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.

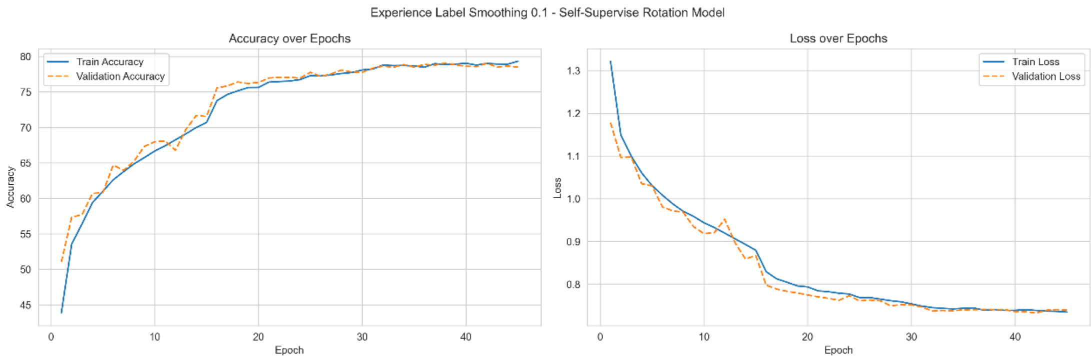
5. Conclusion

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.

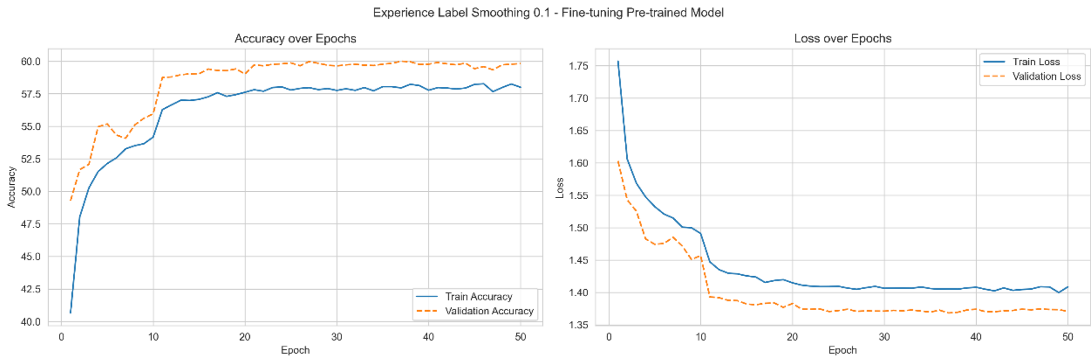
These findings underscore the value of pre-training, particularly in scenarios where labeled data is scarce or the task is complex. (Figure 6)

6. Appendices

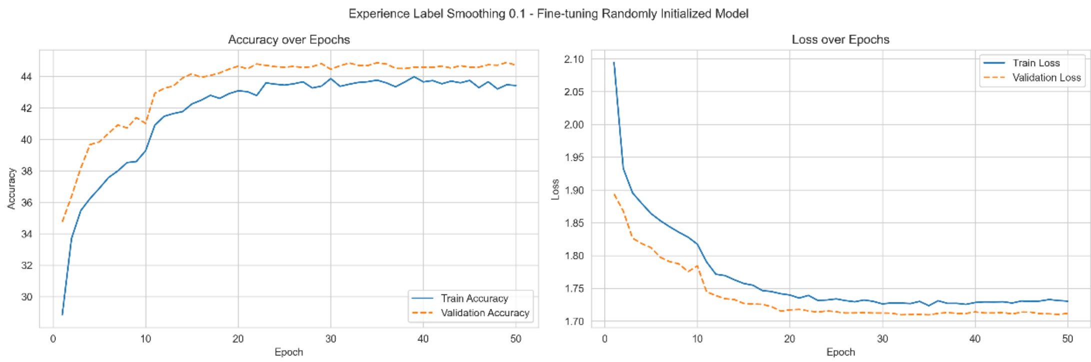
The appendices would include detailed tables and graphs of the experiment results, as well as any additional statistical analyses conducted.



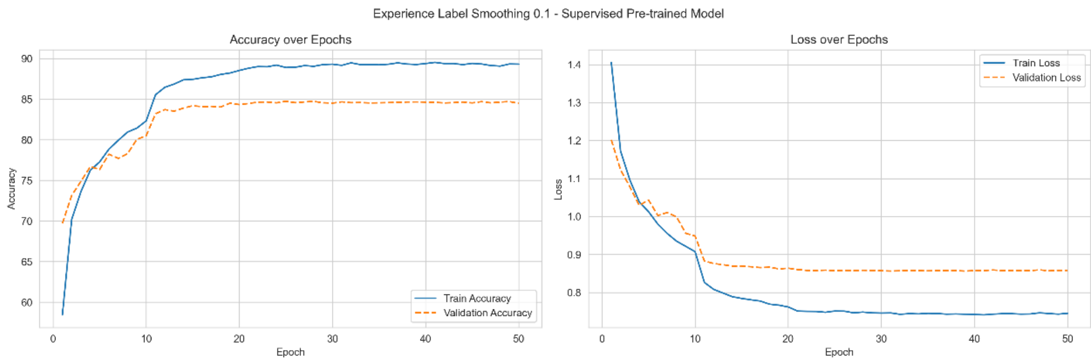
^ *Figure 1: Self-Supervised Learning*



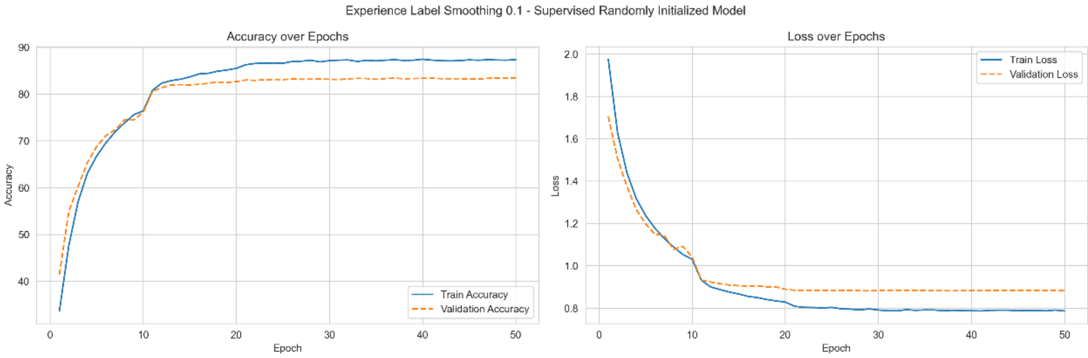
^ *Figure 2: Fine-tuning Pre-trained Model*



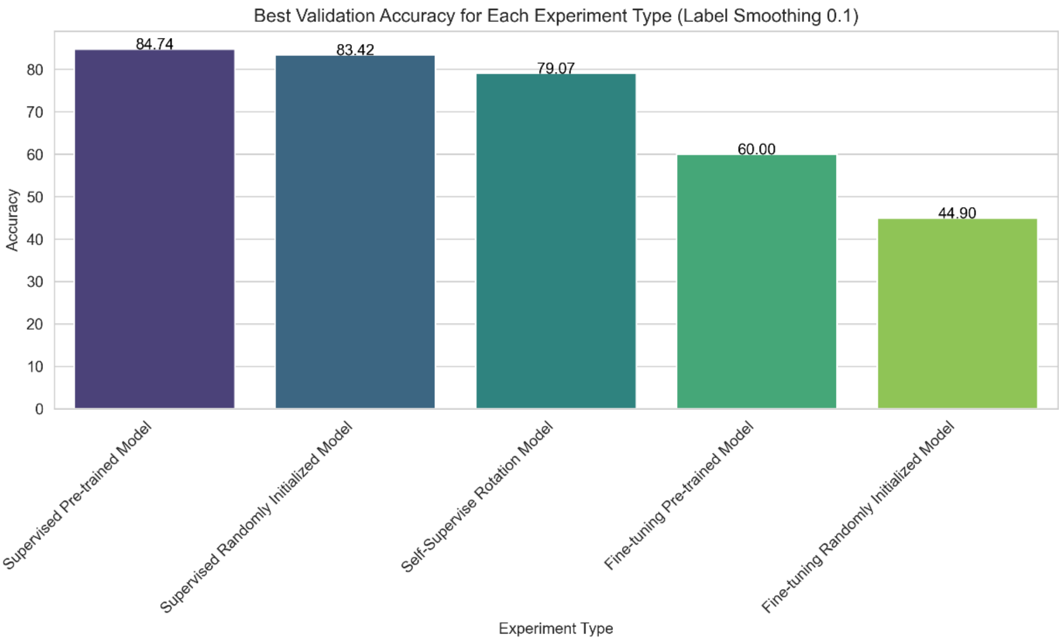
^ *Figure 3: Fine-tuning Randomly Initialized Model*



