SAFLEX: Self-Adaptive Augmentation via Feature Label Extrapolation

Mucong Ding, Bang An, Yuancheng Xu,
Anirudh Satheesh, Furong Huang
University of Maryland, College Park

COMPUTER SCIENCE
UNIVERSITY OF MARYLAND

ICLR

Motivation & Problem

Limitations of Augmentation Methods

Data augmentation is pivotal in enhancing model generalization. However, its limitations, particularly the unintentional introduction of noise, can sometimes outweigh its benefits.

This inherent noise creates a trade-off: under-augmentation may yield insufficient challenging examples, whereas over-augmentation can flood the dataset with misleading samples.

Noises in Augmentations

Noises in augmentation primarily arises from two fundamental challenges:

(1) the deviation of augmented samples x^{aug} from the original data distribution;

(2) the potential mislabeling y^{aug} of augmented samples.

Question: Can we efficiently refine augmented samples to reduce noise and improve model generalization?

Yes, by learning sample weights and soft labels using a bilevel optimization approach called SAFLEX!

Contributions

(1) Novel parametrization and bilevel algorithm for learnable augmentation.

(2) Universally compatible, integrates with various learning processes and augmentation methods.

(3) Empirically validated on diverse datasets and tasks, boosting performance by 1.2% on average

Proposed Method: SAFLEX

We propose SAFLEX (Self-Adaptive Augmentation via Feature Label Extrapolation) which automatically learns the sample weights and soft labels of augmented samples provided by any upstream augmentation pipeline.

Limitations of Existing Learnable Augmentations

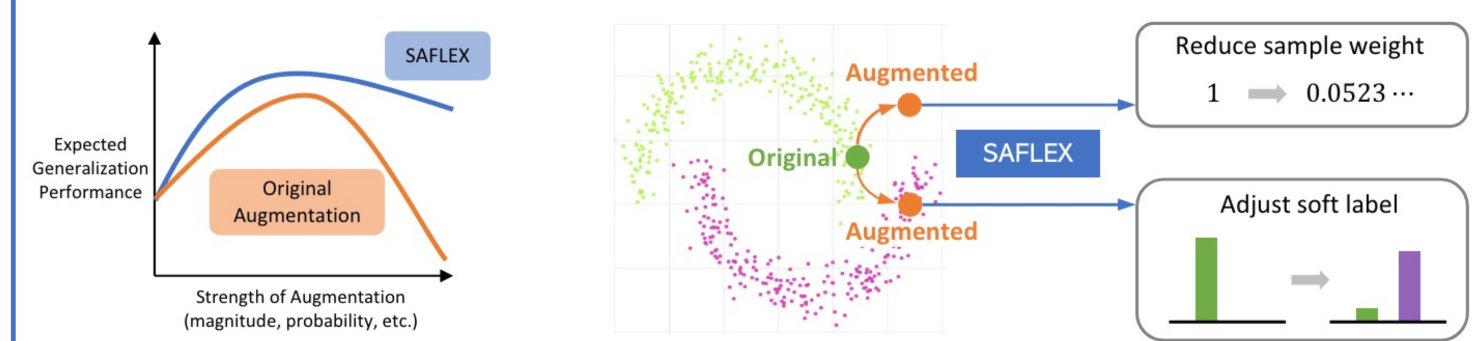
Existing learnable augmentation methods: directly learn in high-dimensional feature spaces

- Restricted augmentation scope due to differentiability needs
- Complicated training process
- Limited generalization across tasks

Feature and Label Extrapolation

In contrast, SAFLEX:

