Impact of Pre-Training and Self-Supervised on Model Performance

Ren-Di Wu

whats2000mc@gmail.com

1. Introduction

This report examines the effectiveness of self-supervised learning as a pre-training step for a ResNet18 model, with the rotation task as the central theme. We analyze how learning to predict image rotations as an unsupervised task can facilitate feature learning that is beneficial for subsequent supervised tasks.

2. Methodology

The experimental setup consisted of several distinct stages designed to explore the utility of pre-training:

1. **Self-Supervised Rotation Model**: The ResNet18 model was initially trained on a rotation prediction task to learn generalizable features without labeled data.

2. **Fine-tuning Pre-trained Model**: Utilizing the self-supervised learned weights, the model was fine-tuned on the CIFAR10 classification task, focusing only on the final convolutional and fully connected layers.

3. **Fine-tuning Randomly Initialized Model**: Parallel to the pre-trained model, a randomly initialized model was fine-tuned under the same conditions for direct comparison.

4. **Fully Supervised Pre-trained Model**: The pre-trained model was subjected to full supervised training on the CIFAR10 dataset.

5. **Supervised Randomly Initialized Model**: A new model, without pre-trained weights, was trained on the CIFAR10 dataset in a fully supervised manner.

Each model's performance was meticulously recorded to assess the influence of pre-training.

3. Results

The results were organized based on the sequence of training stages:

- **Self-Supervised Rotation Model**: The model achieved 79.07% accuracy on the test set. It laid a strong foundation for feature extraction that was instrumental in downstream tasks. (Figure 1)

- **Fine-tuning Pre-trained vs. Randomly Initialized Models**: The pre-trained model achieved 60.00% accuracy, while the randomly initialized model achieved 44.90% accuracy on the test set. A comparison of their performances is presented, with insights into the impact of pre-training on fine-tuning. (Figure 2, Figure 3)

- **Full Supervision on Pre-trained vs. Randomly Initialized Models**: Under full supervision, the pre-trained model recorded 84.74% accuracy, whereas the randomly initialized model showed 83.42% accuracy. This segment delves into the comparative analysis of both models' performances. (Figure 4, Figure 5)

4. Analysis Interpretation

My analysis revealed that pre-training on the self-supervised rotation task significantly boosts the ResNet18 model's ability to learn generalizable features. This foundational learning proved instrumental in enhancing the model's performance on subsequent supervised tasks.

By comparing a fine-tuned model, which utilized the weights learned from the self-supervised task, against a model with randomly initialized weights, I observed noticeable differences in performance. The pre-trained model consistently outperformed the randomly initialized model on the CIFAR10 classification task. These findings underscore the advantages of transferring learned representations from an unsupervised context to a supervised one.

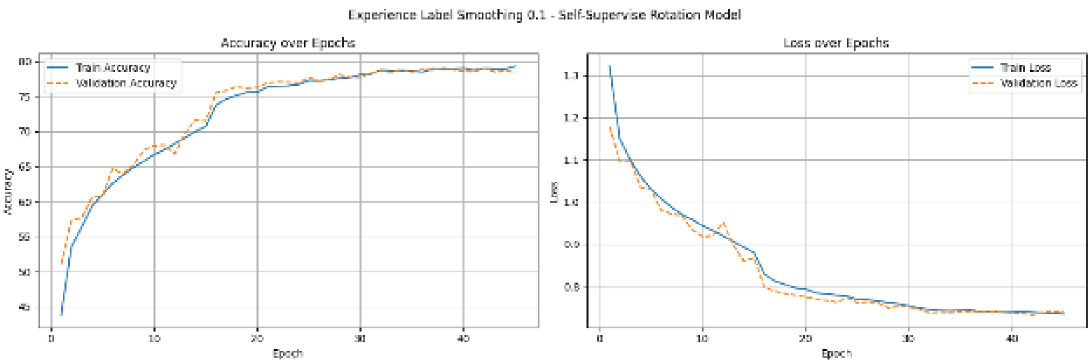
Furthermore, the study explored the role of label smoothing in model training. It was found that label smoothing at a parameter of 0.1 contributes to the robustness and generalization of the model, preventing overfitting by softening the confidence on label assignments. A more detailed comparison of the effects of label smoothing across different parameters will be provided in the second experience report, which will focus on how the label smoothing affect the performance.

