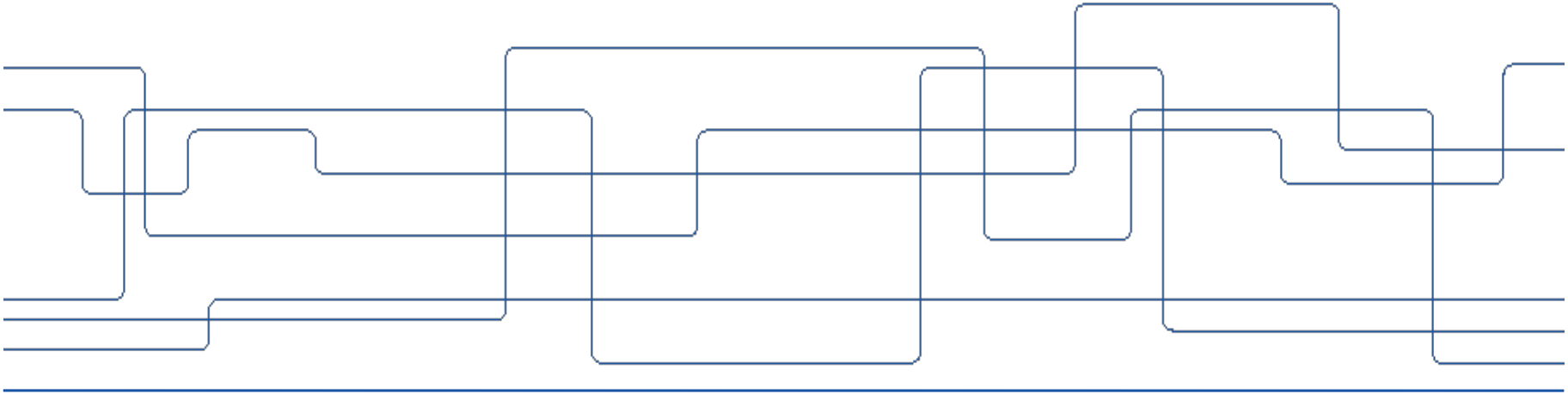
Semi-supervised Learning

Temporal Ensembling for Semi-Supervised Learning - Samuli Laine, Timo Aila - ICLR 2017

Meta Pseudo Labels - Hieu Pham, Zihang Dai, Qizhe Xie, Minh-Thang Luong, Quoc V. Le - CVPR 2021



Semi-supervised learning

Learning from both labeled **and** unlabeled data

Scenario 1:
You have ImageNet, but want to use even more data

Scenario 2:
You have few labels, but loads of unlabeled data

Source: xview2 dataset

How to make use of the unlabeled data?

1. Pre-train model on unlabeled data, then fine-tune on labeled dataset
   → self-supervised learning

2. Train on labeled and unlabeled data **at the same time**.

   a. Assign labels to unlabeled data
      → **Pseudo labels**

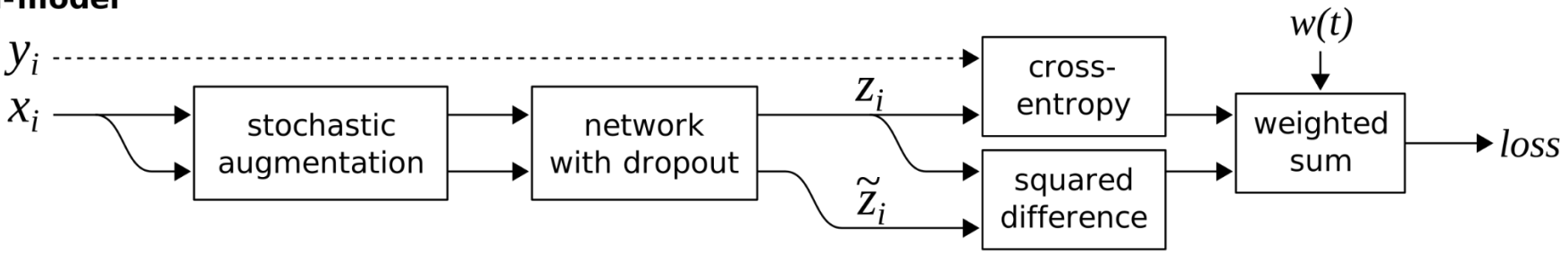
   b. Train unsupervised objective in parallel to supervised
      → **Consistency regularisation / Label propagation**