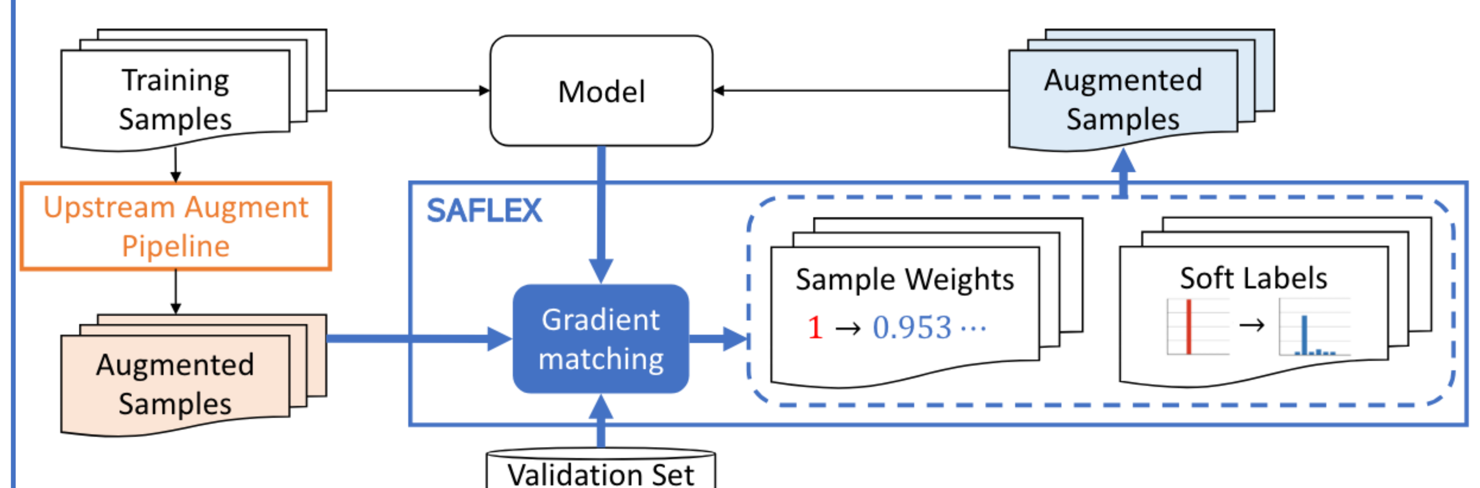
- Learns low-dimensional sample weights and soft labels
- Calibrates weights and labels post-augmentation
- Ensures diverse, distribution-consistent augmentation
- Avoids learning augmentations from scratch



(a) Mitigating trade-offs. (b) Addressing two types of errors.

SAFLEX is a Plug-In to Training Framework

$$(x, y) \xrightarrow{\text{Upstream Augment}} (x^{\text{aug}}, y^{\text{aug}}) \xrightarrow{\text{SAFLEX}} \Big(\underbrace{w^{\text{aug}}}_{\text{sample weight} \in [0,1]}, x^{\text{aug}}, \underbrace{\mathbf{y}^{\text{aug}}}_{\text{soft label} \in \Delta^K} \Big)$$



Bilevel Formulation of Learnable Augmentation

Transforms learnable augmentation into bilevel optimization

- Inner level: conventional model training on $\mathcal{D}_{\text{train}} \cup \mathcal{D}_{\text{aug}}$
- Outer level: optimizing augmented samples (sample weights and soft labels) to minimize validation loss $\mathcal{L}(\mathcal{D}_{\text{val}}, \theta)$

$$\min_{\mathcal{D}_{\text{aug}},\, \theta} \mathcal{L}(\mathcal{D}_{\text{val}}, \theta) \quad \text{s.t.} \quad \theta \in \arg\min_{\theta'} \mathcal{L}\big(\mathcal{D}_{\text{train}} \cup \mathcal{D}_{\text{aug}}, \theta'\big)$$

Efficient Bilevel Approach

Reformulate as greedy approach: optimize augmentation and minimize validation loss after each single update

$$\min_{\mathcal{D}^{\text{batch}}_{\text{aug}},\, \theta_t} \mathcal{L}(\mathcal{D}_{\text{val}}, \theta_t) \quad \text{s.t.} \quad \theta_t = \theta_{t-1} - \alpha \cdot \nabla_\theta \mathcal{L}(\mathcal{D}^{\text{batch}}_{\text{train}} \cup \mathcal{D}^{\text{batch}}_{\text{aug}}, \theta_{t-1})$$

