Variational Adversarial Active Learning

Samarth Sinha *
University of Toronto
samarth.sinha@mail.utoronto.ca

Sayna Ebrahimi *
UC Berkeley
sayna@eecs.berkeley.edu

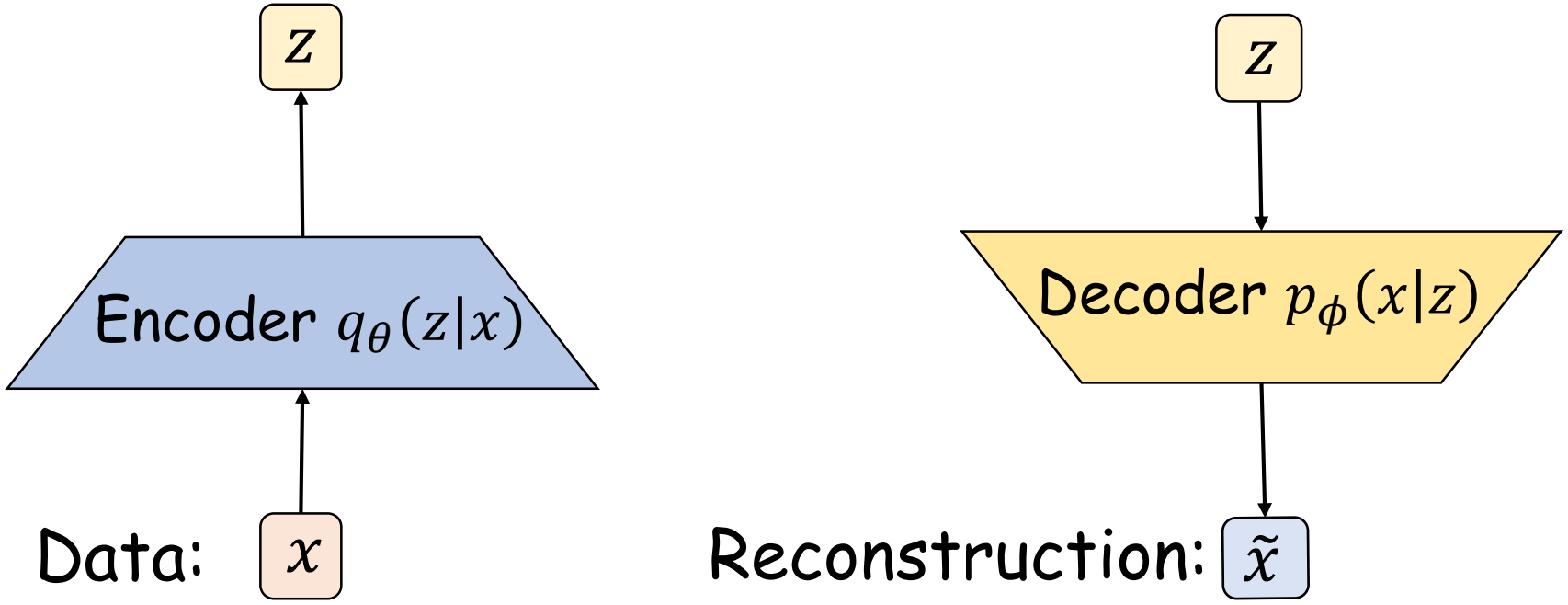
Trevor Darrell
UC Berkeley
trevor@eecs.berkeley.edu

Outline

- Variational Auto-Encoder (VAE)
  - A Neural Net Perspective
  - A Probability Model Perspective
- Variational Adversarial Active Learning (VAAL)

VAE - A Neural Net Perspective

Encoder and Decoder



How much information is lost?

Reconstruction log-likelihood $\log p_\phi(x|z)$

VAE - A Neural Net Perspective

Loss function: the negative log-likelihood with a regularizer

reconstruction loss

regularizer

$$l_i(\theta, \phi) = -\mathbb{E}_{z \sim q_\theta(z|x_i)}[\log p_\phi(x_i \mid z)] + \mathbb{KL}(q_\theta(z \mid x_i) \,||\, p(z))$$