Temporal Ensembling for Semi-Supervised Learning

Samuli Laine, Timo Aila - ICLR 2017

Consistency Regularisation

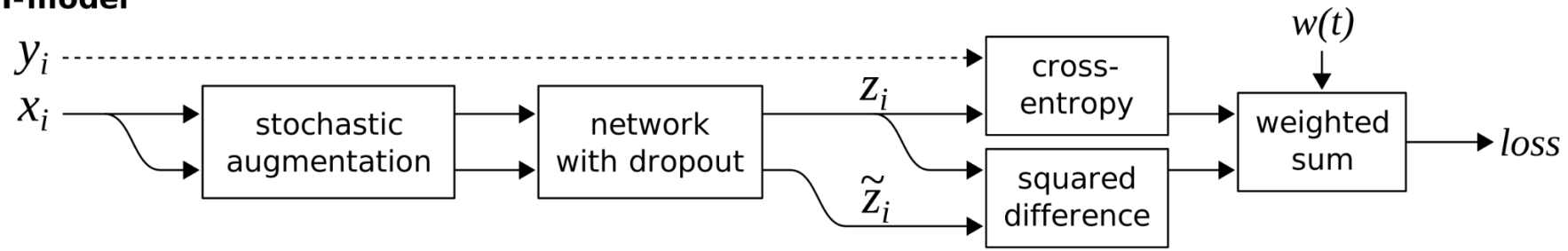
Π-model



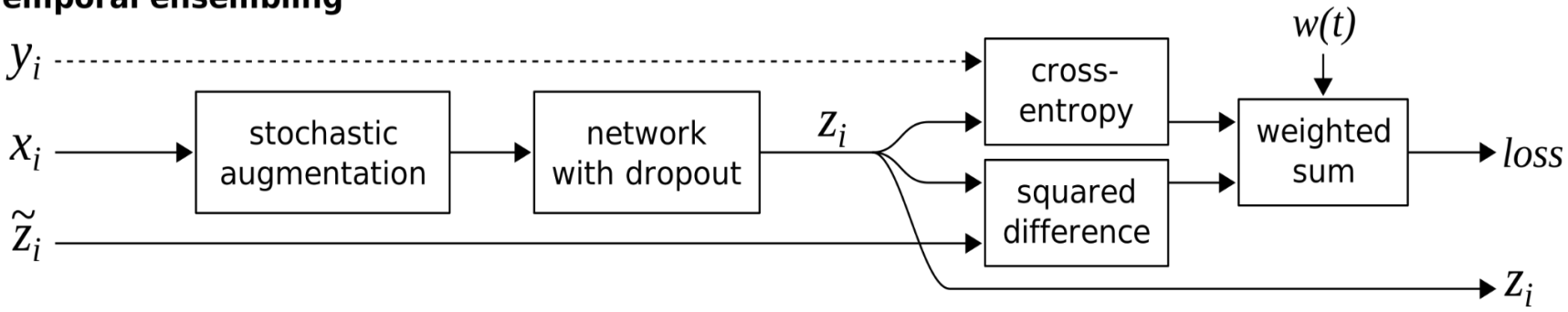
- Loss = Supervised + Unsupervised
- Unsupervised: **Consistency regularisation**. Different stochastic influences should lead to same results
- Unsupervised loss also used for labeled images

