




# **A Guide to Pseudolabelling: How to get a Kaggle medal with only one model**

**Stanley Zheng**



# About Me



Stanley Zheng


Calgary, Alberta, Canada

Joined 6 months ago · last seen in the past day

Followers 60

Following 46



Competitions Expert

Home

Competitions (10)

Datasets (5)

Notebooks (7)

Discussion (272)

Organizations

Followers (60)

...

Edit Profile

Competitions Expert


Current Rank


1188


of 152,297

Highest Rank

1151

 0

 1

 2

Global Wheat D...

4 months ago

Top 4%

73<sup>rd</sup>

of 2245

RSNA STR Pulm...

2 months ago

Top 7%

54<sup>th</sup>

of 784

Halite by Two Si...

3 months ago


Top 6%


65<sup>th</sup>


of 1139

Datasets Contributor

Unranked

 0

 0

 1

RSNA-STR 256...

2 months ago

10 votes

pytorchyolov4

4 months ago

4 votes

darknet

4 months ago

3 votes

Notebooks Expert


Current Rank


368


of 150,741

Highest Rank

360

 2

 0

 3

Baseline NN wit...

3 months ago

85 votes

Multilabel Neura...

3 months ago

61 votes

MOA: Convoluti...

3 months ago

12 votes

Discussion Expert


Current Rank


93


of 173,561

Highest Rank

87

 8

 15

 187

Huge props to t...

4 months ago

23 votes

List of top soluti...

2 months ago

22 votes

Possible cross t...

2 months ago

21 votes

GitHub/Kaggle/LinkedIn:  
@stanleyjzheng

# Contents

1. What are pseudolabels, and why should we use them?
2. Effective use of pseudolabels
3. Applications
4. Minimal code example on MNIST

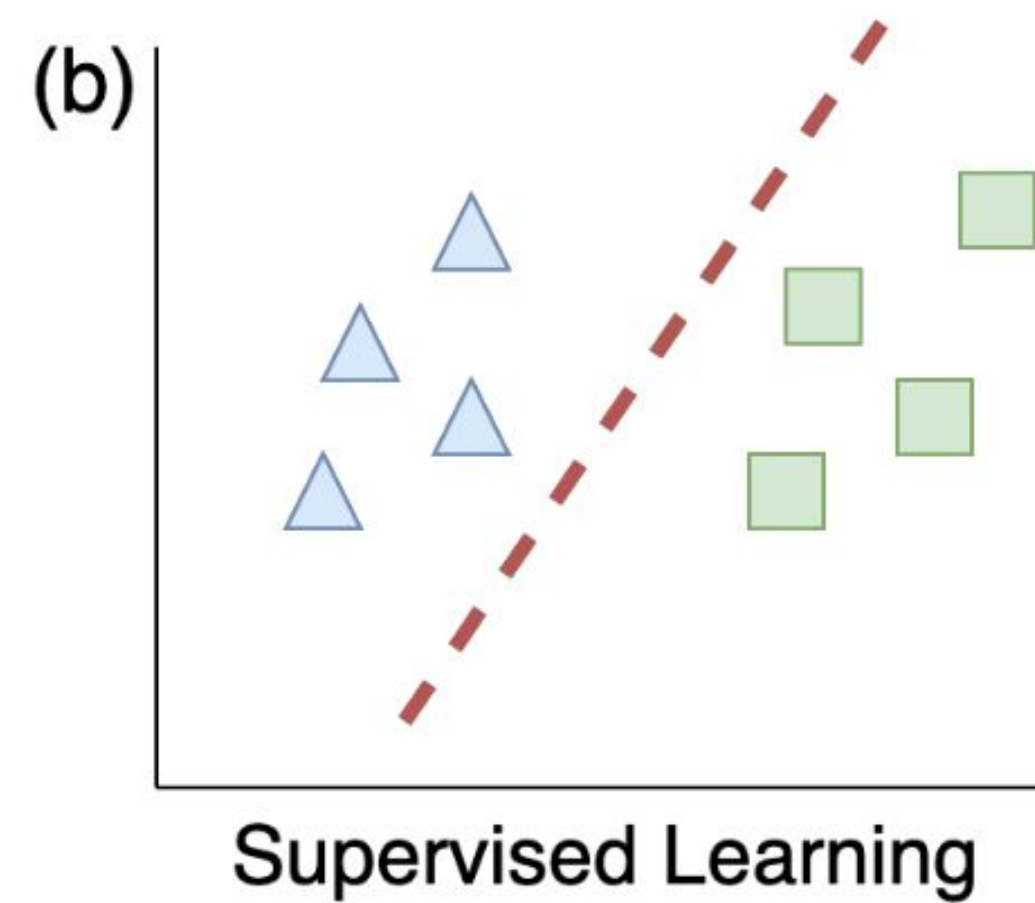
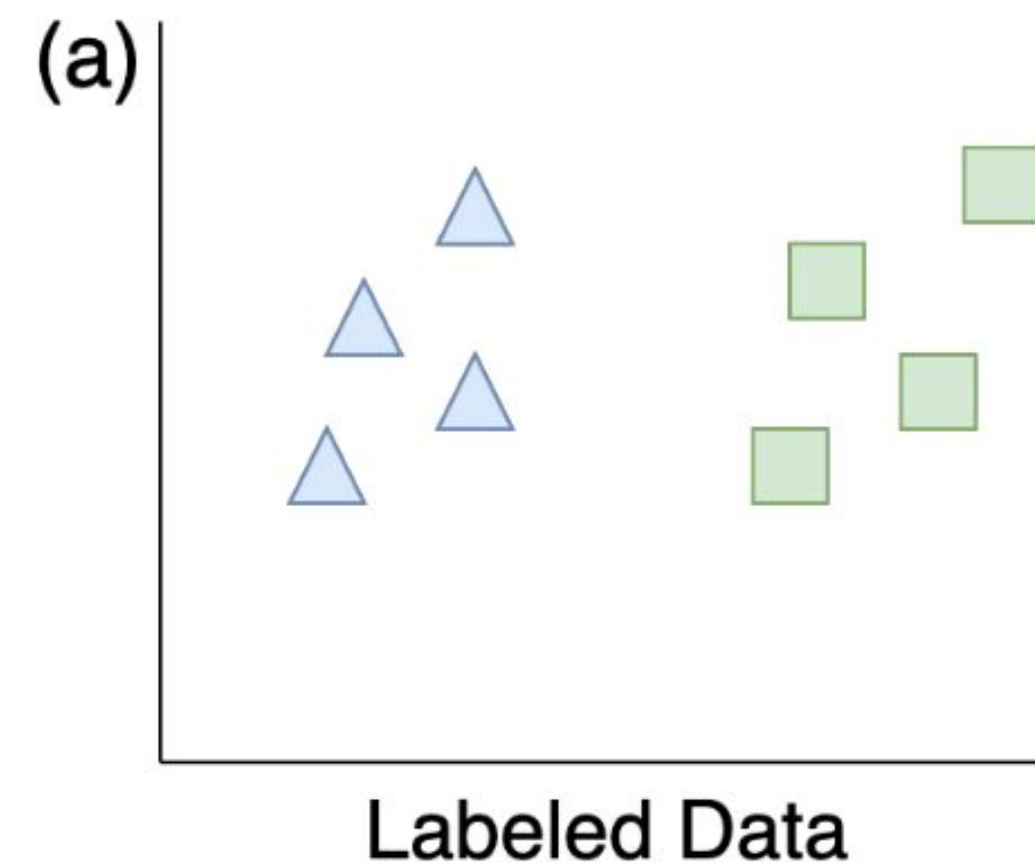
Sources and further reading at [bit.ly/pydata-github](https://bit.ly/pydata-github)

