Weekly Study Report

Wang Ma

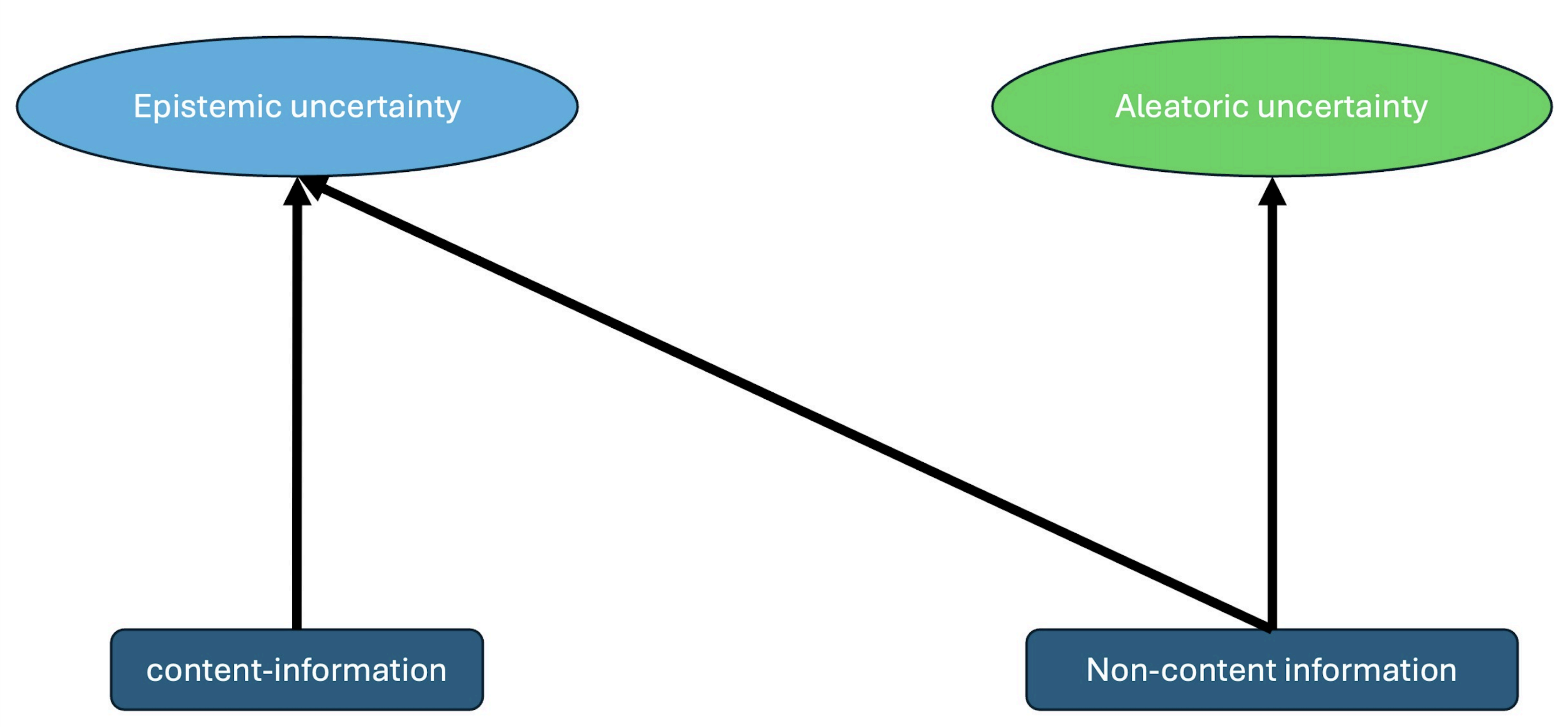
2025-04-22

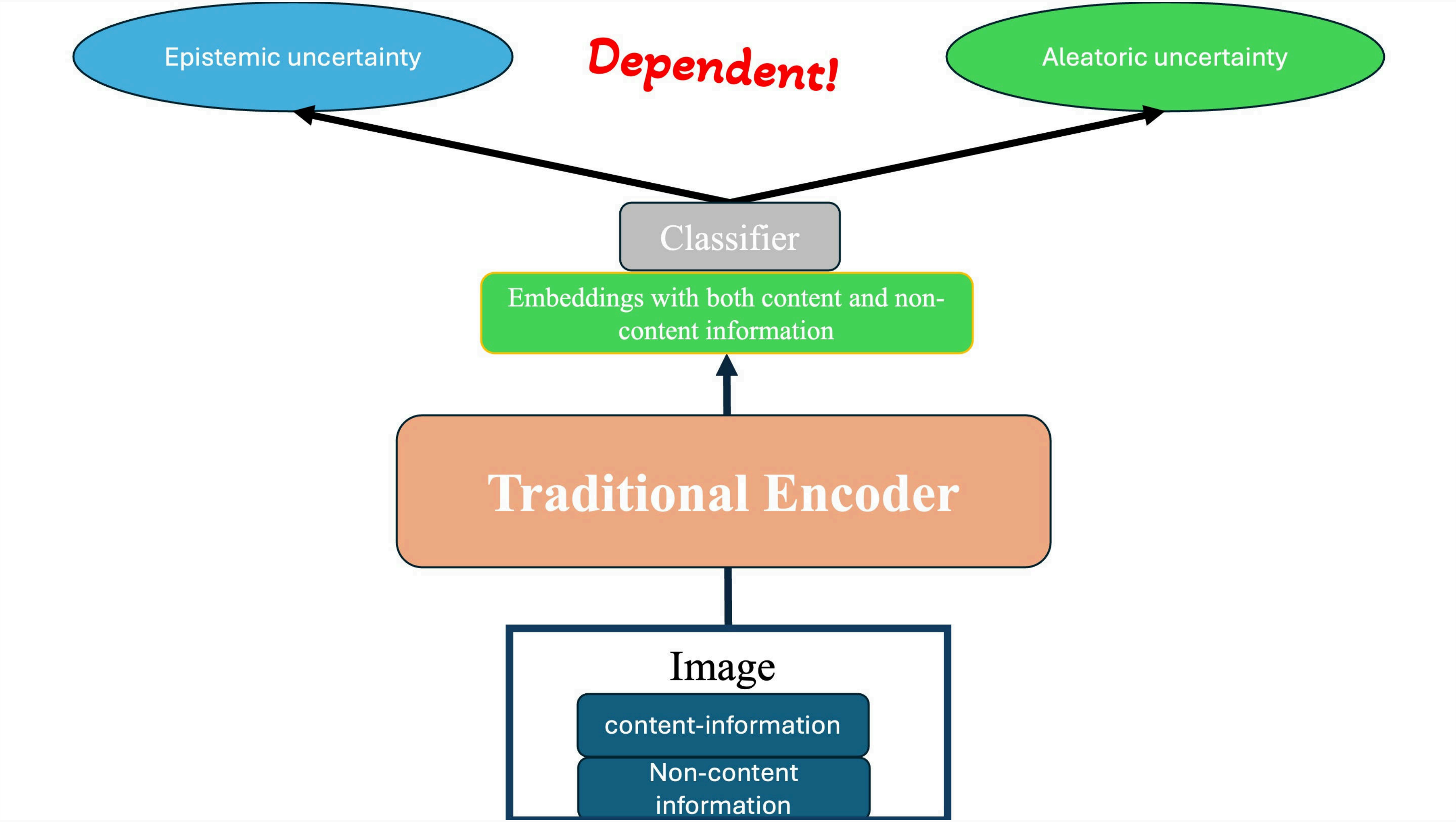
Electrical, Computer, and Systems Engineering Department
Rensselaer Polytechnic Institute