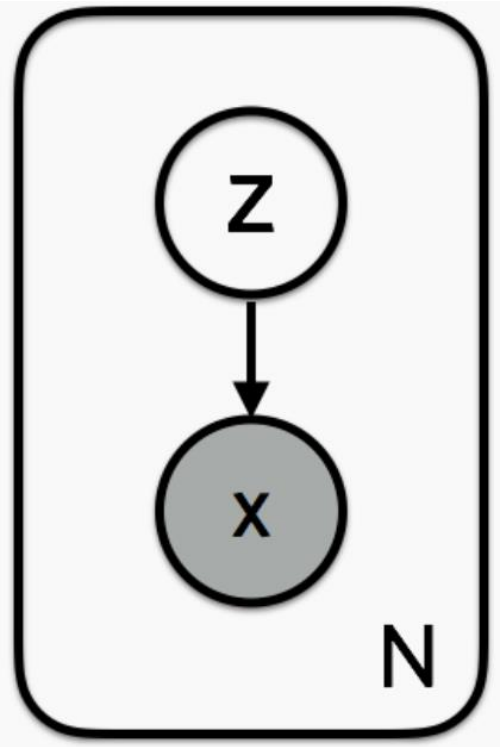
Encourages the decoder parameterizes a likelihood distribution that place much probability mass on the true data

Measures how much information is lost when using q to represent p

The total loss: $\sum_{i=1}^{N} l_i$

VAE - A Probability Model Perspective

The generative process



PRML. Chapter 8

For each datapoint i:

- Draw latent variable $z_i \sim p(z)$
- Draw datapoint $x_i \sim p(x|z)$

x_i ? z_i

Inference: $p(z|x)$

⇕

Infer good values of the latent variables given observed data

VAE - A Probability Model Perspective

How to calculate $p(z|x)$?

Bayes tells us: $$p(z \mid x) = \frac{p(x \mid z)p(z)}{p(x)}.$$

Evidence $p(x) = \int p(x \mid z)p(z)dz$

requires exp time to compute

Variational Inference

Use a family of distribution $q_\lambda(z|x)$ to approximate the true posterior $p(z|x)$

$$q_\lambda^*(z \mid x) = \arg\min_\lambda \mathbb{KL}(q_\lambda(z \mid x) \,||\, p(z \mid x))$$

where

$$\mathbb{KL}(q_\lambda(z \mid x) \,||\, p(z \mid x)) =$$

$$\mathbf{E}_q[\log q_\lambda(z \mid x)] - \mathbf{E}_q[\log p(x, z)] + \log p(x)$$

VAE - A Probability Model Perspective

$$\mathbb{KL}(q_\lambda(z \mid x) \,||\, p(z \mid x)) = \mathbf{E}_q[\log q_\lambda(z \mid x)] - \mathbf{E}_q[\log p(x, z)] + \log p(x)$$

$$ELBO(\lambda) = \mathbf{E}_q[\log p(x, z)] - \mathbf{E}_q[\log q_\lambda(z \mid x)]$$

$$\log p(x) = ELBO(\lambda) + \mathbb{KL}(q_\lambda(z \mid x) \,||\, p(z \mid x))$$

greater than or equal to 0

The Evidence Lower BOund allows us to approximate posterior inference.

The ELBO for a single datapoint

$$ELBO_i(\lambda) = \mathbb{E}q_\lambda(z \mid x_i)[\log p(x_i \mid z)] - \mathbb{KL}(q_\lambda(z \mid x_i) \,||\, p(z))$$

VAE - A Probability Model Perspective

The connection to neural net language

$q_\lambda(z \mid x_i)$ —parameterize→ Encoder with parameters θ

$p(x \mid z)$ —parameterize→ Decoder with parameters ϕ

$$ELBO_i(\theta, \phi) = -l_i(\theta, \phi)$$

$$ELBO_i(\theta, \phi) = \mathbb{E}q_\theta(z \mid x_i)[\log p_\phi(x_i \mid z)] - \mathbb{KL}(q_\theta(z \mid x_i) \mid\mid p(z))$$

reconstruction loss　　regularizer

The probability model makes clear why these terms exist:

to minimize the KL divergence between the approximate posterior $q_\lambda(z|x)$ and the model posterior $p(z|x)$.

