

Lossy compression for lossless prediction

EECS Seminar: Advanced Topics in Machine Learning

Romain Graux

March 17, 2022

Motivation

~ 50 trillion GB data collected per year

Motivation

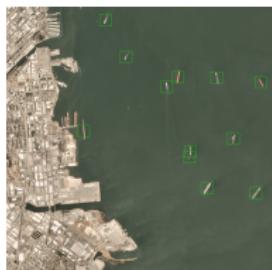
~ 50 trillion GB data collected per year

⇒ But most data is processed by algorithms performing **downstream tasks**.

Motivation

~ 50 trillion GB data collected per year

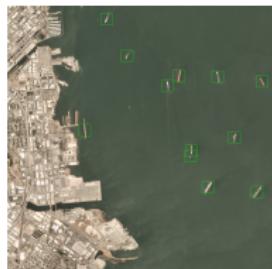
⇒ But most data is processed by algorithms performing **downstream tasks**.



Motivation

~ 50 trillion GB data collected per year

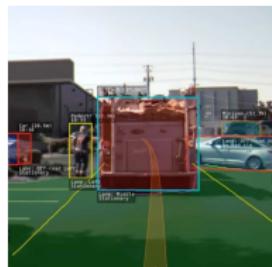
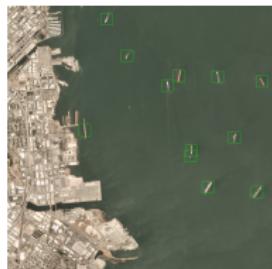
⇒ But most data is processed by algorithms performing **downstream tasks**.



Motivation

~ 50 trillion GB data collected per year

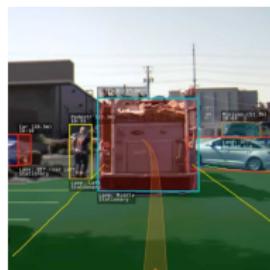
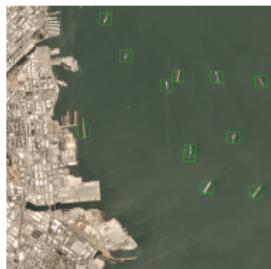
⇒ But most data is processed by algorithms performing **downstream tasks**.



Motivation

~ 50 trillion GB data collected per year

⇒ But most data is processed by algorithms performing **downstream tasks**.



Motivation

~ 50 trillion GB data collected per year

⇒ But most data is processed by algorithms performing **downstream tasks**.

Yet current compressors optimize high **perceptual** fidelity

Motivation

~ 50 trillion GB data collected per year

⇒ But most data is processed by algorithms performing **downstream tasks**.

Yet current compressors optimize high **perceptual** fidelity

- Stores too much not needed information
- Does not ensure good task performance

Motivation

~ 50 trillion GB data collected per year

⇒ But most data is processed by algorithms performing **downstream tasks**.

Yet current compressors optimize high **perceptual** fidelity

- Stores too much not needed information
- Does not ensure good task performance



Source

Motivation

~ 50 trillion GB data collected per year

⇒ But most data is processed by algorithms performing **downstream tasks**.

Yet current compressors optimize high **perceptual** fidelity

- Stores too much not needed information
- Does not ensure good task performance



Source



High bitrate

Motivation

~ 50 trillion GB data collected per year

⇒ But most data is processed by algorithms performing **downstream tasks**.

Yet current compressors optimize high **perceptual** fidelity

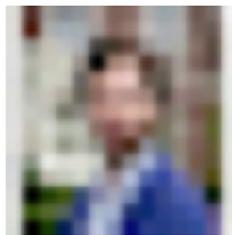
- Stores too much not needed information
- Does not ensure good task performance



Source



High bitrate



Low bitrate

Motivation

~ 50 trillion GB data collected per year

⇒ But most data is processed by algorithms performing **downstream tasks**.

Yet current compressors optimize high **perceptual** fidelity

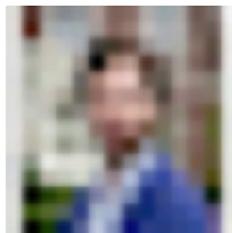
- Stores too much not needed information
- Does not ensure good task performance



Source



High bitrate



Low bitrate



Desired

What they designed

They designed a **task-centric** distortion that ensures good downstream performance

What they designed

They designed a **task-centric** distortion that ensures good downstream performance

- Characterize minimum bit-rate to ensure high performance on desired tasks;

What they designed

They designed a **task-centric** distortion that ensures good downstream performance

- Characterize minimum bit-rate to ensure high performance on desired tasks;
- Derive unsupervised objectives for training **task-centric** compressors;

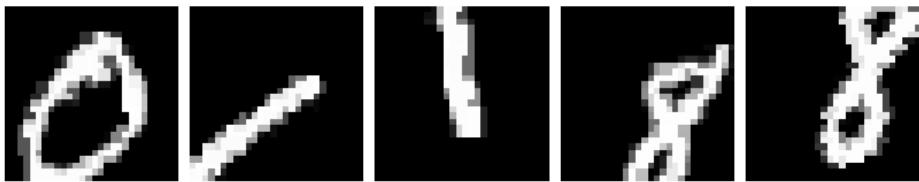
What they designed

They designed a **task-centric** distortion that ensures good downstream performance

- Characterize minimum bit-rate to ensure high performance on desired tasks;
- Derive unsupervised objectives for training **task-centric** compressors;
- > 1000x compression gains on Imagenet compared to JPEG (see Slide ??).

