Introduction to Data Augmentation

Hongyang R. Zhang

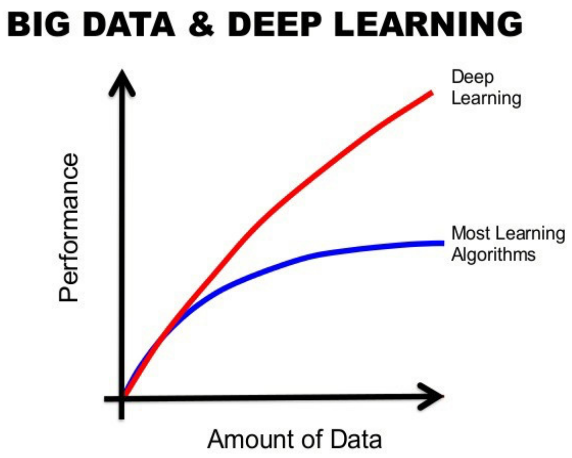
Lecture 10

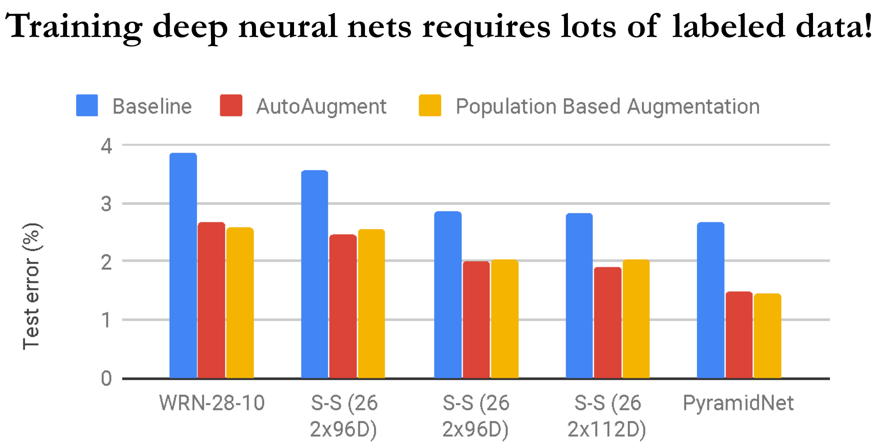
Lecture plan

- An overview of data augmentation
- A theoretical framework that precisely analyzes the generalization properties of data augmentation
- Research trends
  - Semi-supervised learning
  - Text classification

Why data augmentation

Neural net training, getting labeled data, and data augmentation



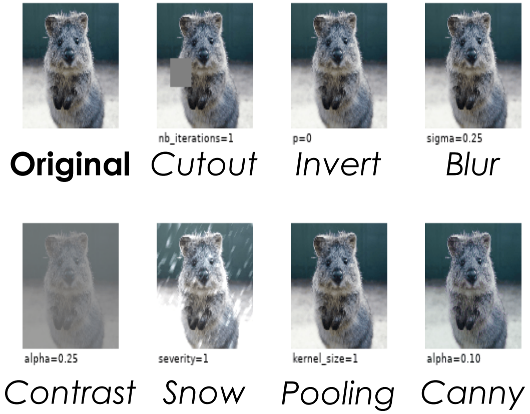


- In image classification, data augmentation has become standard practice [e.g. ResNet and follow-up works, Ratner et al'17, Cubuk et al'18]
- In text classification, reinforcement learning, meta learning etc, data augmentation is an emerging approach!

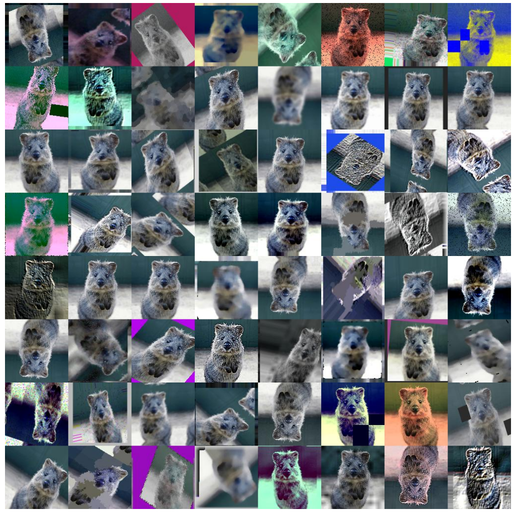
Figure credit: https://github.com/aleju/imgaug, https://towardsdatascience.com/7-practical-deep-learning-tips-97a9f514100e

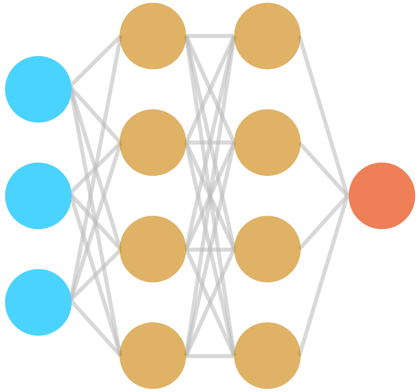
How data augmentation works?