Average predictions over time

Π-model

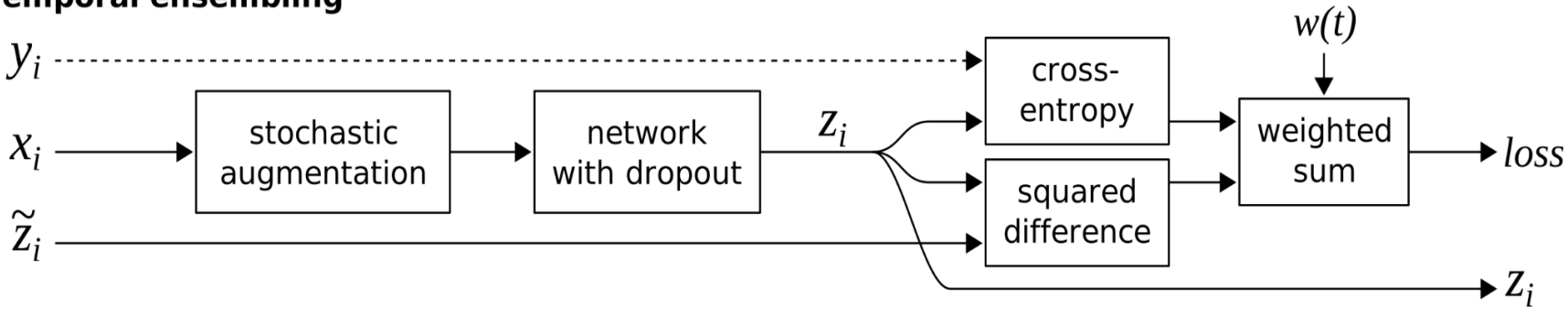


Temporal ensembling



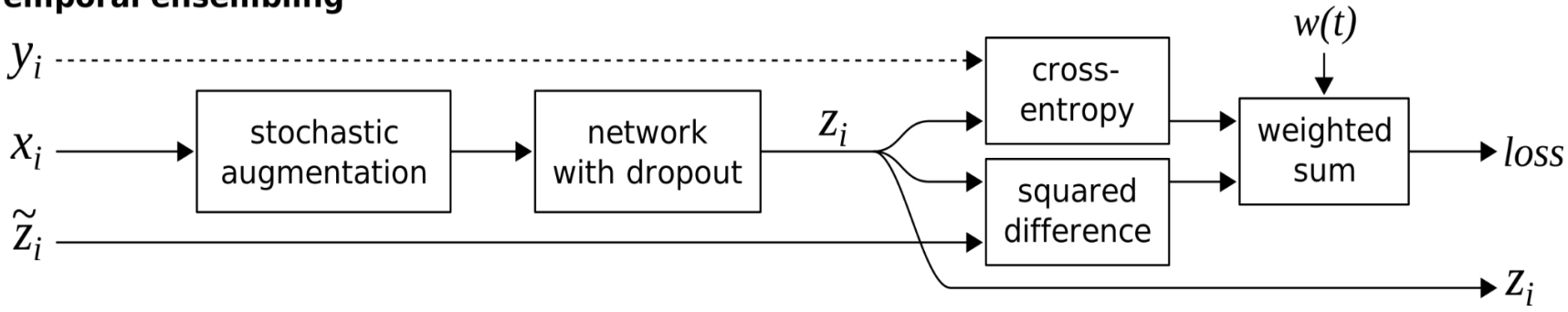
Self-ensembling over time

Temporal ensembling



Self-ensembling over time

Temporal ensembling



Advantages:

- Compare prediction to moving average of **past** predictions
  → Less noisy than current prediction!
- Only one pass per epoch → Faster!

Disadvantage:

- Memory needed to store prediction for whole dataset
- Prediction targets are too old when training on large datasets

Semi-supervised learning on CIFAR-10

	Error rate (%) with # labels	
	4000	All (50000)
Supervised-only	35.56 ± 1.59	7.33 ± 0.04
with augmentation	34.85 ± 1.65	6.05 ± 0.15
Conv-Large, Γ-model (Rasmus et al., 2015)	20.40 ± 0.47	
CatGAN (Springenberg, 2016)	19.58 ± 0.58	
GAN of Salimans et al. (2016)	18.63 ± 2.32	
Π-model	16.55 ± 0.29	6.90 ± 0.07
Π-model with augmentation	12.36 ± 0.31	$\mathbf{5.56 \pm 0.10}$
Temporal ensembling with augmentation	$\mathbf{12.16 \pm 0.24}$	5.60 ± 0.10

Semi-supervised learning on CIFAR-10

	Error rate (%) with # labels	
	4000	All (50000)
Supervised-only	35.56 ± 1.59	7.33 ± 0.04
with augmentation	34.85 ± 1.65	6.05 ± 0.15
Conv-Large, Γ-model (Rasmus et al., 2015)	20.40 ± 0.47	
CatGAN (Springenberg, 2016)	19.58 ± 0.58	
GAN of Salimans et al. (2016)	18.63 ± 2.32	
Π-model	16.55 ± 0.29	6.90 ± 0.07
Π-model with augmentation	12.36 ± 0.31	$\mathbf{5.56 \pm 0.10}$
Temporal ensembling with augmentation	$\mathbf{12.16 \pm 0.24}$	5.60 ± 0.10

Why better than supervised on the full dataset?