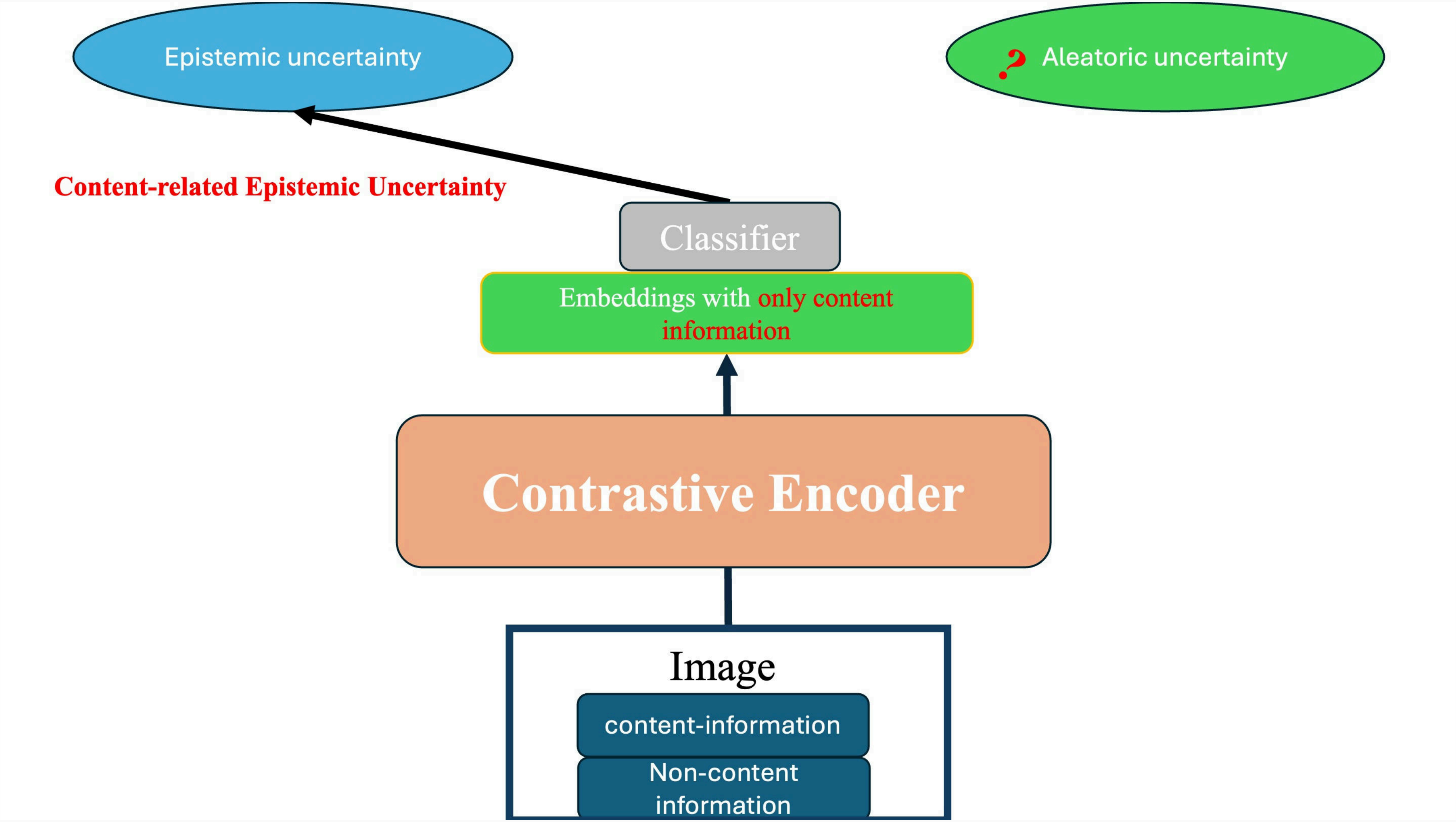
Outline

1. [Updated Theoretical Results] Contrastive Learning to get content-related Epistemic Uncertainty

