UNSUPERVISED DATA AUGMENTATION FOR CONSISTENCY TRAINING

Puzanova Anastasia
HSE
November 28, 2019

Plan

- Introduction. General idea
- Consistency training
- Augmentation strategies:
    - RandAugment
    - Back-translation
    - Word replacing with TF-IDF
- Training Signal Annealing
- Experiments
- Conclusions

Introduction

- Deep learning requires a lot of labeled data to work well
- Semi-supervised learning (SSL) use unlabeled data to address this weakness
- Data augmentation is often limited to supervised learning only
- In UDA we perform data augmentation on unlabeled data to improve SSL

Notation

- solving classification problem
- x - input
- y^*- ground-truth prediction target
- $p_\theta(y \mid x)$ - model, predicting y^* based on x with parameters θ
- L, U - sets of labeled and unlabeled examples respectively

Augmentation

Aim: create novel and realistic-looking data without changing label

- Let $q(\hat{x} \mid x)$ - augmentation transformation to draw $\hat{x}$ from x
- Transformation is valid if any example $\hat{x} \sim q(\hat{x} \mid x)$ shares the same ground-truth label as x
- Minimize divergence metric $D(p_\theta(y \mid x), p_\theta(y \mid \hat{x}))$

UDA

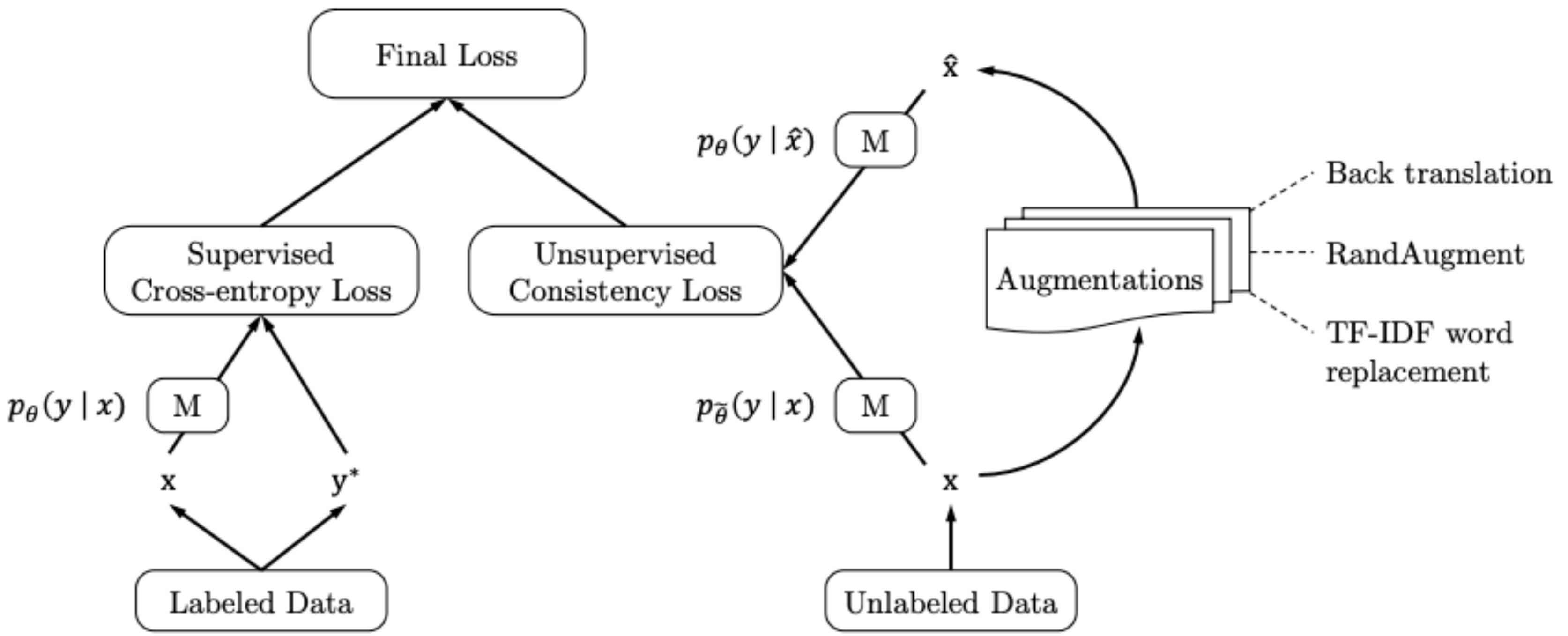


Figure 1: Training objective for UDA, where M is a model that predicts a distribution of y given x.

