#Problem 2

The problem is to design and implement a CNN architecture capable of classifying the CIFAR-10 images into their respective categories. This involves defining the network layers, specifying the loss function, optimizer, and training loop, as well as evaluating the model's performance on the test set.

Methodology

The CNN architecture implemented in this assignment consists of four convolutional layers, each followed by batch normalization, ReLU activation, max pooling, and dropout layers. The final layer is a fully connected layer with 10 output units, representing the confidence scores for the 10 classes.

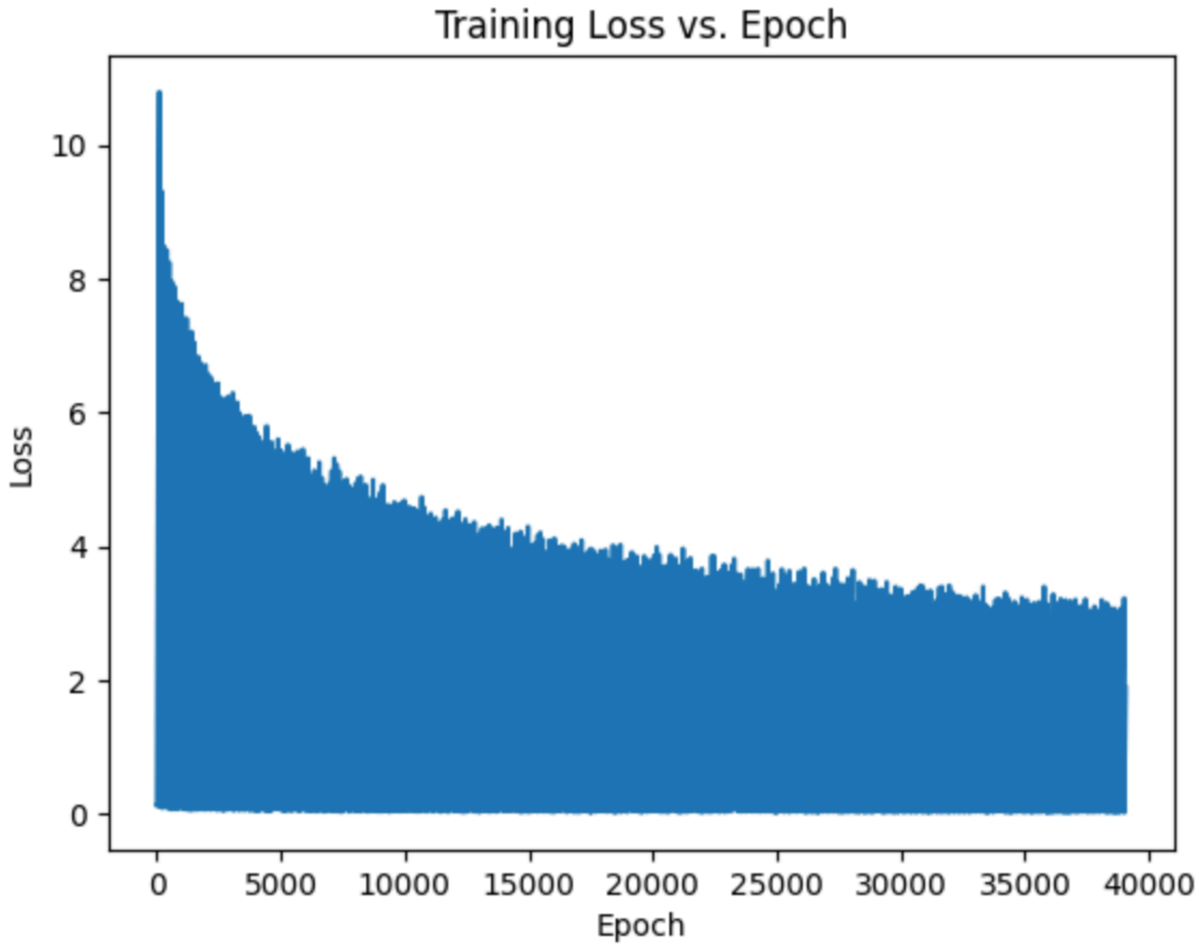
The network is defined in the ConvNet class, where the __init__ method initializes the layers, and the forward method defines the forward pass of the input through the network. The loss function used is the cross-entropy loss, which is suitable for multi-class classification tasks. The optimizer chosen is Stochastic Gradient Descent (SGD) with a momentum of 0.9 and a learning rate of 0.001.

The training process is carried out for 25 epochs, with a batch size of 32. During each epoch, the model is trained on the training set, and the accuracy and loss are computed. After each epoch, the model is evaluated on the test set, and the test accuracy is recorded. The training loss, training accuracy, and test accuracy are plotted over the epochs to visualize the model's performance and convergence.

