Unsupervised Data Augmentation For Consistency Training

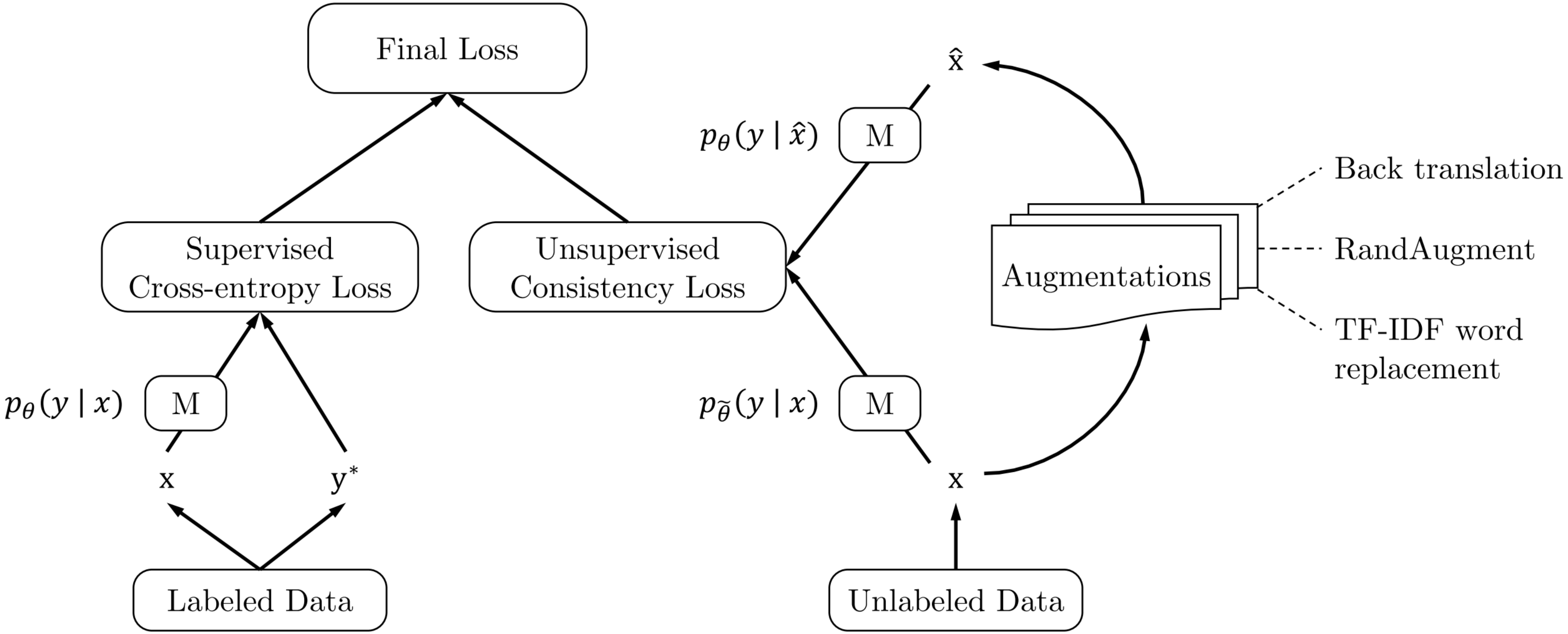
Qizhe Xie[1,2], Zihang Dai[1,2], Eduard Hovy[2], Minh-Thang Luong[1], Quoc V. Le[1]

1 Google Research, Brain Team, 2 Carnegie Mellon University

Motivation

1. Investigate the role of noise injection in consistency training and observe that advanced data augmentation methods, specifically those work best in supervised learning.
2. Despite the promising results, data augmentation is mostly regarded as the “cherry on the cake”, which provides a steady but limited performance boost because these augmentations has so far only been applied to a set of labeled examples which is usually of a small size.

The UDA Framework



$$\min_{\theta} \ \mathcal{J}(\theta) = \mathbb{E}_{x \sim p_L(x)} \left[-\log p_\theta(f^*(x) \mid x) \right] + \lambda \mathbb{E}_{x \sim p_U(x)} \mathbb{E}_{\hat{x} \sim q(\hat{x} \mid x)} \left[\mathrm{CE} \left(p_{\tilde{\theta}}(y \mid x) \| p_\theta(y \mid \hat{x}) \right) \right]$$

Methods of augment on CV

- **AutoAugment**

Use a search method to combine all image processing transformations in the Python Image Library (PIL) to find a good augmentation strategy.

- **RandAugment**

Instead of search, but uniformly sample from the same set of augmentation transformations in PIL

Methods of augment on NLP

- **Back-translation**

Translate an existing example x in language A into another language B and then translating it back into A to obtain an augmented example $\hat{x}$

- **Word replacing with TF-IDF**

Replace uninformative words with low TF-IDF scores while keeping those with high TF-IDF values.

