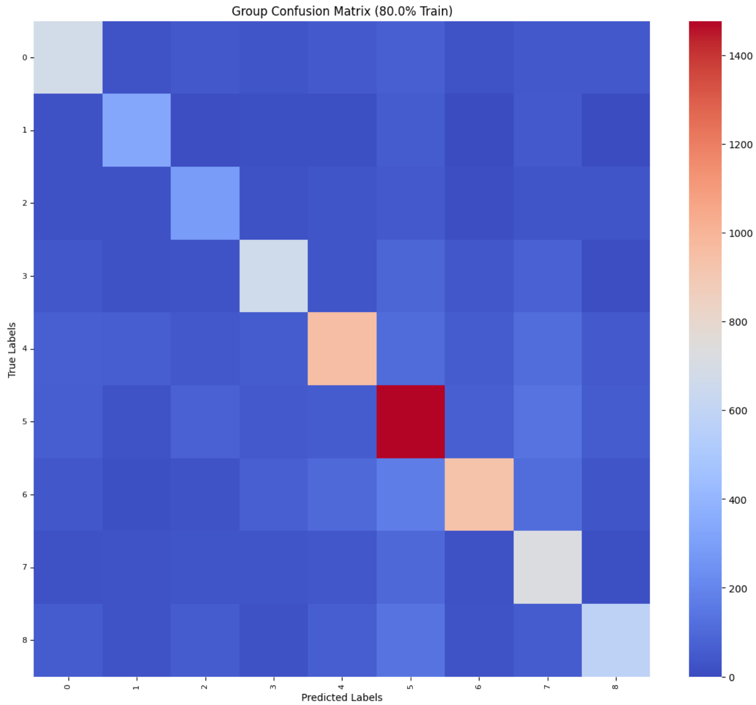

Figure 6: Confusion Matrix for Group (80:20 Split)

5.4.3 Confusion Matrices for 90:10 Split-ratio

Below are the confusion matrices for each classification level (Class, Superclass, and Group) for the 90:10 split:

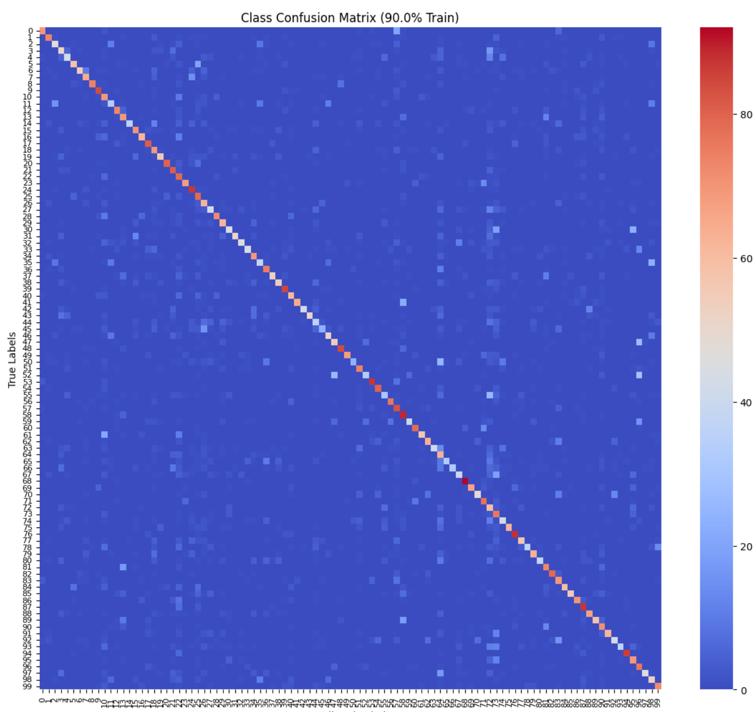

Figure 7: Confusion Matrix for Class (90:10 Split)