Thanks to Yauhen Babakhin for some of the graphics in this presentation [1]

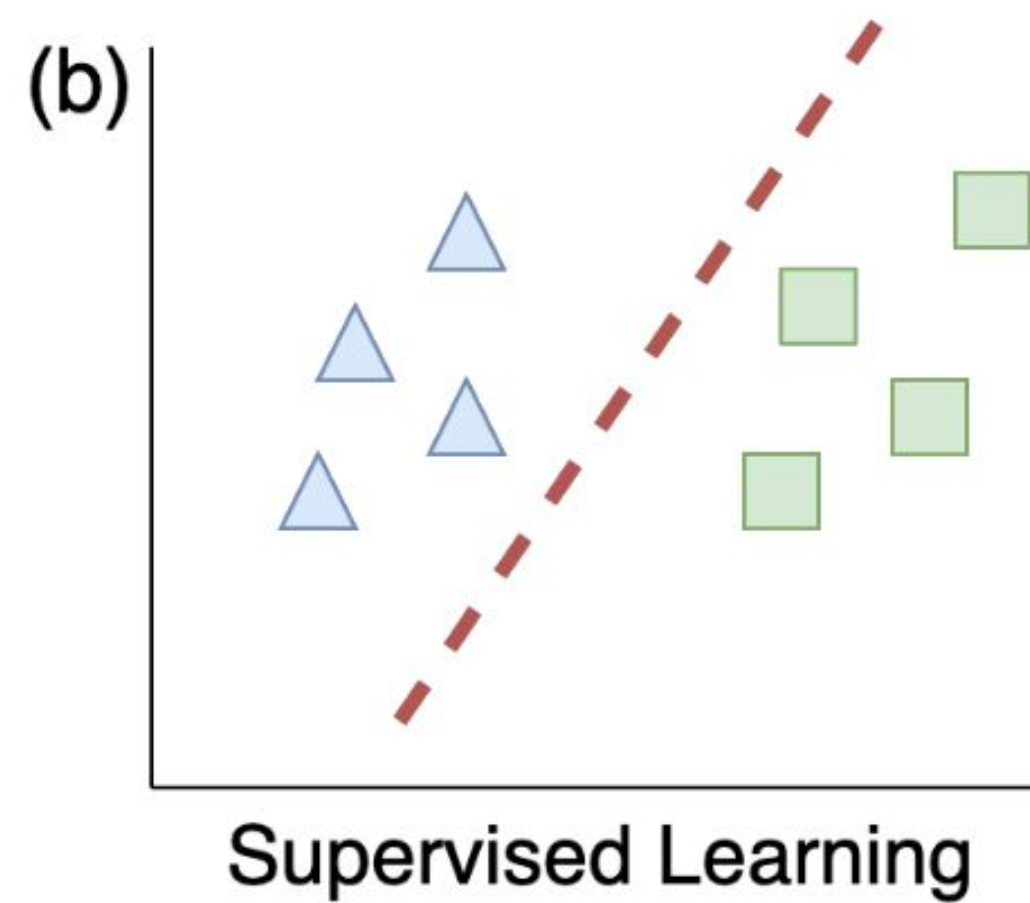
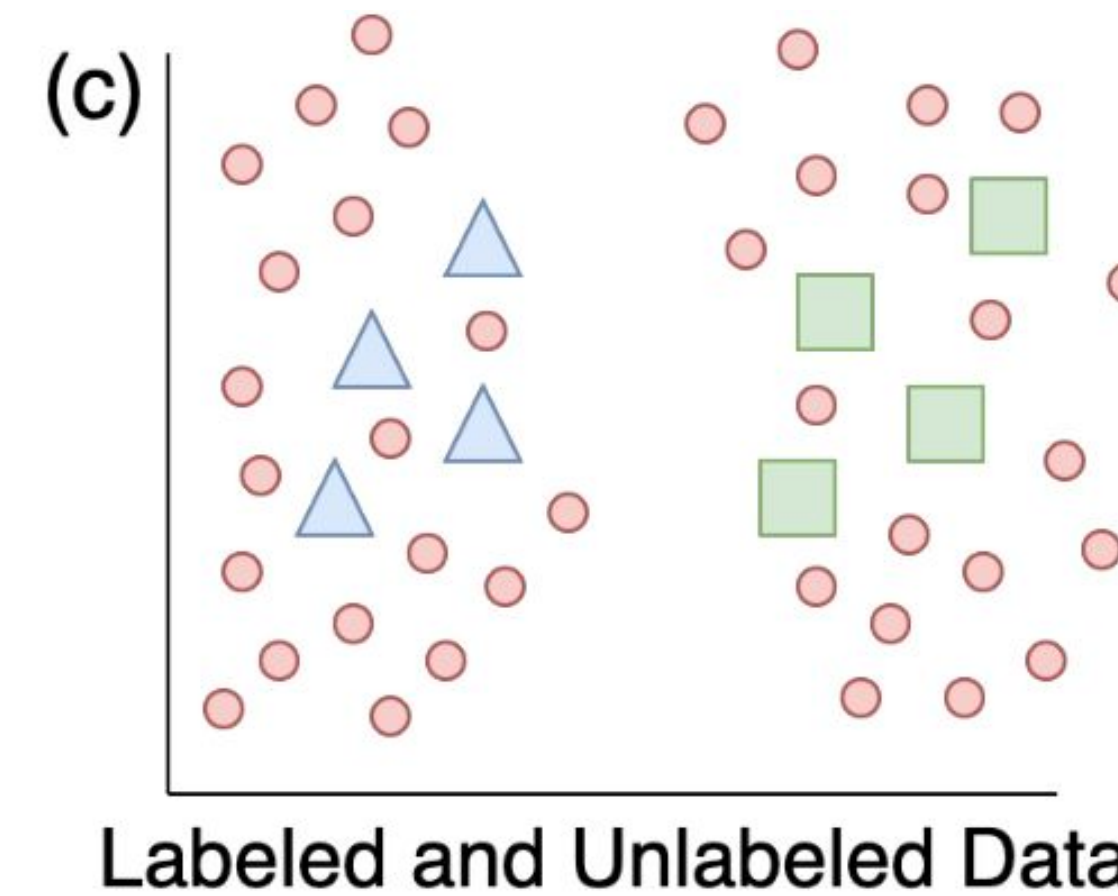
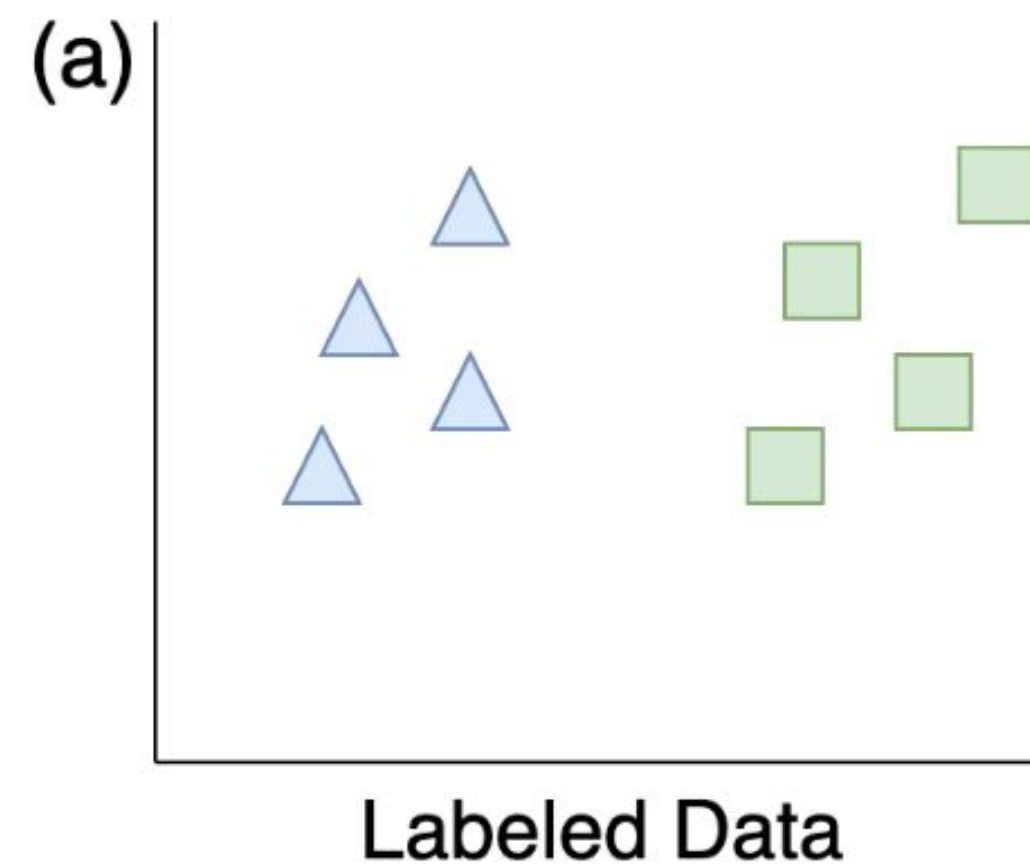
# What is Pseudolabelling?