$$\boxed{\min_{\theta} \; \mathcal{J}(\theta) = \mathbb{E}_{x,y^* \in L}\left[-\log p_\theta(y^* \mid x)\right] + \lambda \mathbb{E}_{x \in U} \mathbb{E}_{\hat{x} \sim q(\hat{x} \mid x)}\left[\mathcal{D}_{\mathrm{KL}}\left(p_{\tilde{\theta}}(y \mid x) \,\|\, p_\theta(y \mid \hat{x})\right)\right]}$$

UDA

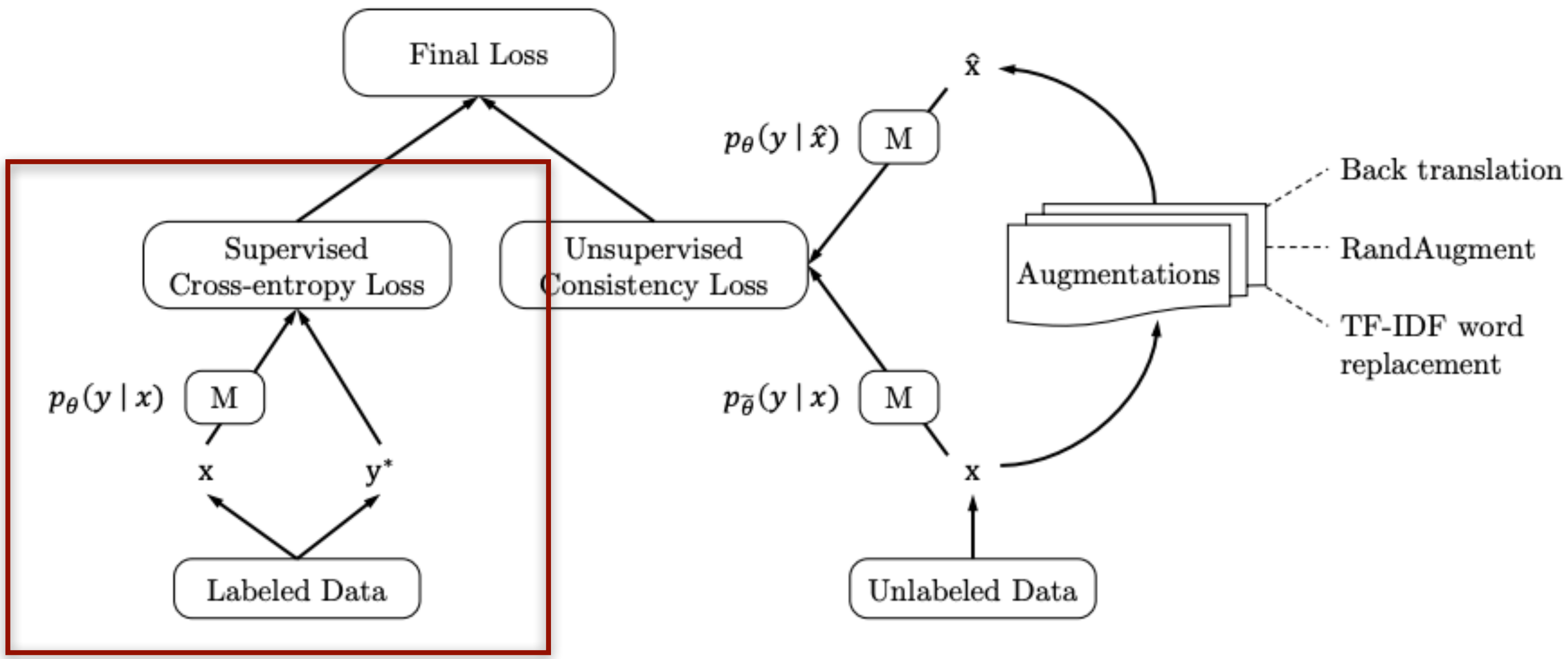


Figure 1: Training objective for UDA, where M is a model that predicts a distribution of y given x.

