Enhancing The Reliability of Out-of-distribution Image Detection in Neural Networks

Shiyu Liang, Yixuan Li, R. Srikant

Grepp
Zayden

목차

딥러닝 모델의 학습

훈련 데이터의 분포와 테스트 데이터의 분포가 비슷할때 테스트 데이터에 대한 모델의 성능이 높다.

실제로 모델을 배포한다면?

- 실제 상황에서는 모델의 **훈련과정에서 접하지 않은 다양한 분포의 데이터**를 마주할 수 있다.
- 새로운 종류[1]의 데이터 (Out-Of-Distribution; OOD)를 모델에 넣는경우 특정 class로 분류되기 때문에 **기대와 다르게 특정 class에 높은 확률값을 보이는 경우가 많다**.

1) 처음보거나, 분류가 불가능하거나, 관련없는 데이터

Out-Of-Distribution (OOD)

예측가능한 분포안에 있는 데이터 (In-Distribution; ID)와 다르게 분류할 수 없는 OOD를 판별하는것이 중요하다.

일반적으로 잘 학습된 딥러닝 모델은 OOD보다 ID에 높은 확률값을 보인다

- softmax에 temperature scaling를 추가
- 입력에 노이즈[1] 추가

→ **ID와 OOD 사이의 softmax score 차이가 커짐**

1) 논문에는 perbutation으로 되어있고 데이터에 포함된 복잡한, 교란시키는 정도의 의미

문제 정의

이미지 $P_{\mathbf{X}}$와 $Q_{\mathbf{X}}$가 $P_{\mathbf{X}}Q_{\mathbf{X}} \in \mathcal{X}$일때
혼합 분포를 가지는 새로운 이미지 $\mathbb{P}_{\mathbf{X}|Z} \in \mathcal{X} \times (0,1)$가 아래와 같을때,

- $\mathbb{P}_{\mathbf{X}|Z=0} = P_{\mathbf{X}}$ → In distribution
- $\mathbb{P}_{\mathbf{X}|Z=1} = Q_{\mathbf{X}}$ → Out distribution

→ $\mathbb{P}_{\mathbf{X}|Z}$ 분포를 가진 이미지 $\mathbf{X}$가 주어지면 이 이미지를 $P_{\mathbf{X}}$인지 아닌지 분류할 수 있을까?

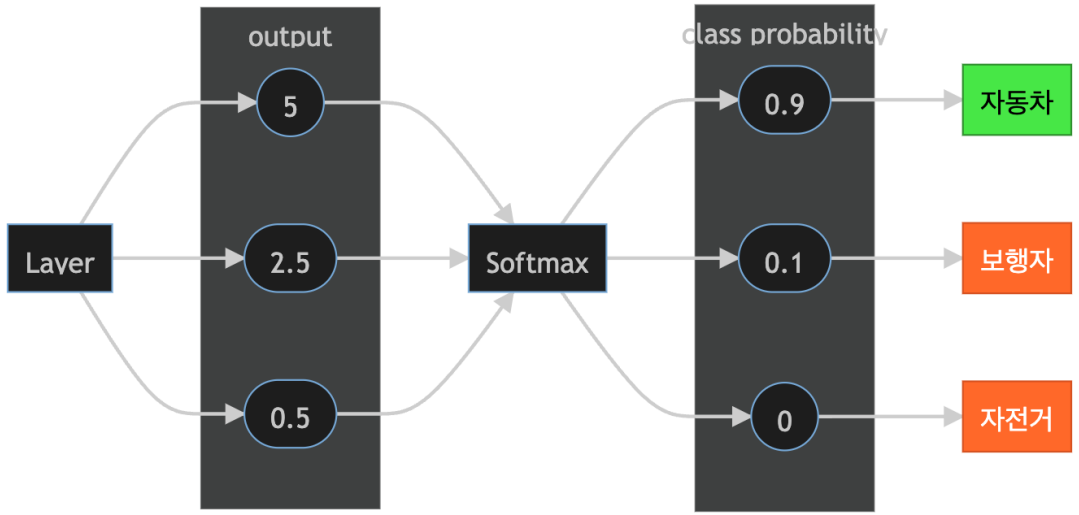
ODIN (Out-of-DIstribution detector for Neural net- works)

Contributions

- 딥러닝 모델의 재학습이 필요없다!
- 최신 모델들에도 쉽게 적용할 수 있다!
- 기존의 baseline model보다 큰폭으로 성능이 향상되었다!
- 하이퍼 파라미터의 직관적 선택이 가능하도록 실험을 많이 진행하였다!