1.1 Background



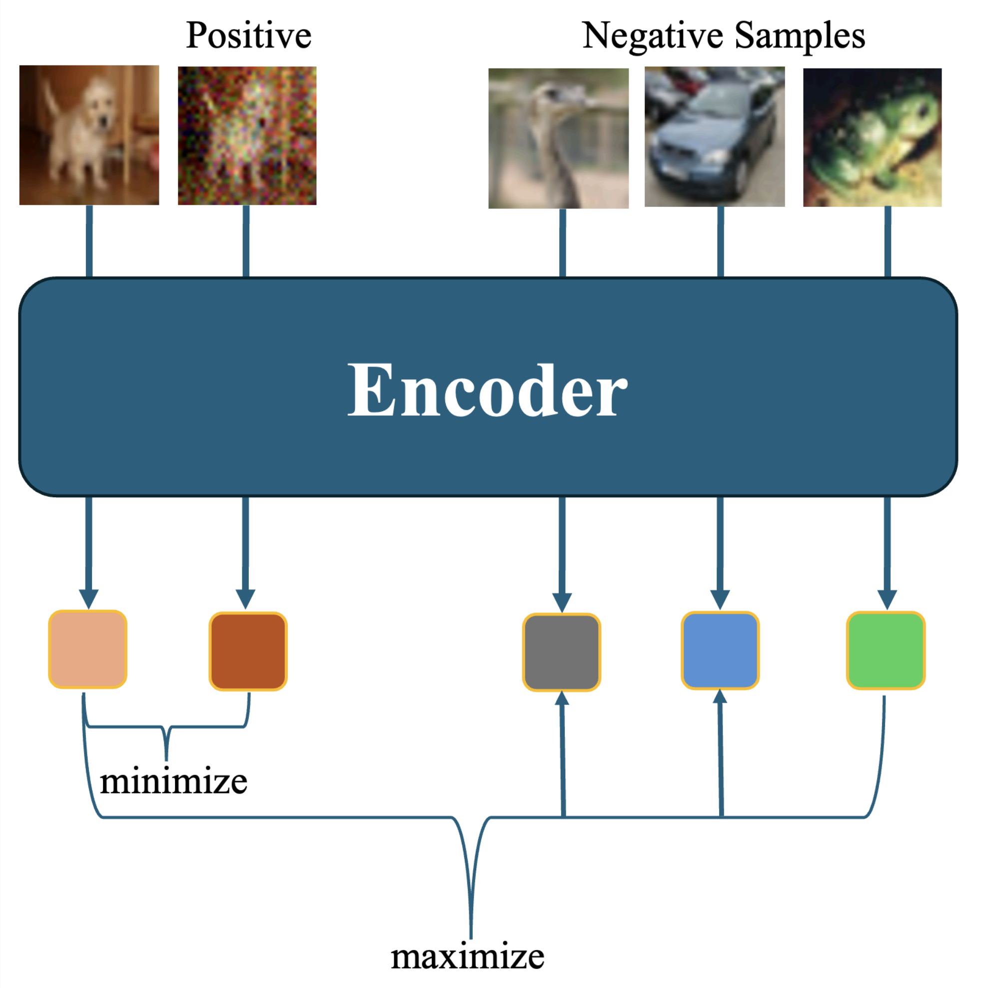

Epistemic uncertainty
Dependent!
Aleatoric uncertainty
Classifier
Embeddings with both content and non-content information
Traditional Encoder
Image
content-information
Non-content information


Epistemic uncertainty
Aleatoric uncertainty
Content-related Epistemic Uncertainty
Classifier
Embeddings with only content information
Contrastive Encoder
Image
content-information
Non-content information

GOAL.

1. Training a contrastive encoder to learn consistent features/embeddings from high- and low- quality input.
2. We want to minimize the influence of non-content information to the epistemic uncertainty.
3. The final goal is to obtain True content-related Epistemic Uncertainty, which can be used to detect in-lier data when both AU and EU are high.

1.3 The Contrastive Learning Model (Encoder Learning)



- Anchor: the clean (high-quality) image
- Positive Samples: the corrupted image
- Negative Samples: other images in batch