Intuition

Intuition: Augmented MNIST

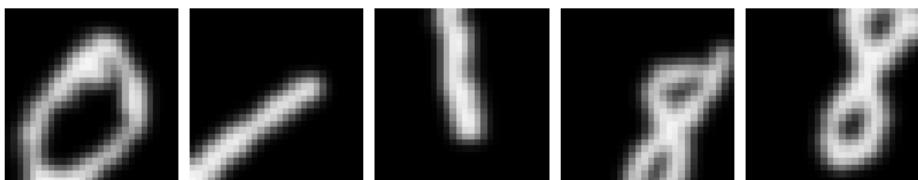


Source: Augmented MNIST

Intuition: Augmented MNIST

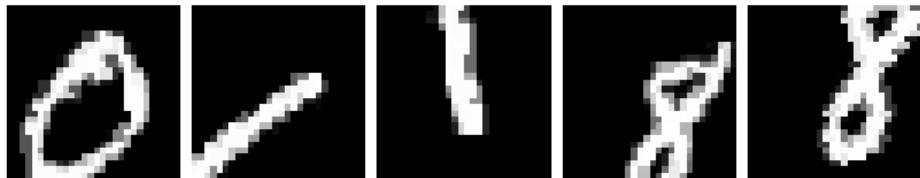


Source: Augmented MNIST

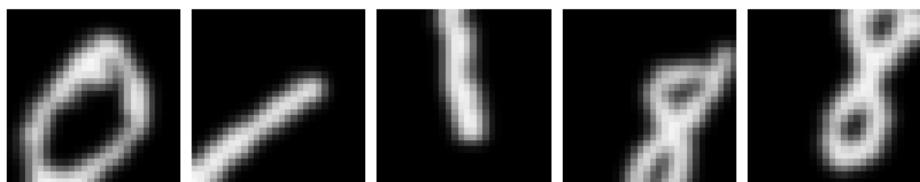


Standard neural compressor: 130 bit-rate

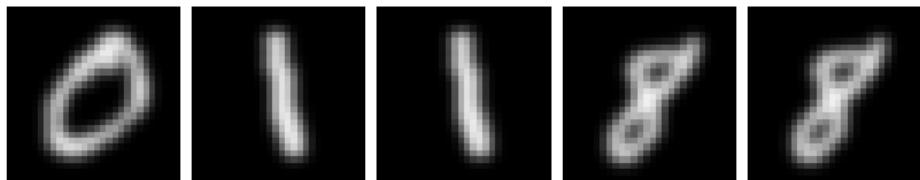
Intuition: Augmented MNIST



Source: Augmented MNIST



Standard neural compressor: 130 bit-rate



Their neural compressor: 48 bit-rate

Intuition: Augmented MNIST



Prototypical digit ensures

- high downstream performance
- good compression rate

Intuition: Augmented MNIST



Prototypical digit ensures → high downstream performance
→ good compression rate

Why not sending directly the labels?

Intuition: Augmented MNIST



Prototypical digit ensures → high downstream performance
→ good compression rate

Why not sending directly the labels?

- Might be interested in multiple downstream tasks;

Intuition: Augmented MNIST



Prototypical digit ensures → high downstream performance
→ good compression rate

Why not sending directly the labels?

- Might be interested in multiple downstream tasks;
- Would require knowing tasks of interest at compression time.

Intuition: Augmented MNIST



Prototypical digit ensures → high downstream performance
→ good compression rate

Why not sending directly the labels?

- Might be interested in multiple downstream tasks;
 - Would require knowing tasks of interest at compression time.
- ⇒ The objective is **unsupervised**

Formalism

Problem setup

Goal:

Problem setup

Goal:

- Minimum achievable bit-rate to store X

Problem setup

Goal:

- Minimum achievable bit-rate to store X
 - e.g. X : all satellite images

Problem setup

Goal:

- Minimum achievable bit-rate to store X
 - e.g. X : all satellite images
 - e.g. X : all images of machine learning researchers

Problem setup

Goal:

- Minimum achievable bit-rate to store X
- While ensuring high performance on any task Y of interest
 $\mathcal{T} = \{Y_1, Y_2, \dots\}$

Problem setup

Goal:

- Minimum achievable bit-rate to store X
- While ensuring high performance on any task Y of interest
 $\mathcal{T} = \{Y_1, Y_2, \dots\}$
 - e.g. Y_1 : how old is the person?

Problem setup

Goal:

- Minimum achievable bit-rate to store X
- While ensuring high performance on any task Y of interest
 $\mathcal{T} = \{Y_1, Y_2, \dots\}$
 - e.g. Y_1 : how old is the person?
 - e.g. Y_2 : does the person wear glasses?

Problem setup

Goal:

- Minimum achievable bit-rate to store X
- While ensuring high performance on any task Y of interest
 $\mathcal{T} = \{Y_1, Y_2, \dots\}$

⇒ looking for a representation Z s.t. predictions are approx. as good as using X

Problem setup

Goal:

- Minimum achievable bit-rate to store X
- While ensuring high performance on any task Y of interest
 $\mathcal{T} = \{Y_1, Y_2, \dots\}$

⇒ looking for a representation Z s.t. predictions are approx. as good as using X

$$\sup_{Y \in \mathcal{T}} \underbrace{R[Y|Z] - R[Y|X]}_{\text{excess Bayes risk}} \leq \delta$$

Problem setup

Goal:

- Minimum achievable bit-rate to store X
- While ensuring high performance on any task Y of interest
 $\mathcal{T} = \{Y_1, Y_2, \dots\}$

⇒ looking for a representation Z s.t. predictions are approx. as good as using X

$$\sup_{Y \in \mathcal{T}} R[Y|Z] - R[Y|X] \underbrace{\leq \delta}_{\text{small}}$$

Problem setup

Goal:

- Minimum achievable bit-rate to store X
- While ensuring high performance on any task Y of interest
 $\mathcal{T} = \{Y_1, Y_2, \dots\}$

⇒ looking for a representation Z s.t. predictions are approx. as good as using X

$$\sup_{\substack{Y \in \mathcal{T} \\ \text{all tasks}}} R[Y|Z] - R[Y|X] \leq \delta$$

Problem setup

Goal:

- Minimum achievable bit-rate to store X
- While ensuring high performance on any task Y of interest
 $\mathcal{T} = \{Y_1, Y_2, \dots\}$

⇒ looking for a representation Z s.t. predictions are approx. as good as using X

$$\sup_{Y \in \mathcal{T}} R[Y|Z] - R[Y|X] \leq \delta$$

$\delta = 0$: lossless prediction regime

Problem setup

Goal:

- Minimum achievable bit-rate to store X
- While ensuring high performance on any task Y of interest
 $\mathcal{T} = \{Y_1, Y_2, \dots\}$

⇒ looking for a representation Z s.t. predictions are approx. as good as using X

$$\sup_{Y \in \mathcal{T}} R[Y|Z] - R[Y|X] \leq \delta$$

$\delta = 0$: lossless prediction regime

Problem: Would assume access \mathcal{T}

Key assumption: Invariance structure

Tasks of interest to humans share structure.