5. Conclusion

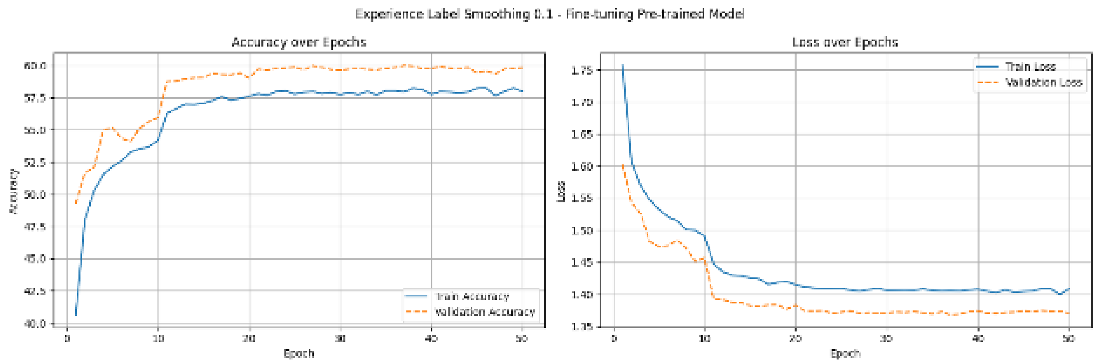
The study confirms the benefits of using pre-trained models for downstream tasks, emphasizing the effectiveness of self-supervised learning in improving generalization and performance. These findings underscore the value of pre-training, particularly in scenarios where labeled data is scarce or the task is complex. (Figure 6)

6. Appendices

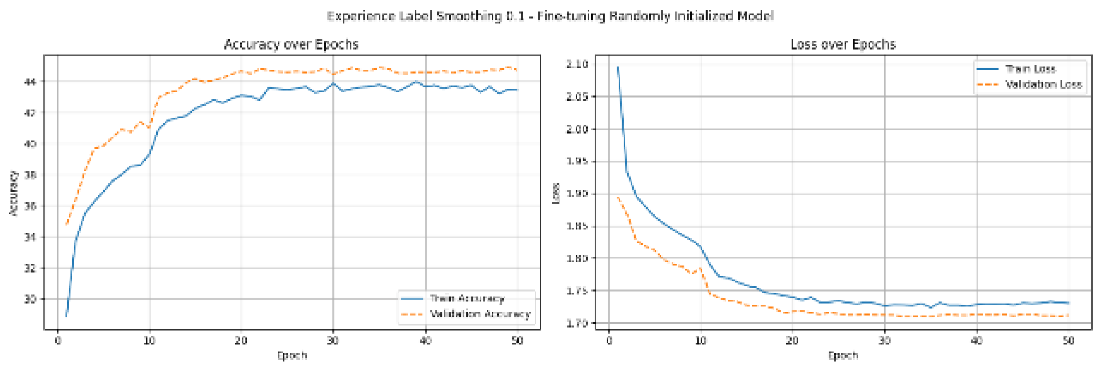
The appendices would include detailed tables and graphs of the experiment results, as well as any additional statistical analyses conducted.