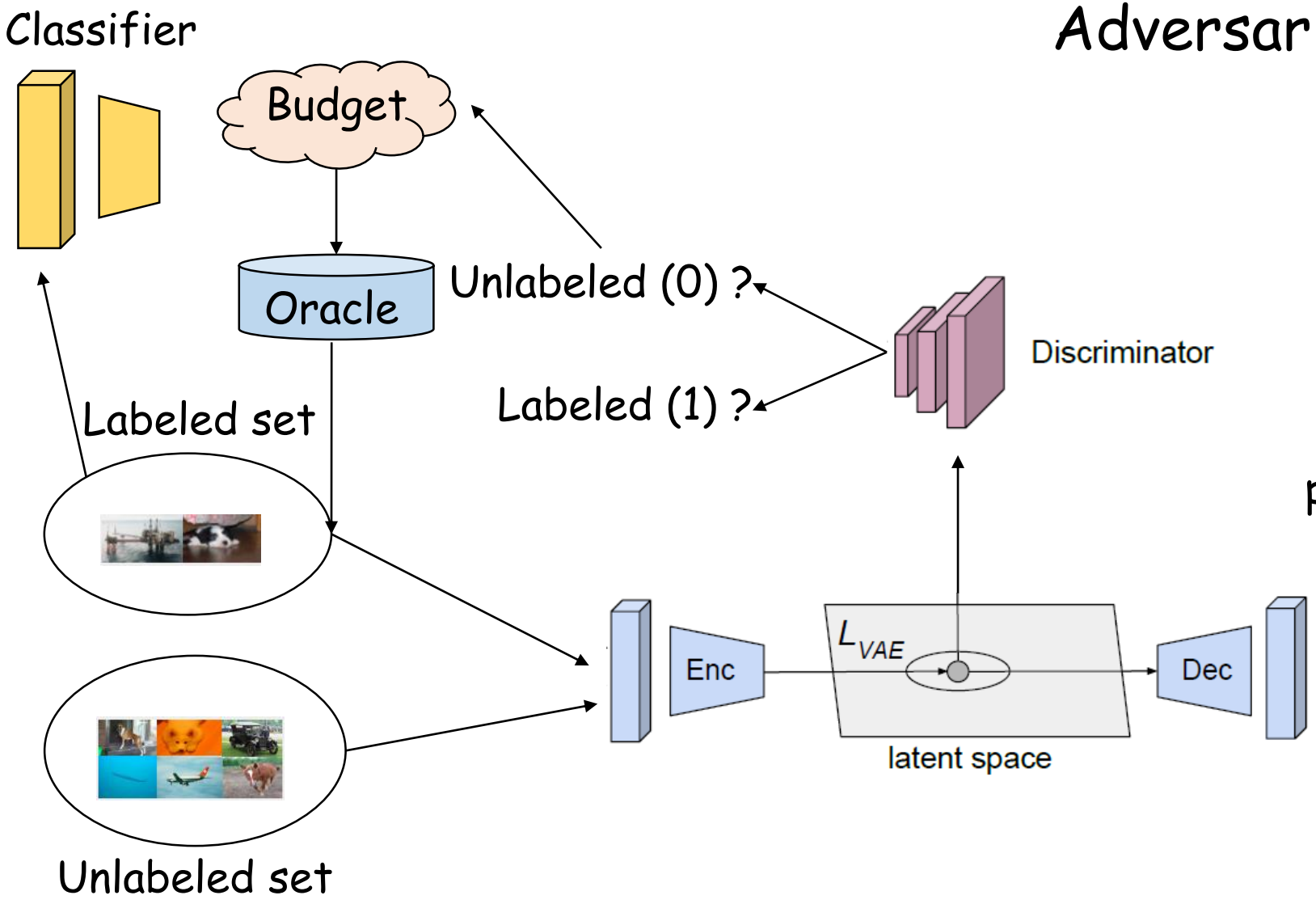
Variational Adversarial Active Learning

Samarth Sinha *
University of Toronto
samarth.sinha@mail.utoronto.ca

Sayna Ebrahimi *
UC Berkeley
sayna@eecs.berkeley.edu

Trevor Darrell
UC Berkeley
trevor@eecs.berkeley.edu

Idea



Adversarial learning

VAE fools the D to classify all inputs as labeled

VAE and D are learned together

D attempts to estimate the probability that the data comes from the unlabeled data

Variational Adversarial Active Learning (VAAL)

The objective function of the β-VAE

$$\begin{aligned}\mathcal{L}_{\text{VAE}}^{trd} = \quad & \mathbb{E}[\log p_\theta(x_L|z_L)] - \beta\, \mathrm{D}_{\text{KL}}(q_\phi(z_L|x_L)||p(z)) \\ & + \mathbb{E}[\log p_\theta(x_U|z_U)] - \beta\, \mathrm{D}_{\text{KL}}(q_\phi(z_U|x_U)||p(z))\end{aligned}$$