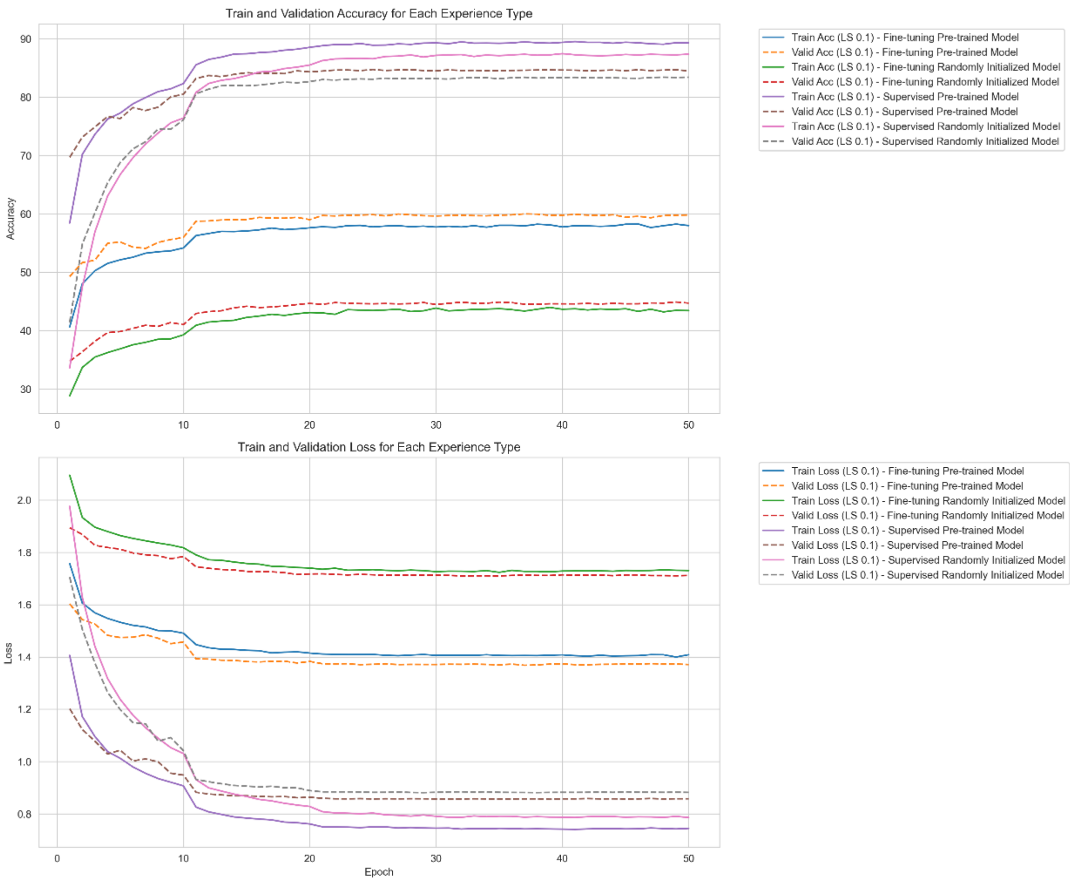
^ *Figure 4: Supervised Learning with Pre-trained Model*



^ *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 comprehensive experiments underscore the significance of label smoothing in model training across various scenarios:

- **Label Smoothing 0.1**: Consistently outperforming other values, label smoothing at 0.1 emerged as the optimal choice, striking a delicate balance between precise label representation and the ability to generalize, leading to the highest accuracy gains in both pre-trained and randomly initialized models across all experiment stages. (Figure 6a)

- **Label Smoothing 0.2 vs. 0**: While label smoothing at 0.2 did demonstrate slight improvements over the absence of smoothing (0) in models initialized with random weights, it could not achieve the performance heights of the 0.1 setting. This outcome indicates that a moderate degree of label smoothing is generally beneficial, enhancing performance in models lacking pre-trained advantages. (Figure 6b)

The findings accentuate the pivotal role of label smoothing in bolstering model performance.

A smoothing parameter set to 0.1 not only facilitated superior accuracy in self-supervised learning contexts but also proved to be more effective during the transfer of knowledge to new tasks, thereby reinforcing its efficacy as a tool for model regularization and performance optimization.

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.

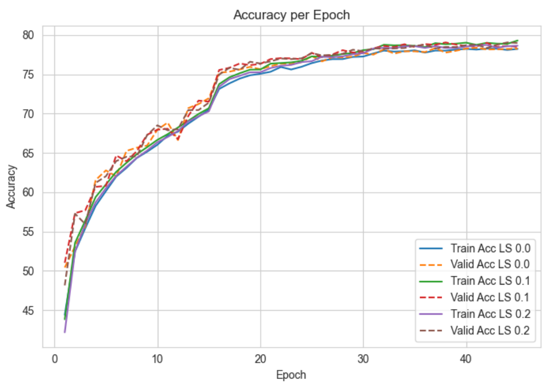
5. Conclusions

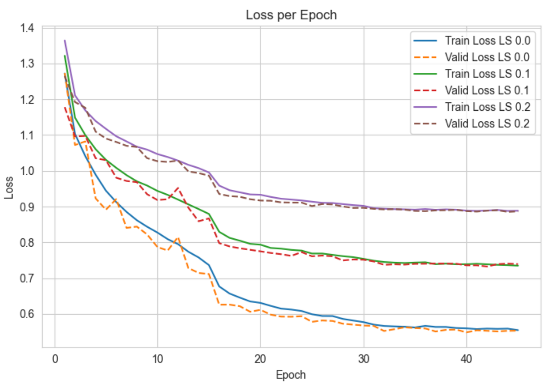
Label smoothing has proven to be a beneficial technique in enhancing the performance of self-supervised and transfer learning tasks. A smoothing value of 0.1 is recommended for similar tasks and architectures. Future experiments might explore:

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

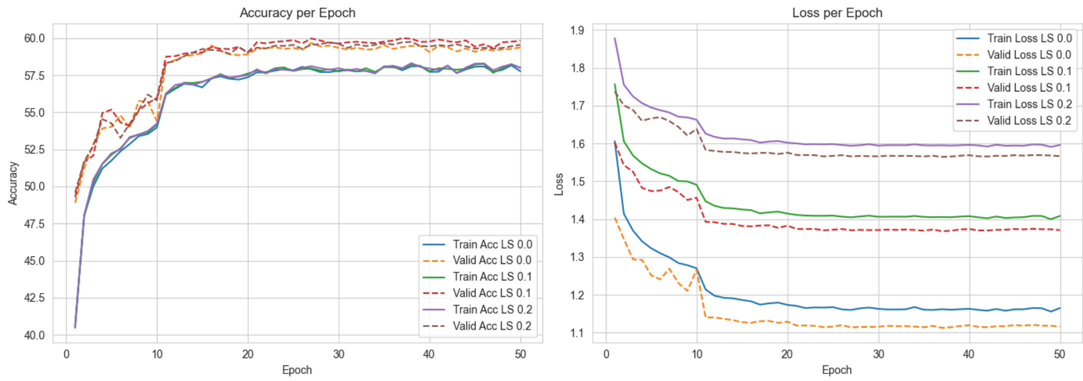
6. Appendices

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

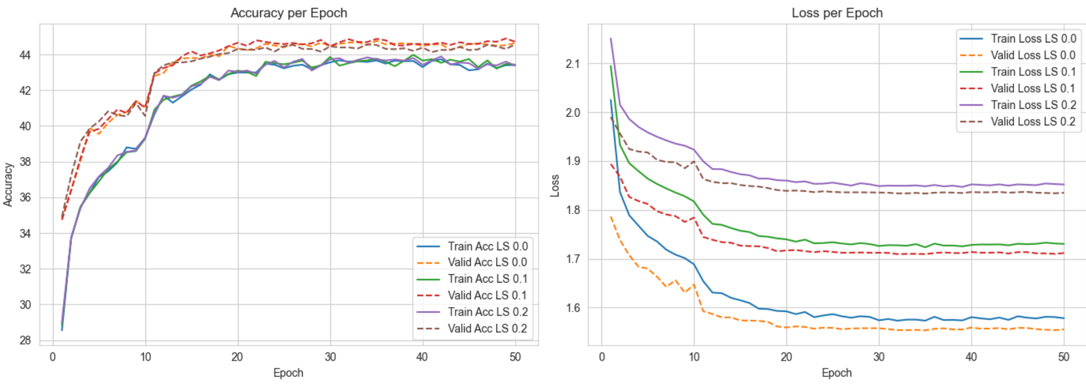




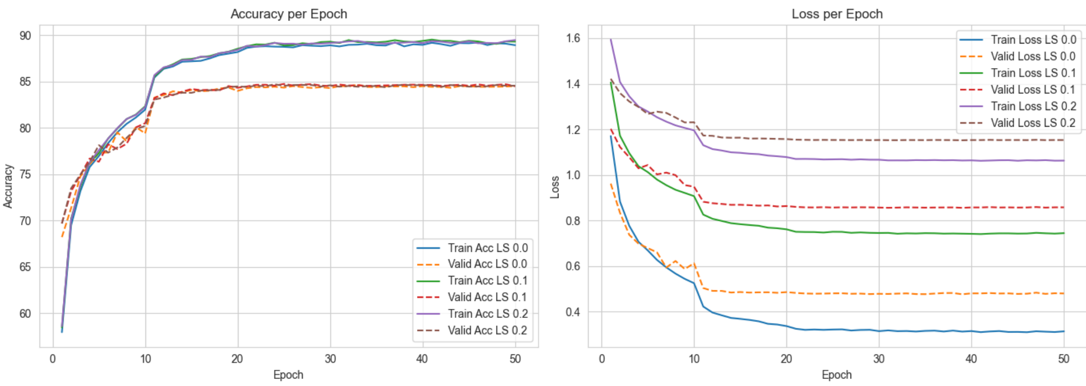
^ *Figure 1: Self-Supervised Learning*



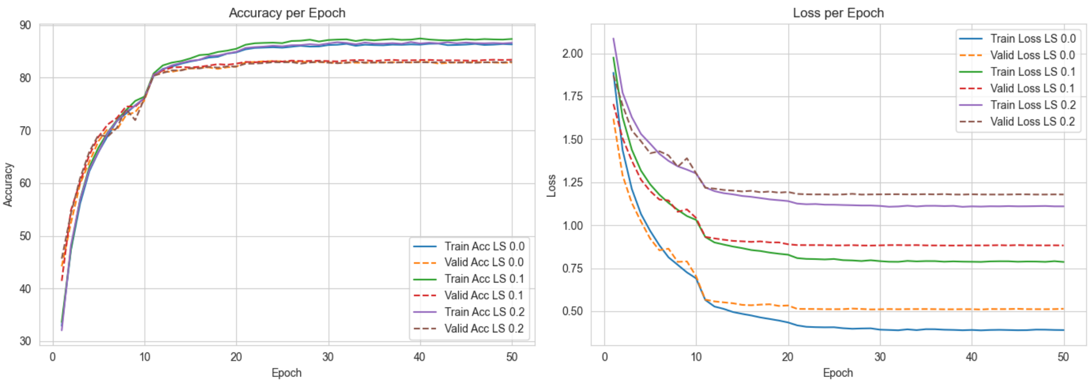
^ *Figure 2: Fine-tuning Pre-trained Model*