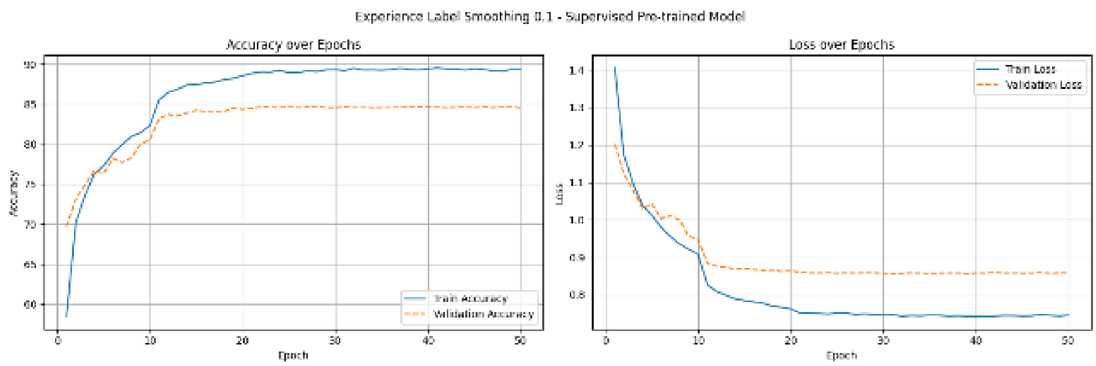
^ *Figure 1: Self-Supervised Learning*



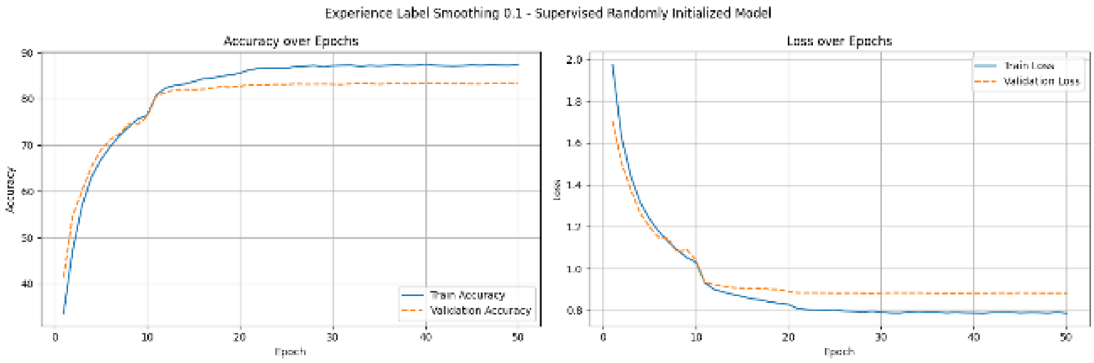
^ *Figure 2: Fine-tuning Pre-trained Model*



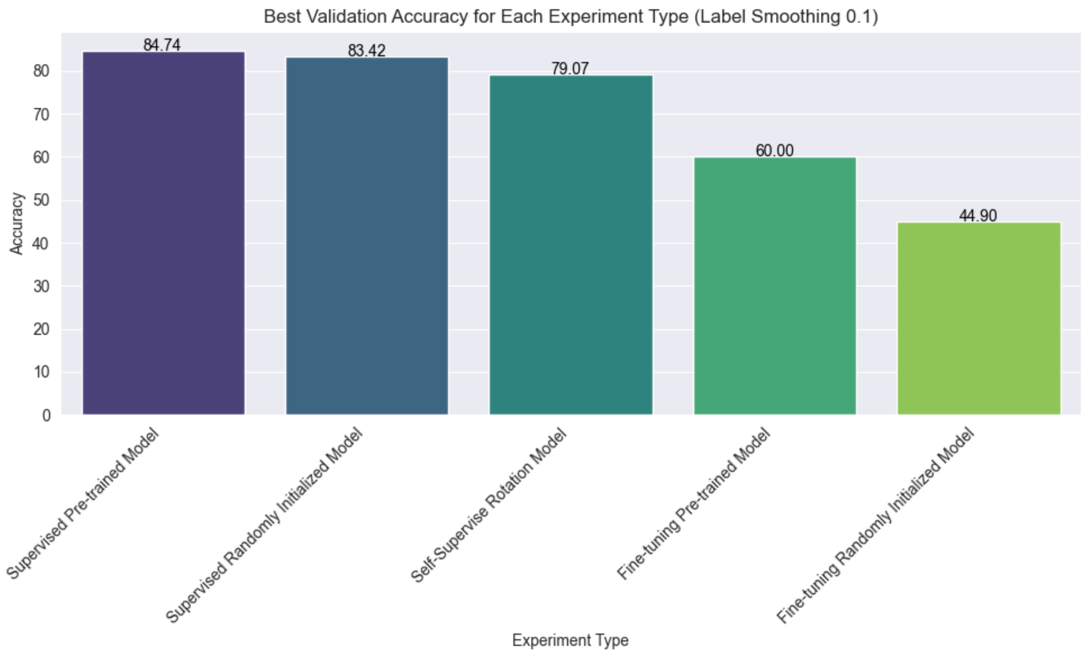
^ *Figure 3: Fine-tuning Randomly Initialized Model*