The objective function for the adversarial role of the VAE

$$\mathcal{L}_{\text{VAE}}^{adv} = \mathcal{L}_{\text{BCE}}(q_\phi(z_L|x_L), \mathbb{1}) + \mathcal{L}_{\text{BCE}}(q_\phi(z_U|x_U)), \mathbb{1})$$

The full objective function for the VAE: $\mathcal{L}_{\text{VAE}} = \lambda_1 \mathcal{L}_{\text{VAE}}^{trd} + \lambda_2 \mathcal{L}_{\text{VAE}}^{adv}$

The objective to train the discriminator

$$\mathcal{L}_{\text{D}} = \mathcal{L}_{\text{BCE}}(q_\phi(z_L|x_L)), \mathbb{1}) + \mathcal{L}_{\text{BCE}}(q_\phi(z_U|x_U)), \mathbb{0})$$

Variational Adversarial Active Learning (VAAL)

Algorithm 1 Variational Adversarial Active Learning

Input: Labeled pool (X_L, Y_L), Unlabeled pool (X_U), Initialized models for θ_T, θ_{VAE}, and θ_D

Input: Hyperparameters: epochs, λ_1, λ_2, α_1, α_2, α_3

for $e = 1$ to epochs **do**

sample $(x_L, y_L) \sim (X_L, Y_L)$

sample $x_U \sim X_U$

mini-batch sampling

Compute $\mathcal{L}_{\text{VAE}}^{trd}$ by using Eq. 1

Compute $\mathcal{L}_{\text{VAE}}^{adv}$ by using Eq. 2

$\mathcal{L}_{\text{VAE}} \leftarrow \lambda_1 \mathcal{L}_{\text{VAE}}^{trd} + \lambda_2 \mathcal{L}_{\text{VAE}}^{adv}$

Update VAE by descending stochastic gradients:

$\theta'_{VAE} \leftarrow \theta_{VAE} - \alpha_1 \nabla \mathcal{L}_{\text{VAE}}$

VAE training

$\mathcal{L}_{\text{D}} \leftarrow \mathcal{L}_{\text{BCE}}(q_\phi(z_L|x_L)), \mathbb{1}) + \mathcal{L}_{\text{BCE}}(q_\phi(z_U|x_U)), \mathbb{0})$

Update D by ascending its stochastic gradient:

$\theta'_D \leftarrow \theta_D - \alpha_2 \nabla \mathcal{L}_{\text{D}}$

Dis training

Train and update T:

$\theta'_T \leftarrow \theta_T - \alpha_3 \nabla \mathcal{L}_{\text{T}}$

update classifier

end for

return Trained $\theta_T, \theta_{VAE}, \theta_D$

Algorithm 2 Sampling Strategy in VAAL

Input: b, X_L, X_U

Output: X_L, X_U

Select samples (X_s) with $\min_b\{\theta_D(z_U)\}$

$y_o \leftarrow \mathcal{ORACLE}(X_s)$

$X_L \leftarrow X_L \cup (X_s, y_o)$

$X_U \leftarrow X_U - X_s$

return X_L, X_U

Experiments

Datasets

- Image Classification
  - CIFAR10 and CIFAR100
  - Caltech-256
  - ImageNet
- Semantic Segmentation
  - BDD100K
  - Cityscapes