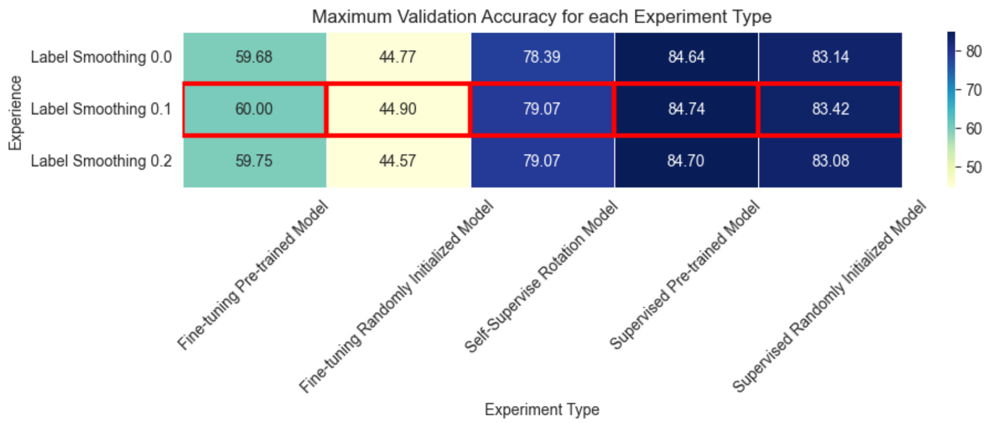
^ *Figure 3: Fine-tuning Randomly Initialized Model*



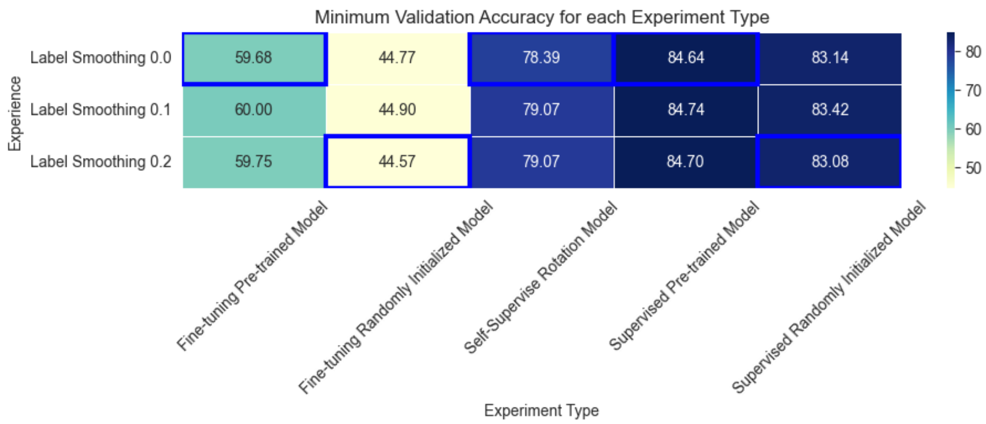
^ *Figure 4: Supervised Learning with Pre-trained Model*



^ Figure 5: Supervised Learning with Randomly Initialized Model



^ Figure 6a: Table for validation accuracy



^ Figure 6b: Table for validation accuracy

Impact of Rotation Counts in Self-Supervised Pre-Training on Downstream Task Performance

1. Introduction

This report presents an analysis of self-supervised learning with a focus on the impact of different rotation counts used during the pre-training phase.

The objective is to evaluate how the granularity of self-supervised tasks affects the performance of a neural network model when fine-tuning for a downstream task.

The rotations considered were 4, 6, and 8 directions, providing a gradient of complexity in self-supervised learning.

2. Methodology

The methodology involved pre-training a ResNet18 model on a self-supervised rotation prediction task with three different rotation granularities: 4, 6, and 8 directions.

Following the self-supervised pre-training, two downstream tasks were performed:

- Fine-tuning the pre-trained model on the CIFAR10 classification task.
- Training the same architecture from scratch (random initialization) on CIFAR10 for comparison.

Each model's performance was evaluated based on test accuracy, providing insights into the efficacy of self-supervised learning with varying rotation counts.

3. Results

The test accuracy results for each experiment type and rotation count were as follows:

- **Self-Supervised Rotation Model**: The highest accuracy was achieved with 6 rotations (85.17%), followed by 4 rotations (79.18%) and 8 rotations (79.87%). (Figure 1)
- **Fine-tuning Pre-trained Model**: The 4 rotations scenario led to the highest accuracy (63.38%), with 8 rotations trailing closely behind (60.46%), and 6 rotations showing a noticeable drop (55.77%). (Figure 2)
- **Fine-tuning Randomly Initialized Model**: All three rotation counts resulted in similar accuracies, hovering around 46%. (Figure 3)
- **Supervised Pre-trained Model**: The models pre-trained with 4 and 8 rotations achieved similar accuracies (84.64% and 84.58%, respectively), while 6 rotations resulted in a slightly lower accuracy

(84.05%). (Figure 4)

- **Supervised Randomly Initialized Model**: All three rotation counts resulted in similar accuracies, hovering around 83%. (Figure 5)

4. Analysis Interpretation

The varying rotation counts in the self-supervised pre-training phase exhibited distinct impacts on the model's performance during fine-tuning.

A higher rotation count did not necessarily translate to better performance, as seen with the 6 rotations count, which, despite having the highest accuracy in the self-supervised task, did not perform as well in the fine-tuning stage.

The results suggest that the complexity of the self-supervised task needs to be carefully balanced to ensure that the learned features are beneficial and generalizable to the downstream task.

5. Conclusions

The investigation into the effects of rotation count in self-supervised learning revealed that more complex self-supervised tasks do not always yield better downstream performance.

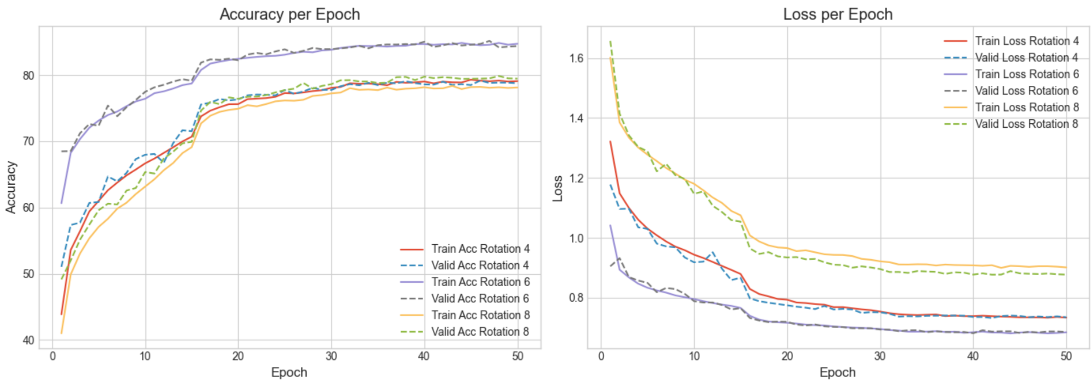
Specifically, the 4 rotations count provided the most effective transfer learning for fine-tuning, indicating that an optimal level of task difficulty might exist for self-supervised pre-training.

These findings underscore the importance of task design in self-supervised learning and its subsequent influence on supervised tasks.

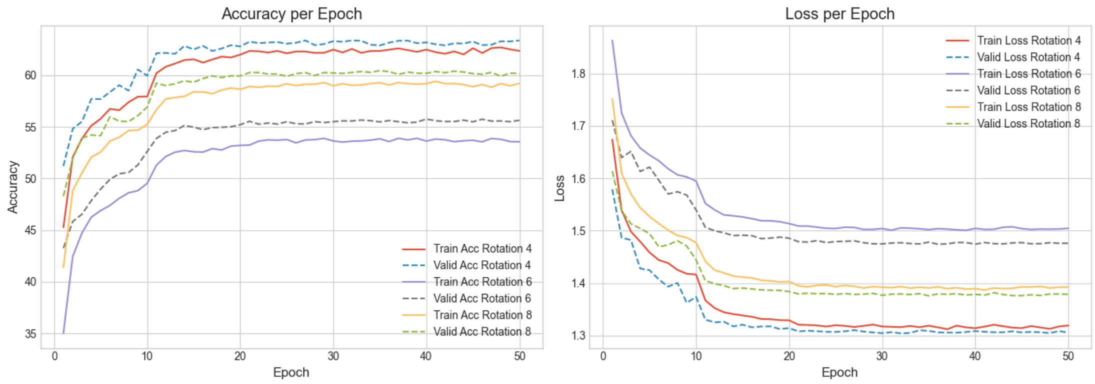
6. Appendices

The appendices would include detailed figures and tables depicting the accuracy results for each experiment type and rotation count.