Additional Training Techniques

- **Confidence-based masking**

In each minibatch, the consistency loss term is computed only on examples whose highest probability among classification categories is greater than a threshold β.

$$\frac{1}{|B|}\sum_{x\in B} I(\max_{y'} p_{\tilde{\theta}}(y' \mid x) > \beta)\mathrm{CE}\left(p_{\tilde{\theta}}^{(sharp)}(y \mid x) \| p_{\theta}(y \mid \hat{x})\right)$$

- **Sharpening Predictions**

Regularizing the predictions to have low entropy has been shown to be beneficial

$$p_{\tilde{\theta}}^{(sharp)}(y \mid x) = \frac{\exp(z_y/\tau)}{\sum_{y'} \exp(z_{y'}/\tau)}$$

- **Domain-relevance Data Filtering**

Use baseline model trained on the in-domain data to infer the labels of data in a large out-of-domain dataset and pick out examples that the model is most confident about.

Correlation Between Supervised And Semi-supervised Performances

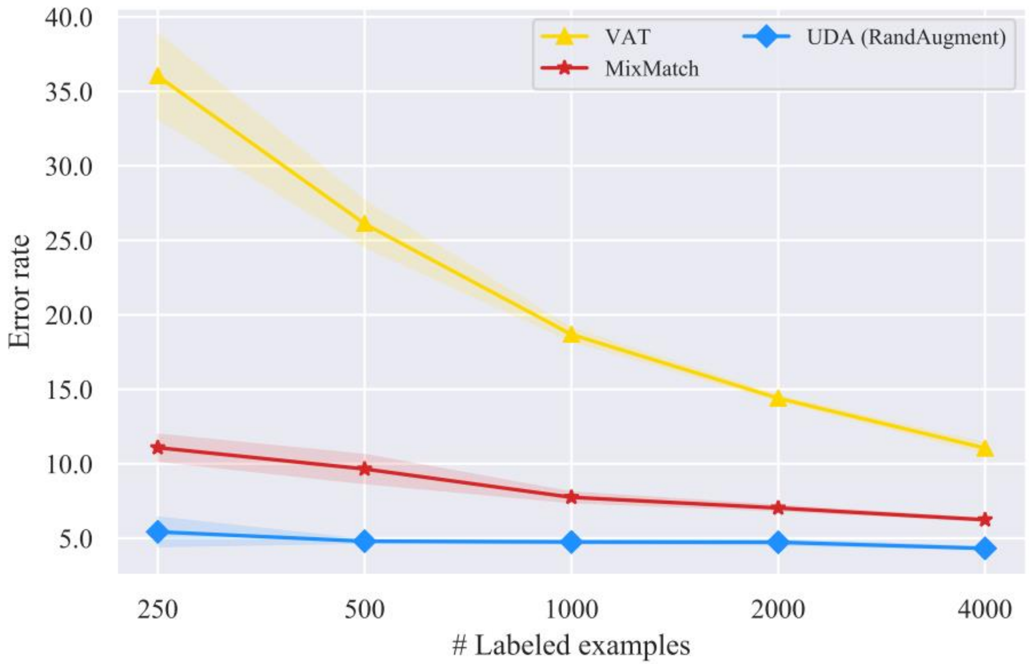
Augmentation (# Sup examples)	Sup (50k)	Semi-Sup (4k)
Crop & flip	5.36	10.94
Cutout	4.42	5.43
RandAugment	**4.23**	**4.32**

Table 1: Error rates on CIFAR-10.

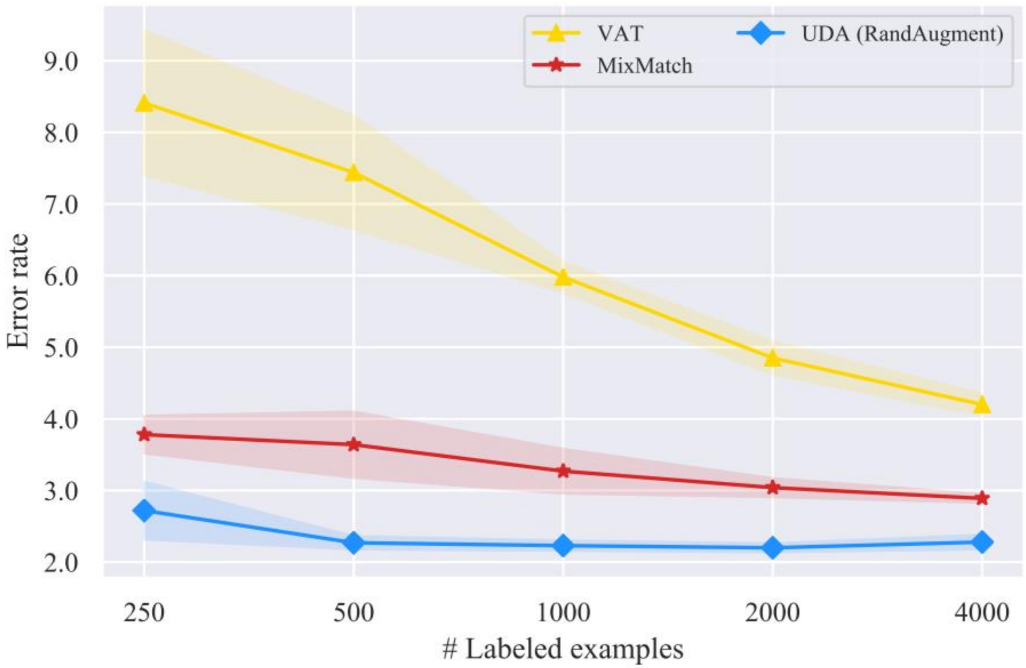
Augmentation (# Sup examples)	Sup (650k)	Semi-sup (2.5k)
✗	38.36	50.80
Switchout	37.24	43.38
Back-translation	**36.71**	**41.35**