1. Temperature scaling

Softmax


output
class probability
Layer
5
2.5
0.5
Softmax
0.9
0.1
0
자동차
보행자
자전거

모델의 출력으로부터 각 class에 대한 class confidence를 구하기위해 사용하는 Softmax는 다음과 같이 정의된다.

$$S_i(x) = \frac{\exp f_i(x)}{\sum_{j=1}^{N} \exp f_j(x)}$$

이때 출력 class는 $argmax_i S_i(x)$로 선택한다.

T가 양의 실수일때 Temperature scaling은 아래와 같이 적용한다.

$$S_i(x, T) = \frac{\exp f_i(x/T)}{\sum_{j=1}^{N} \exp f_j(x/T)}$$

Temperature Scaling의 영향

- softmax의 argmax에 영향이 없이 **Calibration**을 수행할 수 있음.
    - 특정 class에 over confidence하는 현상을 줄일 수 있음. (soften)

Calibration

단순히 높은 Confidence를 선택해서 결정되는 Accuracy가 아닌 실제 class에 대한 confidence를 따라가게 하는것.

Calibration

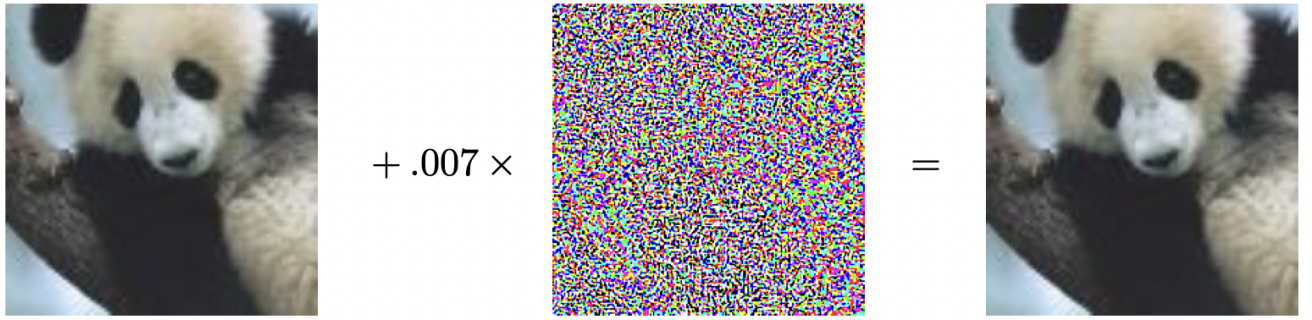
- 강아지 : 머핀
  - `0.9 : 0.1` → 실제 딥러닝 모델의 출력은 맞으나 confidence가 과도함
  - `0.6 : 0.4` → 실제 현실에서의 class confidence와 비슷

2. Input pre-processing

Inspired

Goodfellow et al., 2015의 FGSM에서 영감을 얻음

- gradient에 의한 작은 perturbation을 이용해 softmax score를 낮춰서 입력 이미지를 다른 class로 오해하게 만드는 기법



반대로!

입력 이미지에 대한 softmax score를 높이는 방향으로 perturbation을 추가한다.

$$\tilde{x} = x - \epsilon sign(-\nabla_x log S_{\hat{y}}(x;T)),$$

앞서 두가지 방법을 조합하자!

Out-of-distribution Detector

perturbation을 추가한 데이터에 대한 모델의 출력에 Temperature scaling을 적용

- threshold δ보다 출력 확률이 낮거나 같은 경우 → **OOD**
- threshold δ보다 출력 확률이 높은 경우 → **ID**

$$g(x;\delta;T;\epsilon) = \begin{cases} 1 & \text{if } max_i\ p(\tilde{x};T) \leq \delta, \\ 0 & \text{if } max_i\ p(\tilde{x};T) > \delta. \end{cases}$$

Discussion

Trained models

- DenseNet (Huang et al., 2016)
- Wide ResNet (Zagoruyko & Komodakis, 2016)

Trained model error rate

Architecture	CIFAR-10	CIFAR-100
Dense-BC	4.81	22.37
WRN-28-10	3.71	19.86

Out-of-distribution Datsets

학습된 모델이 CIFAR-10, CIFAR-100에 대해 학습되어 있어서 한번도 보여지지 않은 데이터를 OOD로 테스트 하기위해서 아래와 같은 데이터를 이용함

1. TinyImageNet
2. LSUN
3. Gaussian Noise
4. Uniform Noise

Evaluation Metrics

Predicted \ Real	Positive	Negative
Positive	TP	FP
Negative	FN	TN

- **민감도(Sensitivity, TPR, Recall)**
  실제 Positive중에서 Postive로 예측된 비율
- **특이도(Specificity, TNR)**
  실제 Negative중에서 Negative로 예측된 비율
- **FPR**
  실제 Negative중에서 Posivie로 예측된 비율
- **정밀도 (Precision)**
  Positive로 예측된 값중에서 실제 Positive의 비율