A list of image transformations



Neural net training with automatic labeled data generation





"Quokka"

Data augmentation in text classification

➢ Textual data augmentation example (cf. nlpaug@github)

	Sentence
Original	The quick brown fox jumps over the lazy dog
Synonym (PPDB)	The quick brown fox climbs over the lazy dog
Word Embeddings (word2vec)	The easy brown fox jumps over the lazy dog
Contextual Word Embeddings (BERT)	Little quick brown fox jumps over the lazy dog
PPDB + word2vec + BERT	Little easy brown fox climbs over the lazy dog

➢ Other examples:

- ➢ A concatenation of cased and lowercased training data [ner and pos when nothing is capitalized, Mayhew et al'19]
- ➢ Replacing fragments with other fragments that appear in at least one similar environment [Andreas 20]

Major challenges in data augmentation

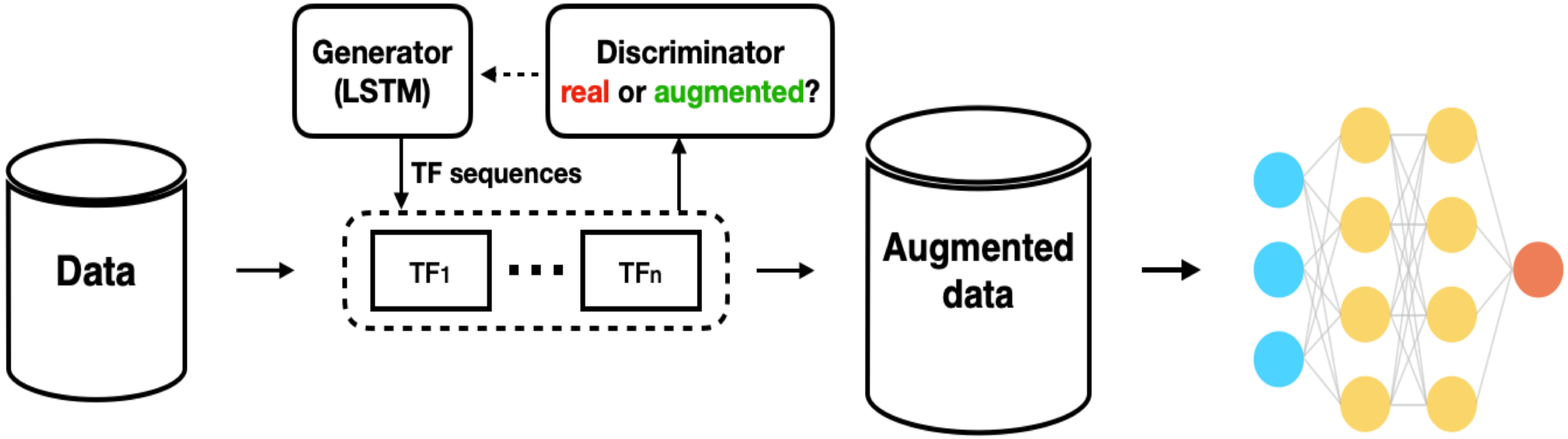
- Some transformations may not help
  - Depends on the dataset and the prediction task
- With composition (of multiple transformations), the search space grows polynomially

- Existing work in this direction
  - RL-based search
  - Random sampling

RL-based search

RL-based search

- Discriminator: is the generated image real or augmented?
- Generator: what kind of images are difficult to recognize by the discriminator?

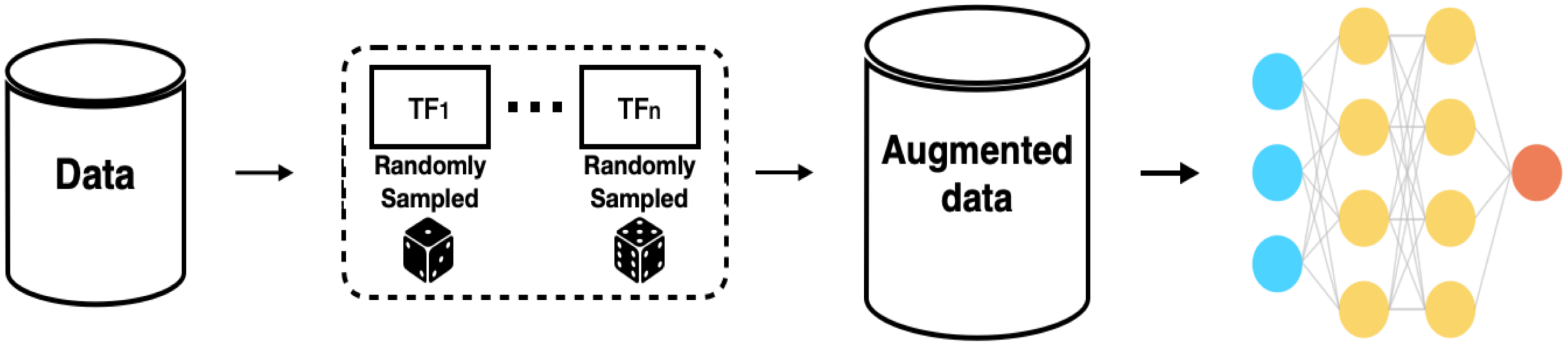


TANDA [Ratner et al.'17]
AutoAugment [Cubuk et al.'18]

