Distance-Based Learning From Errors for Confidence Calibration

Chen Xing*
College of Computer Science,
Nankai University
Tianjin, China

Sercan Ö. Arık
Google Cloud AI
Sunnyvale, CA

Zizhao Zhang
Google Cloud AI
Sunnyvale, CA

Tomas Pfister
Google Cloud AI
Sunnyvale, CA

ICLR 2020

2020.07.16

최영제

Index

1. Introduction & Related work

2. Proposed Approach

3. Experiments

1 SIL Introduction & Related work

Background

- 오늘날 neural net은 과거의 neural net 보다 정확도는 향상되었지만, calibration 이 좋지 않음[1]

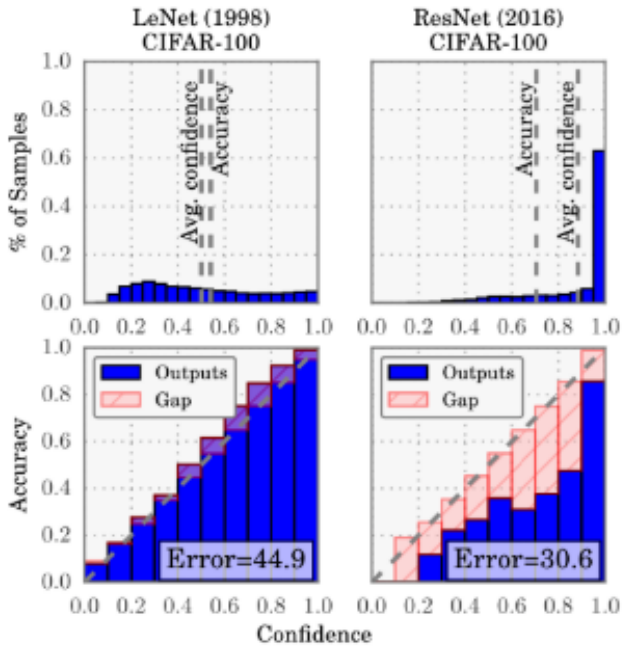
- Calibration 이란 모형의 출력값이 실제 confidence를 반영하도록 만드는 것을 말함
  → 예를 들어, X 의 Y1 에 대한 모형의 출력이 0.8이 나왔을 때, 80 % 확률로 Y1 일 것이라는 의미를 갖도록 만드는 것

- 모형의 출력값이 실제 confidence를 반영한다면 confidence와 accuracy가 일치해야 함
  → 모델이 80%의 confidence로 예측한 sample들의 경우 80%의 acc가 나와야 함

- 그러나 오늘날 neural net은 over confident한 문제를 갖고 있음

- 오른쪽 그림은 LeNet-5와 ResNet의 confidence-accuracy chart임

- Model의 capacity가 낮은 LeNet-5와 달리 ResNet의 경우 over confident가 발생하고 있음



1 SIL Introduction & Related work

Background

- 따라서 model calibration 향상을 위한 노력은 다양하게 존재해왔으며 대표적으로 label smoothing, mix up 등이 있음

- Label smoothing은 label을 0과 1로 두어 학습하는 것이 아닌 smooth하게 부여하여 과도하게 학습하는 것을 막음으로써 해결하고자 함[2]
  → regularization에 도움을 주면서, model generalization 과 calibration 에 도움이 됨

- Mix up은 두 개의 random sample에서 linear interpolation을 적용하여 학습하는 방법론[3]
  → 새로 형성된 label의 경우 label smoothing과 유사하게 1로 할당되지 않아 calibration 에 도움

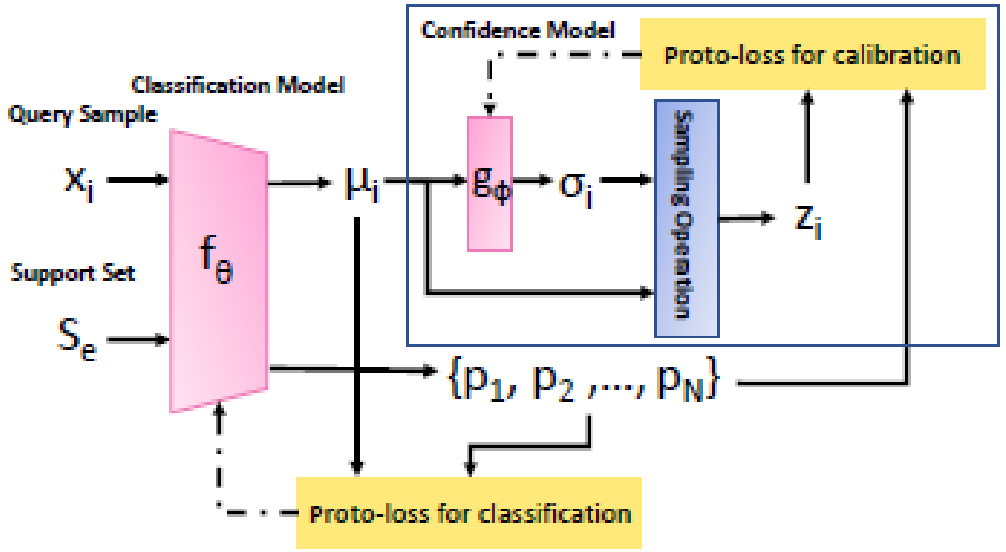
- 그러나 위 방법론 모두 objective function이 confidence estimation을 목표로 삼지 않음

$$\tilde{x} = \lambda x_i + (1-\lambda)x_j, \quad \text{where } x_i, x_j \text{ are raw input vectors}$$
$$\tilde{y} = \lambda y_i + (1-\lambda)y_j, \quad \text{where } y_i, y_j \text{ are one-hot label encodings}$$