- Semi-supervised learning
- Allows models to leverage a large unlabelled dataset

# Visual Example

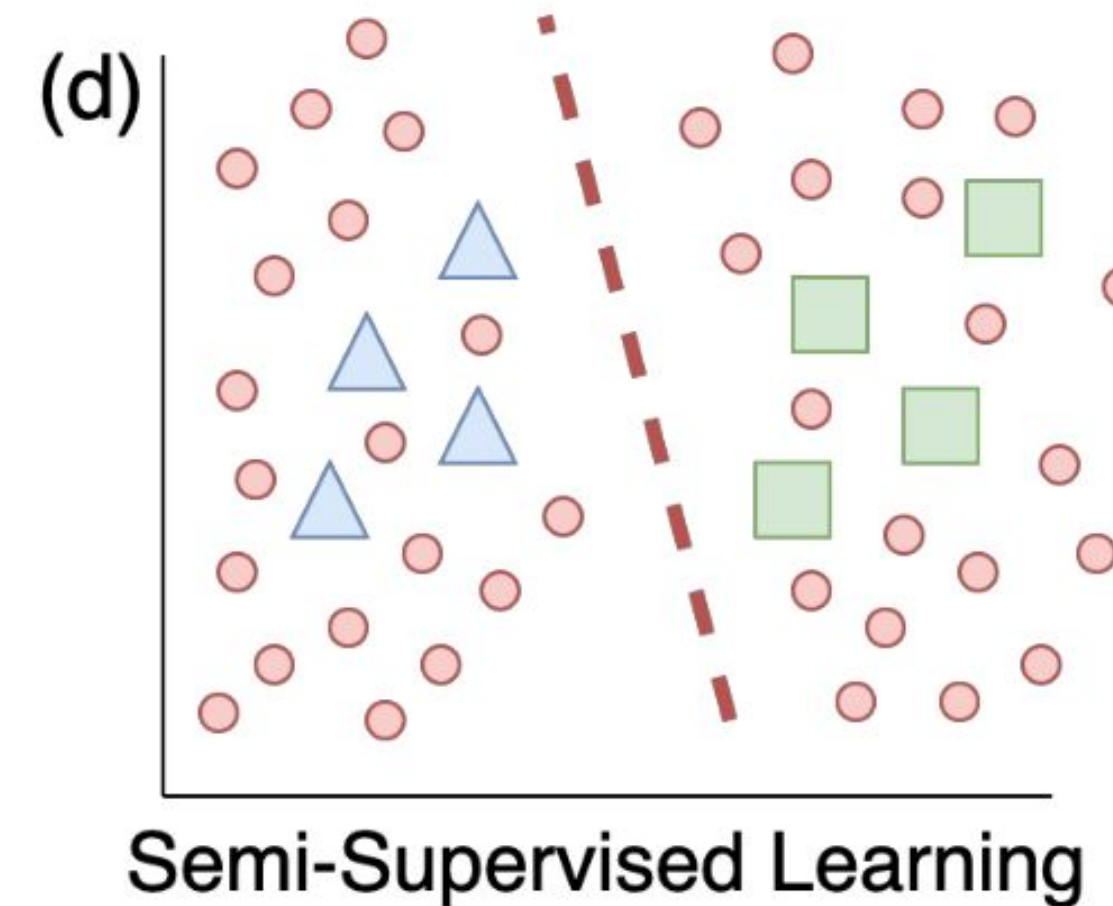