- Exploit linearity of loss function and gradient computation w.r.t. sample weights and soft labels
- First-order approximation of validation loss (outer objective) leads to linear programming

Theorem 1 (Solution of Eq. (3)). *The approximated soft label solution is* $\mathbf{y} = \mathit{OneHot}(\arg\max_k [\mathbf{\Pi}]_k)$, *where* $\mathit{OneHot}(\cdot)$ *denotes one-hot encoding, and the sample weight solution is* $w = 1$ *if* $\sum_{k=1}^{K} [\mathbf{\Pi}]_k \geq 0$; *otherwise,* $w = 0$.

Pseudo-code of SAFLEX

Algorithm 1: SAFLEX (Cross-Entropy Loss, Single batch).

Input: Neural network $f(\cdot): \mathcal{X} \to \Delta^K$ (softmax applied on outputs) with parameters θ, upstream augmented batch $\{(x_1^{\text{aug}}, y_1^{\text{aug}}), \ldots, (x_B^{\text{aug}}, y_B^{\text{aug}})\}$, validation batch $\mathcal{D}^{\text{batch}}_{\text{val}} = \{(x_1^{\text{val}}, y_1^{\text{val}}), \ldots, (x_{B'}^{\text{val}}, y_{B'}^{\text{val}})\}$, penalty coefficient β, temperature τ.

Compute the gradient vector for the validation batch $\nabla_\theta \mathcal{L}(\mathcal{D}^{\text{batch}}_{\text{val}}, \theta)$.

for $i = 1, \ldots, B$ **do** // The actual implementation is vectorized.

- Determine the gradient inner product $\mathbf{\Pi}_i = \nabla_\theta f(x_i^{\text{aug}}) \nabla_\theta \mathcal{L}(\mathcal{D}^{\text{batch}}_{\text{val}}, \theta)$ via Jacobian-vector product.
- Apply Gumbel-SoftMax to get $\mathbf{y}_i = \text{softmax}\big((\mathbf{\Pi}_i + \beta \mathbf{e}_{y_i^{\text{aug}}} + \mathbf{g})/\tau\big)$, where $e_{y_i^{\text{aug}}} \in \mathbb{R}^K$ is one-hot at y_i^{aug}, and $\mathbf{g}$ consists of i.i.d. random variables taken from Gumbel$(0,1)$.
- Set $w_i = 1$ if $\mathbf{\Pi}_i \cdot \mathbf{y}_i \geq 0$, otherwise set $w_i = 0$.

Renormalize the sample weights $w_1, \ldots, w_B$ to sum to 1.

return *Sample weights* $w_1^{\text{aug}}, \ldots, w_B^{\text{aug}}$, *and soft labels* $\mathbf{y}_1^{\text{aug}}, \ldots, \mathbf{y}_B^{\text{aug}}$.

Experiments

Adapting Augmentations to Medical Images

SAFLEX improves the performance of popular augmentation techniques like RandAugment and Mixup on eight medical image datasets.

Method	Path	Derma	Tissue	Blood	OCT	OrganA	OrganC	OrganS
No Aug	94.34 ± 0.18	76.14 ± 0.09	68.28 ± 0.17	96.81 ± 0.19	78.67 ± 0.26	94.21 ± 0.09	91.81 ± 0.12	81.57 ± 0.07
RandAug	93.52 ± 0.09	73.71 ± 0.33	62.03 ± 0.14	95.00 ± 0.21	76.00 ± 0.24	94.18 ± 0.20	91.38 ± 0.14	80.52 ± 0.32
SAFLEX (w/ RandAug)	**95.11 ± 0.14**	76.69 ± 0.33	64.32 ± 0.18	96.91 ± 0.15	**79.63 ± 0.28**	**95.32 ± 0.29**	92.10 ± 0.21	**82.85 ± 0.42**
Mixup	92.98 ± 0.19	75.22 ± 0.45	66.62 ± 0.31	96.28 ± 0.23	77.93 ± 0.41	94.12 ± 0.35	90.76 ± 0.28	80.99 ± 0.21
SAFLEX (w/ Mixup)	93.71 ± 0.37	**76.94 ± 0.51**	**68.31 ± 0.43**	**97.21 ± 0.35**	79.54 ± 0.44	95.06 ± 0.31	**92.73 ± 0.53**	82.14 ± 0.27

Refining Augmentations for Tabular Data

SAFLEX improves the performance of CutMix across diverse tabular datasets.