OOD Test Metric

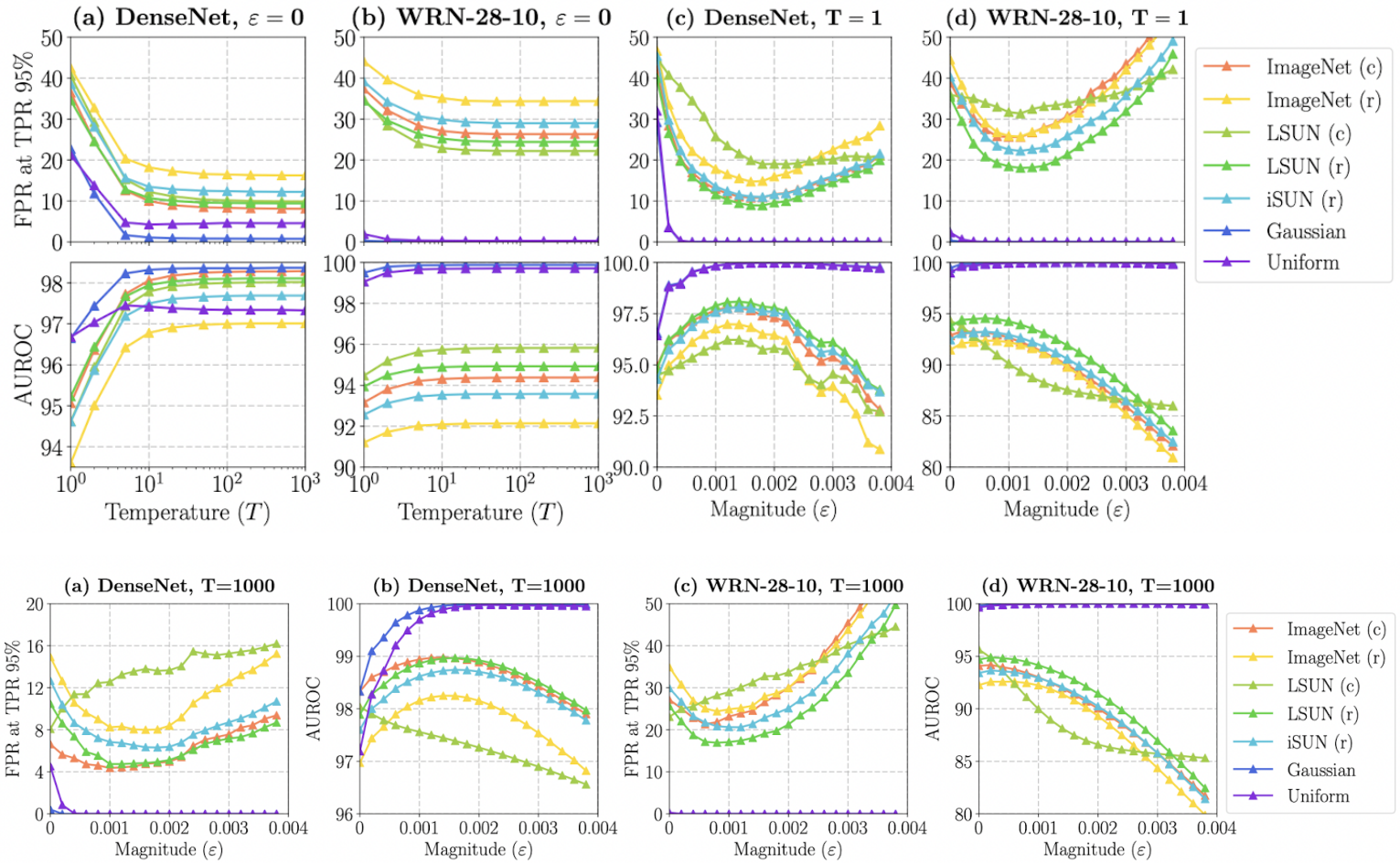
1. **FPR@TPR=0.95**
   True Positive Rate가 95%일때 False Positive Rate
2. **Detection Error**
   TPR이 95%일때 잘못 classify할 확률 ($P_e = 0.5(1 - TPR) + 0.5FPR$)
3. **AUROC (Area Under the Receiver Operating Characteristic curve)**
   Classify Threshold를 조절하면서 TPR/FPR 그래프 아래 면적
4. **AUPR**
   Classify Threshold를 조절하면서 Precision/recall 커브의 아래 면적