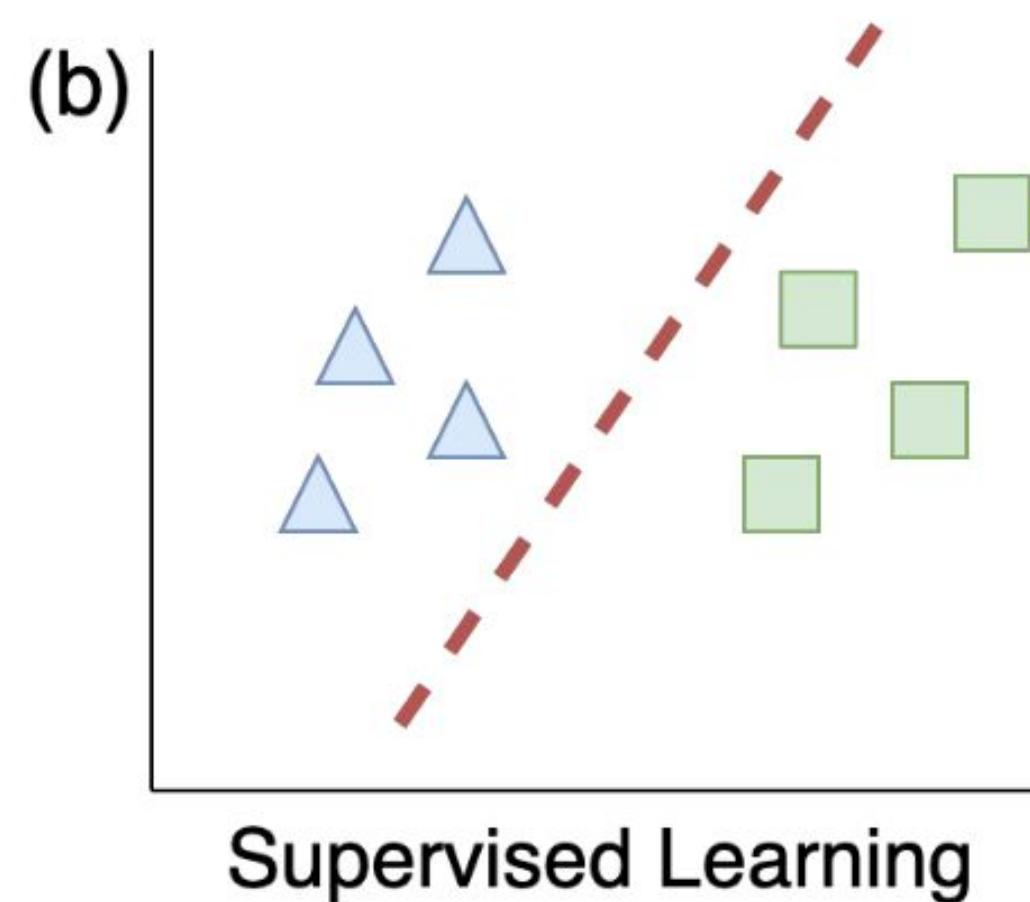
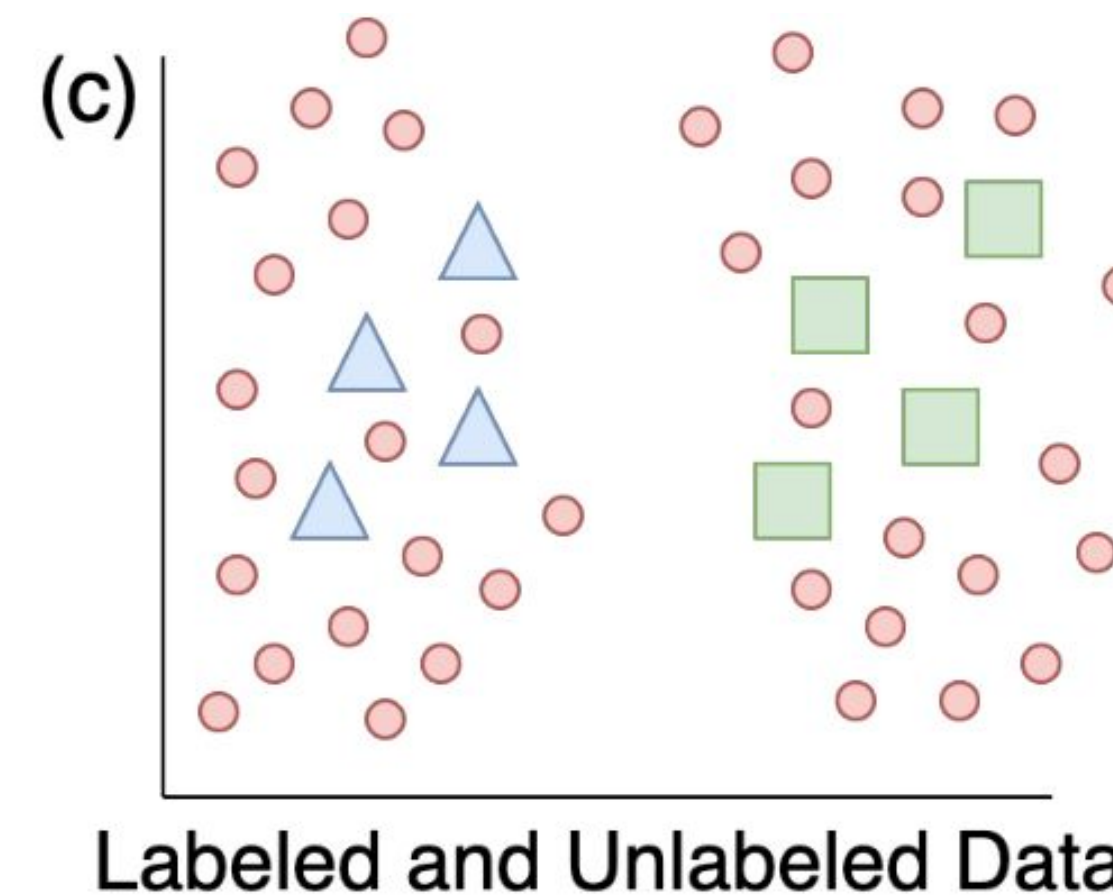
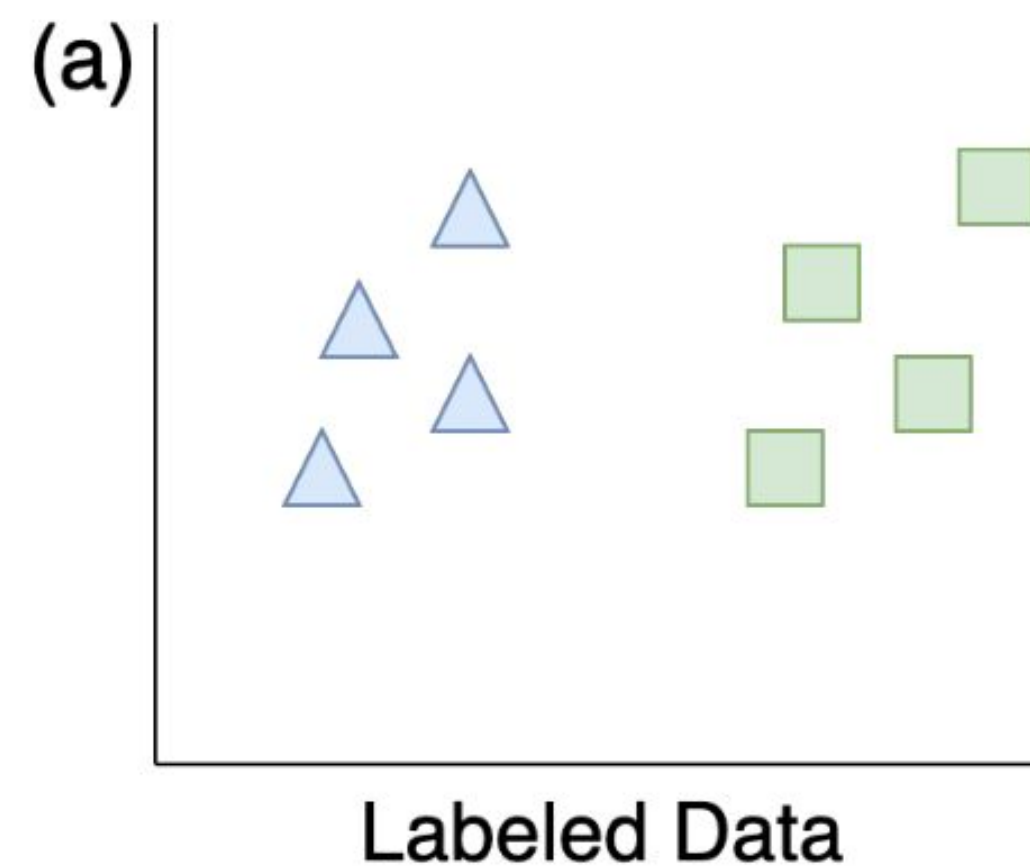