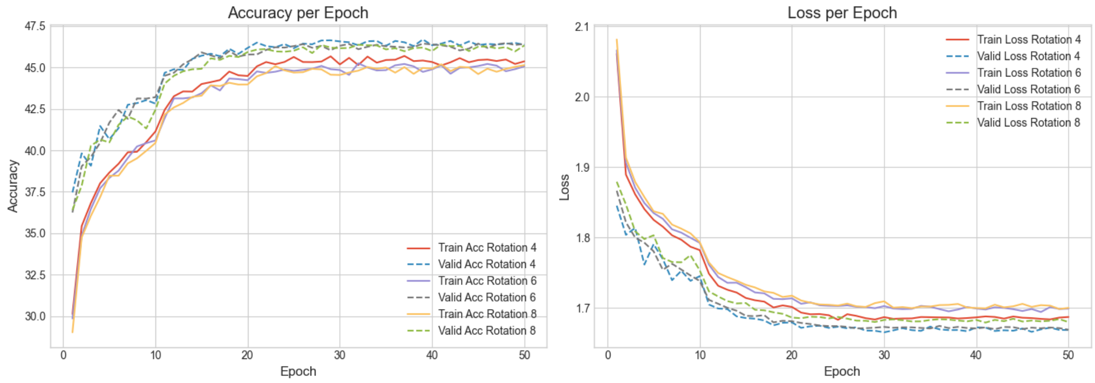
Additional analyses, such as the per-class accuracies, would provide deeper insights into the model's performance.



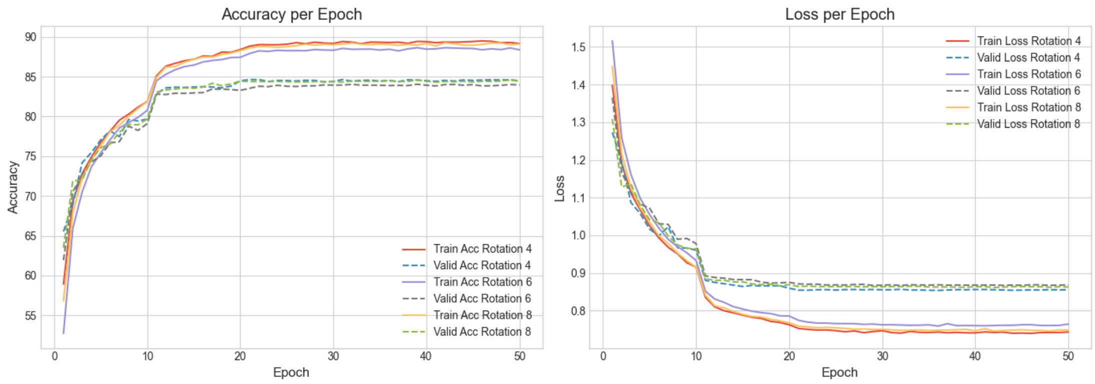
^ *Figure 1: Self-Supervised Learning*



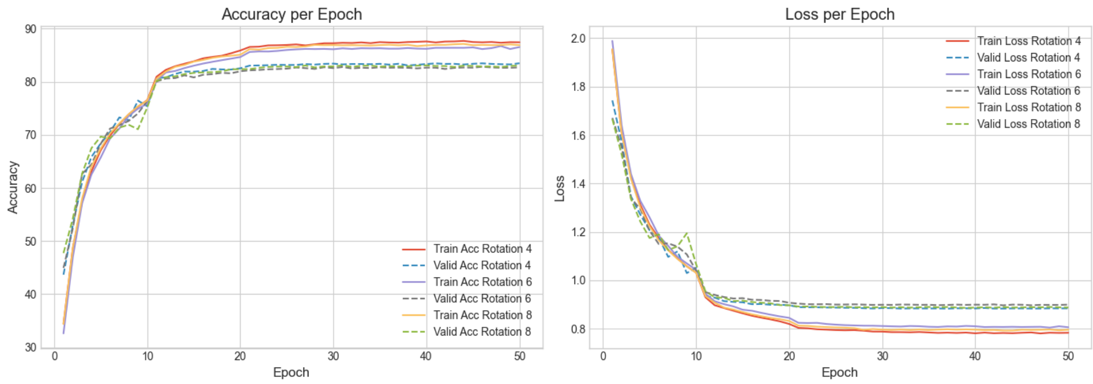
^ *Figure 2: Fine-tuning Pre-trained Model*



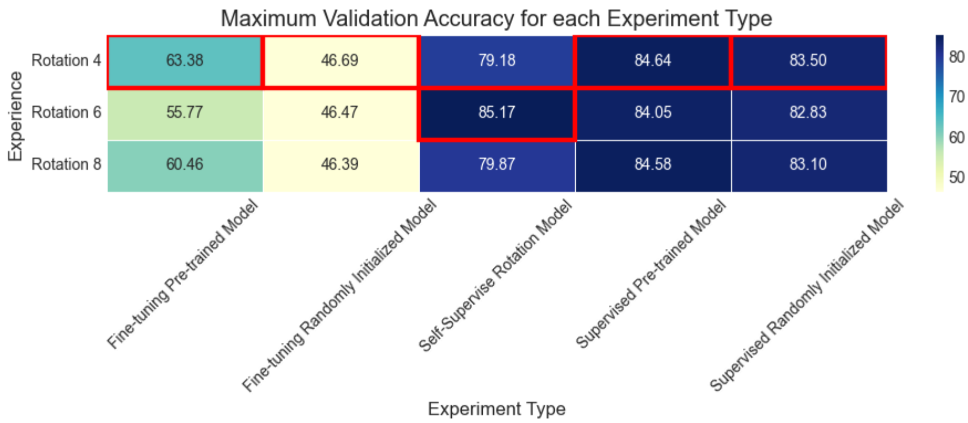
^ *Figure 3: Fine-tuning Randomly Initialized Model*



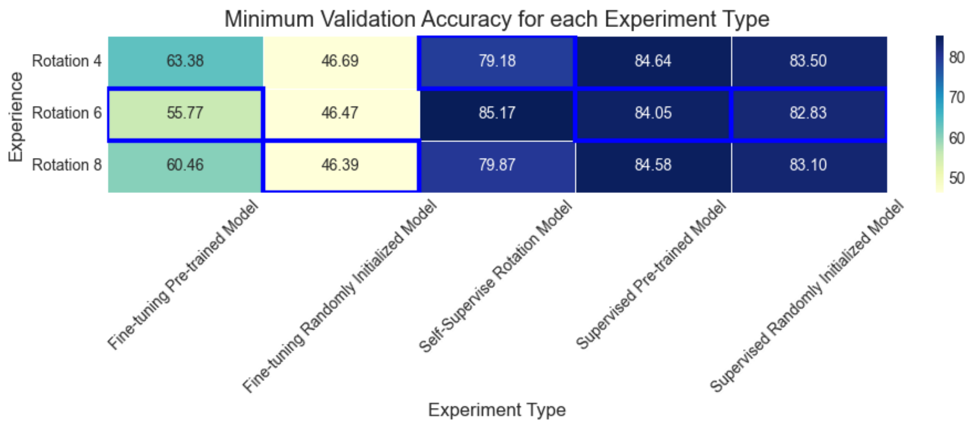
^ *Figure 4: Supervised Learning with Pre-trained Model*



^ Figure 5: Supervised Learning with Randomly Initialized Model



^ Figure 6a: Table for validation accuracy



^ Figure 6b: Table for validation accuracy

Impact of Classification with Limited Labelled Data

1. Introduction

This report presents a comparative analysis of CIFAR10 classification performance using a self-supervised learning model with fine-tuning (specifically, fine-tuned fully connected and layer 4 convolution blocks) versus a fully supervised model initialized randomly. The central focus of this investigation is to understand the efficacy of self-supervised pre-training in scenarios with varying amounts of labeled data.

2. Methodology

Two distinct training approaches were employed:

- **Self-Supervised Pre-training with Fine-Tuning**: A model pre-trained on a rotation prediction task (RotNet) was fine-tuned on CIFAR10 with only the fully connected (fc) and layer 4 convolution blocks being updated.

- **Fully Supervised Training from Random Initialization**: A model was trained from scratch on CIFAR10 with all layers being updated during training.

For both approaches, the models were trained with different numbers of labeled examples per class: 20, 50, 100, 400, 700 and 1000.