Random sampling

Random sampling

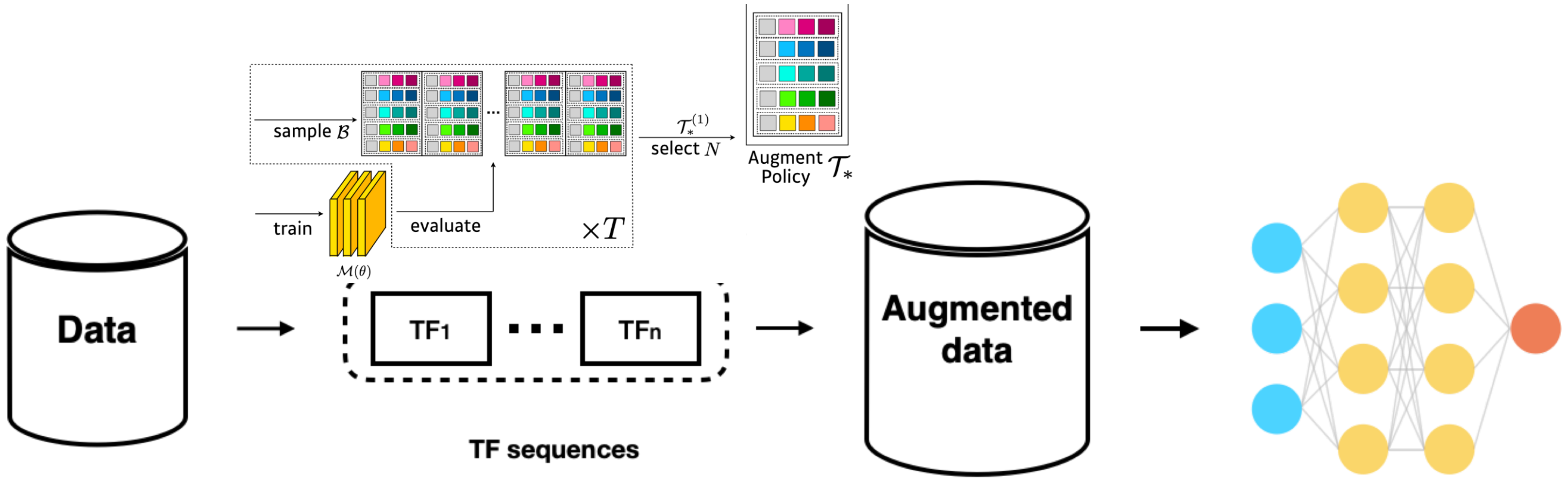
- Generate n new images, randomly sample one for training



RandAugment [Cubuk et al.'19]

Bayesian optimization

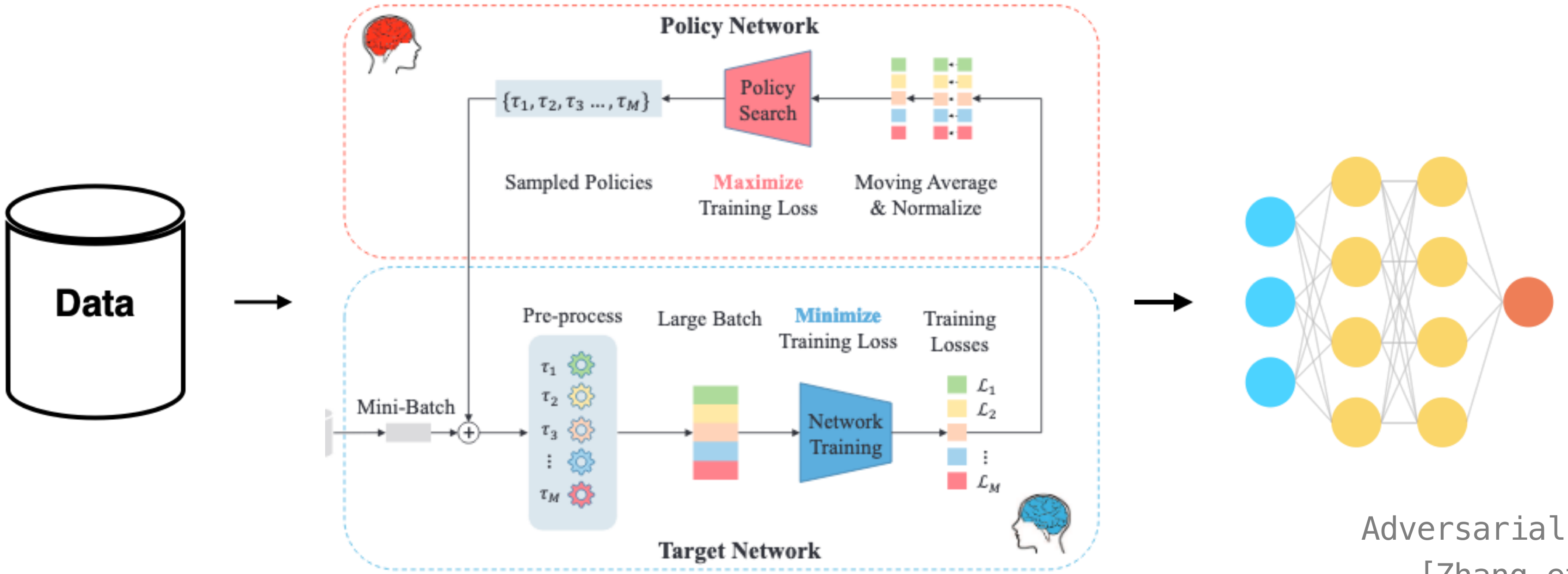
- Imagine that the parameters follow a Gaussian distribution. Can we learn the parameters?
  - Based on a well-known connection between RL and multi-armed bandit [Gaussian Process Optimization in the Bandit Setting: No Regret and Experimental Design, Srinivas et al'10]