Experiments

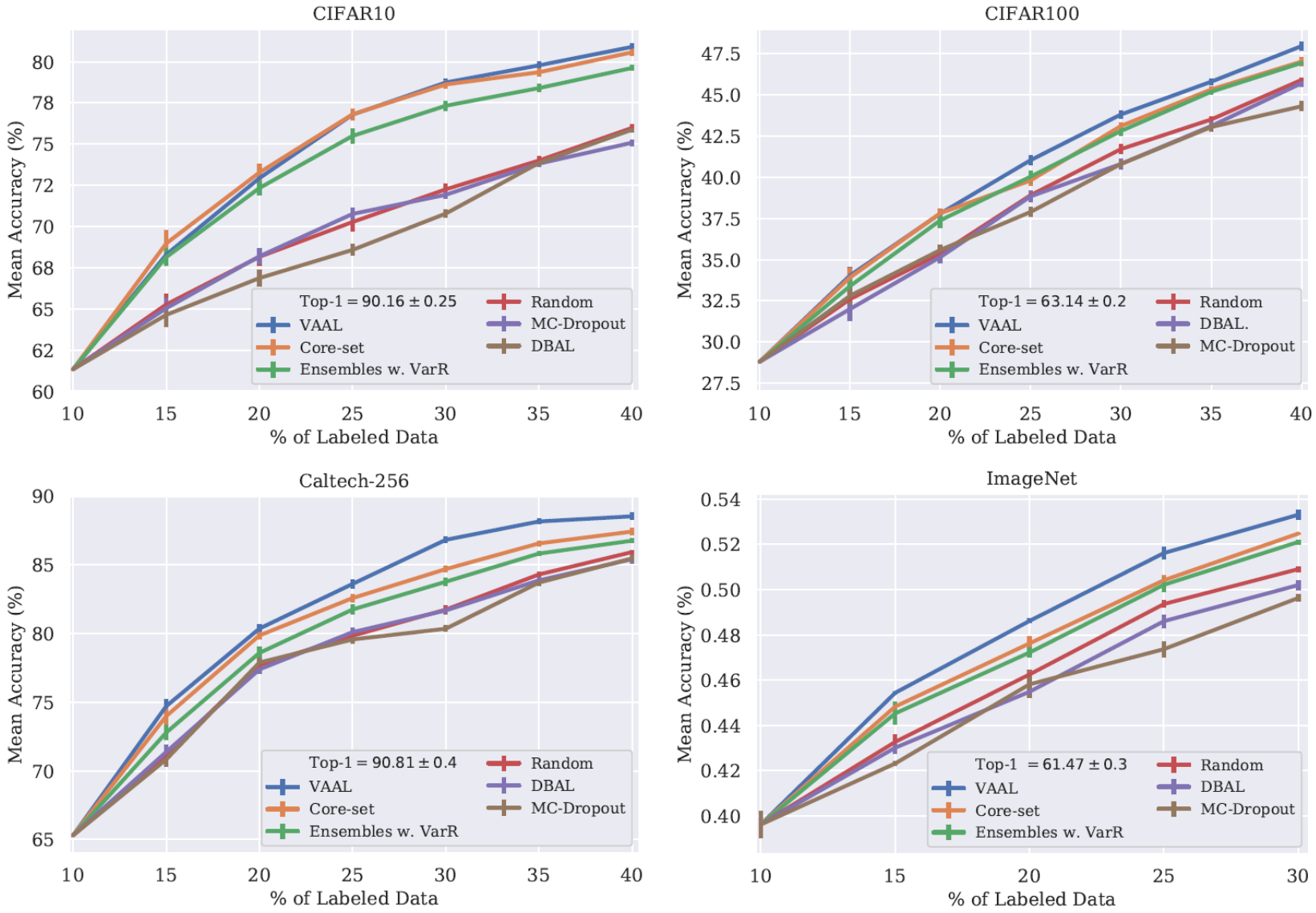