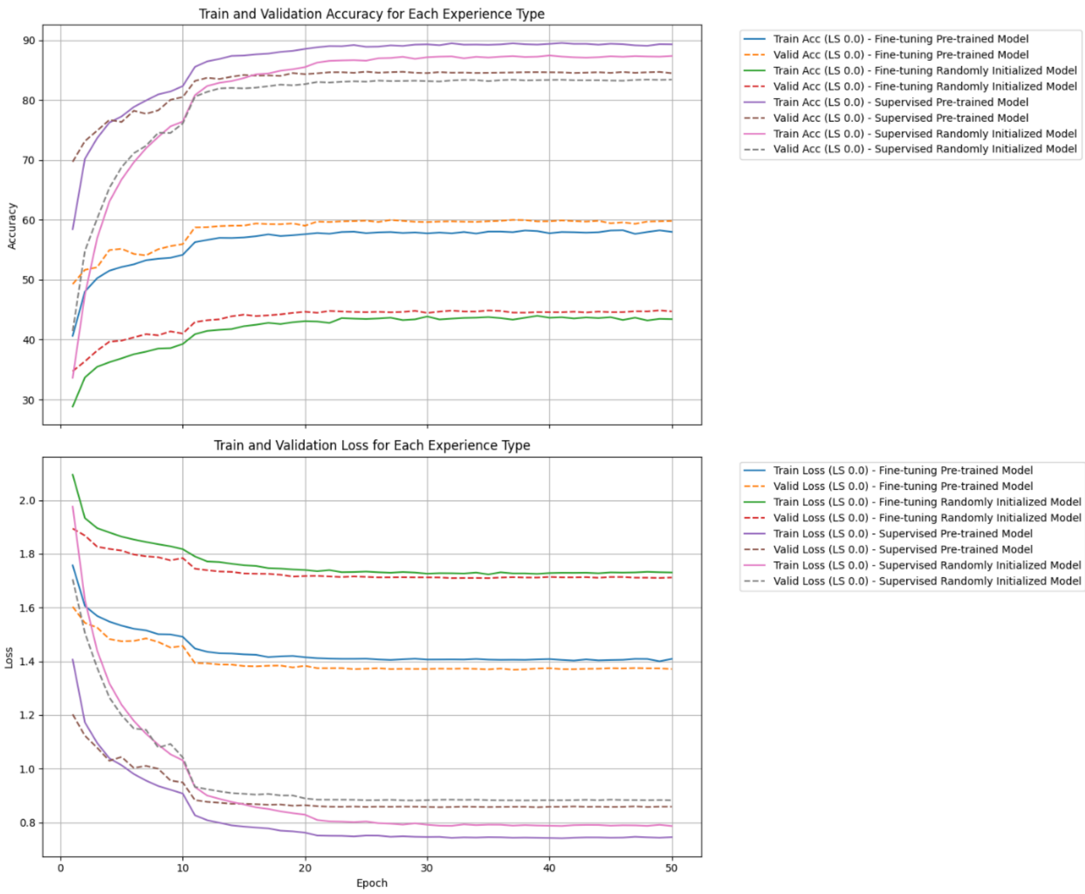
^ *Figure 4: Supervised Learning with Pre-trained Model*



^ *Figure 5: Supervised Learning with Randomly Initialized Model*



^ *Figure 6a: Comparation Between the Models*



^ Figure 6b: Comparation Between the Models

Impact of Label Smoothing on Self-Supervised Learning and Transfer Learning

1. Introduction

The purpose of this series of experiments is to evaluate the effect of label smoothing with values of 0, 0.1, and 0.2 on the performance of self-supervised learning and transfer learning approaches using a ResNet18 model.