$$\boxed{\min_{\theta} \ \mathcal{J}(\theta) = \mathbb{E}_{x,y^* \in L}\left[-\log p_\theta(y^* \mid x)\right] + \lambda \mathbb{E}_{x \in U} \mathbb{E}_{\hat{x} \sim q(\hat{x} \mid x)}\left[\mathcal{D}_{\mathrm{KL}}\left(p_{\tilde{\theta}}(y \mid x) \,\|\, p_\theta(y \mid \hat{x})\right)\right]}$$

UDA

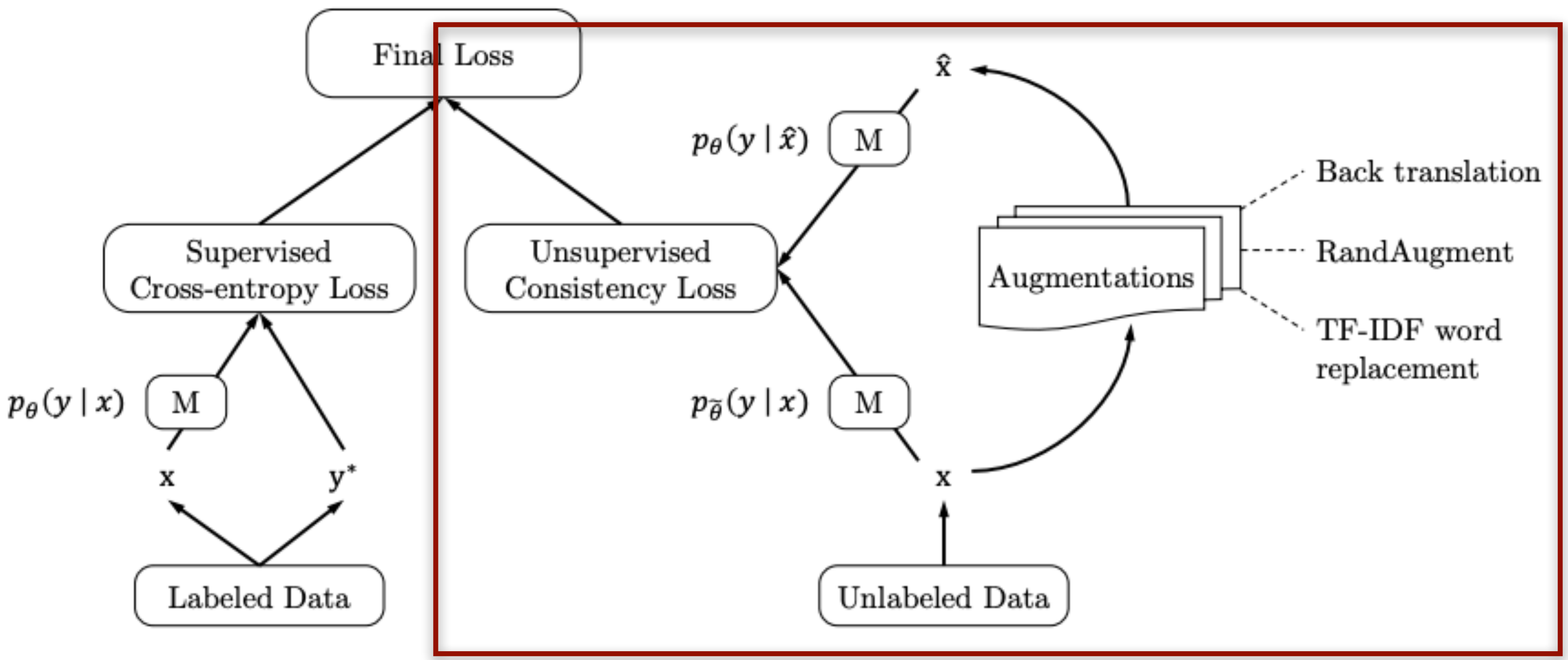


Figure 1: Training objective for UDA, where M is a model that predicts a distribution of y given x.