Fast AutoAugment [Lim et al.'19]

Adversarial training

- Simultaneously optimize a target network for prediction and an (augmentation) policy network
  - Policy network: generate adversarial policies that increase target network's loss
  - Target network: learn from policy network's generated examples



Adversarial AutoAugment [Zhang et al.'20]

Generalization effects of data augmentation

A theoretical framework

A broad context

- Motivating question: a principled understanding of these transformations and search techniques seems mostly unexplored
- Data augmentation allows the model to generalize to unseen data better [SK'19]

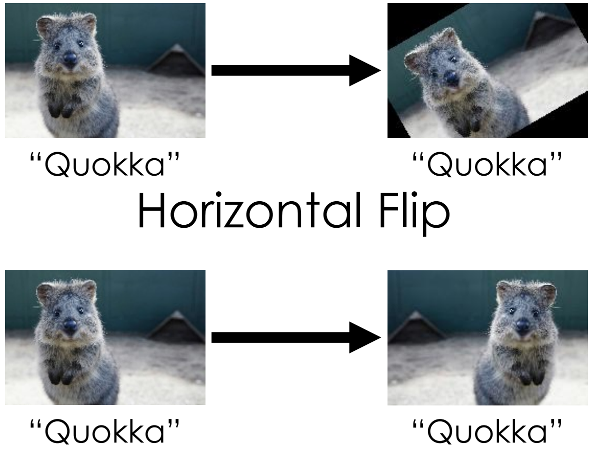
- **This work**
- Goal: a theoretical framework that precisely analyzes the benefit of data augmentation
  - Algorithm: biased sampling that selects useful transformations more efficiently

Theoretical framework

Linear transformations: a large family of image transformations

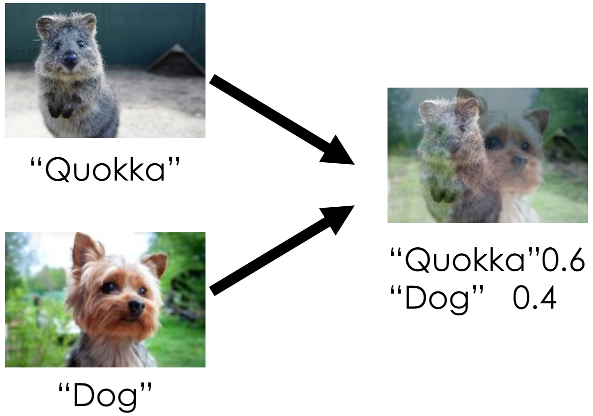
Label-invariant transformations

Rotation



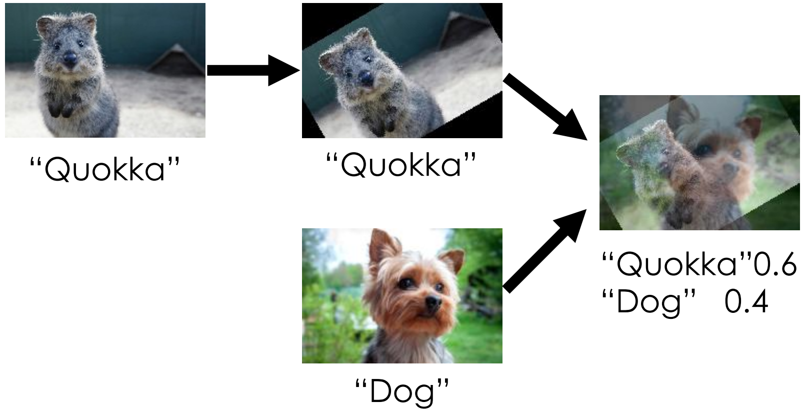
Label-mixing transformations

Mixup [Zhang et al.'17]