3. Results

The performance of both models was evaluated in terms of validation accuracy. (Table 1) The results were plotted to show the relationship between the number of labeled examples per class and the achieved validation accuracy and validation loss. (Figure 1, Figure 2)

Samples Per Class	Fine-Tuned Max Accuracy	Fine-Tuned Min Loss	Random Init Max Accuracy	Random Init Min Loss
20	29.23	2.1380	25.58	2.4882
50	38.74	1.8387	30.34	2.0282
100	45.10	1.7096	39.54	1.8203
250	51.04	1.5691	48.73	1.6271
400	53.50	1.5198	57.61	1.4580
700	55.61	1.4670	66.59	1.2734
1000	56.60	1.4408	66.49	1.2566

Table 1: Maximum Validation Accuracy and Minimum Validation Loss

4. Analysis Interpretation

The fine-tuned model demonstrated higher validation accuracy and lower loss at lower sample sizes (20, 50, 100). However, when the sample size was increased to 400 and above, the fully supervised model started to outperform the fine-tuned model, suggesting that with adequate labeled data, the benefits of self-supervised pre-training become less pronounced. (Figure 3)

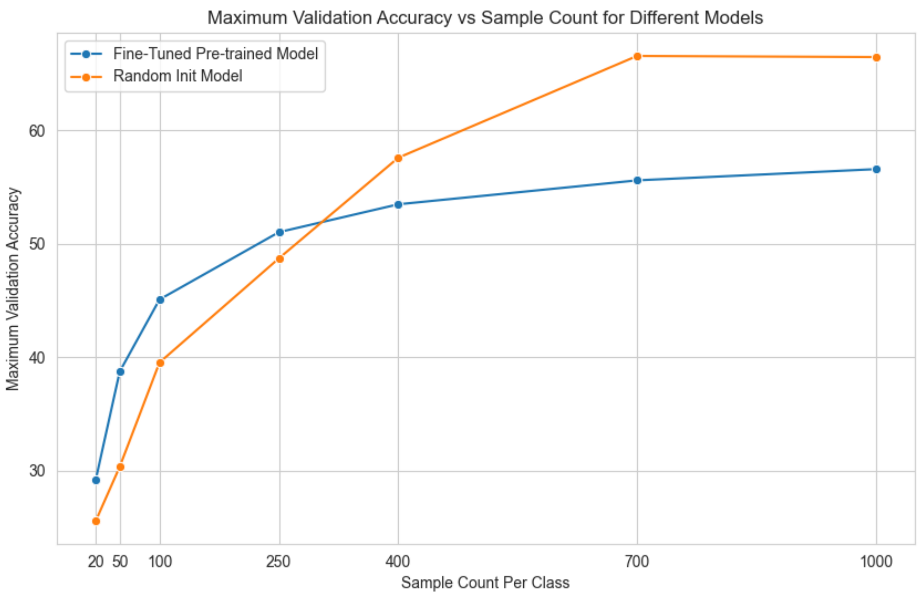
5. Conclusions

The study's findings suggest that self-supervised pre-training with fine-tuning is particularly effective when the availability of labeled data is limited. Notably, in scenarios where labeled data may be scarce or inadvertently leaked, such an approach can be leveraged to enhance model performance. In contrast, with ample labeled data, the advantages of self-supervised pre-training become less pronounced, and a fully supervised approach is shown to be more beneficial.

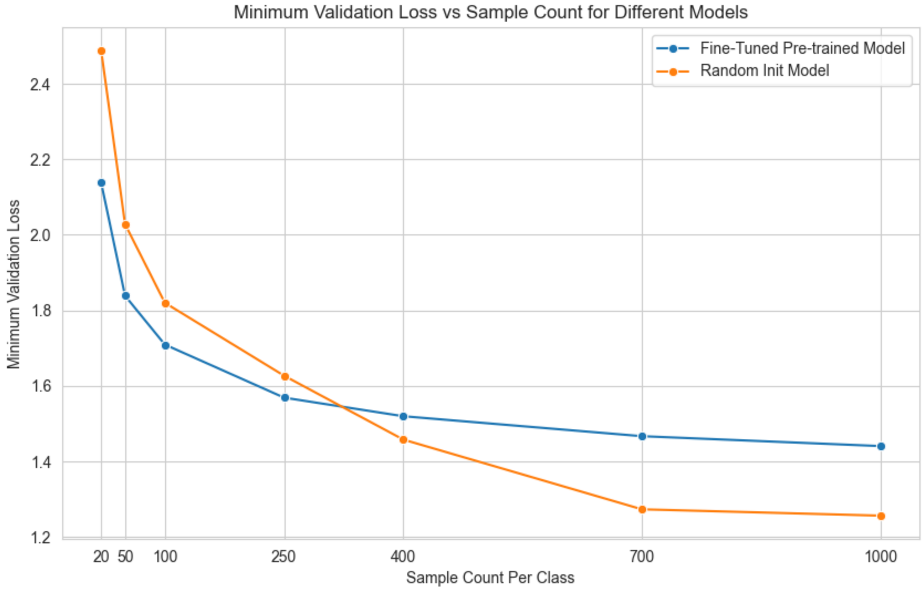
This insight is pivotal for situations where the conservation of labeled data is critical or where data labeling presents a significant cost. Implementing a self-supervised learning paradigm under these circumstances can serve as a strategic method to boost model performance and utilize available data more efficiently.

6. Appendices

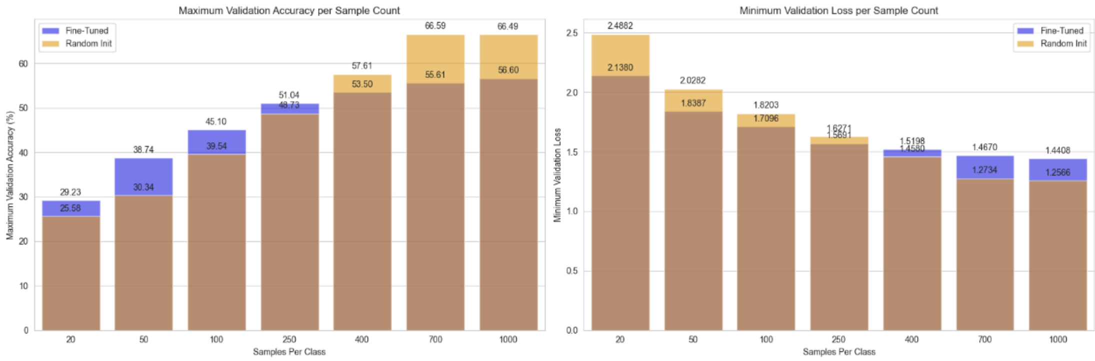
The appendices would include the generated plots that supports the findings of this report.



^ *Figure 1: Test Accuracy Plot*



^ Figure 2: Test Lost Plot



^ Figure 3: Annotated Accuracy Loss Comparison

Performance Evaluation of ResNet Models in Different Learning Contexts

1. Introduction

This report focuses on evaluating the performance of ResNet models (ResNet18, ResNet34, and ResNet50) specifically within the realm of self-supervised learning.

Outline the scope of the evaluation: comparing different ResNet architectures in a self-supervised learning setup to understand their capabilities and limitations in this specific context.

2. Methodology

- **Initial Training with Rotation Task**: The study begins by training the ResNet models (ResNet18, ResNet34, and ResNet50) using a self-supervised learning approach with a rotation task.

- **Subsequent Evaluation in Four Tasks**:

1. Fine-tuning Pre-trained Model: This task assesses the models' fine-tuning capabilities on specific tasks after self-supervised pre-training.

2. Fine-tuning Randomly Initialized Model: Evaluates the models when fine-tuned from a randomly initialized state, without the benefit of pre-training.

3. Supervised Pre-trained Model: Analyzes the models' performance in a supervised setting, leveraging their initial training from the rotation task.

4. Supervised Randomly Initialized Model: Examines how the models perform in a traditional supervised learning scenario from a randomly initialized state.

- **Metrics for Evaluation**: Validation accuracy and loss are the primary metrics for evaluating model performance. These metrics provide insights into the models' ability to generalize to new data and their overall effectiveness in learning from the tasks.

3. Results

The performance of both models was evaluated in terms of validation accuracy. (Table 1) The results were plotted to show the relationship between the different model and the achieved validation accuracy and validation loss. (Figure 1 ~ 6)

Model Type	Resnet18	Resnet34	Resnet50
Experience Type			
Self-Supervise Rotation Model	79.65	77.84	78.80
Fine-tuning Pre-trained Model	61.27	53.03	57.96
Fine-tuning Randomly Initialized Model	46.05	35.31	31.61
Supervised Pre-trained Model	84.70	84.00	84.32
Supervised Randomly Initialized Model	83.06	82.57	80.52

Table 1: Highest Validation Accuracy Across Models and Experience Types

4. Analysis and Interpretation

- **Self-Supervised Rotation Model**: In the self-supervised rotation task, ResNet18 achieved the highest validation accuracy (79.65%), slightly outperforming ResNet50 (78.80%) and ResNet34 (77.84%). This suggests that for self-supervised tasks involving image rotation, the complexity of larger models like ResNet50 might not significantly enhance performance over smaller models like ResNet18.

- **Fine-Tuning Tasks**: When it comes to fine-tuning pre-trained models, all models exhibited a drop in performance compared to the self-supervised and supervised tasks, with ResNet18 (61.27%) still maintaining a lead over ResNet34 (53.03%) and ResNet50 (57.96%). This indicates that while pre-trained models are beneficial, the nature of the fine-tuning task and the dataset specificity can impact performance.

- In fine-tuning from a randomly initialized state, there was a further decline in performance for all models, most notably for ResNet50 (31.61%). This decline underscores the challenges models face when adapting to new tasks without prior training or transfer learning benefits.