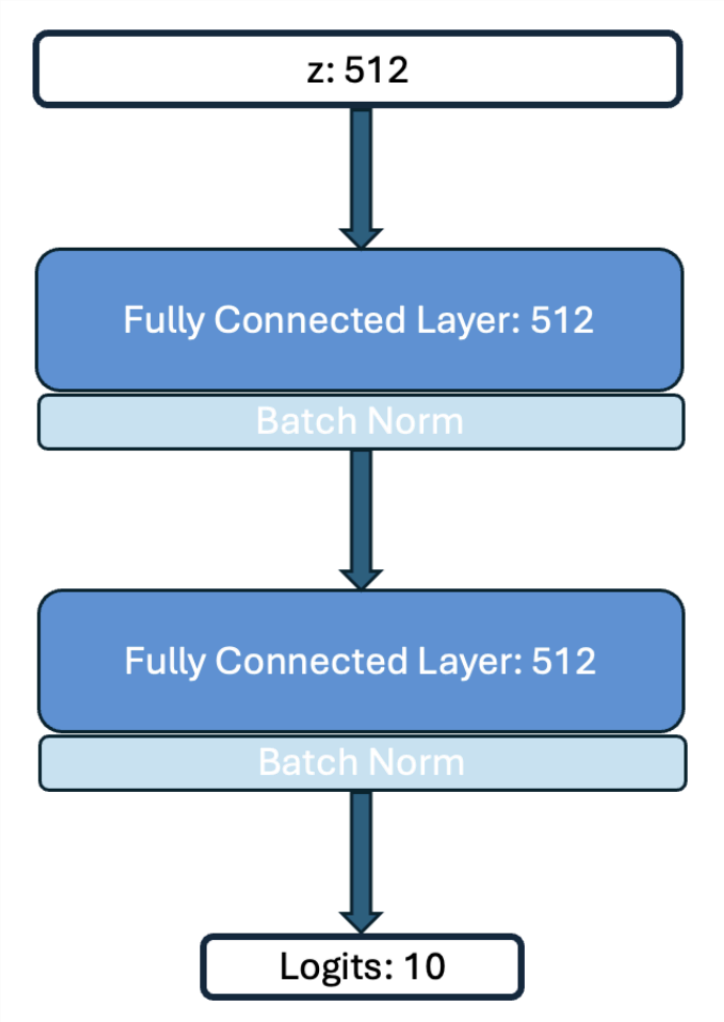
Output of the Encoder: the embedding z

The loss function:

$$L_i = -\log \frac{\exp(z_i z_i^+)}{\sum_{k,k\in z_i^-} \exp(z_i z_k)},$$

where $\|z\|^2 = 1$, so $z_i z_i^j$ is the cosine similarity of the two embeddings.

1.4 The MLP Classifier Model



During the training of the Classifier head, we freeze the encoder part and solely update the parameters in the MLP head.

This training follows a standard Classification task training with cross entropy loss.

1.5 The Idea

There exists a Ground-Truth "Content-based Uncertainty":

$$\mathbb{H}[y|c, D] = \mathbb{I}[y, \theta|c, D] + \mathbb{E}_\theta[\mathbb{H}[y|c, \theta]].$$

Content-based Uncertainty is good at identifying true OOD and Low-quality ID data.

We want to use the contrastive learning to train a model, and let the UQ

$$\mathbb{H}[y|x, D] = \mathbb{I}[y, \theta|x, D] + \mathbb{E}_\theta[\mathbb{H}[y|x, \theta]]$$

approximate the content-based uncertainty.

Especially, we want

$$\| \mathrm{EU}_x - \mathrm{EU}_c \| \to 0.$$

1.6 Content-based Uncertainty is Smaller

Note that

$$p(y|x, D) = \sum_c p(y|c, D)p(c|x, D) = \mathbb{E}_{c \sim p(c|x,D)}[p(y|c, D)].$$

Then unless x perfectly determines c (i.e. $p(c|x, D)$ is a Delta Distribution), we always ave

$$\mathbb{H}[p(y|x, D)] = \mathbb{E}_{c \sim p(c|x,D)}[\mathbb{H}[p(y|c, D)]] + \mathbb{I}[y; c|x, D],$$

which means the total uncertainty measured by the content is smaller than conventional way.

❓Does similar conclusion also holds for Epistemic Uncertainty,

$\mathrm{EU}_x - \mathrm{EU}_c = \mathbb{I}[c; y|x, D] - \mathbb{I}[c; y|x, \theta, D] \leq 0.(?)$

1.6 Content-based Uncertainty is Smaller

1.6.1 Uncertainty Quantification Performance

Contrastive Leared Encoder_not_pretrained		Clean_id	Corrupted_trained	Corrupted_not_trained	OOD
Test Accuracy		0.8863	0.8635	0.6353	\
Total Uncertainty	mean	0.3667	0.4197	0.6592	1.1948
	std	0.4528	0.4756	0.5338	0.4475
Aleatoric Uncertainty	mean	0.2941	0.3335	0.4956	0.9287
	std	0.3661	0.3821	0.411	0.3492
Epistemic Uncertainty	mean	0.0726	0.0862	0.1636	0.2661
	std	0.0998	0.1097	0.158	0.146

ResNet18 Results_not_pretrained		Clean_id	Corrupted_trained	Corrupted_not_trained	OOD
Test Accuracy		0.8784	0.8224	0.6977	\
Total Uncertainty	mean	0.3766	0.5853	0.984	1.429
	std	0.4611	0.5655	0.6472	0.4831
Aleatoric Uncertainty	mean	0.3132	0.509	0.8692	1.3167
	std	0.3866	0.5034	0.5852	0.4625
Epistemic Uncertainty	mean	0.0634	0.0763	0.1148	0.1123
	std	0.0851	0.082	0.0904	0.0558

1.7 A lemma

Let L_{InfoNCE} be optimized with anchor-positive pairs (x, x_1) which share the same c but independent non-content factor n, n_1. Then for the learned encoder $z = f_\varphi(x)$,

$\mathbb{I}[c; z_1]$ is maximized and ($\mathbb{I}[n; z_1|c]$ is minimized).