Key assumption: Invariance structure

Tasks of interest to humans share structure.

Assumption: tasks of interest are invariant to some transformation

Key assumption: Invariance structure

Tasks of interest to humans share structure.

Assumption: tasks of interest are invariant to some transformation

→ ubiquitous in ML: data augmentation

Key assumption: Invariance structure

Tasks of interest to humans share structure.

Assumption: tasks of interest are invariant to some transformation

→ ubiquitous in ML: data augmentation

→ formalized using invariance to an equivalence relationship \sim

Key assumption: Invariance structure

Tasks of interest to humans share structure.

Assumption: tasks of interest are invariant to some transformation

→ ubiquitous in ML: data augmentation

→ formalized using invariance to an equivalence relationship \sim

Proposition

Exists a "worst task" $M(X)$ that has all information to predict any invariant task $Y \in \mathcal{T}$

$$R[M(X)|Z] = \sup_{Y \in \mathcal{T}} R[Y|Z] - R[Y|X] \leq \delta$$

What we want:

$$x \sim x^+ \iff M(x) = M(x^+) \text{ for any } x, x^+ \in X$$



Key assumption: Invariance structure

Tasks of interest to humans share structure.

Assumption: tasks of interest are invariant to some transformation

→ ubiquitous in ML: data augmentation

→ formalized using invariance to an equivalence relationship \sim

Proposition

Exists a "worst task" $M(X)$ that has all information to predict any invariant task $Y \in \mathcal{T}$

$$R[M(X)|Z] = \sup_{Y \in \mathcal{T}} R[Y|Z] - R[Y|X] \leq \delta$$

What we want:

$$x \sim x^+ \iff M(x) = M(x^+) \text{ for any } x, x^+ \in X$$



Key assumption: Invariance structure

Tasks of interest to humans share structure.

Assumption: tasks of interest are invariant to some transformation

→ ubiquitous in ML: data augmentation

→ formalized using invariance to an equivalence relationship \sim

Proposition

Exists a "worst task" $M(X)$ that has all information to predict any invariant task $Y \in \mathcal{T}$

$$R[M(X)|Z] = \sup_{Y \in \mathcal{T}} R[Y|Z] - R[Y|X] \leq \delta$$

What we want:

$$x \sim x^+ \iff M(x) = M(x^+) \text{ for any } x, x^+ \in X$$



Worst task: data augmentation example



Lenna: source image

Worst task: data augmentation example



x_0 : gray scale

Worst task: data augmentation example



x_0 : gray scale

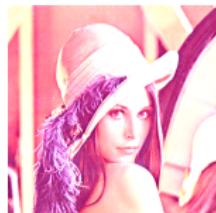


x_1 : brightness

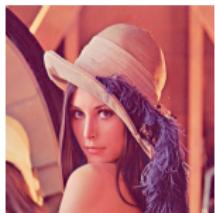
Worst task: data augmentation example



x_0 : gray scale



x_1 : brightness

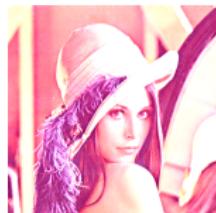


x_2 : horizontal flip

Worst task: data augmentation example



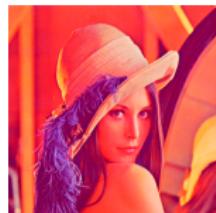
x_0 : gray scale



x_1 : brightness



x_2 : horizontal flip

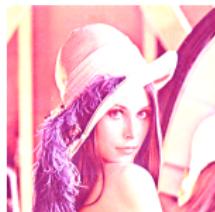


x_3 : saturation

Worst task: data augmentation example



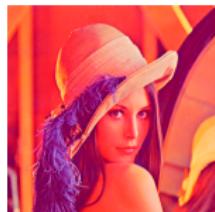
x_0 : gray scale



x_1 : brightness



x_2 : horizontal flip



x_3 : saturation



$M(x)$: unaugmented

Theorem: Rate-Invariance

Using the rate-distortion theorem [Shannon, 1959] with $R[M(X)|Z]$ as distortion:

Theorem (Rate-Invariance)

The minimum achievable bit-rate for transmitting Z s.t. for any invariant $Y \in \mathcal{T}$ we have an excess Bayes risk $R[Y|Z] - R[Y|X]$ upper bounded by δ is

$$\text{Rate}(\delta) = \underbrace{H[M(x)]}_{\text{Minimum bit-rate}} + \underbrace{-\delta}_{\text{Predictive loss}}$$

Theorem: Rate-Invariance

Using the rate-distortion theorem [Shannon, 1959] with $R[M(X)|Z]$ as distortion:

Theorem (Rate-Invariance)

The minimum achievable bit-rate for transmitting Z s.t. for any invariant $Y \in \mathcal{T}$ we have an excess Bayes risk $R[Y|Z] - R[Y|X]$ upper bounded by δ is

$$\text{Rate}(\delta) = \underbrace{H[M(x)]}_{\text{Minimum bit-rate}} + \underbrace{-\delta}_{\text{Predictive loss}}$$

Theorem: Rate-Invariance

Using the rate-distortion theorem [Shannon, 1959] with $R[M(X)|Z]$ as distortion:

Theorem (Rate-Invariance)