Result

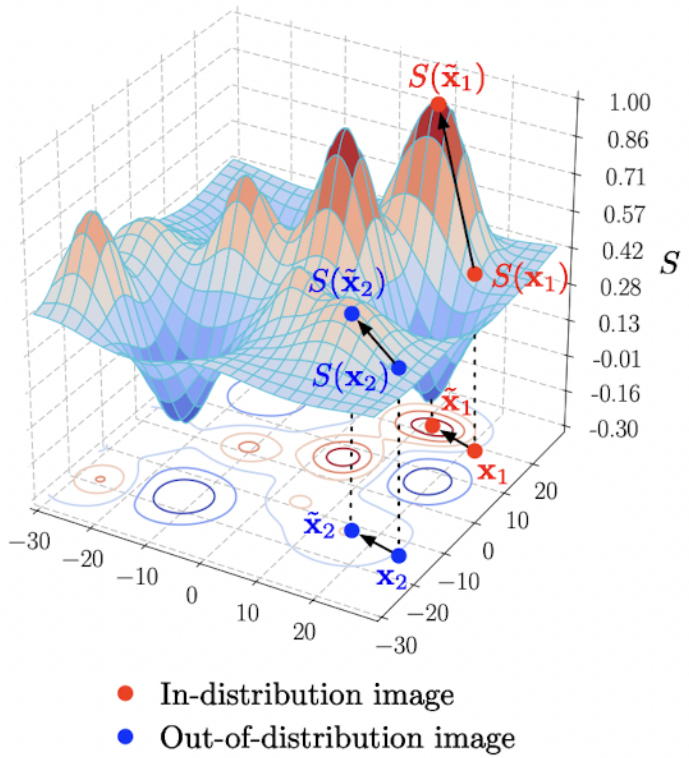
	Out-of-distribution dataset	FPR (95% TPR) ↓	Detection Error ↓	AUROC ↑	AUPR In ↑	AUPR Out ↑
		Baseline (Hendrycks & Gimpel, 2017) / ODIN				
Dense-BC CIFAR-10	TinyImageNet (crop)	34.7/**4.3**	10.0/**4.7**	95.3/**99.1**	96.4/**99.1**	93.8/**99.1**
	TinyImageNet (resize)	40.8/**7.5**	11.5/**6.1**	94.1/**98.5**	95.1/**98.6**	92.4/**98.5**
	LSUN (crop)	39.3/**11.4**	10.2/**7.2**	94.8/**97.9**	96.0/**98.0**	93.1/**97.9**
	LSUN (resize)	33.6/**3.8**	9.8/**4.4**	95.4/**99.2**	96.4/**99.3**	94.0/**99.2**
	Uniform	23.5/**0.0**	5.3/**0.5**	96.5/**99.0**	97.8/**100.0**	93.0/**99.0**
	Gaussian	12.3/**0.0**	4.7/**0.2**	97.5/**100.0**	98.3/**100.0**	95.9/**100.0**
Dense-BC CIFAR-100	TinyImageNet (crop)	67.8/**26.9**	36.4/**12.9**	83.0/**94.5**	85.3/**94.7**	80.8/**94.5**
	TinyImageNet (resize)	82.2/**57.0**	43.6/**22.7**	70.4/**85.5**	71.4/**86.0**	68.6/**84.8**
	LSUN (crop)	69.4/**18.6**	37.2/**9.7**	83.7/**96.6**	86.2/**96.8**	80.9/**96.5**
	LSUN (resize)	83.3/**58.0**	44.1/**22.3**	70.6/**86.0**	72.5/**87.1**	68.0/**84.8**
	Uniform	100.0/**100.0**	35.86/**17.9**	43.1/**99.5**	63.2/**87.5**	41.9/**65.1**
	Gaussian	100.0/**100.0**	41.2/**38.0**	30.6/**40.5**	53.4/**60.5**	37.6/**40.9**

Discussion

The effects of T



The effects of gradient $\nabla_x log S(x;T)$



The effects of ϵ

- ϵ이 작은 값일때는 큰 영향이 없지만 무시할 수 없을정도의 값을 가지는 경우 preprocessing을 거친 이미지는 $||\nabla_x log S(x;T)||_1$의 영향을 받는다.
- 하지만 ϵ이 너무 큰경우 classification 성능이 감소한다.

Q&A

References

- Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. ICLR, 2015.
- Gao Huang, Zhuang Liu, and Kilian Q Weinberger. Densely connected convolutional networks. arXiv preprint arXiv:1608.06993, 2016.
- Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In CVPR, 2016.