# Visual Example





# Visual Example



# Why use pseudolabelling?

- Data is expensive to label
  - Requires expert labelling or is time consuming



# Why use pseudolabelling?

- Data is expensive to label
  - Requires expert labelling or is time consuming
- Data is time-sensitive
  - Model needs to be made now but data is time consuming to label

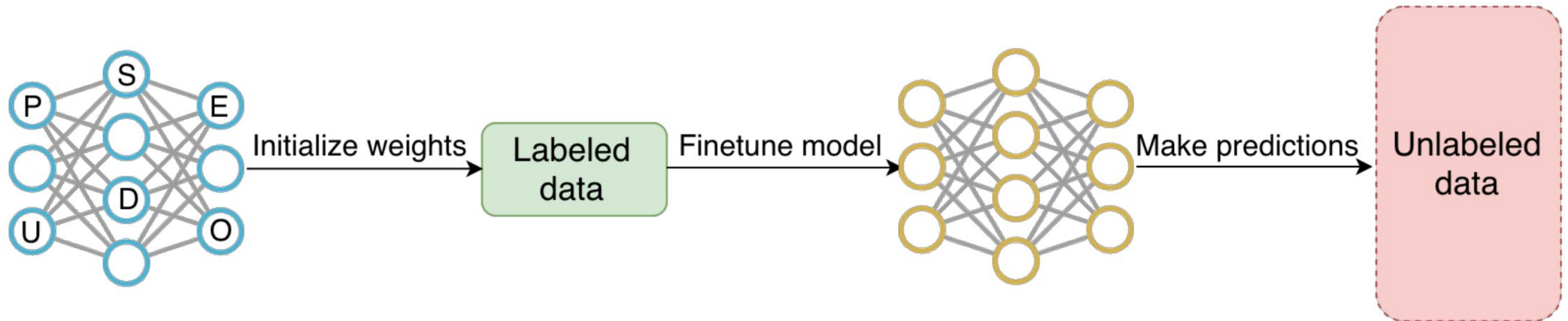
# Why use pseudolabelling?

- Data is expensive to label
  - Requires expert labelling or is time consuming
- Data is time-sensitive
  - Model needs to be made now but data is time consuming to label
- There is a large diversity in the datasets
  - Training images are from one use case, while test images are from another

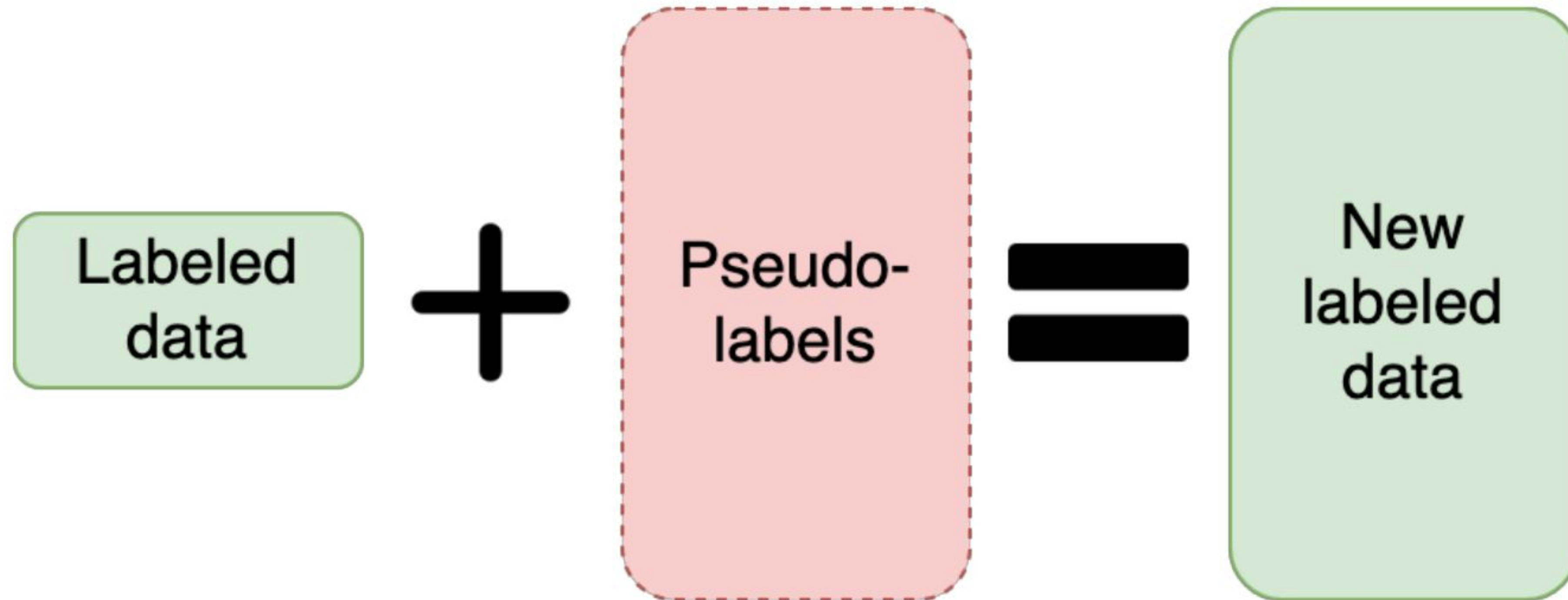
# Why use pseudolabelling? [2]

- Data is expensive to label
  - Requires expert labelling or is time consuming
- Data is time-sensitive
  - Model needs to be made now but data is time consuming to label
- There is a large diversity in the datasets
  - Training images are from one use case, while test images are from another
- Any application with a small training set and a large test set