Table 2: Error rate on Yelp-5.

Algorithm Comparison On Vision Semi-supervised Learning Benchmarks



(a) CIFAR-10



(b) SVHN

Method	**Model**	**# Param**	CIFAR-10 (4k)	SVHN (1k)
Π-Model (Laine & Aila, 2016)	Conv-Large	3.1M	12.36 ± 0.31	4.82 ± 0.17
Mean Teacher (Tarvainen & Valpola, 2017)	Conv-Large	3.1M	12.31 ± 0.28	3.95 ± 0.19
VAT + EntMin (Miyato et al., 2018)	Conv-Large	3.1M	10.55 ± 0.05	3.86 ± 0.11
SNTG (Luo et al., 2018)	Conv-Large	3.1M	10.93 ± 0.14	3.86 ± 0.27
VAdD (Park et al., 2018)	Conv-Large	3.1M	11.32 ± 0.11	4.16 ± 0.08
Fast-SWA (Athiwaratkun et al., 2018)	Conv-Large	3.1M	9.05	-
ICT (Verma et al., 2019)	Conv-Large	3.1M	7.29 ± 0.02	3.89 ± 0.04
Pseudo-Label (Lee, 2013)	WRN-28-2	1.5M	16.21 ± 0.11	7.62 ± 0.29
LGA + VAT (Jackson & Schulman, 2019)	WRN-28-2	1.5M	12.06 ± 0.19	6.58 ± 0.36
mixmixup (Hataya & Nakayama, 2019)	WRN-28-2	1.5M	10	-
ICT (Verma et al., 2019)	WRN-28-2	1.5M	7.66 ± 0.17	3.53 ± 0.07
MixMatch (Berthelot et al., 2019)	WRN-28-2	1.5M	6.24 ± 0.06	2.89 ± 0.06
Mean Teacher (Tarvainen & Valpola, 2017)	Shake-Shake	26M	6.28 ± 0.15	-
Fast-SWA (Athiwaratkun et al., 2018)	Shake-Shake	26M	5.0	-
MixMatch (Berthelot et al., 2019)	WRN	26M	4.95 ± 0.08	-
UDA (RandAugment)	WRN-28-2	1.5M	4.32 ± 0.08	$\mathbf{2.23} \pm \mathbf{0.07}$
UDA (RandAugment)	Shake-Shake	26M	3.7	-
UDA (RandAugment)	PyramidNet	26M	**2.7**	-

EVALUATION ON TEXT CLASSIFICATION DATASETS

Fully supervised baseline							
Datasets (# Sup examples)		IMDb (25k)	Yelp-2 (560k)	Yelp-5 (650k)	Amazon-2 (3.6m)	Amazon-5 (3m)	DBpedia (560k)
Pre-BERT SOTA		*4.32*	2.16	29.98	3.32	34.81	0.70
BERT$_{\text{LARGE}}$		4.51	*1.89*	*29.32*	*2.63*	*34.17*	*0.64*
Semi-supervised setting							
Initialization	**UDA**	IMDb (20)	Yelp-2 (20)	Yelp-5 (2.5k)	Amazon-2 (20)	Amazon-5 (2.5k)	DBpedia (140)
Random	✗	43.27	40.25	50.80	45.39	55.70	41.14
	✓	25.23	8.33	41.35	16.16	44.19	7.24
BERT$_{\text{BASE}}$	✗	18.40	13.60	41.00	26.75	44.09	2.58
	✓	5.45	2.61	33.80	3.96	38.40	1.33
BERT$_{\text{LARGE}}$	✗	11.72	10.55	38.90	15.54	42.30	1.68
	✓	4.78	2.50	33.54	3.93	37.80	1.09
BERT$_{\text{FINETUNE}}$	✗	6.50	2.94	32.39	12.17	37.32	-
	✓	**4.20**	**2.05**	**32.08**	**3.50**	**37.12**	-

SCALABILITY TEST ON THE IMAGENET DATASET

Methods	SSL	10%	100%
ResNet-50 w. RandAugment	✗	55.09 / 77.26 58.84 / 80.56	77.28 / 93.73 78.43 / 94.37
UDA (RandAugment)	✓	**68.78 / 88.80**	**79.05 / 94.49**