Composition of transformations

Rotation @ Mixup



Problem formulation

- Label-invariant (base) transformation $\boldsymbol{F} \in \mathbb{R}^{\boldsymbol{d} \times \boldsymbol{d}}$ and a training sample $(\boldsymbol{x}, \boldsymbol{y})$
  - Transformed sample: $(\boldsymbol{F}\boldsymbol{x}, \boldsymbol{y})$
- Label-mixing transformation mixup [Zhang et al. '17] and two training samples $(\boldsymbol{x}_1, \boldsymbol{y}_1), (\boldsymbol{x}_2, \boldsymbol{y}_2)$
  - Transformed sample: $(\boldsymbol{\alpha} \cdot \boldsymbol{x}_1 + (\mathbf{1} - \boldsymbol{\alpha}) \cdot \boldsymbol{x}_2,\ \boldsymbol{\alpha} \cdot \boldsymbol{y}_1 + (\mathbf{1} - \boldsymbol{\alpha}) \cdot \boldsymbol{y}_2)$
- Composition of two label-invariant transformations $\boldsymbol{F}_1 \in \mathbb{R}^{\boldsymbol{d} \times \boldsymbol{d}}, \boldsymbol{F}_2 \in \mathbb{R}^{\boldsymbol{d} \times \boldsymbol{d}}$
  - Transformed sample: $(\boldsymbol{F}_1 \boldsymbol{F}_2 \boldsymbol{x},\ \boldsymbol{y})$

Problem formulation (cont'd)

- Setting: over-parametrized linear regression
- Training data: feature vectors $\boldsymbol{X} = [\boldsymbol{x}_1 \in \mathbb{R}^p, \boldsymbol{x}_2 \in \mathbb{R}^p, \dots, \boldsymbol{x}_n \in \mathbb{R}^p]$, labels $\boldsymbol{Y} = \boldsymbol{X\beta} + \boldsymbol{\varepsilon}$.
- Assumption: # parameters $\boldsymbol{p} > \boldsymbol{n}$ #samples
- Ridge estimator: add an $\boldsymbol{\ell_2}$ regularization w/ parameter $\boldsymbol{\lambda}$

$$L(\hat{\beta}) = \|X\hat{\beta} - Y\|_2^2 + \lambda \cdot \|\hat{\beta}\|_2^2$$

- Question: how does adding transformed samples impact the ridge estimator's generalization error?

Provable improvement

- **Question:** how does the estimation error of the ridge estimator $\hat{\boldsymbol{\beta}}(\boldsymbol{X}, \boldsymbol{Y})$ compare to the augmented ridge estimator $\hat{\boldsymbol{\beta}}_{\boldsymbol{aug}} = \hat{\boldsymbol{\beta}}(\boldsymbol{X}_{\boldsymbol{aug}}, \boldsymbol{Y}_{\boldsymbol{aug}})$?

- **Result 1:** For one sample x and a label-invariant transformation F, adding the transformed sample reduces the estimation error of the ridge estimator

$$\boldsymbol{e}(\hat{\boldsymbol{\beta}}) - \boldsymbol{e}(\hat{\boldsymbol{\beta}}_{\boldsymbol{aug}}) \geq \frac{\left(\boldsymbol{\beta}^\top \boldsymbol{P}_{\boldsymbol{X}}^{\perp} \boldsymbol{F} \boldsymbol{x}\right)^2}{\boldsymbol{n}}$$

- **Intuition:** The transformed sample adds a new direction outside the span of the training data, which does not cover the entire space because # samples < dimension.

Notation: $P_X^{\perp}$ denotes the projection to the orthogonal subspace of X

Provable improvement

- **Question:** how does the estimation error of the ridge estimator $\widehat{\boldsymbol{\beta}}(\boldsymbol{X}, \boldsymbol{Y})$ compare to the augmented ridge estimator $\widehat{\boldsymbol{\beta}}_{\boldsymbol{aug}} = \widehat{\boldsymbol{\beta}}(\boldsymbol{X}_{\boldsymbol{aug}}, \boldsymbol{Y}_{\boldsymbol{aug}})$?

- **Result 2:** For two random samples x_1, x_2, adding the mixup samples $x^{aug} = \alpha x_1 + (1-\alpha)x_2$ reduces estimation error

$$\boldsymbol{e}(\widehat{\boldsymbol{\beta}}) - \boldsymbol{e}(\widehat{\boldsymbol{\beta}}_{\boldsymbol{aug}}) \geq \frac{\boldsymbol{\lambda}^2 \|\boldsymbol{X}\boldsymbol{\beta}\|^2}{\boldsymbol{n}^2}$$

- **Intuition:** Regularization via shrinking the training data

Using $\mathbb{E}\left[\boldsymbol{x}^{aug} \boldsymbol{x}^{aug\top}\right] = (\mathbf{1} - \mathbf{2}\boldsymbol{\alpha})^2 \frac{\boldsymbol{X}^\top \boldsymbol{X}}{\boldsymbol{n}}$

$$\Rightarrow \mathbb{E}\left[\frac{\boldsymbol{X}^\top \boldsymbol{X} + \boldsymbol{x}^{aug} \boldsymbol{x}^{aug\top}}{\boldsymbol{n}+\mathbf{1}}\right] = \underbrace{\left(\frac{\boldsymbol{n}}{\boldsymbol{n}+\mathbf{1}} + \frac{(\mathbf{1}-\mathbf{2}\boldsymbol{\alpha})^2}{(\boldsymbol{n}+\mathbf{1})}\right)}_{\text{Less than one!}} \frac{\boldsymbol{X}^\top \boldsymbol{X}}{\boldsymbol{n}}$$

Provable improvement

- **Question:** how does the estimation error of the ridge estimator $\hat{\beta}(X, Y)$ compare to the augmented ridge estimator $\hat{\beta}_{aug} = \hat{\beta}(X_{aug}, Y_{aug})$?