- **Supervised Learning with Pre-Trained and Randomly Initialized Models**: In supervised scenarios, both pre-trained and randomly initialized models showed strong performance, with pre-trained models slightly outperforming their randomly initialized counterparts. This again highlights the advantage of transfer learning and the efficacy of pre-training in supervised tasks.

  Interestingly, the gap between the models was narrower in the supervised setting, particularly in the randomly initialized models.

This suggests that the inherent capabilities of the models are more evenly matched when they are trained from scratch in a supervised context.

- **Comparative Insights**: Across all tasks, ResNet18 consistently demonstrated strong performance, challenging the assumption that larger, more complex models always yield better results. This finding suggests that for certain tasks, especially those involving self-supervised learning or specific fine-tuning, smaller models can be equally or more effective.

5. Conclusions

- **Effectiveness of Self-Supervised Learning**: The study highlights the effectiveness of self-supervised learning, especially in the context of the rotation task. ResNet18, with its simpler architecture, demonstrated comparable or even superior performance to its more complex counterparts, challenging the notion that increased model complexity always correlates with better performance in self-supervised tasks.

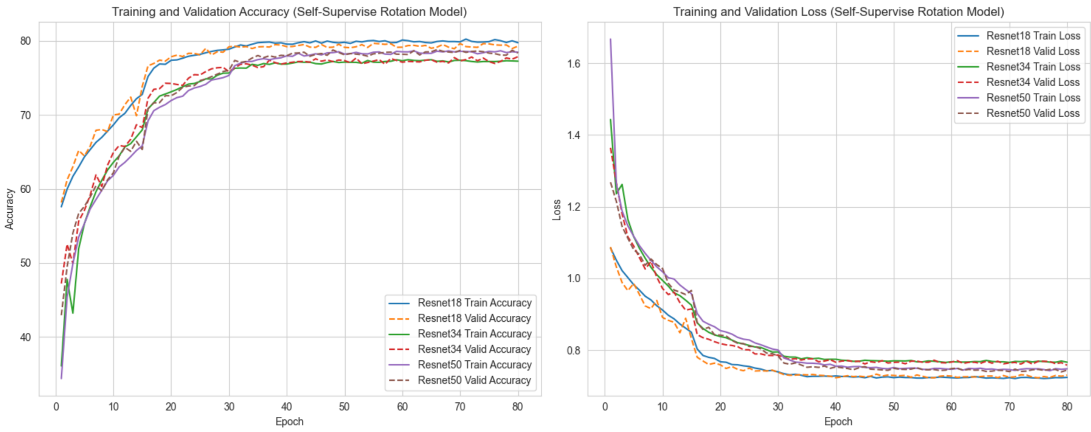
- **Importance of Model Selection**: The results underscore the importance of careful model selection based on the specific requirements of a task. In scenarios where resources are limited or where tasks do not benefit significantly from more complex models, simpler architectures like ResNet18 can be more suitable and efficient.

- **Role of Pre-Training in Model Performance**: Pre-training has a significant impact on model performance, as evidenced by the generally higher validation accuracies in both supervised and fine-tuned pre-trained models compared to their randomly initialized counterparts. This emphasizes the value of transfer learning and the utility of pre-trained models in diverse tasks.

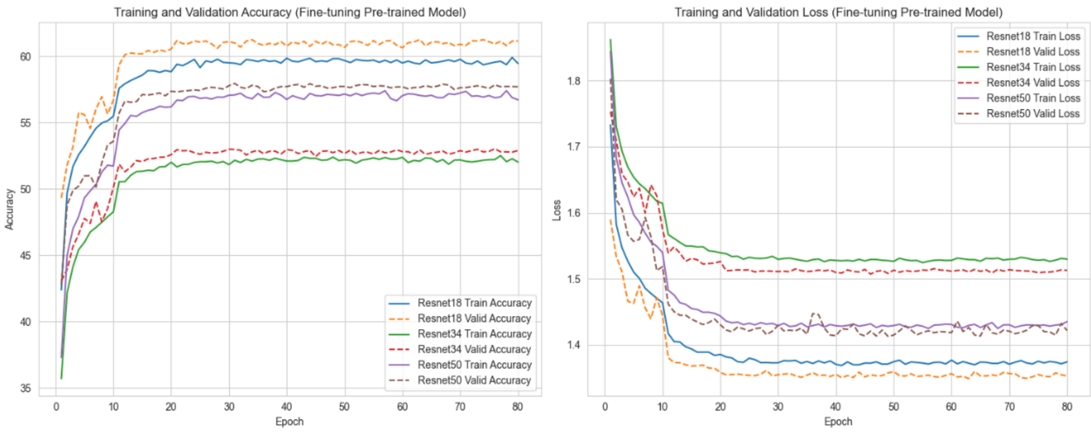
- **Narrower Performance Gap in Supervised Learning**: In supervised learning scenarios, the performance gap between the models was less pronounced, particularly when models were trained from scratch. This suggests that in fully supervised contexts, the intrinsic learning capabilities of the models are more evenly matched, regardless of their complexity.

6. Appendices

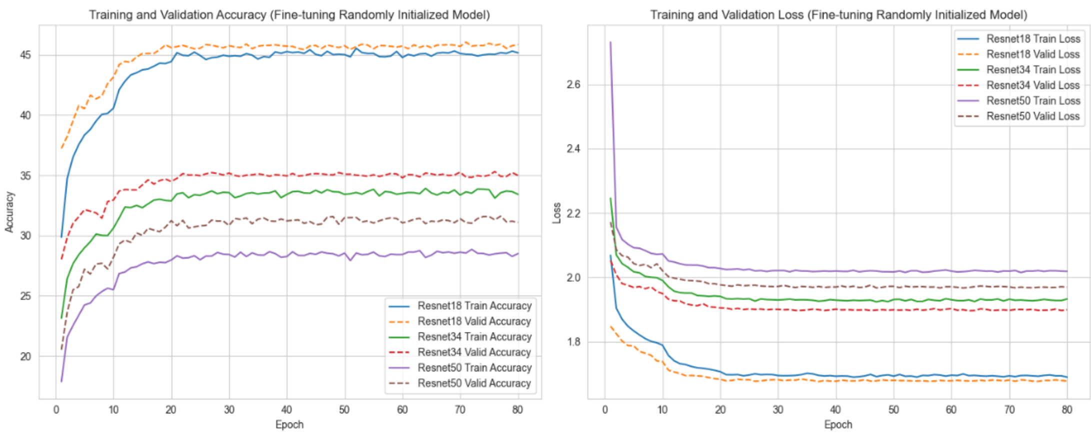
The appendices would include the generated plots that supports the findings of this report.



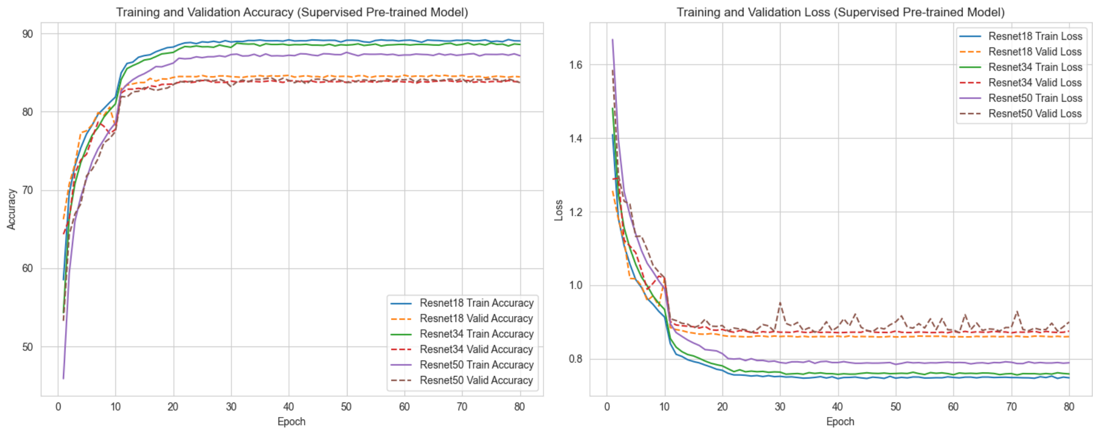
^ Figure 1: Self-Supervised Learning



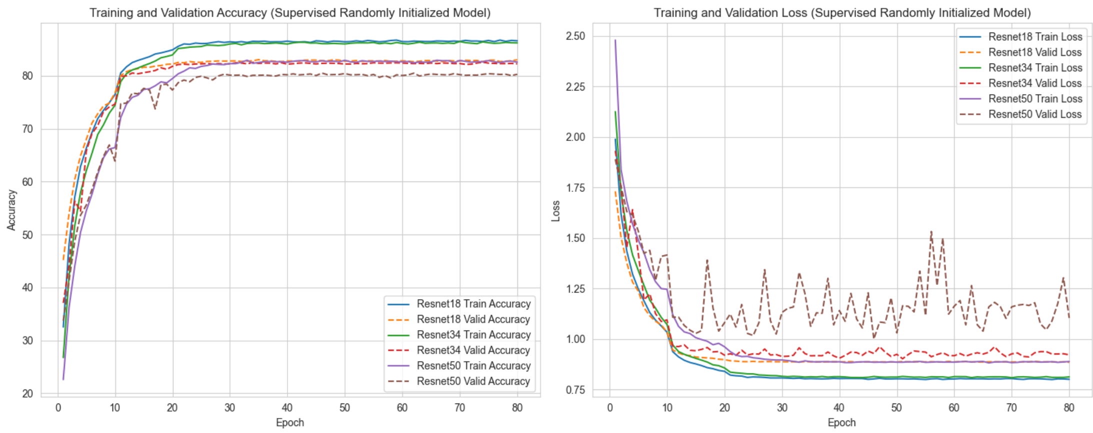
^ Figure 2: Fine-tuning Pre-trained Model