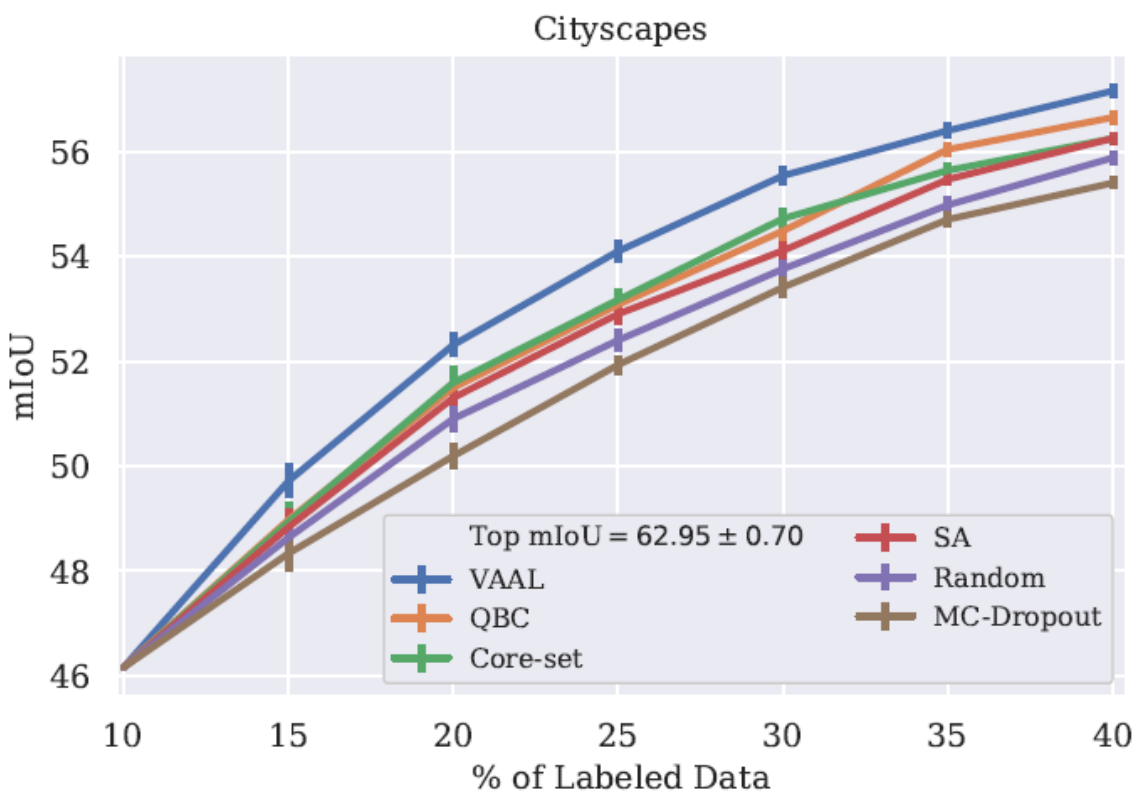
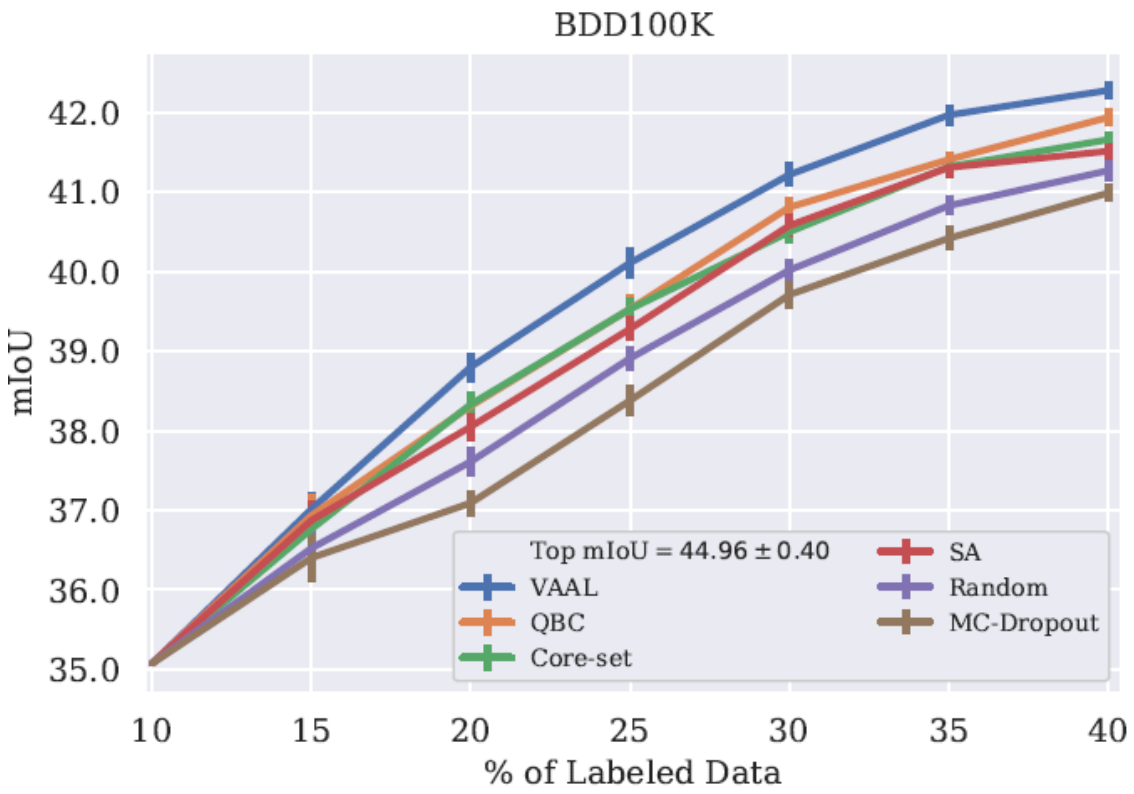