Semi-supervised learning on CIFAR-10

	Error rate (%) with # labels	
	4000	All (50000)
Supervised-only	35.56 ± 1.59	7.33 ± 0.04
with augmentation	34.85 ± 1.65	6.05 ± 0.15
Conv-Large, Γ-model (Rasmus et al., 2015)	20.40 ± 0.47	
CatGAN (Springenberg, 2016)	19.58 ± 0.58	
GAN of Salimans et al. (2016)	18.63 ± 2.32	
Π-model	16.55 ± 0.29	6.90 ± 0.07
Π-model with augmentation	12.36 ± 0.31	$\mathbf{5.56 \pm 0.10}$
Temporal ensembling with augmentation	$\mathbf{12.16 \pm 0.24}$	5.60 ± 0.10

Why better than supervised on the full dataset?
→ Consistency regularisation

Semi-supervised learning on CIFAR-100

Table 3: CIFAR-100 results with 10000 labels, averages of 10 runs (4 runs for all labels).

	Error rate (%) with # labels	
	10000	All (50000)
Supervised-only	51.21 ± 0.33	29.14 ± 0.25
with augmentation	44.56 ± 0.30	26.42 ± 0.17
Π-model	43.43 ± 0.54	29.06 ± 0.21
Π-model with augmentation	39.19 ± 0.36	26.32 ± 0.04
Temporal ensembling with augmentation	$\mathbf{38.65 \pm 0.51}$	$\mathbf{26.30 \pm 0.15}$

Semi-supervised learning on CIFAR-100 + TinyImages

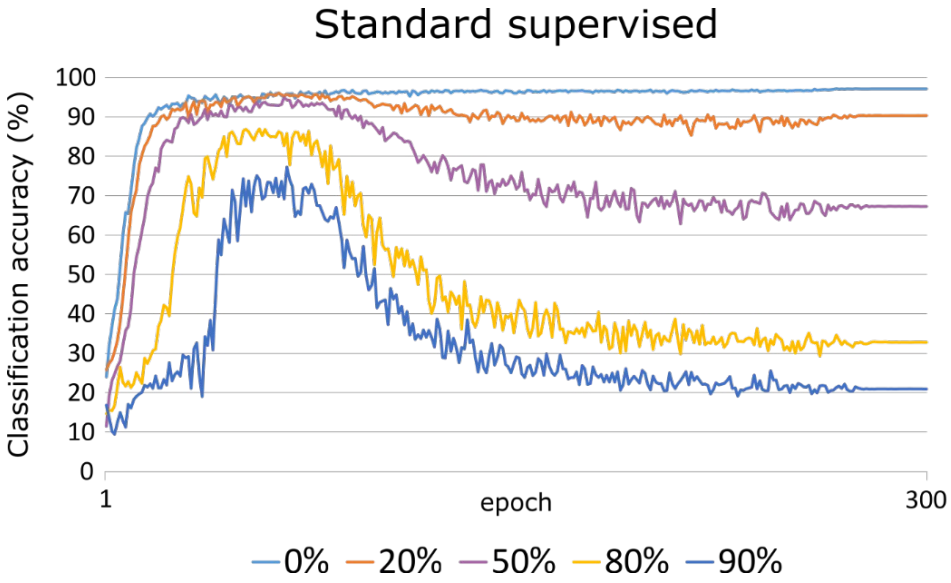
Table 3: CIFAR-100 results with 10000 labels, averages of 10 runs (4 runs for all labels).

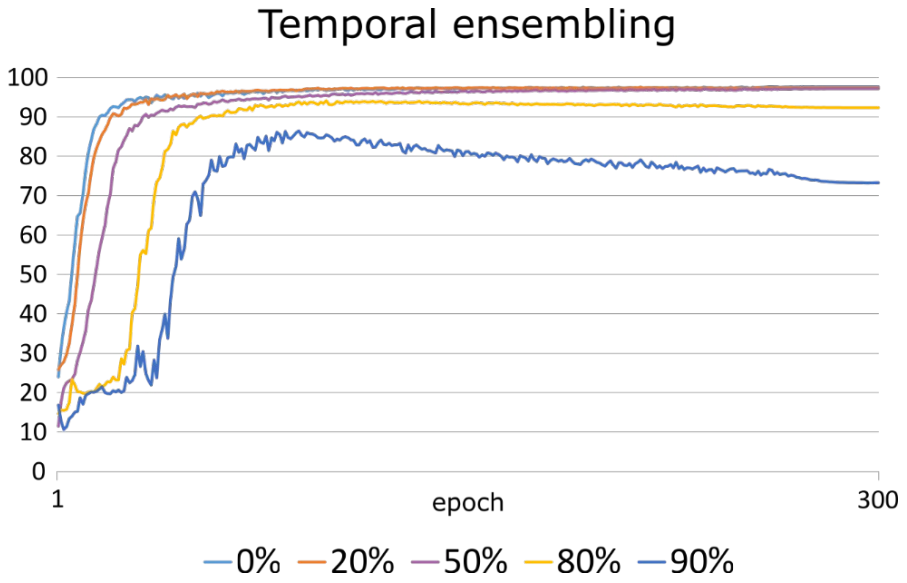
	Error rate (%) with # labels	
	10000	All (50000)
Supervised-only	51.21 ± 0.33	29.14 ± 0.25
with augmentation	44.56 ± 0.30	26.42 ± 0.17
Π-model	43.43 ± 0.54	29.06 ± 0.21
Π-model with augmentation	39.19 ± 0.36	26.32 ± 0.04
Temporal ensembling with augmentation	$\mathbf{38.65 \pm 0.51}$	$\mathbf{26.30 \pm 0.15}$