The minimum achievable bit-rate for transmitting Z s.t. for any invariant $Y \in \mathcal{T}$ we have an excess Bayes risk $R[Y|Z] - R[Y|X]$ upper bounded by δ is

$$\text{Rate}(\delta) = \underbrace{H[M(x)]}_{\text{Minimum bit-rate}} + \underbrace{-\delta}_{\text{Predictive loss}}$$

Theorem: Rate-Invariance

Using the rate-distortion theorem [Shannon, 1959] with $R[M(X)|Z]$ as distortion:

Theorem (Rate-Invariance)

The minimum achievable bit-rate for transmitting Z s.t. for any invariant $Y \in \mathcal{T}$ we have an excess Bayes risk $R[Y|Z] - R[Y|X]$ upper bounded by δ is

$$\text{Rate}(\delta) = \underbrace{H[x]}_{\text{Standard compression}} + \underbrace{-H[x|M(x)]}_{\text{Gains due to Invariance}} + \underbrace{-\delta}_{\text{Predictive loss}}$$

Theorem: Rate-Invariance

Using the rate-distortion theorem [Shannon, 1959] with $R[M(X)|Z]$ as distortion:

Theorem (Rate-Invariance)

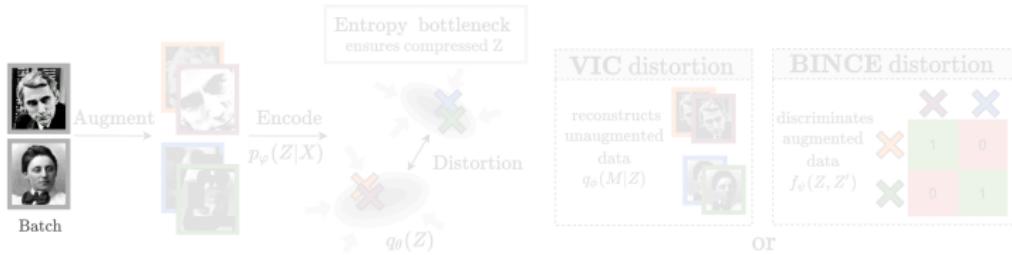
The minimum achievable bit-rate for transmitting Z s.t. for any invariant $Y \in \mathcal{T}$ we have an excess Bayes risk $R[Y|Z] - R[Y|X]$ upper bounded by δ is

$$\text{Rate}(\delta) = \underbrace{H[x]}_{\text{Standard compression}} + \underbrace{-H[x|M(x)]}_{\text{Gains due to Invariance}} + \underbrace{-\delta}_{\text{Predictive loss}}$$

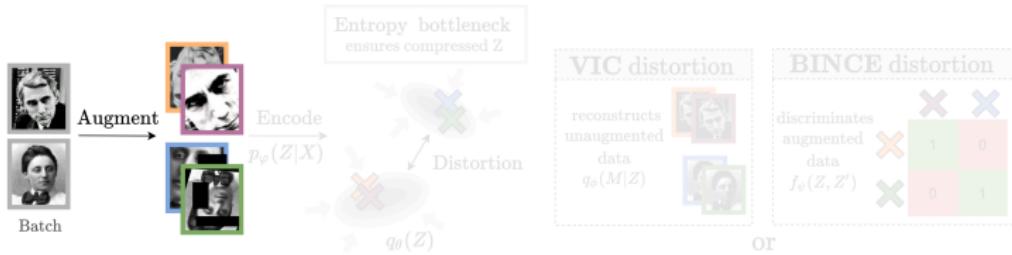
\Rightarrow a δ decrease in *log-loss* save exactly δ bits