- **Result 3:** For a sample x and two label-invariant transformations, adding the transformed sample reduces estimation error

$$e(\hat{\beta}) - e(\hat{\beta}_{aug}) \geq \frac{\left(\beta^{\top} P_X^{\perp} F_1 F_2 x\right)^2}{n}$$

- **Intuition:** Further expands search space

Bias and variance metrics

- **Question:** How do we measure generalization effects in a practical scenario?
- **Idea:** Separate the randomness from the deterministic part. Train an ensemble of models.

$\hat{y}_1$	$\hat{y}_2$	$\hat{y}_3$	$\hat{y}_4$	$\hat{y}_5$	majority	true
+	-	-	+	+	+	+

- **Error score:** measure acc. of majority label. Ex. correct
- **Instability score:** measure % of mislabels compared to majority label. Ex. 40%

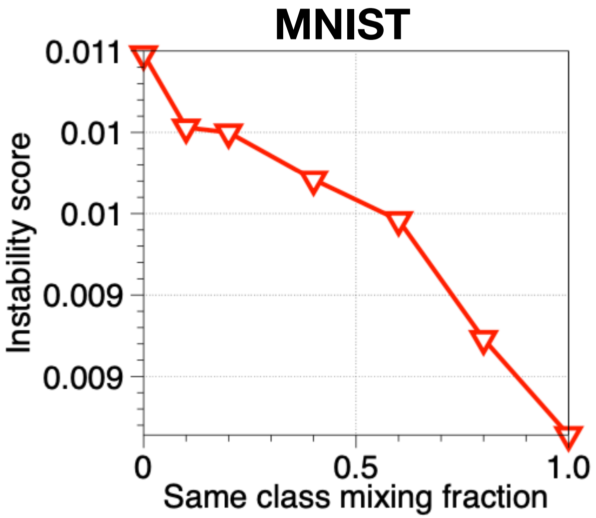
Validation on MNIST

- **Observation 1:** Label-invariant transformations indeed reduce the error score!

	Avg. Acc.	Error Score	Instab. Score
Baseline	98.08%	1.52%	0.95%
Cutout	98.31%	1.43%	0.86%
RandCrop	98.61%	**1.01%**	0.88%
Rotation	**98.65%**	1.08%	**0.77%**

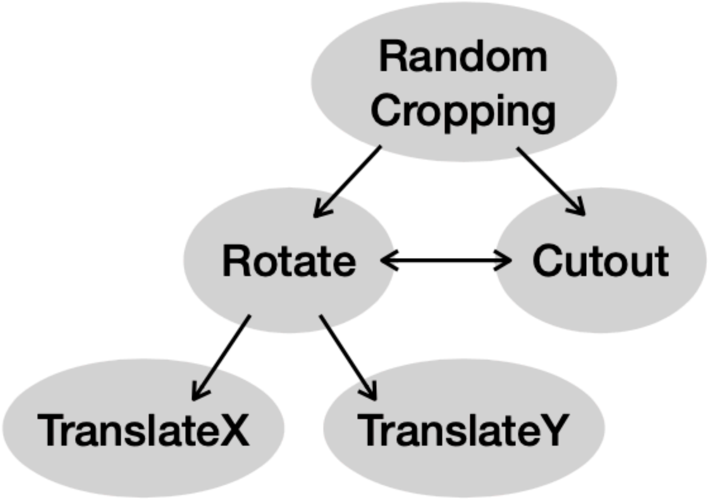
Validation on MNIST

- **Observation 2:** Mixup reduces the instability score as we increase the fraction of mixing same-class digits



Validation on MNIST

- **Observation 3:** On MNIST, translations do not add new information on top of rotate, cutout, and random cropping