Table 4: CIFAR-100 + Tiny Images results, averages of 10 runs.

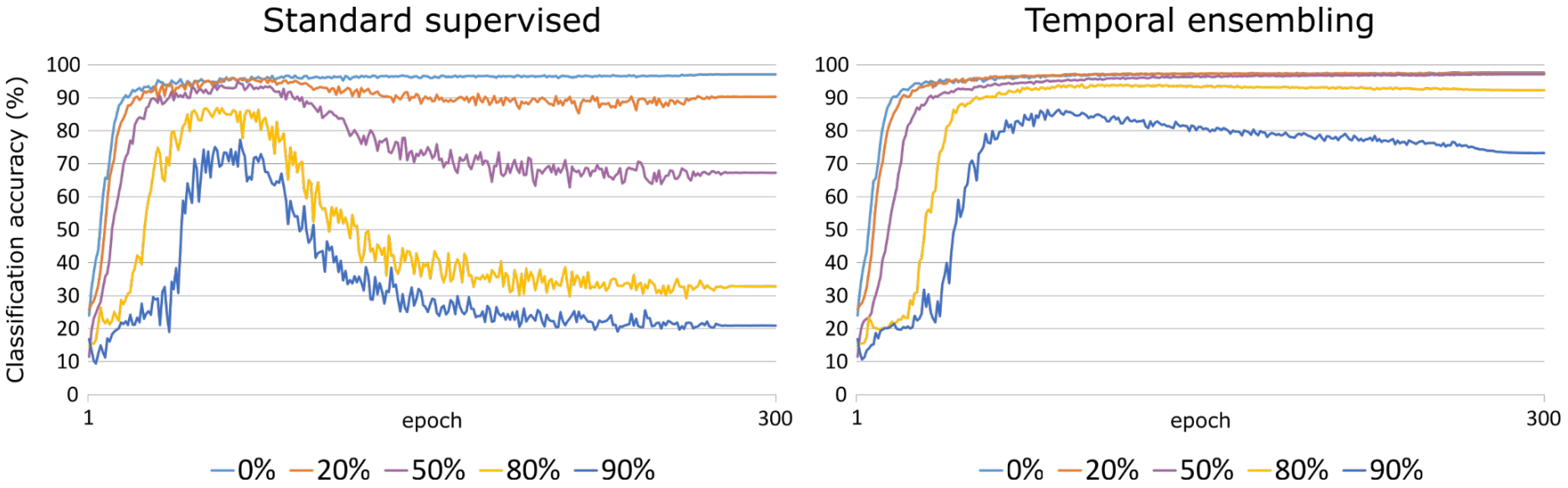
	Error rate (%) with # unlabeled auxiliary inputs from Tiny Images	
	Random 500k	Restricted 237k
Π-model with augmentation	25.79 ± 0.17	25.43 ± 0.32
Temporal ensembling with augmentation	$\mathbf{23.62 \pm 0.23}$	$\mathbf{23.79 \pm 0.24}$

Robustness to noisy labels on SVHN dataset





Robustness to noisy labels on SVHN dataset



Temporal ensembling gets >90% accuracy with 80% false labels.
Discussion: How can that be?