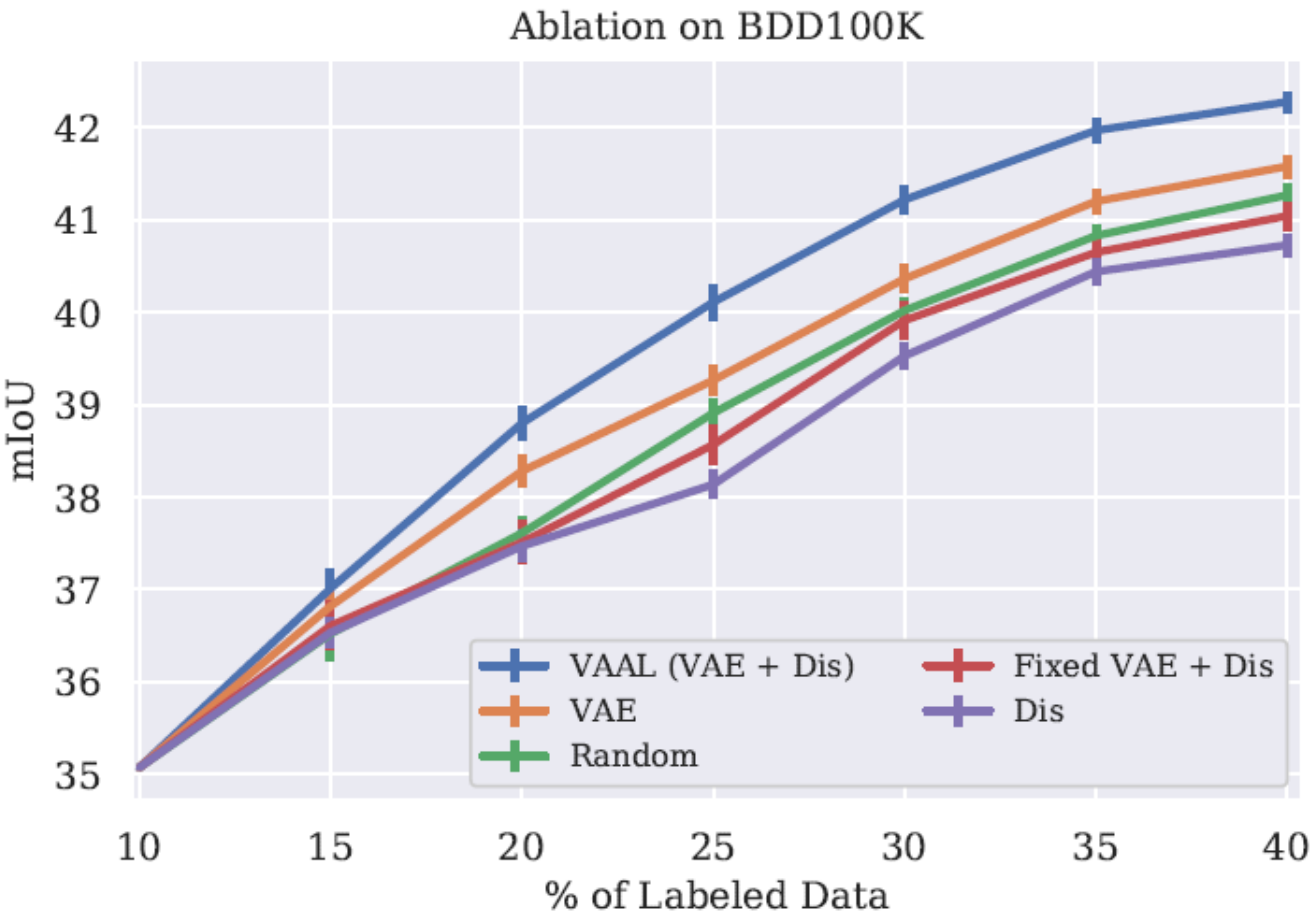
Figure 2. VAAL performance on classification tasks using CIFAR10, CIFAR100, Caltech-256, and ImageNet compared to Core-set [40], Ensembles w. VarR [1], MC-Dropout [13], DBAL [14], and Random Sampling.