# Training Schemes - Self Training

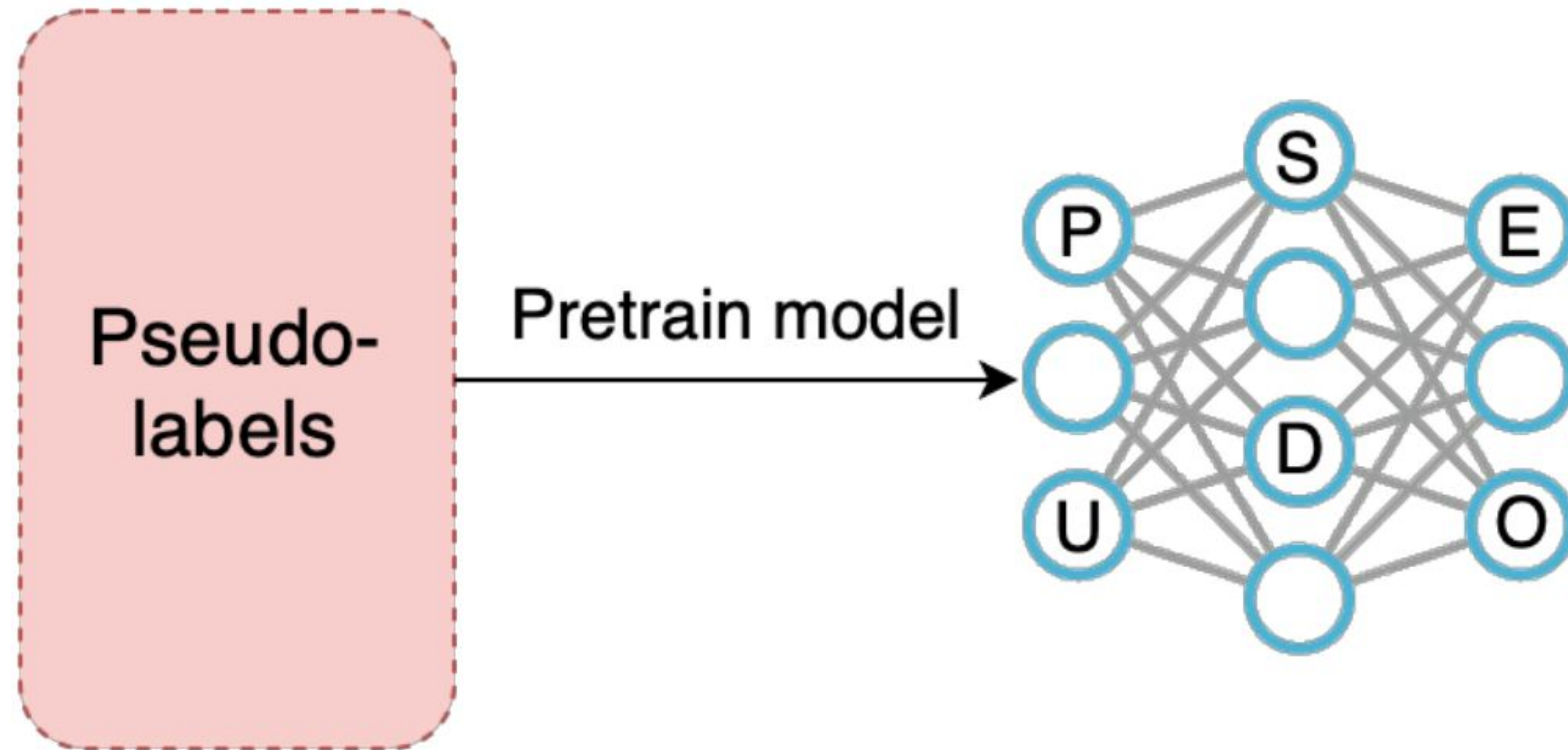


# Training Schemes - Simultaneous training





# Training Schemes - Pretraining [3]





# General tips and tricks with pseudolabels

- For uncleaned datasets, use confidence thresholds to ensure labels are less noisy

# General tips and tricks with pseudolabels

- For uncleaned datasets, use confidence thresholds to ensure labels are less noisy
- Where possible, use soft pseudolabels over hard pseudolabels
- Reduce variance with ensemble

# General tips and tricks with pseudolabels [4]

- For uncleaned datasets, use confidence thresholds to ensure labels are less noisy
- Where possible, use soft pseudolabels over hard pseudolabels
- Reduce variance with ensemble
- Use iterative pseudolabels, not one shot
  - Make sure to reinitialize the weights after each iteration
- Take measures to reduce overfitting (dropout/stochastic depth, augmentations)

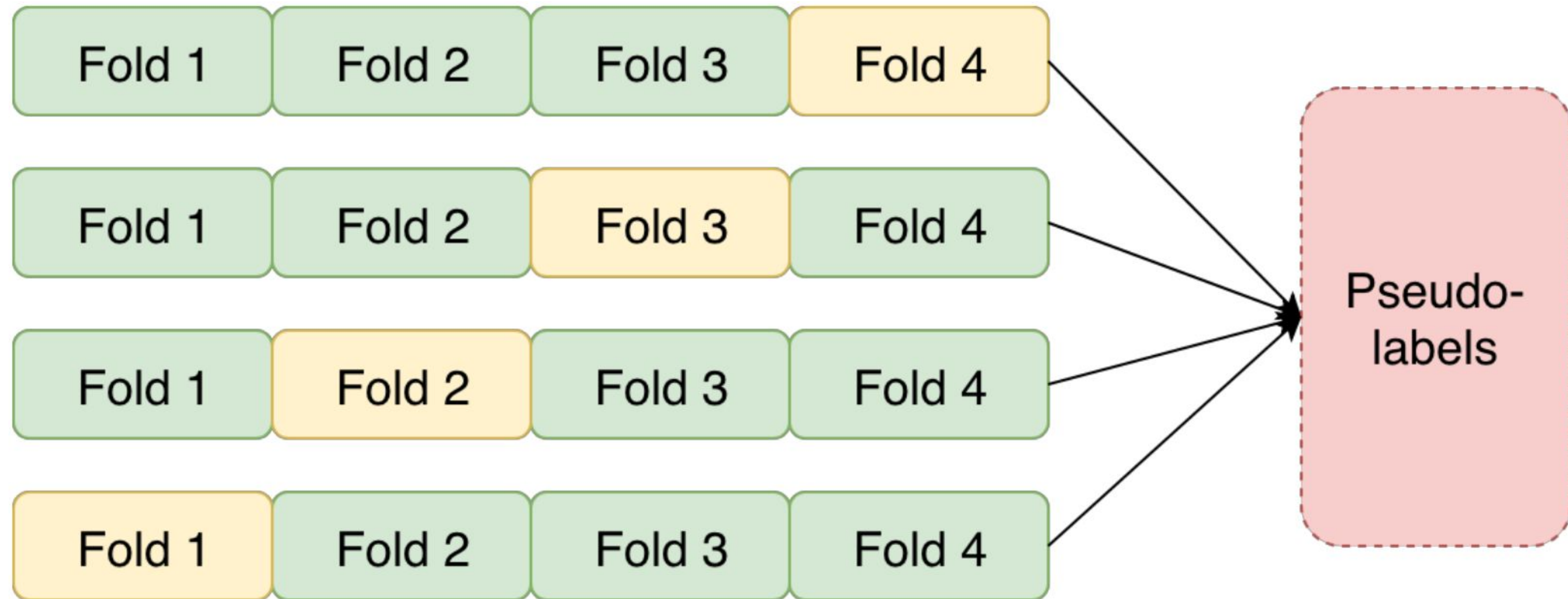
# K-fold Cross Validation

1. Split the dataset into  $k$  groups
2. For each unique group:
  - i. Take the group as a test data set
  - ii. Take the remaining groups as a training data set
  - iii. Fit a model on the training set and evaluate it on the test set
3. Find model's out-of-fold performance and save the model.