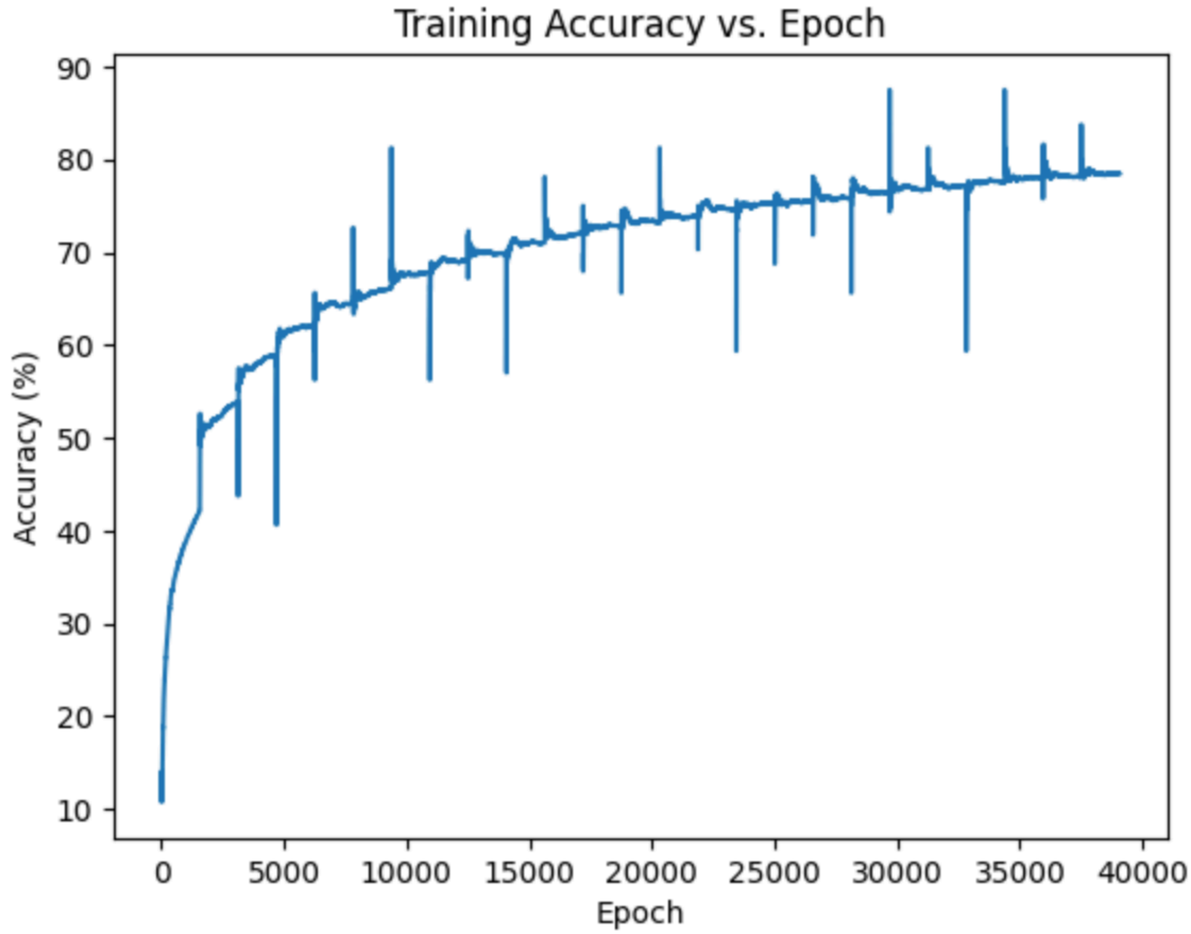
Results

The code provided achieves a final testing accuracy of 80.65% on the CIFAR-10 dataset after 25 epochs of training. The training accuracy and loss plots indicate that the model converges reasonably well, with the training accuracy reaching around 90% and the training loss decreasing steadily.