In practice

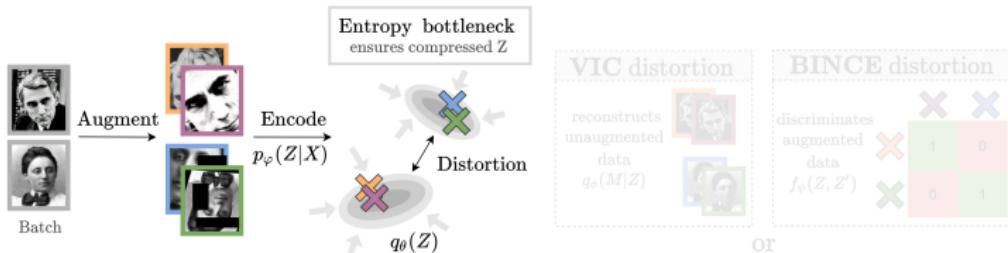
Training architecture



Training architecture

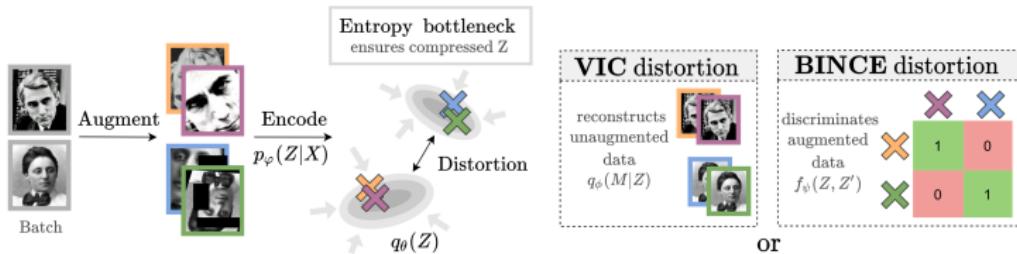


Training architecture



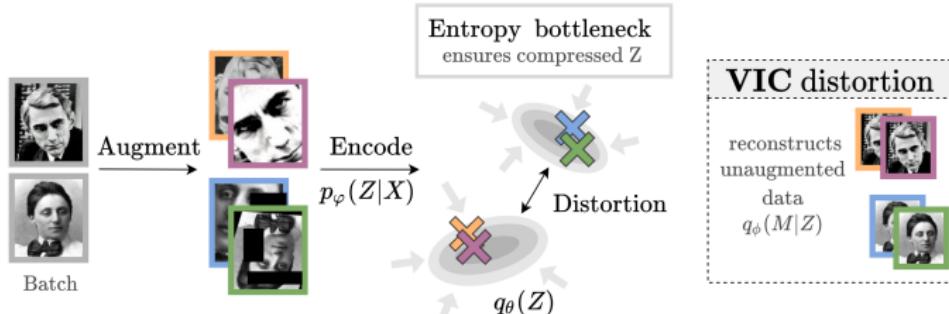
Entropy bottleneck: compressed Z

Training architecture

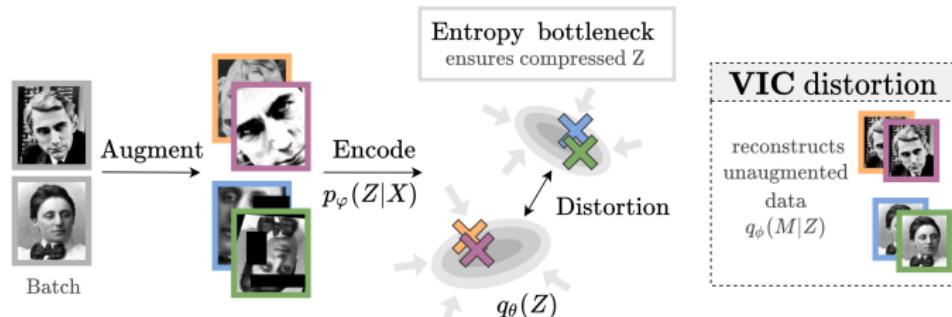


Entropy bottleneck: compressed Z

Variational Invariant Compressor

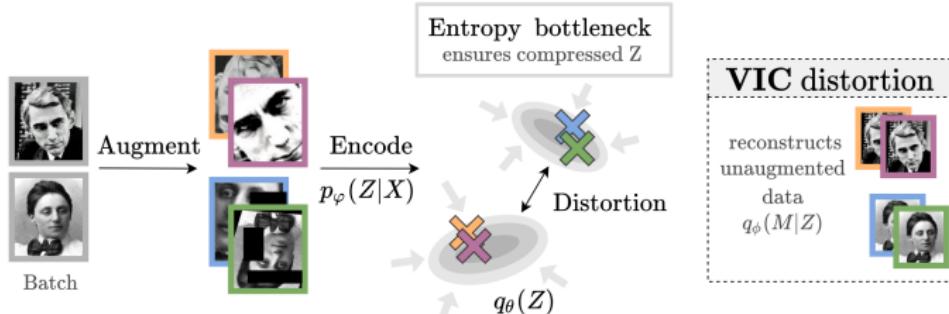


Variational Invariant Compressor



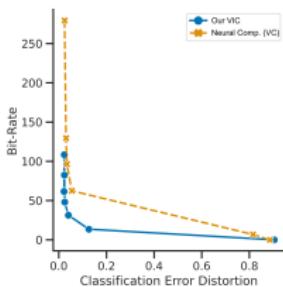
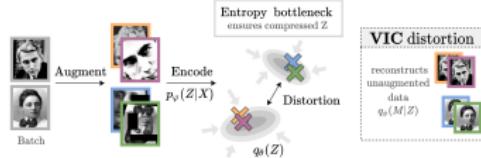
- Reconstruct the prototypicals

Variational Invariant Compressor



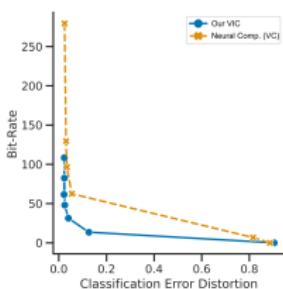
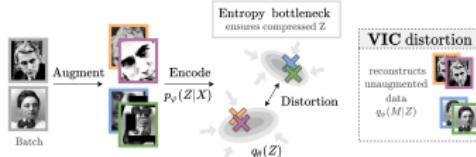
- Reconstruct the prototypical
- Unaugmented images