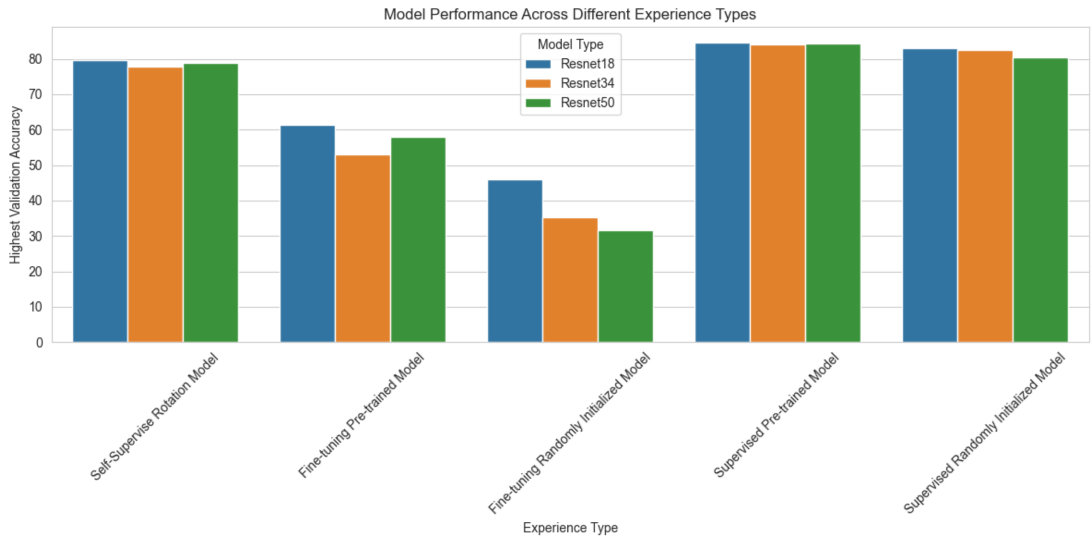
^ Figure 3: Fine-tuning Randomly Initialized Model



^ Figure 4: Supervised Learning with Pre-trained Model



^ Figure 5: Supervised Learning with Randomly Initialized Model



^ Figure 6: Models Accuracy Loss Comparison

Evaluating Self-Supervised and Supervised Learning on ImageNette with ResNet Architectures

1. Introduction

- **Overview of ImageNette Dataset**: ImageNette, a strategically curated subset of the larger ImageNet dataset, serves as the focus of our experiments.

  This subset comprises 10 classes, each chosen for their ease of classification. The dataset is partitioned into 9,469 training images and 3,925 validation images. This division allows for comprehensive training and robust validation of the models.

  The classes encompass a diverse range of objects, including tench, English springer, cassette player, chain saw, church, French horn, garbage truck, gas pump, golf ball, and parachute, providing a balanced mix of natural and man-made objects.

- **Purpose of Experiments**: The primary objective of our experiments is to assess the efficacy of self-supervised learning methods, specifically those utilizing rotation-based tasks, when applied to a complex dataset like ImageNette.

  This experiment aims to determine how well these representations, derived from self-supervised pre-training, adapt to the classification tasks in ImageNette and how they compare with traditional supervised methods.

2. Methodology

Self-Supervised Training Phase: In the initial phase of our experiment, I employed self-supervised learning with a focus on rotation-based tasks. Notably, we ensured a 0% noise label rate, meaning all training labels accurately reflected the content of the images.

Training Parameters

1. **Batch Size**: Set at 16, this parameter determines the number of training samples processed before the model's internal parameters are updated.

2. **Learning Rate**: A learning rate of 0.001 was chosen to balance the speed of convergence with the risk of overshooting minimal loss points.

3. **Label Smoothing**: Implemented at a rate of 0.1, label smoothing helps in regularizing the model, mitigating overconfidence in its predictions.

4. **Optimizer**: Adam optimizer was

used for its efficiency in handling sparse gradients and adapting the learning rate during training.

Detailed Description of Each Experiment:

- **Fine-Tuning Final 2 Layers of Pre-Trained Model**: This experiment involved fine-tuning a pre-trained model by unfreezing and training the fully connected (fc) layer and the last convolutional layer, allowing these layers to adapt to the specific features of the ImageNette dataset.

- **Fine-Tuning on a Randomly Initialized Model**: Following the same procedure as the previous experiment, except starting with a model that has randomly initialized weights, to evaluate how well a non-pre-trained model adapts to the dataset after fine-tuning.

- **Full Supervised Training on the Pre-Trained Model**: This approach involves training the entire pre-trained model, updating gradients across all layers, to leverage the pre-learned features while allowing complete adaptability to the new dataset.

- **Full Supervised Training on a Randomly Initialized Model**: Unlike the previous methods, this experiment involves training a completely randomly initialized model in a fully supervised manner, providing a baseline to evaluate the effectiveness of pre-training and fine-tuning strategies.

3. Results

The performance of both models was evaluated in terms of validation accuracy. (Table 1) The results were plotted to show the relationship between the different model and the achieved validation accuracy and validation loss. (Figure 1 ~ 6)

Model Type Experience Type	Resnet18	Resnet34	Resnet50
Self-Supervise Rotation Model	61.32	58.42	63.21
Fine-tuning Pre-trained Model	72.59	62.24	66.83
Fine-tuning Randomly Initialized Model	75.41	66.98	67.72
Supervised Pre-trained Model	86.29	84.64	82.37
Supervised Randomly Initialized Model	86.34	85.30	84.05

Table 1: Highest Validation Accuracy Across Models and Experience Types In ImageNette Dataset

4. Analysis and Interpretation

- **Learning Curve Analysis:** The self-supervised learning phase exhibited unstable learning curves, particularly in validation accuracy, which oscillated between 35% and 55%. This variability suggests a disparity between the features learned during training and those necessary for effective validation performance. Several factors could contribute to this phenomenon.

  - **Feature Misalignment**: The features learned during the self-supervised rotation task may not align well with the discriminative features needed for classifying the ImageNette dataset. If the rotation task encourages the model to focus on features that do not generalize well to the validation set, the model's performance could suffer.

  - **Overfitting to Training Data**: The model may be overfitting to the training data, learning features that are too specific to the training set and not applicable to the validation set. This would result in high training accuracy but poor validation accuracy.

- **Implications for Self-Supervised Learning:** The unpredictable validation accuracy in the self-supervised phase raises questions about the utility and robustness of self-supervised pre-training for this dataset. It implies that while self-supervised learning can provide a solid foundation, it must be carefully tailored to ensure that the learned features are beneficial for the downstream task.

- **Self-Supervised Learning vs. Random Initialization:** Contrary to what might be expected, the performance of the randomly initialized ResNet18 model surpassed that of the self-supervised pre-trained models. This raises questions about the nature of the representations learned during the self-supervised phase and their relevance to the ImageNette dataset.

  The rotation prediction task might have led the model to learn features that were not discriminative enough for the classification task at hand or were too specific to the self-supervised learning context.

  The pre-trained models could also have suffered from a form of representation bias, where the features learned were overly specialized to the pre-text task, limiting their utility for the subsequent classification task. In contrast, the randomly initialized model would not have this limitation and could learn directly and solely from the labeled data, possibly leading to more relevant feature representations for the specific task of classifying ImageNette images.

- **Supervised Learning on Random Initialization:** The results indicate a noteworthy performance of the randomly initialized model when fully trained in a supervised manner. This could suggest that for certain datasets like ImageNette, starting from scratch—without any pre-learned biases or representations—allows the model to adapt more closely to the unique characteristics of the dataset.

  It also implies that the initialization of the network can play a significant role in the performance, and in some cases, starting with a clean slate might be advantageous.

5. Conclusions

The experiments on the ImageNette dataset reveal that self-supervised learning, despite its potential, can yield inconsistent results if the pre-text task does not effectively translate to the target task.

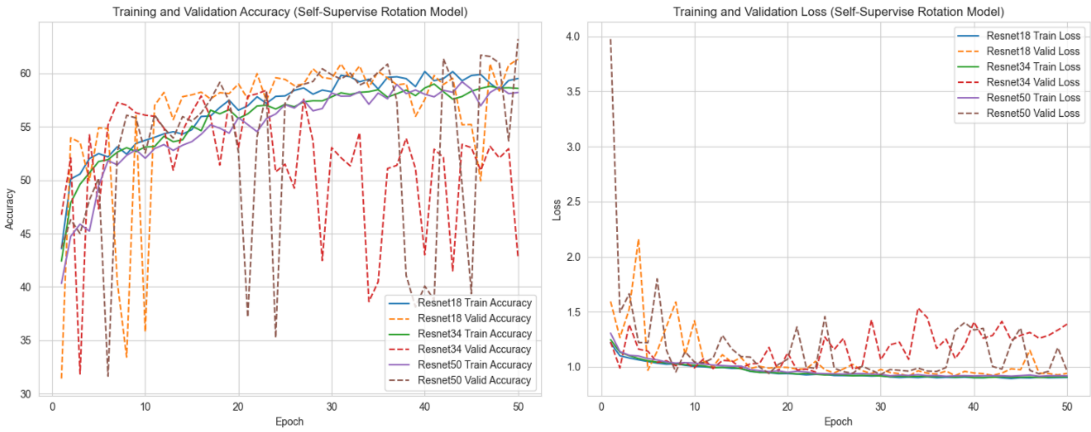
The observed erratic validation accuracy underscores the necessity for careful design of self-supervised tasks, especially when the end goal is to perform classification on complex datasets.