# Ensembling folds to produce pseudolabels

1. Split the dataset into  $k$  groups
2. For each unique group:
  - i. Take the group as a test data set
  - ii. Take the remaining groups as a training data set
  - iii. Fit a model on the training set
3. Ensemble all models together to produce one set of pseudolabels
4. Train models on pseudolabels and evaluate

# Ensembling folds to produce pseudolabels

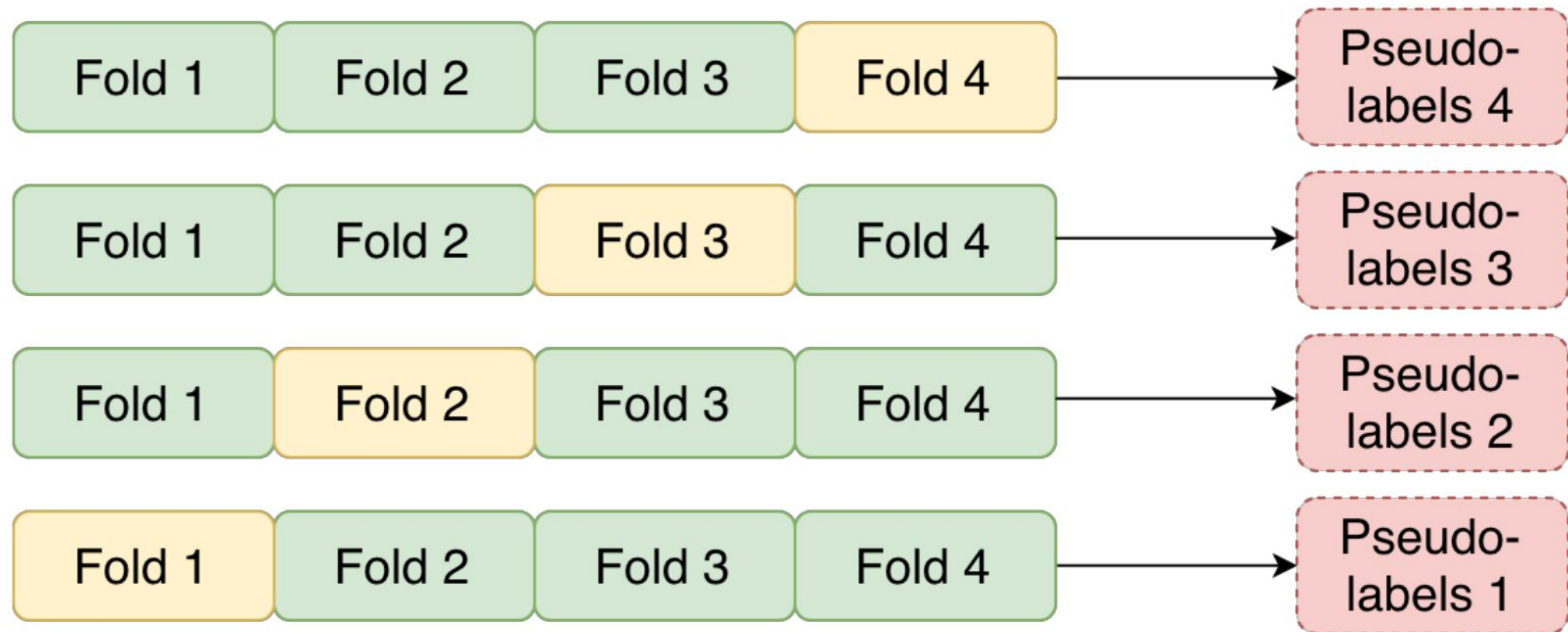




# Non-leaky approach

1. Split the dataset into  $k$  groups
2. For each unique group:
  - i. Take the group as a test data set
  - ii. Take the remaining groups as a training data set
  - iii. Fit a model on the training set
  - iv. Infer pseudolabels for the test set and train on it
  - v. Evaluate the model
3. Find model's out-of-fold performance and save the model.

# Non-leaky approach





# Global Wheat Detection [5]

Scores of various models (Intersection over Union, higher is better)

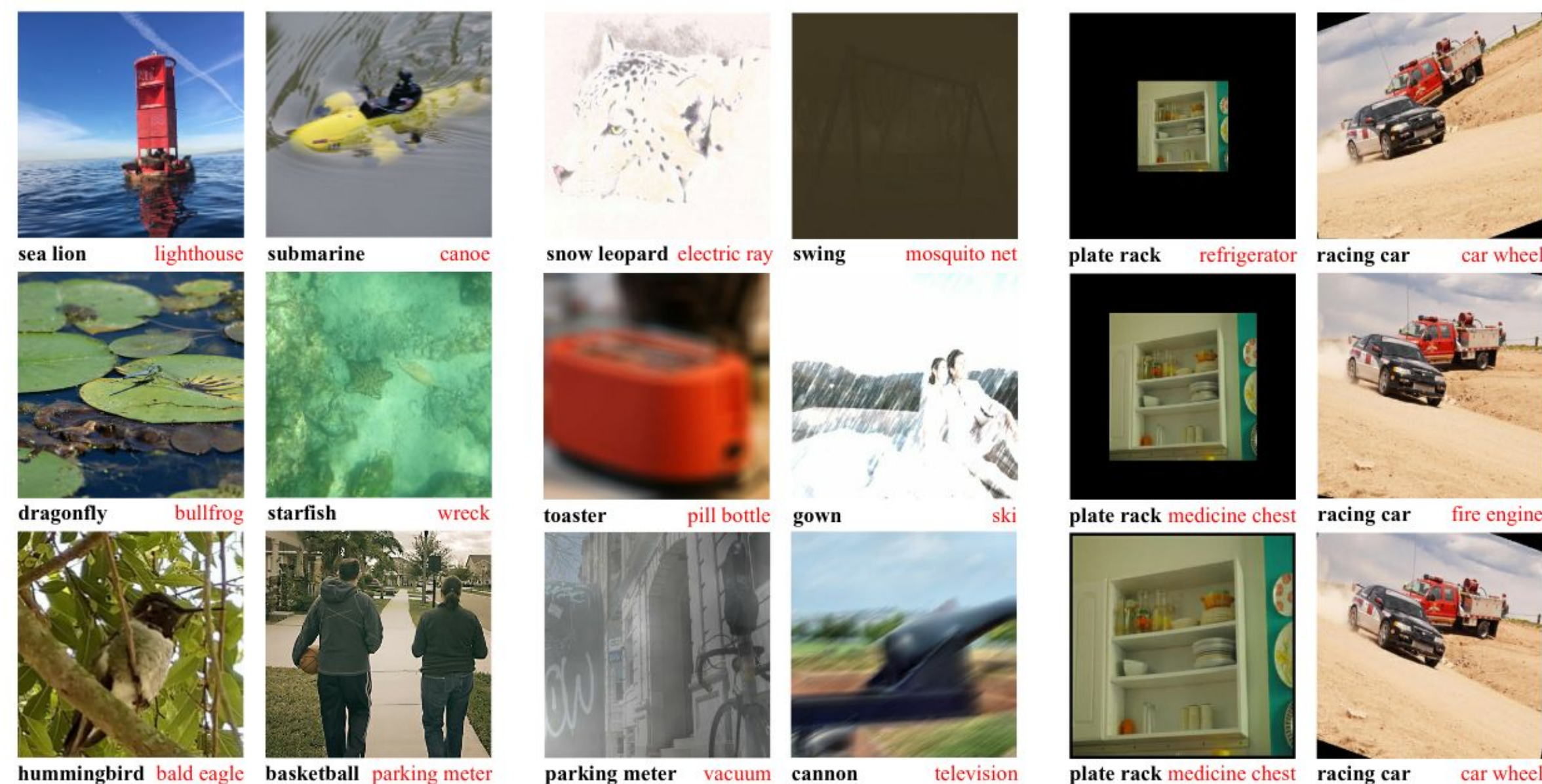
Technique	Public Leaderboard	Private Leaderboard
No pseudolabelling	0.7115 (821st)	0.6371 (327th)
Self-training	0.7406 (+4%, 207th)	0.6625 (+4%, 36th)
Pretraining	0.7562 (+6%, 61st)	0.6668 (+5%, 22nd)