Variational Invariant Compressor

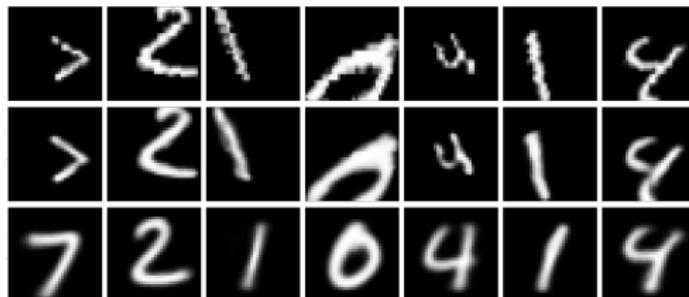


Rate-Error curve

Variational Invariant Compressor

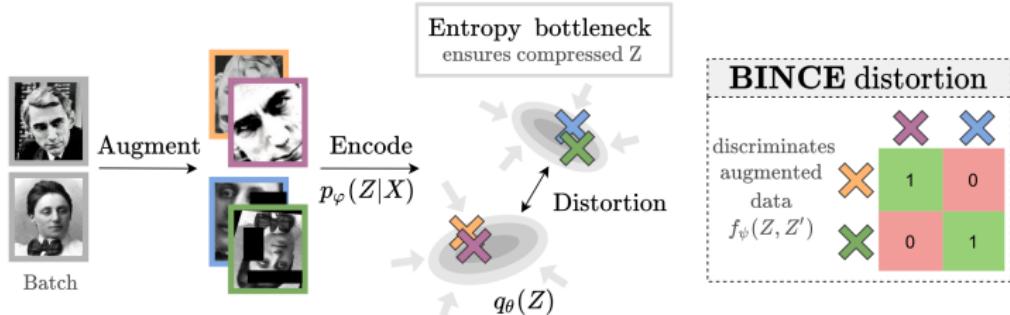


Rate-Error curve

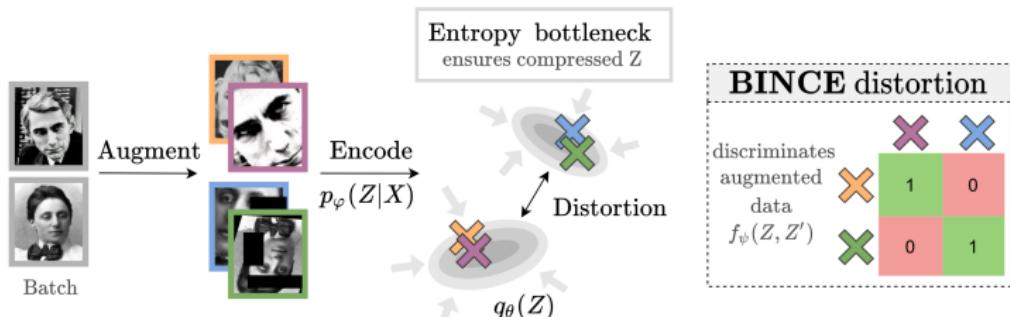


Reconstructions that allow 99% downstream accuracy
(1) Source (2) Standard compression [130 bits] (3) Invariant compression [48 bits]

Bottleneck InfoNCE: Intuition



Bottleneck InfoNCE: Intuition

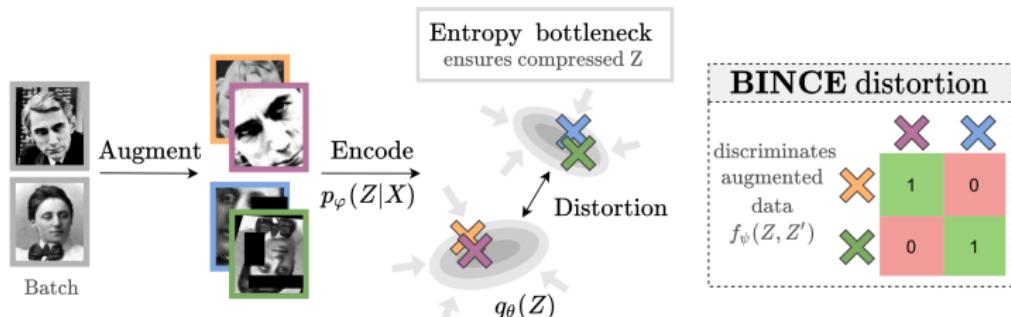


Contrastive learning



Latent representation

Bottleneck InfoNCE: Intuition



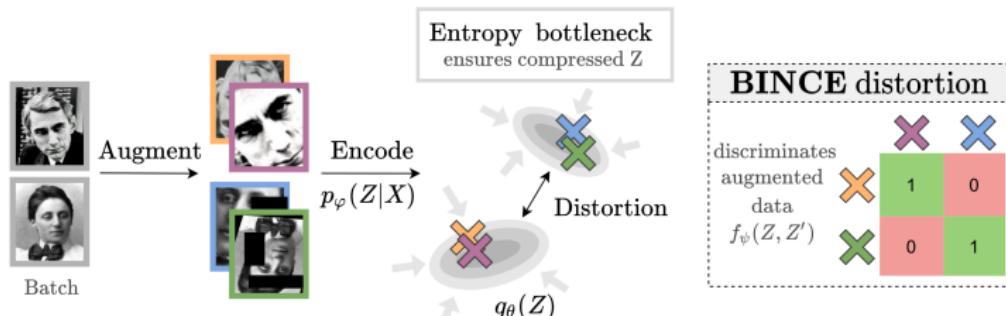
Contrastive learning



Latent representation

Bottleneck

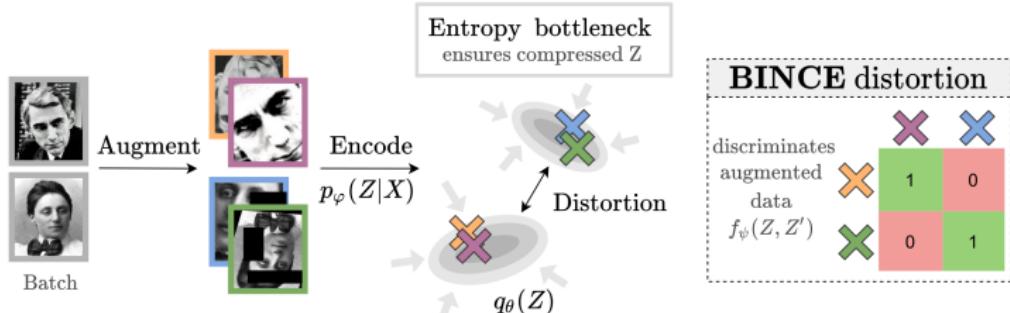
Bottleneck InfoNCE: Intuition



Contrastive learning

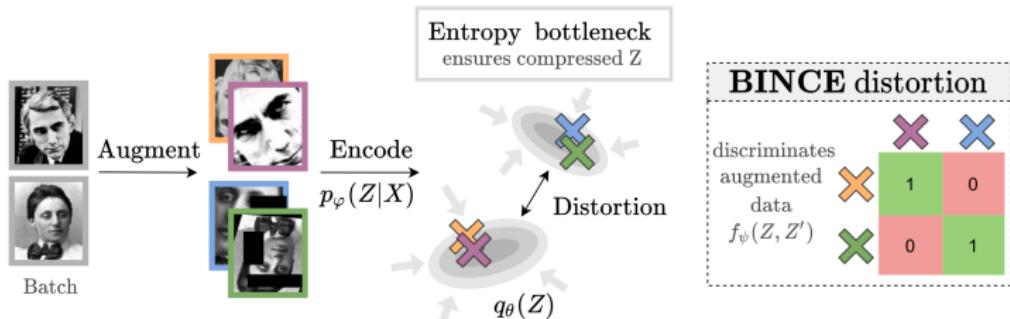


Bottleneck InfoNCE: Intuition



Pros:

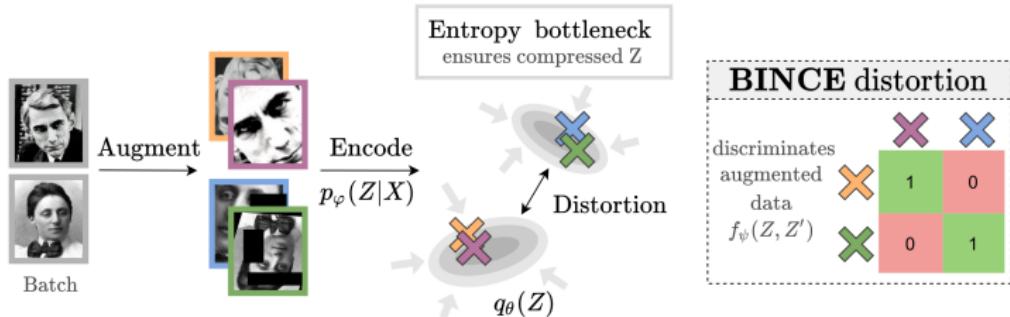
Bottleneck InfoNCE: Intuition



Pros:

- does not have to reconstruct high dimensional data

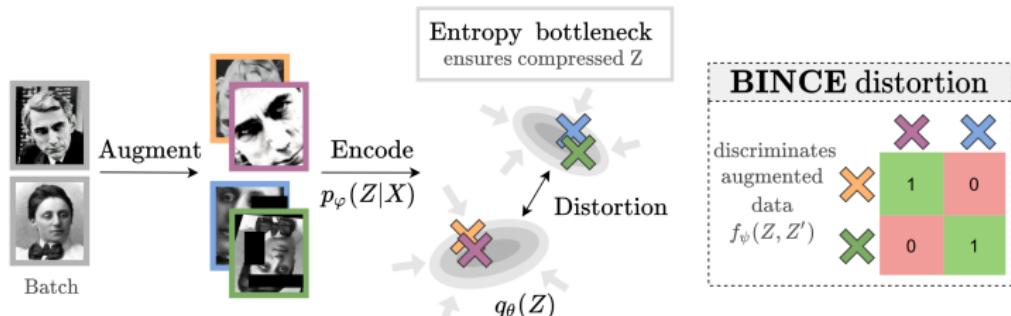
Bottleneck InfoNCE: Intuition



Pros:

- does not have to reconstruct high dimensional data
- gives representations that are approx. linearly separable

Bottleneck InfoNCE: Intuition

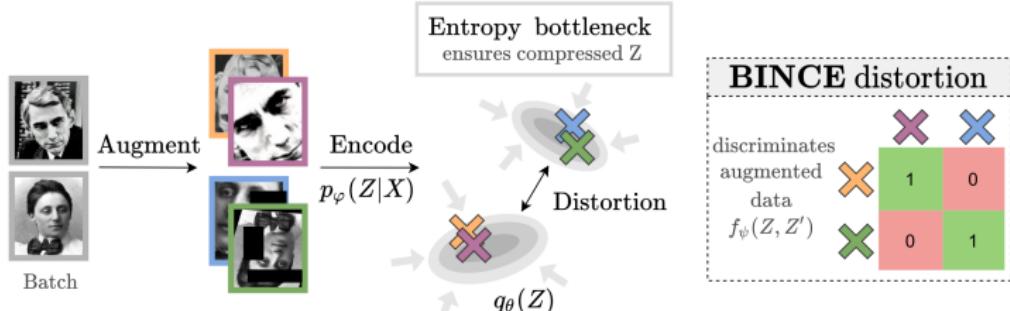


Pros:

- does not have to reconstruct high dimensional data
- gives representations that are approx. linearly separable

Cons:

Bottleneck InfoNCE: Intuition



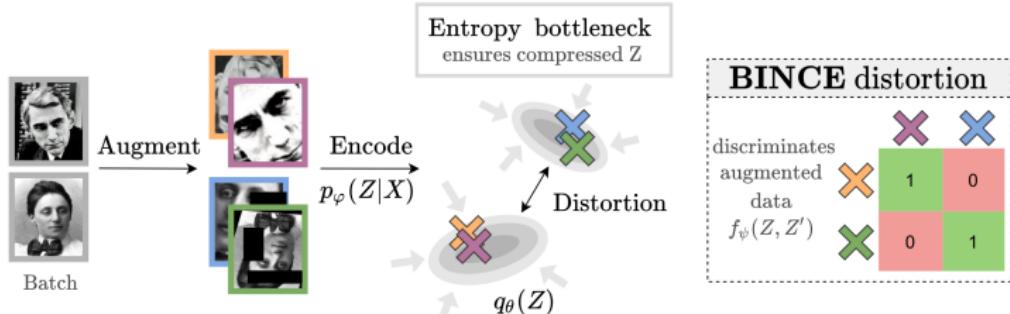
Pros:

- does not have to reconstruct high dimensional data
- gives representations that are approx. linearly separable

Cons:

- diminishes interpretability

Bottleneck InfoNCE: Intuition



Pros:

- does not have to reconstruct high dimensional data
- gives representations that are approx. linearly separable

Cons:

- diminishes interpretability
- has a high bias

Trained encoder : CLIP

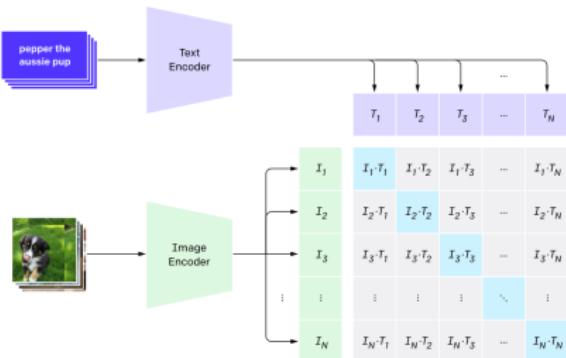
Idea: Can simply add an entropy bottleneck to a SOTA SSL