Connections to other approaches

- Ensembling over past predictions takes a lot of memory, changes slowly
  → Ensemble over past weights of the model: **Mean Teacher**

- Making outputs for differently augmented versions match:
  Consistency Regularisation, **FixMatch**, **Contrastive Learning**

- Soft knowledge of old model (temporal ensemble prediction) gets transferred to new model: ~ **knowledge distillation**, **teacher-student** approaches

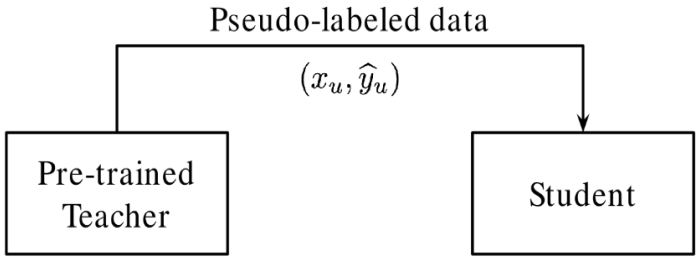
- Model's predictions are used as soft labels for the next time step.
  Hard labels → **Pseudo Label / Self-Training**, a form of **Entropy Regularization**

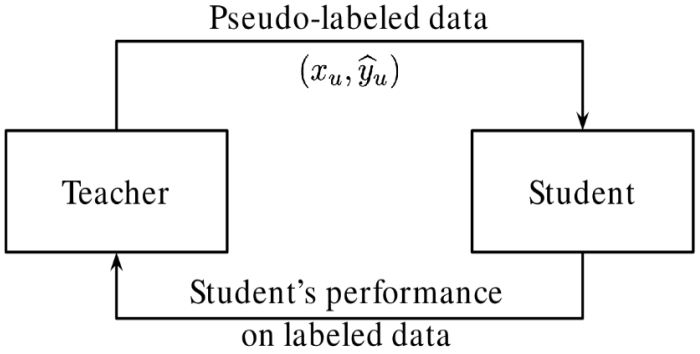
- Encouraging local smoothness by adding noise:
  Virtual Adversarial Training (**VAT**), **Noisy Student**

Meta Pseudo Labels

Hieu Pham, Zihang Dai, Qizhe Xie, Minh-Thang Luong, Quoc V. Le - CVPR 2021

Meta Pseudo Labels





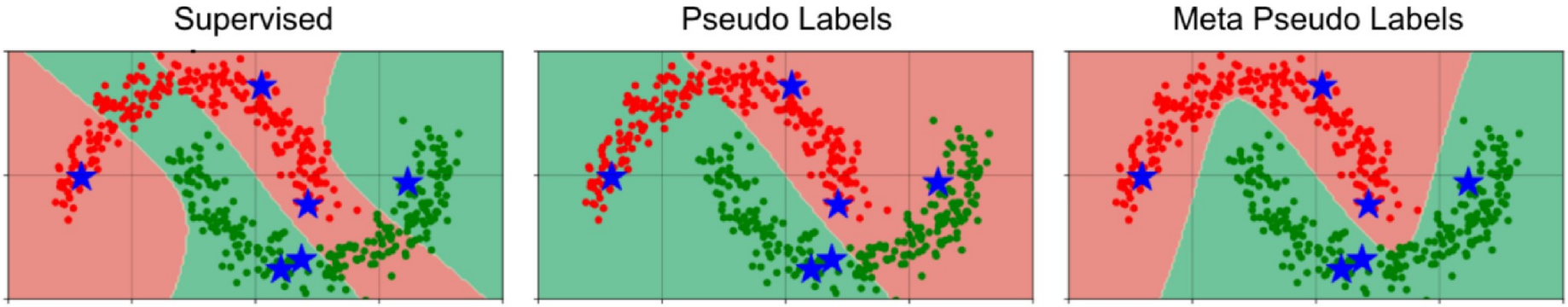
Problem in regular pseudo-labels: **Confirmation bias**

Solution: Use labeled data as validation of student's learning success, adjust the teacher to construct better pseudo-labels