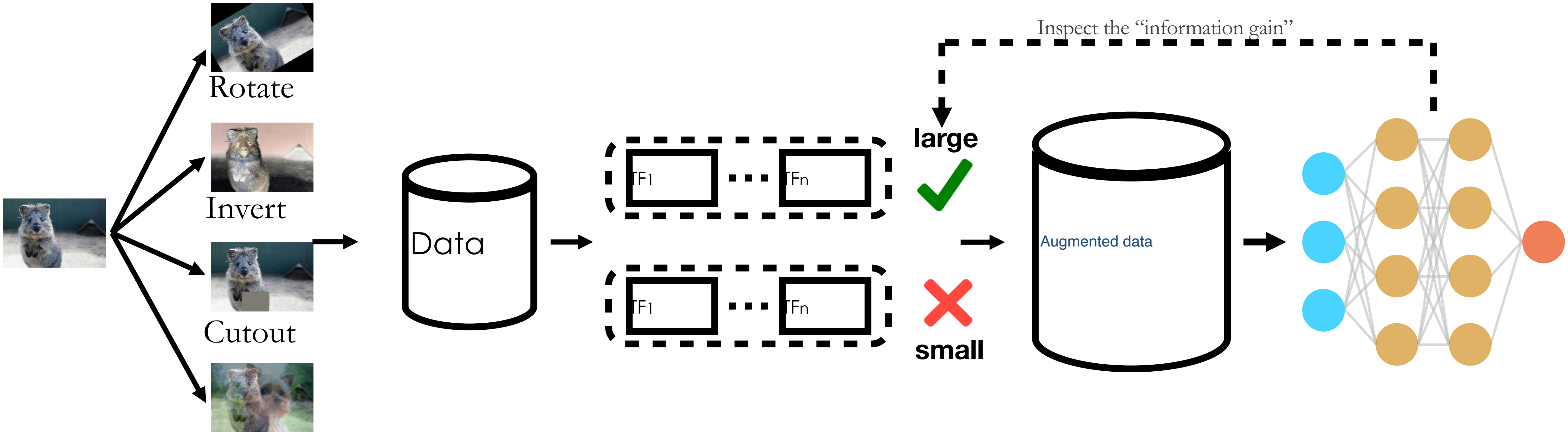


Uncertainty-based sampling

Step 1: Users provide transformation functions

Step 2: Randomly sample K transformation functions

Step 3: Model selects TFs with the highest loss during training

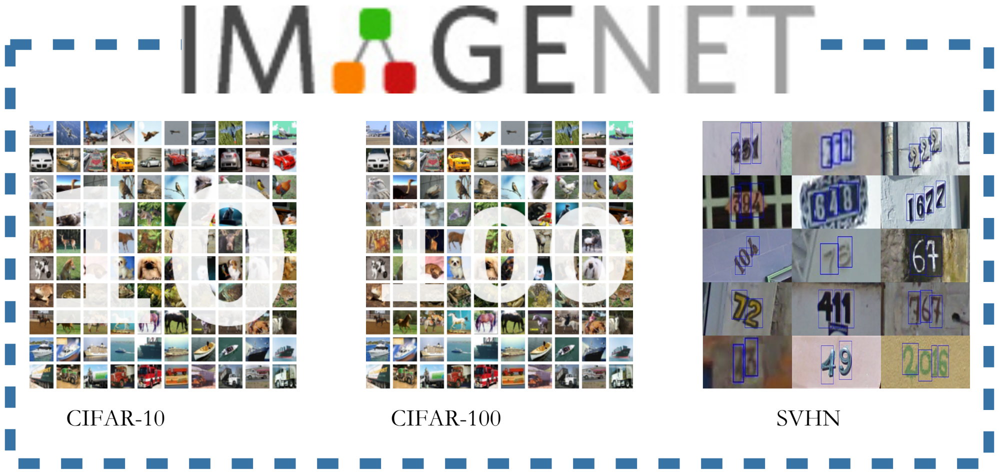


conceptually similar to Adversarial AutoAugment [ZWZZ, ICLR'20] but simpler

Experimental results

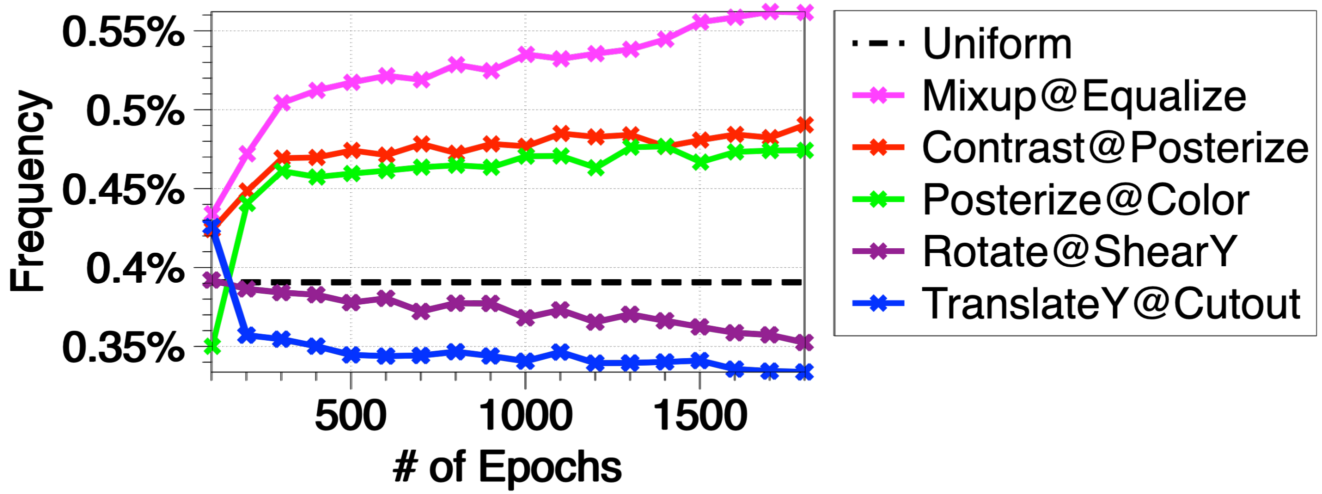
- Evaluation on multiple image classification datasets and models

- **Highlight 1:** 79% accuracy on ImageNet using ResNet-50, comparable to SoTA with less computation
- **Highlight 2:** By increasing # augmented samples, accuracy 85% on CIFAR-100 using WideResNet



Ablation studies

- ➢Why it works?
- ➢Our method learns and reduces the frequencies of the better performing transformations during training!