Experiments





Experiments



Dis: eliminate VAE

Fixed VAE + Dis: frozen VAE with D

VAE: eliminate D

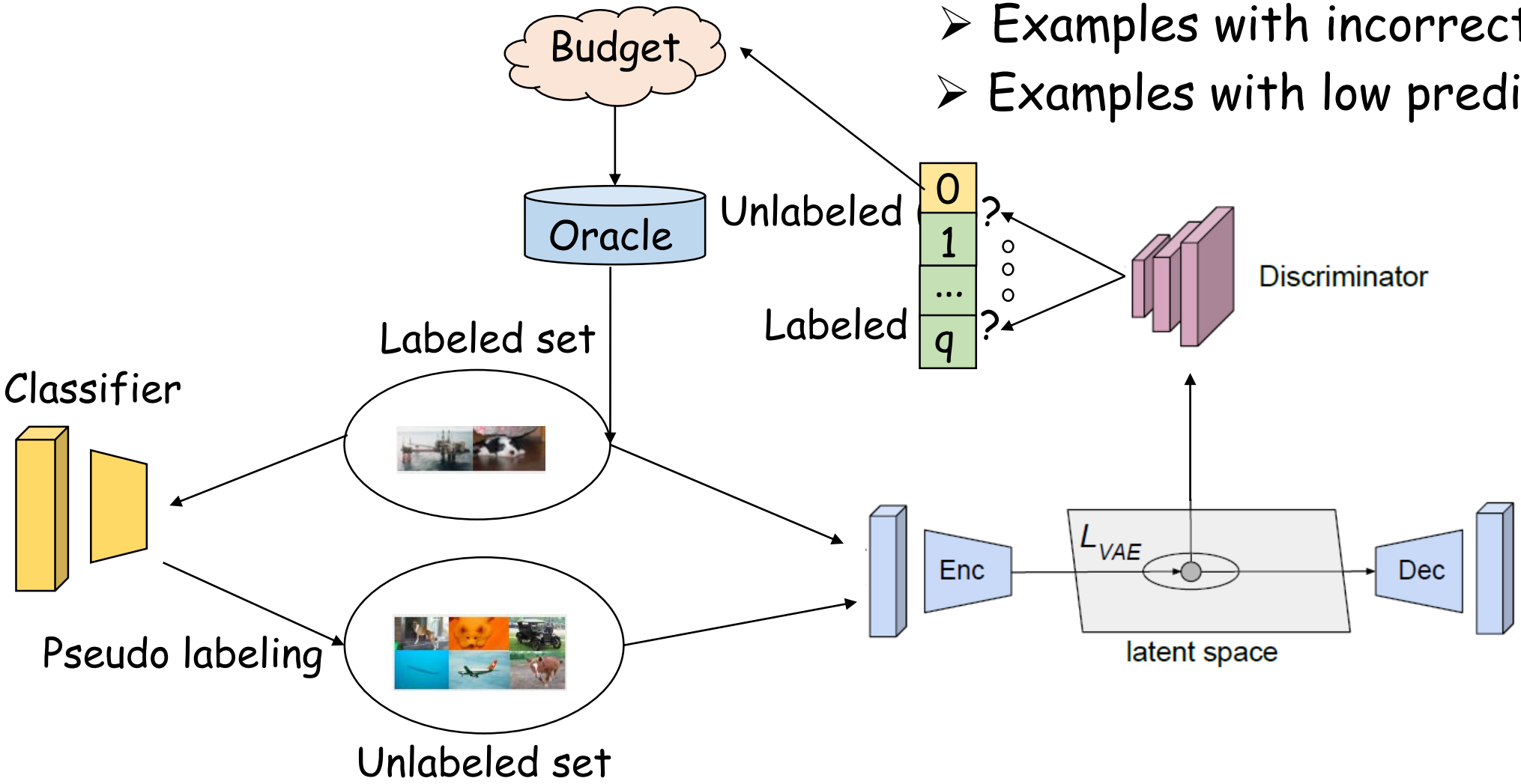
Use 2-Wasserstein distance from the cluster-centroid of the labeled dataset as a heuristic to explicitly measure uncertainty.

Improvement

Uncertainty

Tow kinds of examples will be identified:

- Examples with incorrect pseudo labels
- Examples with low prediction confidence



Thanks