Meta Pseudo Labels: Algorithm

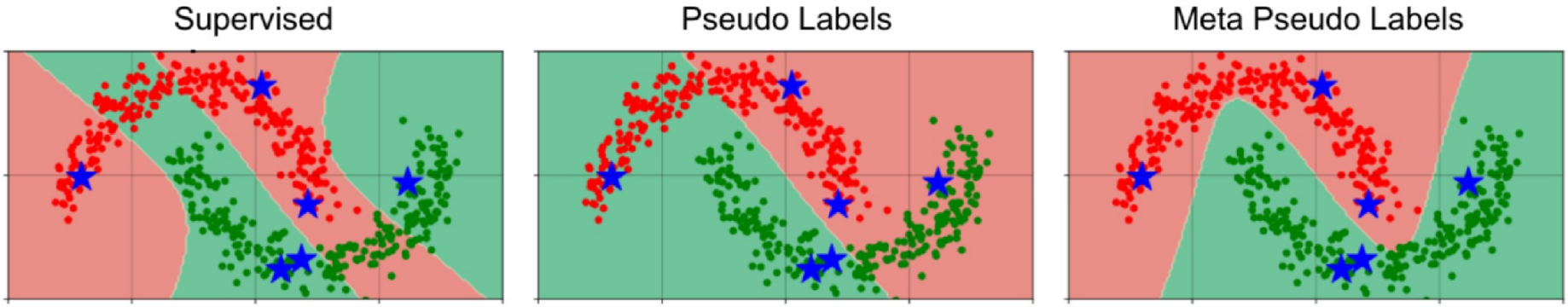
1. Pre-train the teacher model on the labeled dataset
2. Create hard pseudo labels from teacher's prediction on unlabeled data
3. Train student on this batch of pseudo-labeled data
4. Check student's learning progress on batch *L* of labeled data
5. Compute teacher's gradient based on:
    a. Student's performance on *L*
    b. Teacher's performance on the labeled batch *L*
    c. Teacher's performance on consistency regulariser on *L*
6. Update teacher's parameters
7. Go to 2.

Comparison on TwoMoon dataset



- MLP with two hidden layers, eight hidden nodes each
- 1000 points per cluster, three points labeled per cluster
- Pseudo labels based on supervised model

Comparison on TwoMoon dataset



- MLP with two hidden layers, eight hidden nodes each
- 1000 points per cluster, three points labeled per cluster
- Pseudo labels based on supervised model

Discussion: Why does the pseudo labels model not learn the supervised decision boundary?

Experimental settings

Training details

- Architectures that are commonly used in previous work
- Hyperparameters the same as in previous work
- Only dropped some augmentations in RandAugment that don’t make sense
- Fine-tune on labeled data after teacher-student-training
- Last model checkpoint is evaluated, not chosen via validation set, since labeled data is limited

Experimental settings

Training details

- Architectures that are commonly used in previous work
- Hyperparameters the same as in previous work
- Only dropped some augmentations in RandAugment that don’t make sense
- Fine-tune on labeled data after teacher-student-training
- Last model checkpoint is evaluated, not chosen via validation set, since labeled data is limited

Comparisons

- Only compare to methods using the same architecture
- No comparison to methods using self-distillation or distillation from bigger teacher

  “[...] since it is known that larger architectures and distillation can improve any method, possibly including Meta Pseudo Labels.”

Experiments on small datasets

	Method	CIFAR-10-4K (mean ± std)	SVHN-1K (mean ± std)	ImageNet-10% Top-1	ImageNet-10% Top-5
Label Propagation Methods	Temporal Ensemble [35]	83.63 ± 0.63	92.81 ± 0.27	–	–
	Mean Teacher [64]	84.13 ± 0.28	94.35 ± 0.47	–	–
	VAT + EntMin [44]	86.87 ± 0.39	94.65 ± 0.19	–	83.39
	LGA + VAT [30]	87.94 ± 0.19	93.42 ± 0.36	–	–
	ICT [71]	92.71 ± 0.02	96.11 ± 0.04	–	–
	MixMatch [5]	93.76 ± 0.06	96.73 ± 0.31	–	–
	ReMixMatch [4]	94.86 ± 0.04	97.17 ± 0.30	–	–
	EnAET [72]	94.65	97.08	–	–
	FixMatch [58]	95.74 ± 0.05	97.72 ± 0.38	71.5	89.1
	UDA* [76]	94.53 ± 0.18	97.11 ± 0.17	68.07	88.19
Self-Supervised Methods	SimCLR [8, 9]	–	–	71.7	90.4
	MOCOv2 [10]	–	–	71.1	–
	PCL [38]	–	–	–	85.6
	PIRL [43]	–	–	–	84.9
	BYOL [21]	–	–	68.8	89.0
	Meta Pseudo Labels	**96.11 ± 0.07**	**98.01 ± 0.07**	**73.89**	**91.38**
	Supervised Learning with full dataset*	94.92 ± 0.17	97.41 ± 0.16	76.89	93.27