$$\boxed{\min_\theta \ \mathcal{J}(\theta) = \mathbb{E}_{x,y^* \in L}\left[-\log p_\theta(y^* \mid x)\right] + \lambda \mathbb{E}_{x \in U} \mathbb{E}_{\hat{x} \sim q(\hat{x}|x)}\left[\mathcal{D}_{\mathrm{KL}}\left(p_{\tilde{\theta}}(y \mid x) \,\|\, p_\theta(y \mid \hat{x})\right)\right]}$$

Objective

$$\min_{\theta} \mathcal{J}(\theta) = \mathbb{E}_{x,y^* \in L}\left[-\log p_\theta(y^* \mid x)\right] + \lambda \mathbb{E}_{x \in U} \mathbb{E}_{\hat{x} \sim q(\hat{x} \mid x)} \left[\mathcal{D}_{\mathrm{KL}}\left(p_{\tilde{\theta}}(y \mid x) \,\|\, p_\theta(y \mid \hat{x})\right)\right]$$

- $q(\hat{x} \mid x)$ - data augmentation transformation
- $\hat{\theta}$ - fixed copy of current parameters (gradient is not propagated)
- $\lambda = 1$ at most of experiments

Advanced augmentation intuition

- **Valid noise** - advanced augmentation generate realistic examples
- **Diverse noise** - advanced augmentation can make larger modifications
- **Targeted inductive biases**