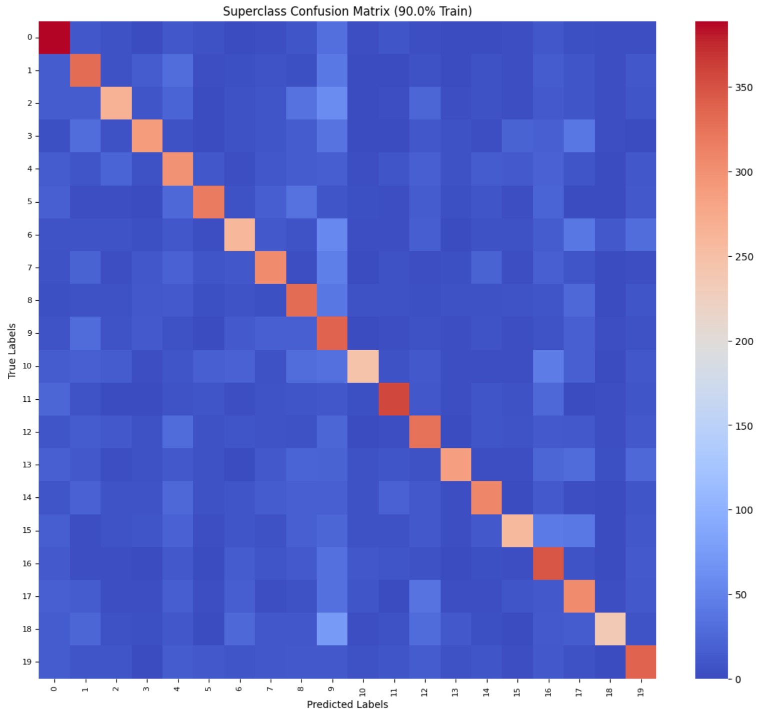

Figure 8: Confusion Matrix for Superclass (90:10 Split)

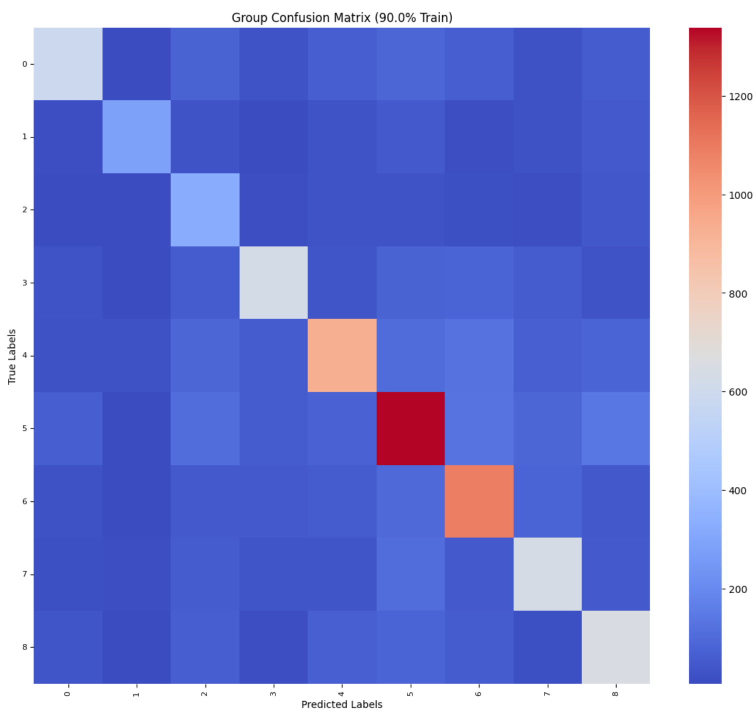

Figure 9: Confusion Matrix for Group (90:10 Split)

6 Results and Findings

The performance of the Convolutional Neural Network (CNN) for CIFAR100 dataset classification is analyzed based on different train-test splits (70:30, 80:20, and 90:10). The findings are summarized below:

6.1 Performance across Train-Test Splits

The model was trained with randomized splits of 70:30, 80:20, and 90:10. The key performance metrics observed include training loss, validation accuracy at class, superclass, and group levels.

- **70:30 Split:**
  - Training Loss decreased from 4.5274 to 0.2327 across 10 epochs.
  - Validation Class Accuracy peaked at 67.76%.

- Superclass and Group Accuracy reached 66.81% and 68.73%, respectively.

- **80:20 Split:**
  - Training Loss reduced from 1.3536 to 0.3341.
  - Maximum Validation Class Accuracy observed: 85.27%.
  - Superclass Accuracy reached 85.56%, and Group Accuracy was 86.64%.
- **90:10 Split:**
  - Training Loss decreased from 0.7468 to 0.4795.
  - Maximum Validation Class Accuracy achieved: 93.66%.
  - Superclass Accuracy peaked at 94.32%, and Group Accuracy reached 94.18%.

6.2 Insights from Confusion Matrices

Confusion matrices were analyzed for class, superclass, and group classifications across different splits:

- The model demonstrates better performance when classifying groups compared to superclass or class level, indicating the hierarchical grouping is effective.
- Misclassifications are more frequent between classes within the same superclass, validating the hierarchical taxonomy approach.
- Group-level predictions are more robust across different splits.

6.3 Impact of Train-Test Split Ratio

- Increasing training data improves accuracy, as seen in the 90:10 split achieving the highest performance metrics.
- Overfitting tendencies were observed with the 80:20 and 90:10 splits, where validation accuracy declined in later epochs.
- The 70:30 split provided a balanced performance, with stable metrics across epochs.

6.4 Additional Observations

- The ResNet architecture effectively captures class-level distinctions despite high inter-class similarities.

Overall, the CNN model successfully performed hierarchical classification across classes, superclasses, and groups with promising results across different train-test splits.

Google Collab link: M24AIR006.