2. Methodology

The experimental framework is divided into five stages:

1. **Self-Supervised Learning**: Train a ResNet18 model using a rotation task to learn feature representations without labeled data.
2. **Fine-tuning Pre-trained Model**: Fine-tune the model trained in Stage 1 by unfreezing the last fully connected layer and the final convolution layer.
3. **Fine-tuning Randomly Initialized Model**: Initialize a new ResNet18 model and fine-tune it similarly to Stage 2.
4. **Supervised Learning with Pre-trained Model**: Perform fully supervised training on the model obtained from Stage 1.
5. **Supervised Learning with Randomly Initialized Model**: Train a randomly initialized ResNet18 model in a fully supervised manner.

In stages 2 and 3, the models are fine-tuned, while stages 4 and 5 involve full training sessions. The pre-trained models for supervised learning stages are derived from the self-supervised learning in Stage 1.

3. Results

The experiments revealed that:

- **Label Smoothing 0.1**: This level of label smoothing achieved the best results across all stages of the experiments, indicating a favorable balance between label precision and generalization. (Figure 6)
- **Label Smoothing 0.2 vs. 0**: Although label smoothing with a factor of 0.2 did not outperform the 0.1 setting, it won in most cases against the no smoothing condition (0), suggesting that some degree of label smoothing generally aids in model performance. (Figure 6)

The experiments suggest that label smoothing improves model performance in both self-supervised learning scenarios and when transferring learned features to new tasks. Specifically, a label smoothing value of 0.1 consistently enhanced model accuracy across different stages and learning paradigms.

4. Analysis Interpretation

Label smoothing serves as a regularization technique, potentially preventing overfitting by discouraging the model from becoming too confident about its predictions. The optimal smoothing value of 0.1 could be providing a sweet spot where the model is regularized enough to generalize well without being too penalized for confident predictions.

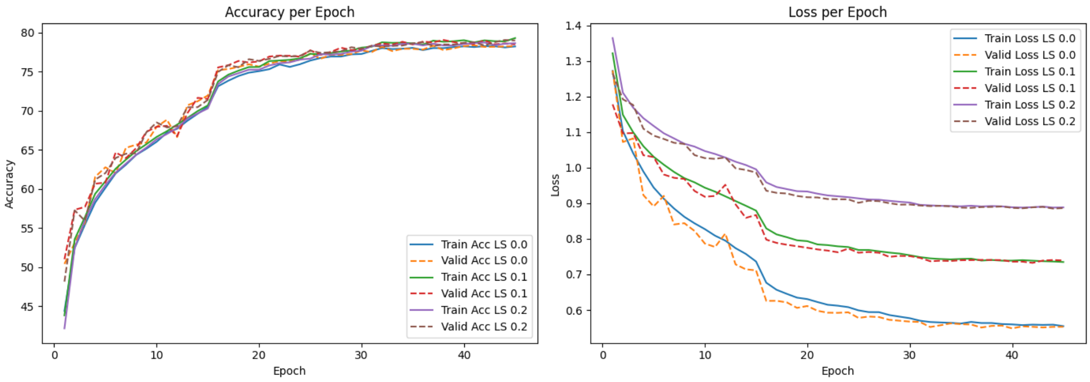
5. Conclusions

Label smoothing has proven to be a beneficial technique in enhancing the performance of self-supervised and transfer learning tasks. A smoothing value of 0.1 is recommended for similar tasks and architectures. Future experiments might explore:

