AAIT – HW2

Luca Marian

Task1

Initially, I aimed to create a model that performs well on a test set. The original dataset was divided into two parts: training and validation. In each split, I ensured a balanced representation from each of the 100 classes. The split ratio was 80% for training and 20% for validation.

The model used was ResNet50 with ImageNet weights. Initially, I experimented with a low batch size, which yielded unsatisfactory results. However, increasing the batch size to 64 or more significantly improved the model's performance, likely because a larger batch size aids in better generalization.

After establishing a good baseline, I labeled all the unlabeled data using this model and attempted a teacher-student training procedure. I implemented a noisy student training method using two models based on ResNet50.

The first model, the teacher, was the baseline model, and the second was another ResNet50 model with identical parameters.

I employed two methods for this approach:

1. Using the baseline teacher model, I generated pseudo-labels. I applied a threshold that varied across multiple training iterations and experiments. The pseudo-labels were augmented and combined with the labeled dataset, shuffled, and then used to train the student model. If the student model outperformed the teacher model in terms of accuracy, I updated the teacher model with the student's parameters and generated new pseudo-labels. This process was repeated for 6 iterations.
2. The second method involved training both unlabeled and labeled data in the same training loop. I trained the teacher model on labeled data and used it to label the unlabeled data in that batch. The student model was first trained on labeled data and then on unlabeled data, compared with the teacher model. Losses for both labeled and unlabeled data were computed. If the student model demonstrated higher accuracy during evaluation, the teacher model parameters were updated with those of the student. This approach yielded the best results.

The final approach I attempted involved creating an ensemble of three models: ResNet50, ResNet34, and ResNet18. I trained each of these three models on the labeled dataset. Subsequently, I combined them into a single model. Using a threshold of 0.90, I generated pseudo-labels and created a new dataset, incorporating the labeled data. This ensemble model was then trained on this augmented dataset. However, the results did not surpass those of the baseline ResNet50 model used alone.

During the training, I chose the Stochastic Gradient Descent (SGD) optimizer as the hyperparameter because I believe it is more robust for fine-tuning pre-trained models. Additionally, I selected Cross-Entropy Loss as it is well-suited for multi-label classification tasks. Furthermore, I opted for a learning rate scheduler designed to reduce the learning rate as training progresses.

Notable results on leatherboard:

Model	BaselineResnet50	NoisyStudentV1	NoisyStudentV2	EnsembleModels
Results	71.92	73.26	74.06	53.46

Task2

For this task, I followed a similar approach to Task 1. Initially, I trained a ResNet model on the entire dataset, which served as a good baseline.

I then implemented a co-teaching method to improve accuracy on the noisy label dataset. The co-teaching algorithm is a method where two deep neural networks are trained simultaneously on a dataset with noisy labels. The core idea is to enable each network to guide the other in focusing on the most reliable parts of the data (those with the smallest loss), thereby reducing the impact of label noise on training. The reduction rate (rt), which determines the proportion of data to focus on, is dynamically adjusted throughout the training.

Unfortunately, I was unable to surpass the baseline model with these approaches. The best result for this task was achieved by the model trained in Task 1 and fine-tuned on the second dataset from Task 2.

For the training parameters, I employed the same hyperparameters as used in Task 1. However, for the co-teaching approach, I opted for an Adam optimizer instead of the SGD optimizer that was used in the first task.

Notable results on leatherboard:

Model	Baseline- Resnet50	Coteaching-2Resnet
Results	71.32	62.74

References

[1] https://www.ai-contentlab.com/2023/03/self-training-with-noisy-student.html

[2] https://medium.com/@bhatiatrj/deep-learning-dealing-with-noisy-labels-51961ad0f140

[3] Han, B., Yao, Q., Yu, X., Niu, G., Xu, M., Hu, W., ... & Sugiyama, M. (2018). Co-teaching: Robust training of deep neural networks with extremely noisy labels. *Advances in neural information processing systems*, *31*.

[4] Xie, Q., Luong, M. T., Hovy, E., & Le, Q. V. (2020). Self-training with noisy student improves imagenet classification. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 10687-10698).

[5] Pham, H., Dai, Z., Xie, Q., & Le, Q. V. (2021). Meta pseudo labels. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 11557-11568).