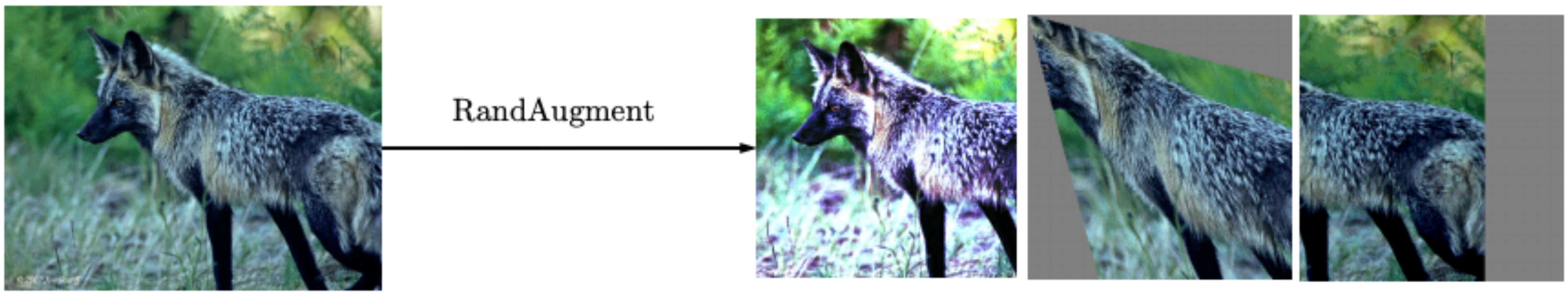
Augmentation strategies: RandAugment

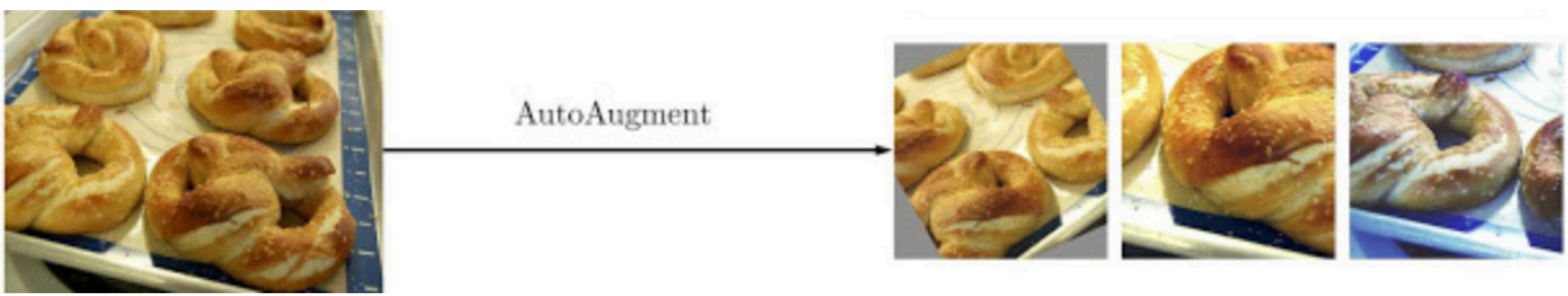
- Image classification task
- Inspired by AutoAugment
- Uniformly sample from the set of augmentation transformations in PIL
- Requires no labeled data

RandAugment details

- two operations represented by transformation name, probability and magnitude [(Sharpness, 0.6, 2), (Posterize, 0.3, 9)]
- 15 possible transformation
- magnitude from 1 to 9
- probability 0.5
- tuning hyperparameters might result in higher accuracy

Augmentation strategies: RandAugment

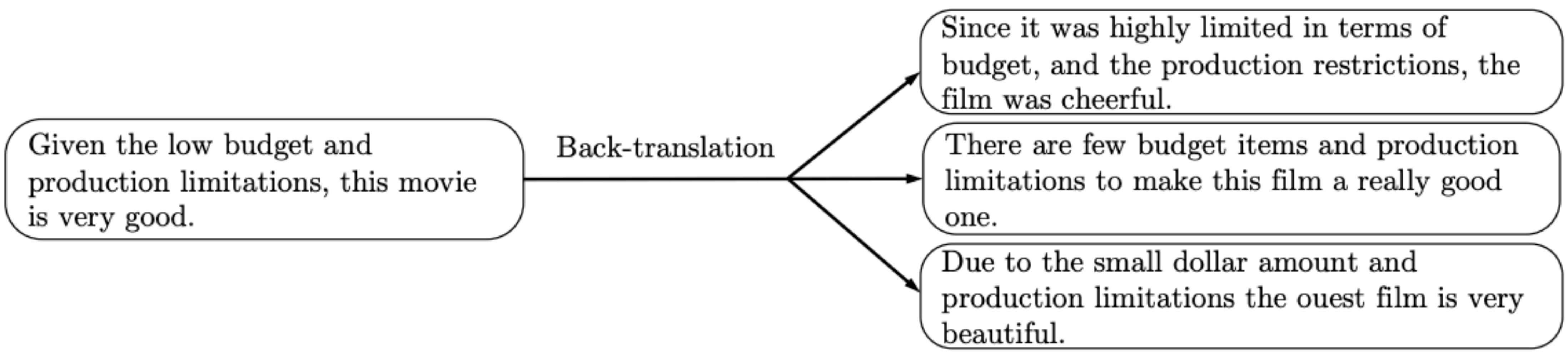




Augmentation strategies: Back-translation

- Text classification task
- Translate example in language *A* to language *B* and translate back to *A*
- Can generate diverse paraphrases while preserving semantic
- Diversity is more important then quality or validity
- Tunable temperature instead of beam search