Model: PyramidNet + ShakeDrop
Dataset: CIFAR-10

Composition of transformations

Summary

- **Takeaway:** We provide a theoretical framework to understand data augmentation better, and a new augmentation sampling algorithm.

- **Theory & Intuition:** geometric intuition formalized via the span of training data.
- **Algorithm:** Uncertainty-based augmentation sampling by inspecting how large the losses of the transformed samples are!
- **Experiments:** SoTA quality on several image classification benchmarks.

Further reading

- Arxiv: 2005.00695
- Blog post: http://hazyresearch.stanford.edu/data-aug-part-3
- Code release: http://github.com/SenWu/dauphin

Data augmentation in semi-supervised learning

Combining unlabeled and labeled data

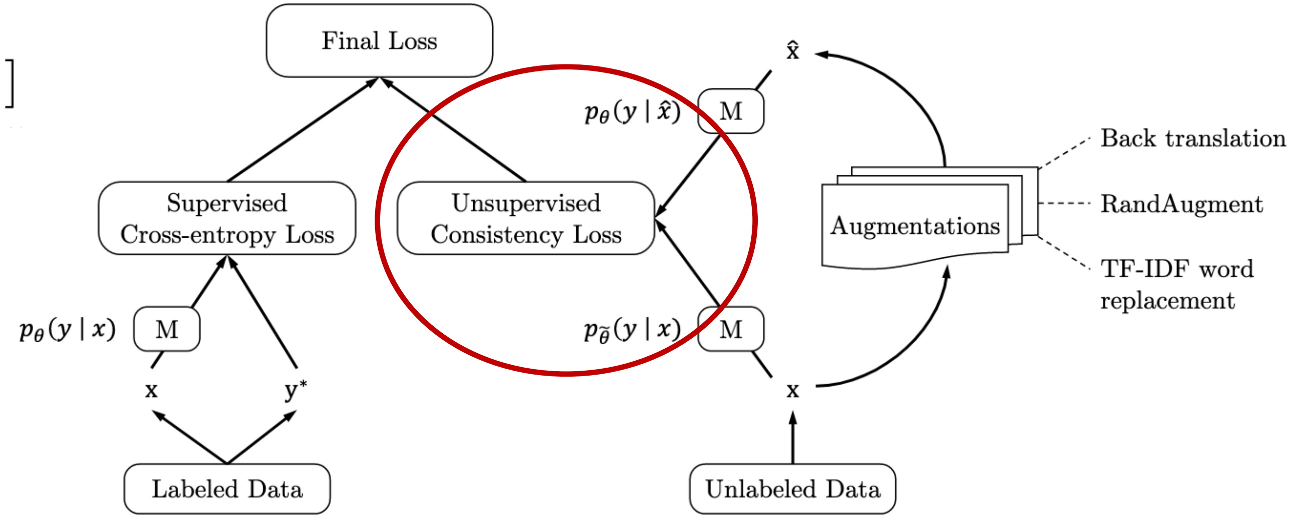
- Motivation: training neural networks requires lots of labeled data
- Semi-supervised learning: combine both labeled and unlabeled data together
- Examples: imagine having both labeled and unlabeled images

- Approaches:
  - **Label propagation:** assign labels to previously unlabeled data points
  - **Self-training:** first a supervised learning algorithm is trained based on the labeled data. This classifier is then applied to the unlabeled data to generate more *labels*
- **Intuition:** unlabeled data helps by estimating the features more accurately

Data augmentation

- Data augmentation is naturally suited for semi-supervised learning
- **Consistency regularization** is a method for using data augmentation in semi-supervised learning [Unsupervised data augmentation for consistency training, Xie et al'20]
- Encourages the labels of the original data x and the augmented data $\hat{x}$ to be similar:

$$\lambda \mathbb{E}_{x \sim p_U(x)} \mathbb{E}_{\hat{x} \sim q(\hat{x} \mid x)} \left[\mathrm{CE}\left(p_{\tilde{\theta}}(y \mid x) \| p_\theta(y \mid \hat{x}) \right) \right]$$



Data augmentation in text classification

Composition of fragments

- Good-enough compositional data augmentation, Andreas'20: provide a compositional bias in conditional and unconditional sequence models
- Motivation: we often want models to generalize beyond training dataset
- Examples:

(1) a. *The cat sang.*
b. *The wug sang.*
c. *The cat daxed.*

(a)

(2) a. *The wug daxed.*
b. * *The sang daxed.*

(c)

- Approach

She picks the wug up in Fresno. ≋ *Pat picks cats up.*

≀ ⇩

She puts the wug down in Tempe. → *Pat puts cats down.*

(b) (d)

Recap

- An overview of data augmentation
  - Motivation
  - How data augmentation works
  - Major challenges
  - Previous work
- A theoretical framework that precisely analyzes the generalization properties of data augmentation
  - Three categories of linear transformations in an over-parametrized setting
  - Uncertainty-based sampling
- Research trends
  - Semi-supervised learning: consistency regularization
  - Text classification: composition of sentence fragments