Trained encoder : CLIP

Idea: Can simply add an entropy bottleneck to a SOTA SSL

CLIP: trained on +400M of image/text pairs

1. Contrastive pre-training

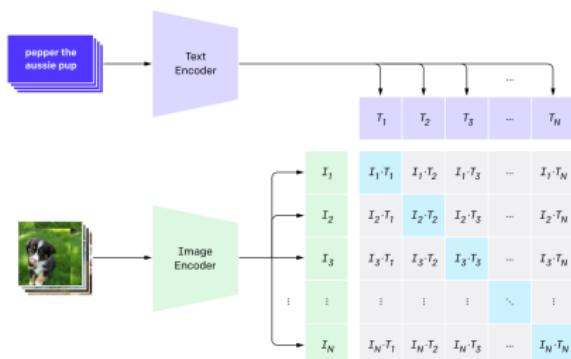


Trained encoder : CLIP

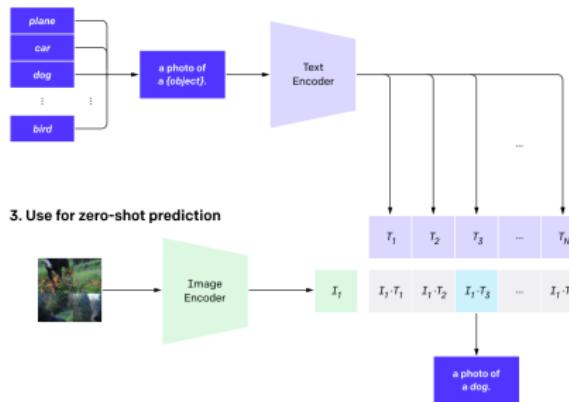
Idea: Can simply add an entropy bottleneck to a SOTA SSL

CLIP: trained on +400M of image/text pairs

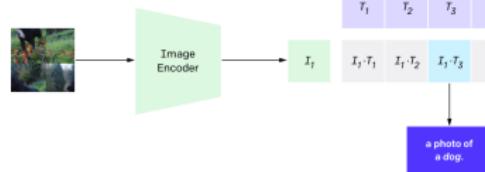
1. Contrastive pre-training



2. Create dataset classifier from label text



3. Use for zero-shot prediction

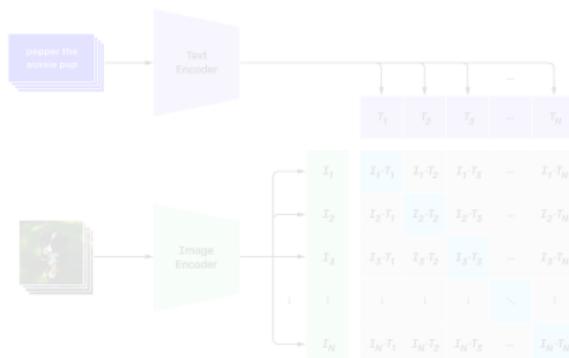


Trained encoder : CLIP

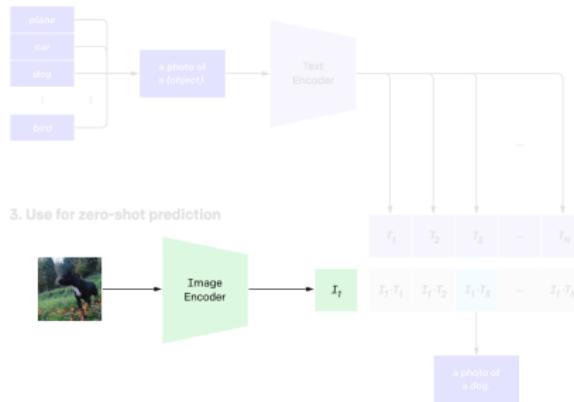
Idea: Can simply add an entropy bottleneck to a SOTA SSL

CLIP: trained on +400M of image/text pairs

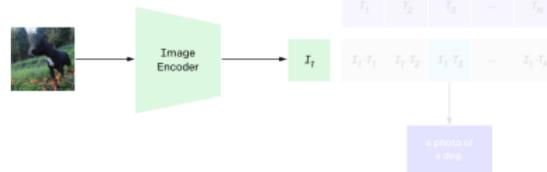
1. Contrastive pre-training



2. Create dataset classifier from label text



3. Use for zero-shot prediction



Bottleneck InfoNCE: Experiment

1. Download SOTA SLL (CLIP) and freeze it

Bottleneck InfoNCE: Experiment

1. Download SOTA SLL (CLIP) and freeze it
2. Train an entropy bottleneck on some dataset (MSCOCO) < 1 hour

Bottleneck InfoNCE: Experiment

1. Download SOTA SLL (CLIP) and freeze it
2. Train an entropy bottleneck on some dataset (MSCOCO) < 1 hour
3. Evaluate on very different tasks/datasets **never seen before**

Bottleneck InfoNCE: Experiment

1. Download SOTA SLL (CLIP) and freeze it
2. Train an entropy bottleneck on some dataset (MSCOCO) < 1 hour
3. Evaluate on very different tasks/datasets **never seen before**

	ImageNet	STL	PCam	Cars	CIFAR10	Food	Pets	Caltech
Rate gains vs JPEG	1104×	35×	64×	131×	7×	109×	150×	126×
Our Acc. [%]	76.3	98.7	80.9	79.6	95.2	88.3	89.5	93.4
Supervised Acc. [%]	76.1	99.0	82.6	49.1	96.7	81.8	90.4	94.5

Discussion

Discussion

Strengths

- Rates

Weaknesses

Discussion

Strengths

- Rates
- Zero-Shot

Weaknesses

Discussion

Strengths

- Rates
- Zero-Shot

Weaknesses

- Irrecoverable loss

Discussion

Strengths

- Rates
- Zero-Shot

Weaknesses

- Irrecoverable loss
- Interpretability of compressed data

Discussion

Strengths

- Rates
- Zero-Shot

Weaknesses

- Irrecoverable loss
- Interpretability of compressed data
- Set of transformations hard to find

⇒ Achieved orders of magnitude improvements in compression for predictions.

Thank you!