Large-scale experiment: Adding unlabeled data

Method	# Params	Extra Data	ImageNet		ImageNet-ReaL [6]
			Top-1	Top-5	Precision@1
ResNet-50 [24]	26M	–	76.0	93.0	82.94
ResNet-152 [24]	60M	–	77.8	93.8	84.79
DenseNet-264 [28]	34M	–	77.9	93.9	–
Inception-v3 [62]	24M		78.8	94.4	83.58
...					
EfficientNet-B7 [63]	66M	–	85.0	97.2	–
EfficientNet-B7 + FixRes [70]	66M	–	85.3	97.4	–
EfficientNet-L2 [63]	480M	–	85.5	97.5	–
ResNet-50 Billion-scale SSL [79]	26M	3.5B labeled Instagram	81.2	96.0	–
ResNeXt-101 Billion-scale SSL [79]	193M	3.5B labeled Instagram	84.8	–	–
ResNeXt-101 WSL [42]	829M	3.5B labeled Instagram	85.4	97.6	88.19
FixRes ResNeXt-101 WSL [69]	829M	3.5B labeled Instagram	86.4	98.0	89.73
Big Transfer (BiT-L) [33]	928M	300M labeled JFT	87.5	98.5	90.54
Noisy Student (EfficientNet-L2) [77]	480M	300M unlabeled JFT	88.4	98.7	90.55
Noisy Student + FixRes [70]	480M	300M unlabeled JFT	88.5	98.7	–
Vision Transformer (ViT-H) [14]	632M	300M labeled JFT	88.55	–	90.72
EfficientNet-L2-NoisyStudent + SAM [16]	480M	300M unlabeled JFT	88.6	98.6	–
Meta Pseudo Labels (EfficientNet-B6-Wide)	390M	300M unlabeled JFT	90.0	98.7	**91.12**
Meta Pseudo Labels (EfficientNet-L2)	480M	300M unlabeled JFT	**90.2**	**98.8**	91.02

Meta Pseudo Labels - lite version

Reduced Meta Pseudo Labels

1. Train large teacher model T until convergence
2. Pre-compute all *soft* pseudo-labels
3. Small MLP as teacher T' that is trained together with the student:
   Input: pre-computed distributions
   Output: adjusted distributions
4. Train T' and student with Meta Pseudo Labels

Performs slightly better than Noisy Student (+1% on ImageNet, less on smaller datasets)

86.9% on ImageNet instead of 90.2%, but smaller architecture and smaller, different unlabeled dataset.

Conclusion

Pros:

- Semi-supervised method that works on rather small and very large datasets
- Can beat pure supervised learning while using much fewer labels (!)
- Very interesting way to use labeled data as validation set
- Lite version for ‘regular’ users

Cons:

- Unclear how good the reduced model is, since it uses a different experimental setup
- Possibly unfair comparisons on small datasets, since they have the advantage of distillation
- Unclear why they don’t use soft pseudo labels and gradient descent

References

Temporal Ensembling for Semi-Supervised Learning - Samuli Laine, Timo Aila - ICLR 2017 - https://arxiv.org/abs/1610.02242

Meta Pseudo Labels - Hieu Pham, Zihang Dai, Qizhe Xie, Minh-Thang Luong, Quoc V. Le - CVPR 2021 - https://arxiv.org/abs/2003.10580v4

Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results - Antti Tarvainen, Harri Valpola - NIPS 2017 - https://arxiv.org/abs/1703.01780

FixMatch: Simplifying Semi-Supervised Learning with Consistency and Confidence - Kihyuk Sohn, David Berthelot, Chun-Liang Li, Zizhao Zhang, Nicholas Carlini, Ekin D. Cubuk, Alex Kurakin, Han Zhang, Colin Raffel - NeurIPS 2020 - https://arxiv.org/abs/2001.07685

Virtual Adversarial Training: A Regularization Method for Supervised and Semi-Supervised Learning - Takeru Miyato, Shin-Ichi Maeda, Masanori Koyama, Shin Ishii - IEEE Transactions on Pattern Analysis and Machine Intelligence (Volume: 41, Issue: 8, Aug. 1 2019) - https://ieeexplore.ieee.org/abstract/document/8417973

Self-training with Noisy Student improves ImageNet classification - Qizhe Xie, Minh-Thang Luong, Eduard Hovy, Quoc V. Le - CVPR 2020 - https://arxiv.org/abs/1911.04252

Semi-supervised Learning by Entropy Minimization - Yves Grandvalet, Yoshua Bengio - NIPS 2014 - https://www.researchgate.net/profile/Y_Bengio/publication/221618545_Semi-supervised_Learning_by_Entropy_Minimization/links/546b702a0cf2f5eb18091df0.pdf