Method	Model	Appetency	Arrhythmia	Click	Credit	QASR	Shrutime	Volkert
No Aug	MLP	49.03 ± 0.01	81.53 ± 0.03	52.54 ± 0.04	66.91 ± 0.03	91.84 ± 0.02	86.27 ± 0.04	61.14 ± 0.05
CutMix	MLP	48.98 ± 0.03	81.57 ± 0.05	52.59 ± 0.09	73.68 ± 0.08	91.87 ± 0.02	86.39 ± 0.05	61.20 ± 0.02
SAFLEX (w/ CutMix)	MLP	**51.04 ± 0.09**	**83.02 ± 0.06**	**52.81 ± 0.06**	**74.61 ± 0.15**	**92.69 ± 0.13**	**86.90 ± 0.10**	**61.51 ± 0.05**
No Aug	SAINT	78.90 ± 0.03	83.90 ± 0.01	65.72 ± 0.06	79.49 ± 0.05	98.18 ± 0.04	87.53 ± 0.04	66.82 ± 0.05
CutMix	SAINT	81.05 ± 0.07	**85.32 ± 0.09**	65.77 ± 0.04	79.71 ± 0.08	**98.61 ± 0.06**	87.61 ± 0.07	68.23 ± 0.10
SAFLEX (w/ CutMix)	SAINT	**81.33 ± 0.14**	85.27 ± 0.14	**66.12 ± 0.09**	**79.93 ± 0.17**	98.59 ± 0.21	**87.93 ± 0.13**	**68.91 ± 0.17**

Purifying Diffusion-Model-Generated Augments

SAFLEX consistently improves the performance of all three diffusion-model-generated augmentation techniques, across both fine-grained classification and out-of-distribution generalization tasks.

Task	No Aug	RandAug	Text2Img	InstructPix2Pix		Img2Img (w/o finetune)		Img2Img (w/ finetune)	
	—	—	—	w/o SAFLEX	w/ SAFLEX	w/o SAFLEX	w/ SAFLEX	w/o SAFLEX	w/ SAFLEX
Fine-Grained Classification	68.60 ± 0.16	71.26 ± 0.52	69.68 ± 0.97	71.38 ± 0.91	**72.34 ± 0.59**	71.25 ± 0.86	**73.22 ± 0.63**	72.01 ± 1.24	**73.61 ± 0.78**
OOD Generalization	57.19 ± 1.13	61.34 ± 2.72	64.53 ± 3.01	67.29 ± 1.96	**69.92 ± 0.88**	70.65 ± 1.50	**72.61 ± 1.44**	70.49 ± 1.21	**72.83 ± 0.92**

Applying to Contrastive Fine-Tuning

SAFLEX enhances the performance of standard image augmentations like RandAugment when applied to the contrastive fine-tuning.

Task		Zero-Shot	LP-FT	FLYP	FLYP+RandAug	FLYP+SAFLEX (w/ RandAug)
ID	w/o Ensembling	8.7 ± 0.0	49.7 ± 0.5	52.2 ± 0.6	52.4 ± 0.8	**52.7 ± 0.7**
OOD	w/o Ensembling	11.0 ± 0.0	34.7 ± 0.4	35.6 ± 1.2	36.3 ± 1.4	**36.9 ± 1.5**
ID	w/ Ensembling	8.7 ± 0.0	50.2 ± 0.5	52.5 ± 0.6	52.6 ± 1.0	**53.0 ± 0.7**
OOD	w/ Ensembling	11.0 ± 0.0	35.7 ± 0.4	37.1 ± 1.2	37.6 ± 0.9	**37.8 ± 1.1**