The superior performance of the randomly initialized model in the supervised learning context suggests that, at least for this dataset, starting with a clean slate may be more beneficial.

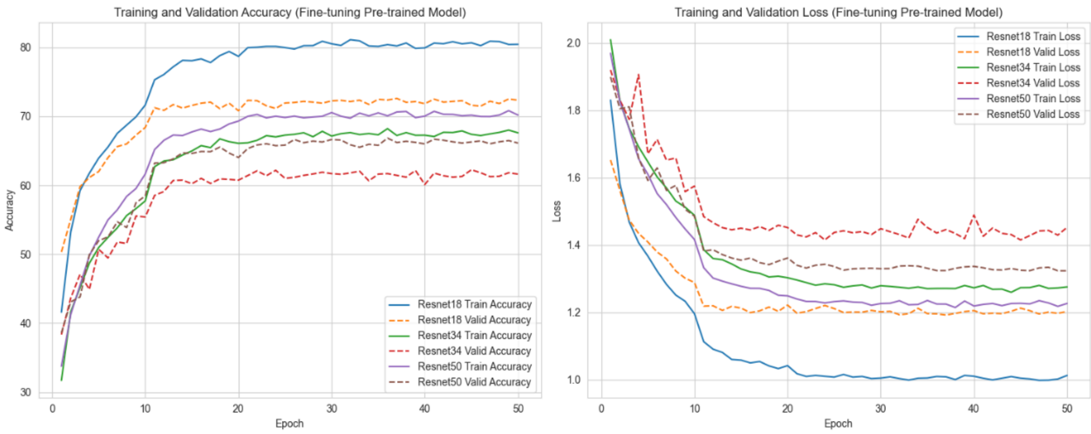
These insights pave the way for future explorations into optimizing self-supervised tasks to achieve better alignment with specific classification goals.

6. Appendices

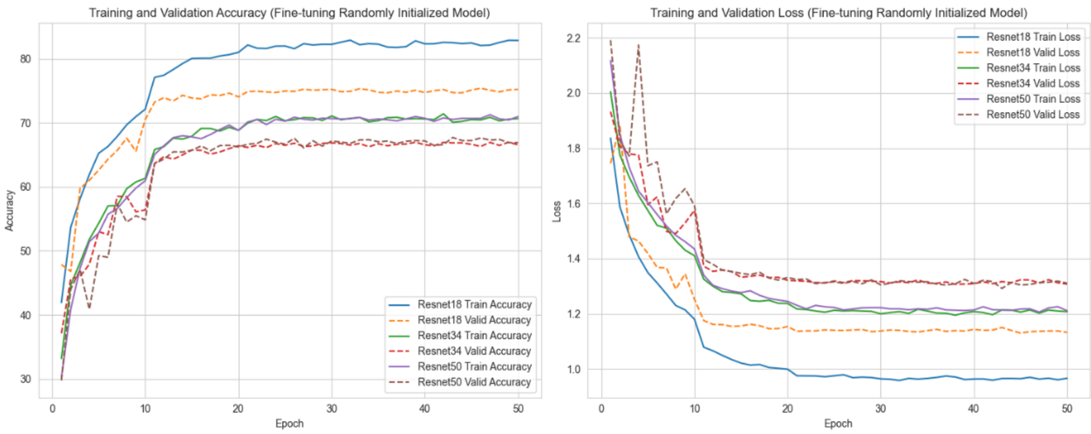
The appendices would include the generated plots that supports the findings of this report.



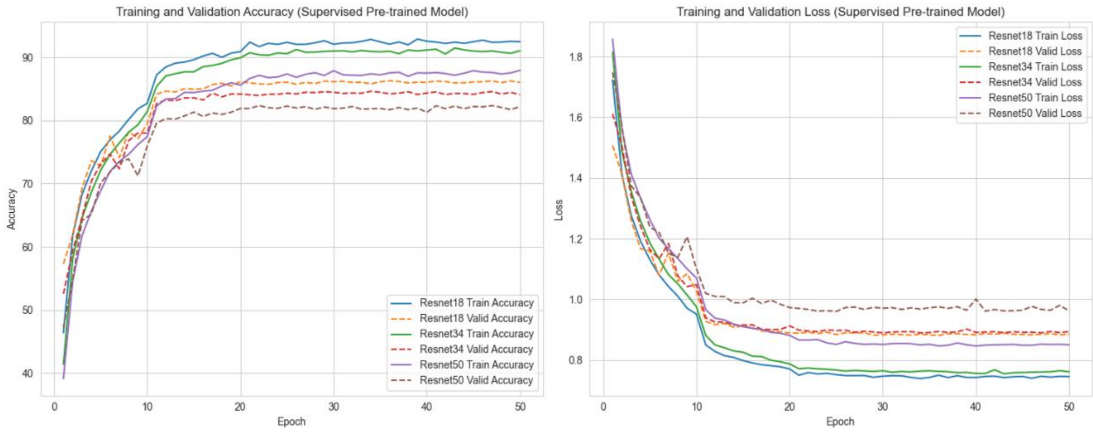
^ Figure 1: Self-Supervised Learning



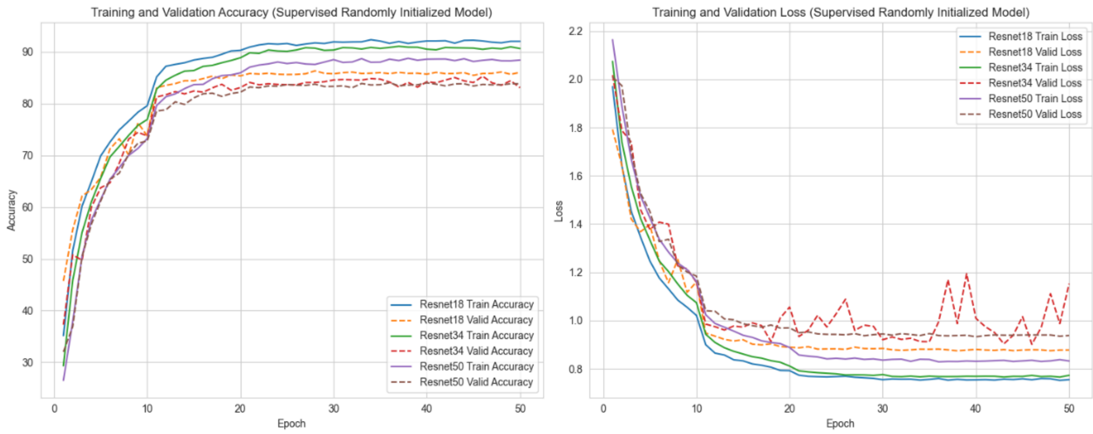
^ Figure 2: Fine-tuning Pre-trained Model



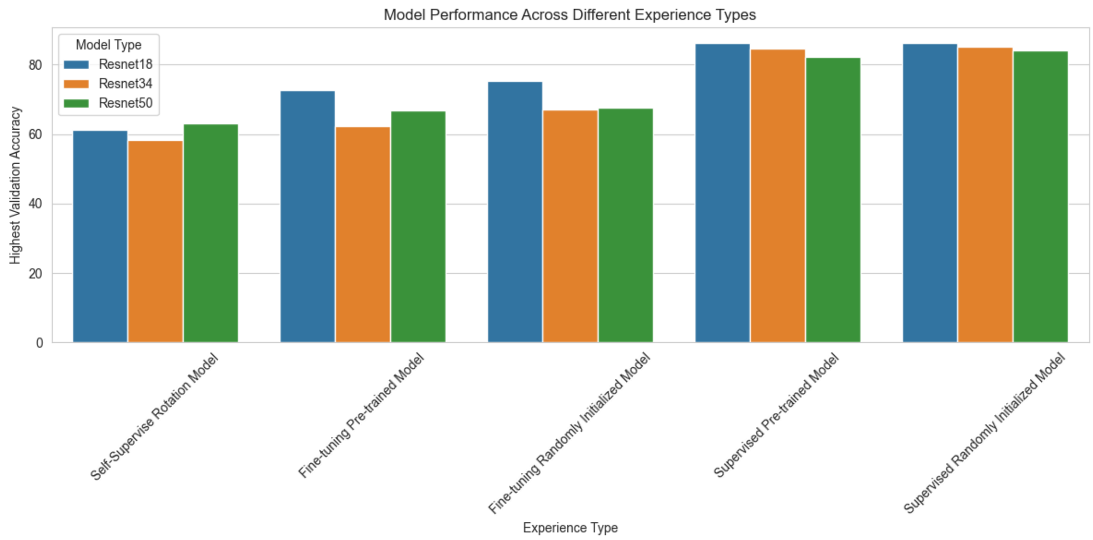
^ Figure 3: Fine-tuning Randomly Initialized Model



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



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



^ *Figure 6: Models Accuracy Loss Comparison*

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) Retrieved from paper

[3] Müller, R., Kornblith, S., & Hinton, G. E. (2019). When Does Label Smoothing Help? In Advances in Neural Information Processing Systems 32 (NeurIPS 2019). Retrieved from paper