Augmentation strategies: Back-translation



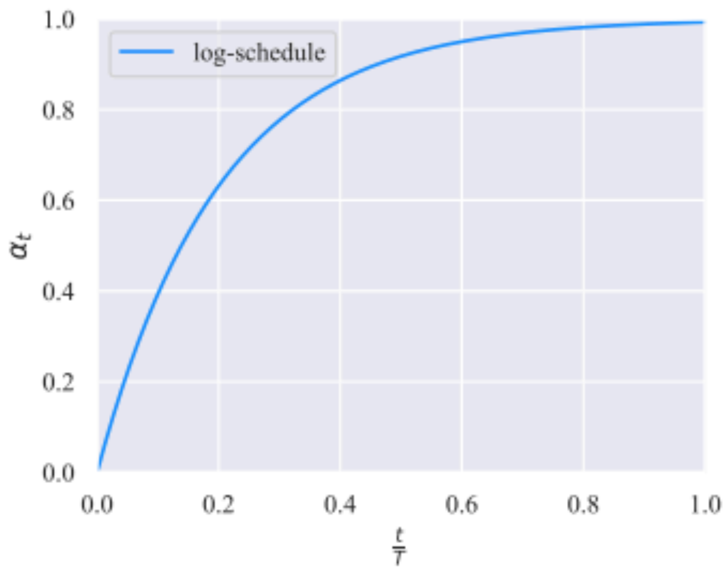
Augmentation strategies: Word replacing with TF-IDF

- Back-translation has little control over which words will be retained
- May be important for topic classification tasks
- Replace words with low TF-IDF and keep words with high TF-IDF

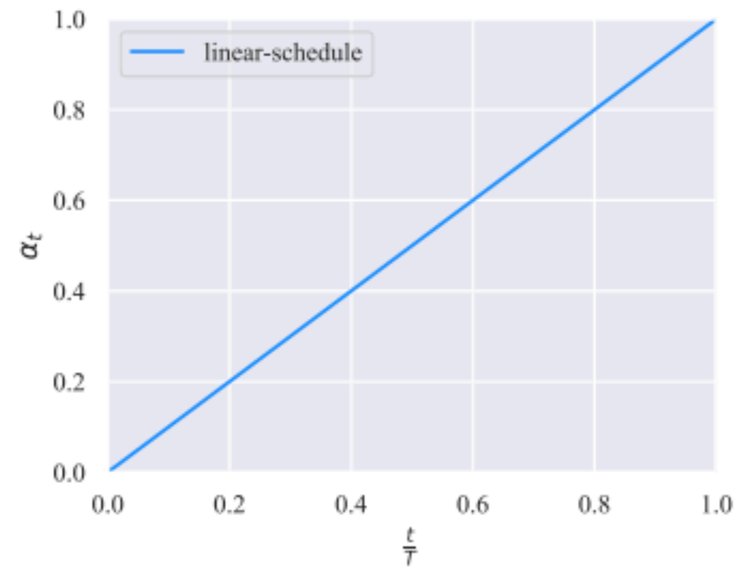
Training Signal Annealing

- Often there is much more unlabeled data
- Model quickly overfits labeled data while underfitting unlabeled
- Utilize labeled example if model's confidence lower then threshold η_t