From Poole et al. 2019, we get $-L_{\text{InfoNCE}} \leq \mathbb{I}[z; z_1] = \mathbb{I}[z_1; c] - \mathbb{I}[z_1; c|z] \leq \mathbb{I}[z_1; c]$ ◦

So optimizing the InfoNCE loss is increasing the lower bound of $\mathbb{I}[z_1; c]$, which means pushing the learned embedding of a corrupted/augmented samples to best align with the content information shared with the clean anchor image x.

From Wang & Isola 2020, we have $L_{\text{InfoNCE}} = \frac{\|z_1 - z_2\|^2}{2\tau} + \log Z'$

And it is shown that $\mathbb{E}[\![|z_1 - z_2\|^2]\!] \to 0 \Rightarrow D_{\text{KL}}(p(z|c,n)\|p(z|c))0 \to 0$ by Pinsker's and Jensen's inequality.

Then since $\mathbb{I}[n; z|c] = \mathbb{E}_{c,n}[D_{\text{KL}}(p(z|c,n)\|p(z|c))]$, we obtain that during the training, $\mathbb{I}[n; z|c] \to 0$.

1.8 Contrastive Trained Encoder Can Learn Content-related Epistemic Uncertainty

$$\mathrm{EU}_x = \mathbb{I}[y;\theta|x,D] = \mathbb{I}[y;\theta|c,n,D] = \mathbb{I}[y;\theta|c,D] - \mathbb{I}[y;n|c,D] + \mathbb{I}[y;n|c,\theta,D]$$

$$\mathrm{EU}_c = \mathbb{I}[y;\theta|c,D]$$

Then

$$\begin{aligned}\|\mathrm{EU}_x - \mathrm{EU}_c\| &= \| - \mathbb{I}[y;n|c,D] + \mathbb{I}[y;n|c,\theta,D]\| \\ &\leq \mathbb{I}[y;n|c,\theta,D] \\ &\leq \mathbb{I}[z;n|c,\theta,D] \\ &\leq \mathbb{I}[n;z|c]\end{aligned}$$ ⬇

1.9 The Efficiency of Content-based Methods

Contrastive Learned Encoder_not_pretrained		Corrupted_id vs Corrupted_ood	Corrupted_id vs SVHN (OOD)	Corrupted_ood vs SVHN(OOD)
Total Uncertainty	AUROC	0.6388	0.8709	0.7742
	AUPR	0.6201	0.9325	0.877
Aleatoric Uncertainty	AUROC	0.6268	0.864	0.7844
	AUPR	0.5971	0.9236	0.88
Epistemic Uncertainty	AUROC	0.6563	0.8503	0.7031
	AUPR	0.6653	0.9238	0.8207

Resnet 18 _not_pretrained		Corrupted_id vs Corrupted_ood	Corrupted_id vs SVHN (OOD)	Corrupted_ood vs SVHN(OOD)
Total Uncertainty	AUROC	0.6767	0.8586	0.7008
	AUPR	0.682	0.9263	0.818
Aleatoric Uncertainty	AUROC	0.6776	0.8681	0.7201
	AUPR	0.6782	0.9357	0.8433
Epistemic Uncertainty	AUROC	0.6831	0.6831	0.5228
	AUPR	0.776	0.776	0.6862

Conclusion:

1. The Contrastive Learned Encoder is consistent to Corrupted_id data. And for Corrupted_ood data, the AUROC is low meaning the model does not really see difference in highly-corrupted data.
2. Meanwhile, the model can identify true OOD data well, even the test accuracy is not good. But it seems the EU does not work well on SVHN detection.
3. The Resnet-18 almost failed on the OOD detection, which means it cannot capture significant features. This means I need to check the model training and retrain the resnet18 models.
4. The Contrastive Learned Encoder has the ability to identify corrupted_ood from SVHN, that is, the low-quality version from the true OOD. But Resnet-18 failed to do this, especially for the un-pretrained resnet 18.