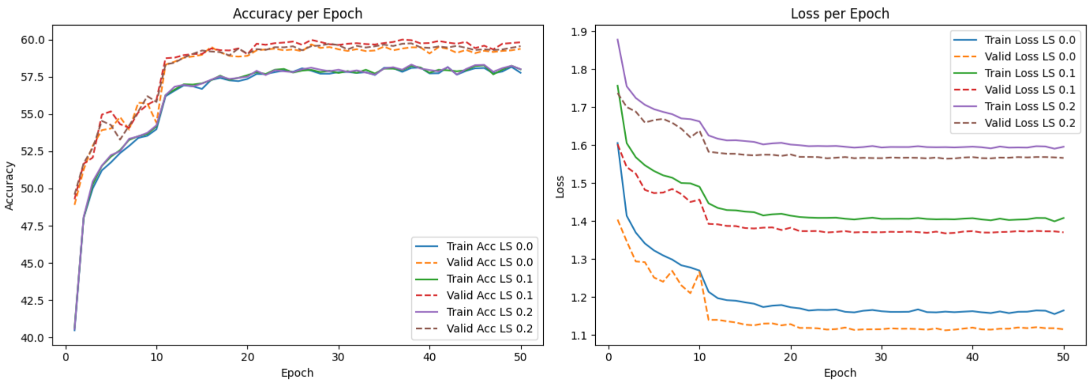
- The impact of label smoothing on different neural network architectures.
- The effects of label smoothing in larger, more complex datasets.

6. Appendices

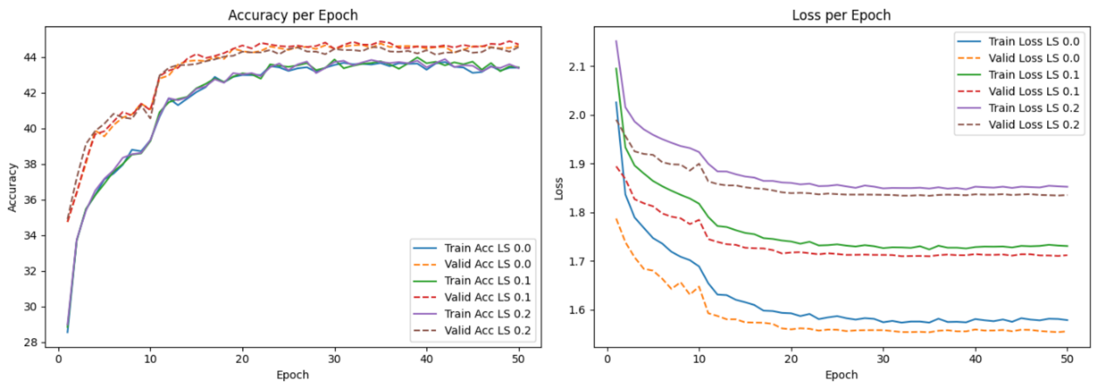
The appendices would include detailed tables and graphs of the experiment results, as well as any additional statistical analyses conducted.