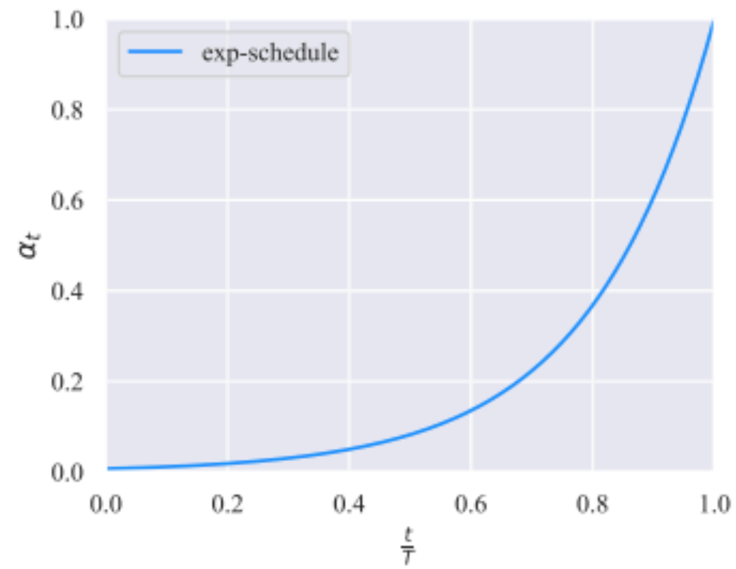
Training Signal Annealing



$$\alpha_t = 1 - \exp(-\frac{t}{T} \cdot 5)$$



$$\alpha_t = \frac{t}{T}$$



$$\alpha_t = \exp((\frac{t}{T} - 1) \cdot 5)$$

$$\eta_t = \alpha_t \cdot (1 - \frac{1}{K}) + \frac{1}{K}$$

T - number of training steps

t - current step

K - number of categories

Experiments:semi-supervised vs fully-supervised

Augmentation (# Sup examples)	Sup (50k)	Semi-Sup (4k)
Crop & flip	5.36	16.17
Cutout	4.42	6.42
RandAugment	**4.23**	**5.29**

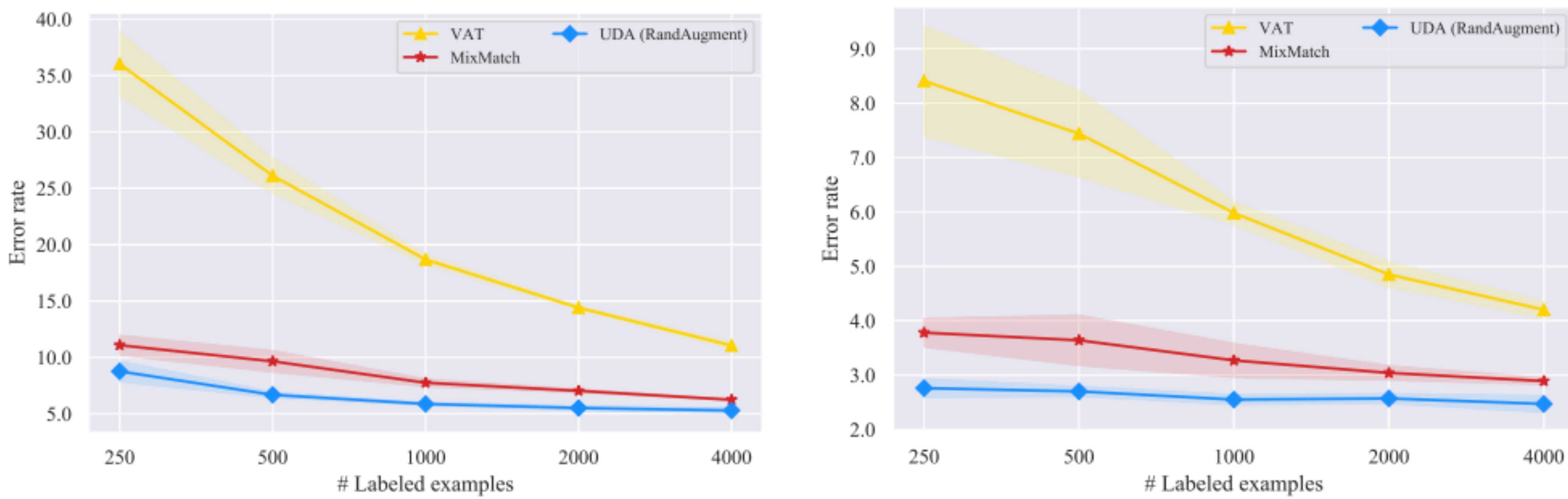
Table 1: Error rates on CIFAR-10.

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.