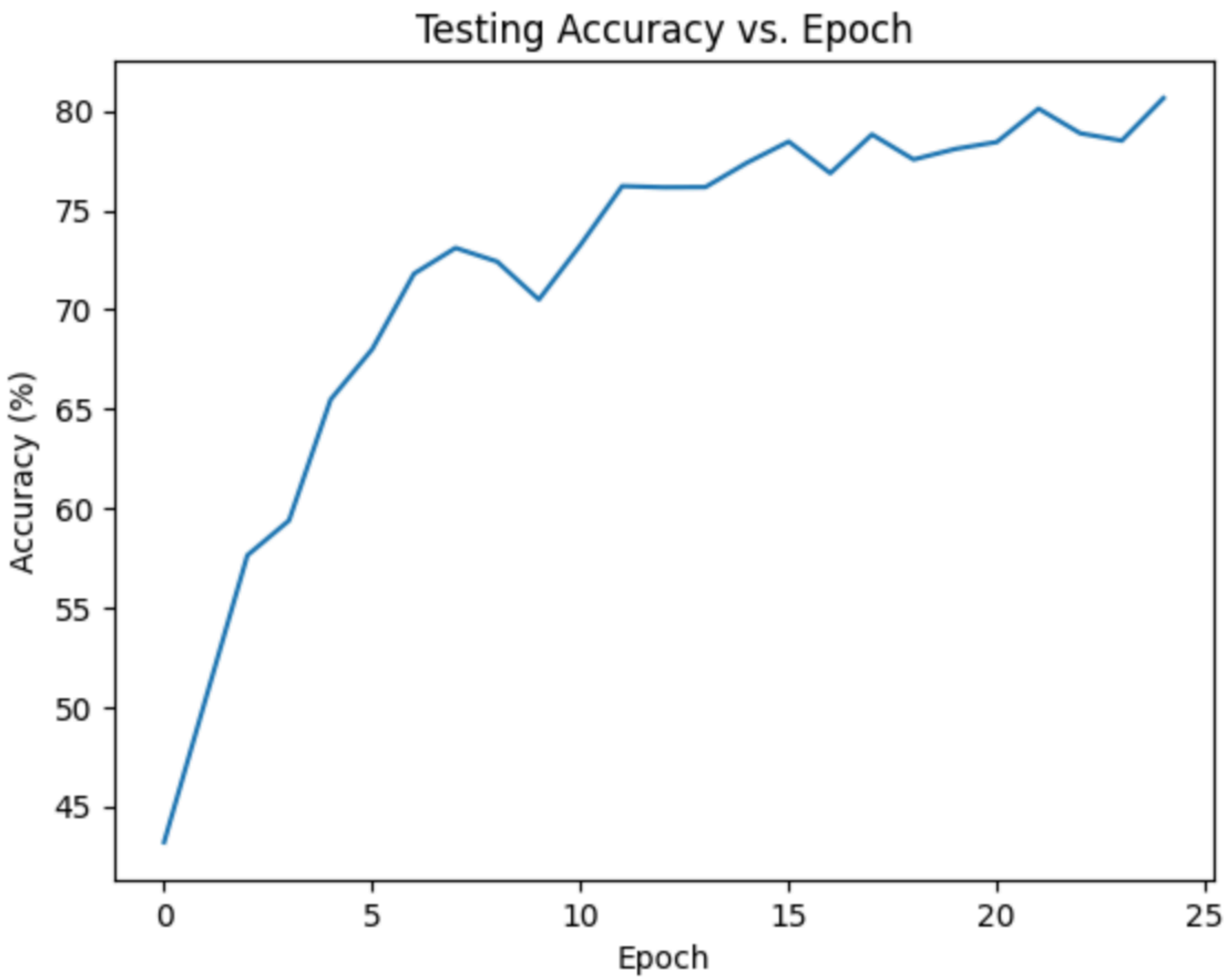
By analyzing the training and test accuracy plots, it can be inferred that the model is neither severely overfitted nor underfitted. The training and test accuracies are reasonably close, suggesting that the model has generalized well to the test set.



As we can see from the graph the loss decreases as the number of epoch got increased



As we can see from the graph the accuracy increased as the number of epochs increased.



Also we can see from the graph that testing accuracy tends to increase as the number of epochs increases.

Conclusion

In this problem, a 4-layer CNN was successfully trained from scratch on the CIFAR-10 dataset, achieving a respectable testing accuracy of 80.65%. The code implements the necessary components, including the CNN architecture, loss function, optimizer, and training loop.

#Problem 3

The problem is to conduct zero-shot classification on the CIFAR-100 dataset using the pre-trained CLIP (Contrastive Language-Image Pre-training) model. Zero-shot classification is a challenging task that involves classifying images into categories without any labeled training data for those categories. Instead, the model relies on its ability to associate images with text descriptions, leveraging the knowledge acquired during pre-training on a large-scale dataset of image-text pairs.

Methodology:

- ***Model Selection:*** The ViT-B/32 variant of the CLIP model was chosen for this task. This model has approximately 151 million parameters and expects input images with a resolution of 224x224 pixels.
- ***Data Preparation:***
    - The CIFAR-100 dataset was downloaded and preprocessed using the provided torchvision.transforms.
    - Text descriptions for each of the 100 classes in CIFAR-100 were created in the format "This is a photo of a {label}".
    - The text descriptions were tokenized using the CLIP tokenizer, and their text features were computed using the CLIP model's text encoder.
    -
- ***Zero-Shot Classification:***
    - A custom PyTorch dataset and dataloader were created to stream the CIFAR-100 images in batches of 256.
    - For each batch of images, their image features were computed using the CLIP model's image encoder.
    - The cosine similarity between the image features and text features for each class was calculated, treating these similarity scores as logits.
    - The class for each image was predicted by taking the argmax of the logits, effectively assigning the image to the class with the highest cosine similarity between its image features and the text features.
    - The number of correct predictions was tracked, and the overall classification accuracy was computed at the end.

Results:

The zero-shot classification accuracy achieved on the CIFAR-100 dataset using the ViT-B/32 variant of the CLIP model is 58.85%.

Conclusion:

The pre-trained CLIP model demonstrated its capability to perform zero-shot classification on the challenging CIFAR-100 dataset, which has a diverse set of 100 classes. However, the achieved accuracy of 58.85% is lower than the reported accuracy of around 65% in the CLIP paper. This discrepancy could be due to various factors, such as the choice of the CLIP model variant, differences in preprocessing or evaluation methodology, or the inherent difficulty of the CIFAR-100 dataset for zero-shot classification.