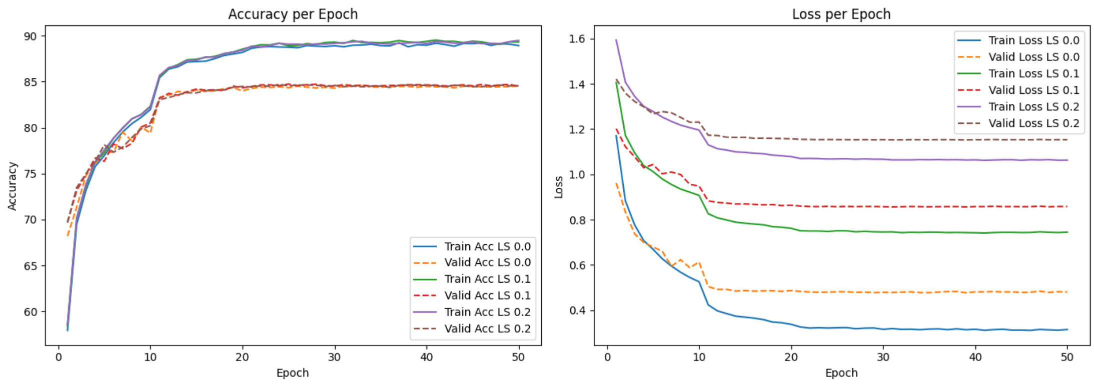
^ Figure 1: Self-Supervised Learning



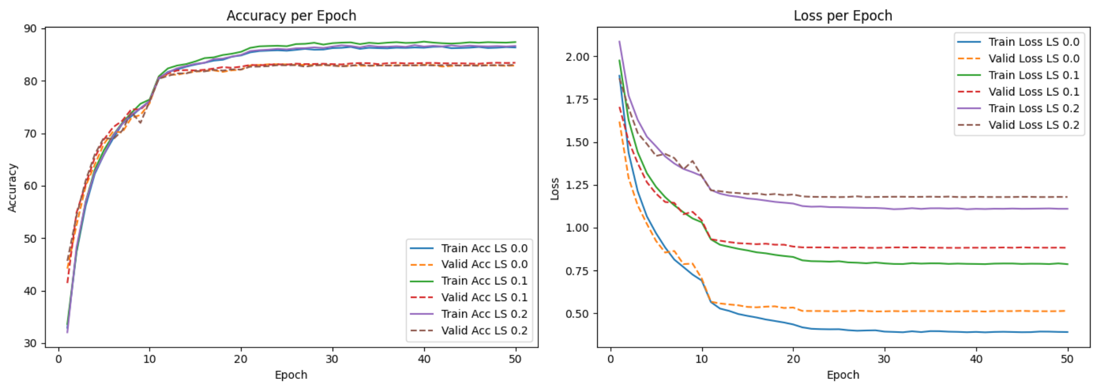
^ Figure 2: Fine-tuning Pre-trained Model



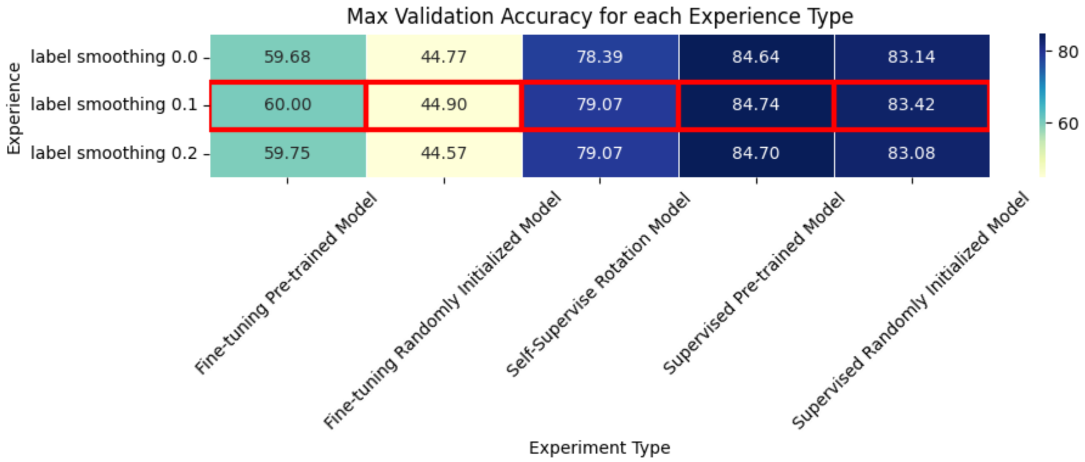
^ Figure 3: Fine-tuning Randomly Initialized Model



^ Figure 4: Supervised Learning with Pre-trained Model



^ Figure 5: Supervised Learning with Randomly Initialized Model



^ *Figure 6: Table for validation accuracy*

References

[1] OpenAI. (2023). *ChatGPT* [Large language model]. https://chat.openai.com
[2] Feng, Z., Xu, C., & Tao, D. (2019). Self-Supervised Representation Learning by Rotation Feature Decoupling. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (pp. 10364-10373)