Correlation of augmentation effectiveness in supervised and semi-supervised learning

Experiments: vision semi-supervised benchmarks



(a) CIFAR-10 (b) SVHN

- UDA outperforms two baselines
- VAT differs from UDA essentially in the noise process

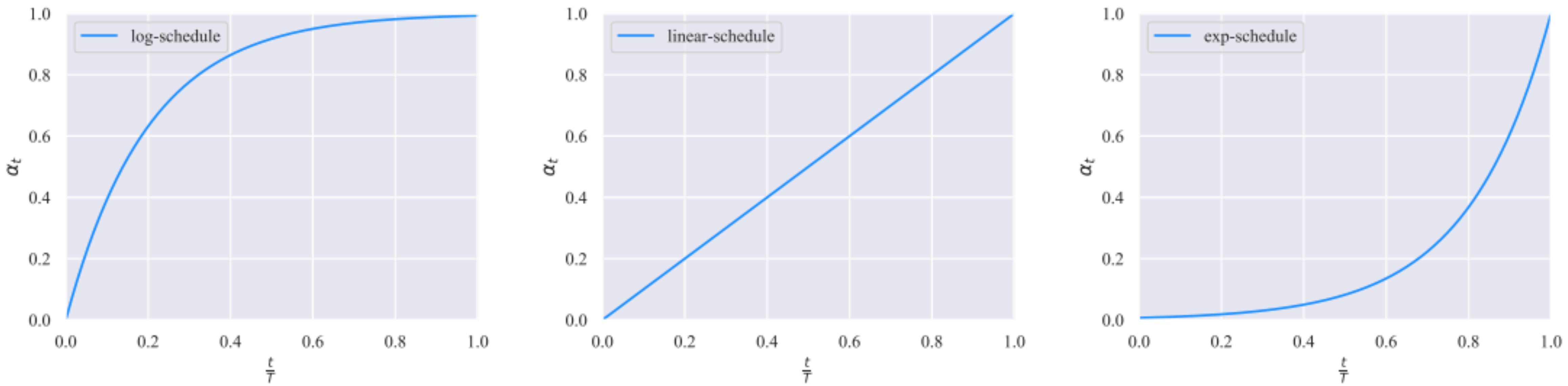
Experiments: text classification

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
$\text{BERT}_{\text{LARGE}}$		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
$\text{BERT}_{\text{BASE}}$	✗	18.40	13.60	41.00	26.75	44.09	2.58
	✓	5.45	2.61	33.80	3.96	38.40	1.33
$\text{BERT}_{\text{LARGE}}$	✗	11.72	10.55	38.90	15.54	42.30	1.68
	✓	4.78	2.50	33.54	3.93	37.80	1.09
$\text{BERT}_{\text{FINETUNE}}$	✗	6.50	2.94	32.39	12.17	37.32	-
	✓	**4.20**	**2.05**	**32.08**	**3.50**	**37.12**	-

- UDA is complimentary to transfer learning
- Competitive results for binary classification

Experiments: TSA

- Yelp-5 has 2.5k labeled examples 6m unlabeled
- CIFAR-10 has 4k labeled examples 50k unlabeled



Which is better?

Experiments: TSA

TSA schedule	Yelp-5	CIFAR-10
✗	50.81	5.67
log-schedule	49.06	5.67
linear-schedule	45.41	**5.29**
exp-schedule	**41.35**	7.81

- Yelp-5 has 2.5k labeled examples 6m unlabeled
- CIFAR-10 has 4k labeled examples 50k unlabeled

Conclusions

- Better data augmentation can lead to significantly better semi-supervised learning
- State-of-the-art supervised augmentations performs good at semi-supervised learning
- UDA can match and outperform purely supervised learning
- For text classification great results on IMDb with only 20 labeled examples
- For vision tasks nearly matches the performance of fully supervised models

Questions

1. Запишите функционал, оптимизируемый в consistency training, поясните формулу.
2. Какие 3 основных вида аугментации были рассмотрены в статье? Кратко опишите одну из них.
3. Назовите пороги для TSA, рассмотренные в статье. В каком случае стоит применять каждый из них?

Links

- Unsupervised data augmentation for consistency training: https://arxiv.org/pdf/1904.12848.pdf
- Interpolation consistency training for semi-supervised learning: https://arxiv.org/pdf/1903.03825.pdf
- Github: https://github.com/google-research/uda