# EfficientNet Noisy Student [4]

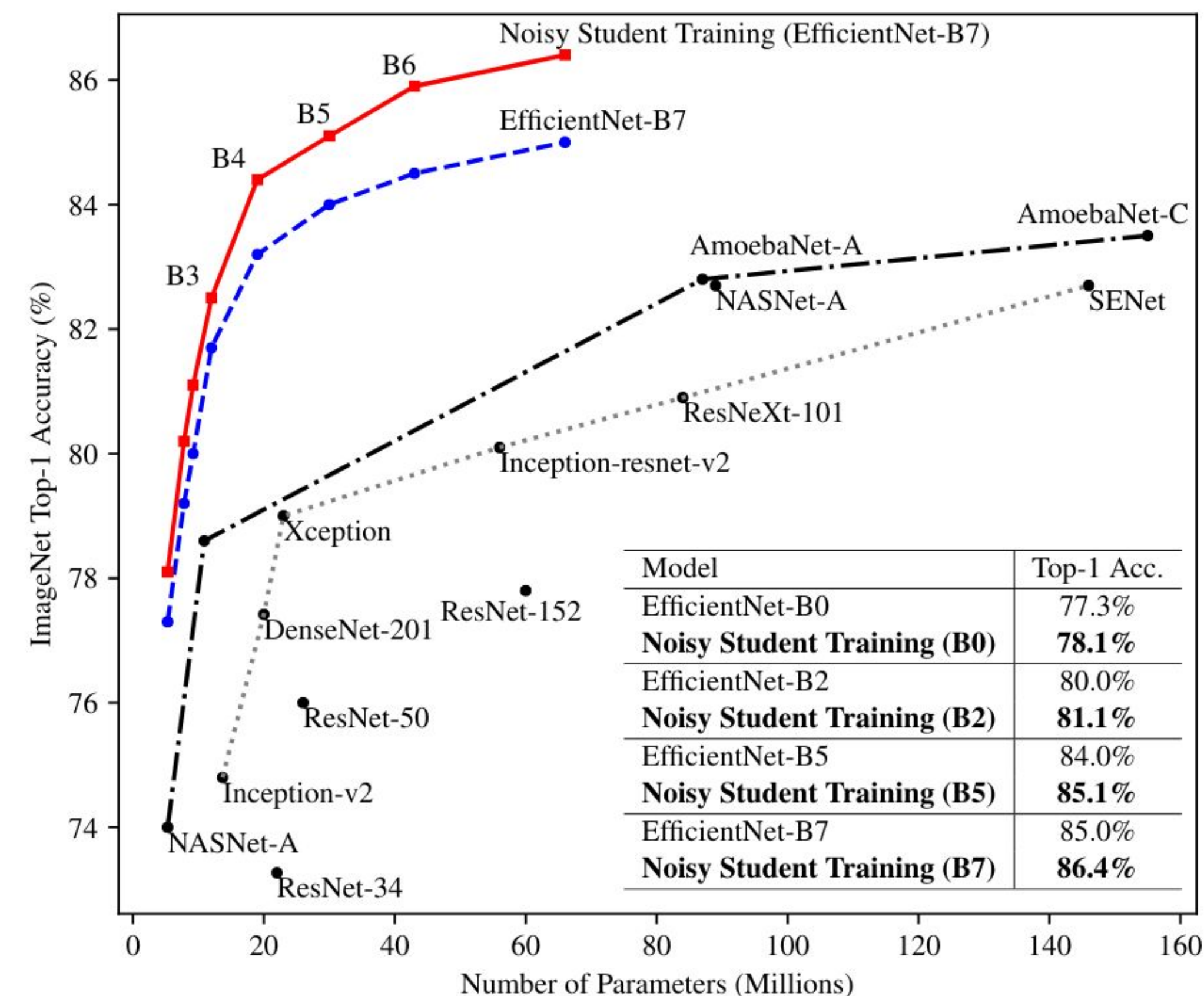
Method	# Params	Extra Data	Top-1 Acc.	Top-5 Acc.
ResNet-50 Billion-scale [93]	26M	3.5B images labeled with tags	81.2%	96.0%
ResNeXt-101 Billion-scale [93]	193M		84.8%	-
ResNeXt-101 WSL [55]	829M		85.4%	97.6%
FixRes ResNeXt-101 WSL [86]	829M		86.4%	98.0%
Big Transfer (BiT-L) [43] <sup>†</sup>	928M	300M weakly labeled images from JFT	87.5%	98.5%
<b>Noisy Student Training (EfficientNet-L2)</b>	480M	300M unlabeled images from JFT	<b>88.4%</b>	<b>98.7%</b>



(a) ImageNet-A

(b) ImageNet-C

(c) ImageNet-P





# Simple Code Example on MNIST [6]

[bit.ly/pydata-notebook](http://bit.ly/pydata-notebook)

# Thanks!

I'm looking for paid or unpaid remote internships during Winter 2020 or Summer 2021 - [szheng3@athabasca.edu](mailto:szheng3@athabasca.edu)

Find me on GitHub/Kaggle/LinkedIn: @stanleyjzheng