- Confidence scoring을 직접적으로 학습에 활용하는 방법으로는 Temperature scaling 등이 있음
  → but, classification과 calibration 두가지 task로 데이터를 분할해야하기 때문에 data의 분배량에 따라 task의 성능이 trade-off 성질을 띔

2 Proposed Approach

전체 concept



1) 먼저 classification을 진행하며 이 때 개별 sample들을 각 class 중심점으로 mapping 시켜 분류에 용이하도록 학습을 진행함
   → support set으로 class의 중심점을 구하고, query sample이 중심점과의 distance가 최소화 되도록 mapping

2) 그 후 classification에서 오분류된 data를 이용하여 confidence model을 학습함
   → confidence model의 역할은 ground-truth가 없는 test data에 대한 confidence를 추정하는 것

2 Proposed Approach

Episodic training

- Batch 단위 samples를 학습하는 일반적인 training 방식과는 다르게 episodic training은 매 update마다 K-shot, N-way samples 를 사용함
- 매 episode마다 전체 class M에서 N개의 sampled class를 가져온 후, N개의 class들을 K개의 데이터를 지닌 support set과 query set으로 분리함
  → they are containing different examples from the same N classes

$$\mathcal{S}_e = \{(\mathbf{s}_j, y_j)\}_{j=1}^{N \times K} \qquad \mathcal{Q}_e = \{(\mathbf{x}_i, y_i)\}_{i=1}^{Q}$$

- Support set의 역할은 N개의 class들의 중심점(p_i)을 구하는 것이며 query set의 sample 들은 classification model을 거쳐서 mapping됨
  → embedding space로 mapping된 x_i들은 μ_i로 표시함
- N개의 class들의 중심점은 support sample들의 평균을 통해서 구해짐

$$\mathbf{p}_c = \frac{1}{|S_e^c|} \sum_{(\mathbf{s}_j, y_j) \in \mathcal{S}_e^c} f_\theta(\mathbf{s}_j)$$

- Episodic training의 loss는 다음과 같음

$$\mathcal{L}(\theta) = \mathop{\mathbb{E}}_{(\mathcal{S}_e, \mathcal{Q}_e)} - \sum_{i=1}^{Q_e} \log p(y_i | \mathbf{x}_i, \mathcal{S}_e; \theta)$$

2 SIL Proposed Approach

Prototypical Learning for Classification

- 먼저 prototypical loss는 embedding된 query sample μ_i과 class 중심점 p_i과의 거리를 softmax를 취하는 것으로 설정

$$p(y_i|\mathbf{x}_i, S_e; \theta) = \frac{\exp(-d(\mu_i, \mathbf{p}_{y_i}))}{\sum_k \exp(-d(\mu_i, \mathbf{p}_k))}$$

- 학습을 거듭하면서 embedding space 상에서 inter-class distance는 커지고, intra-class distance는 작아짐
  → 즉 같은 class는 모이고 다른 class는 서로 밀어내는 효과를 보임

- Inference 시에는 각 class의 중심점은 training samples를 이용하여 계산 후 mapping된 μ_i 와 가장 거리가 가까운 label로 예측을 진행함

$$\mathbf{p}_c^{test} = \frac{1}{|\mathcal{T}_c|} \sum_{(x_i, y_i) \in \mathcal{T}_c} f_\theta(x_i) \qquad y_t' = \arg\min_c \{d(\mu_t, \mathbf{p}_c^{test})\}_{c \in \mathcal{M}}$$

- 따라서 mapping된 μ_i가 ground-truth center와 거리가 멀 수록 오분류 될 가능성이 커짐, 즉 거리에 비례하여 모델의 성능이 결정됨

  → distance가 곧 model의 calibrated confidence를 표현한다고 볼 수 있음

2 SIL Proposed Approach

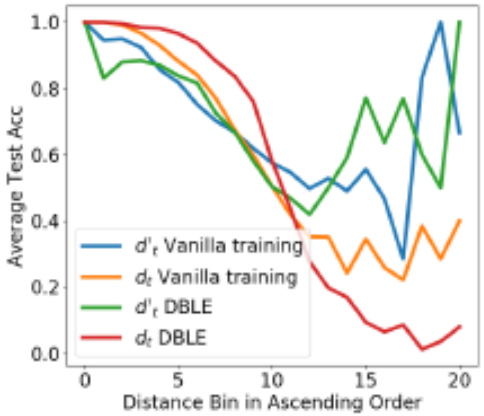
Calibrated confidence of DBLE

- 아래 그림의 x축은 개별 sample들의 class 중심점과의 거리를 뜻하며 y축은 그 때의 test acc들의 평균을 뜻함

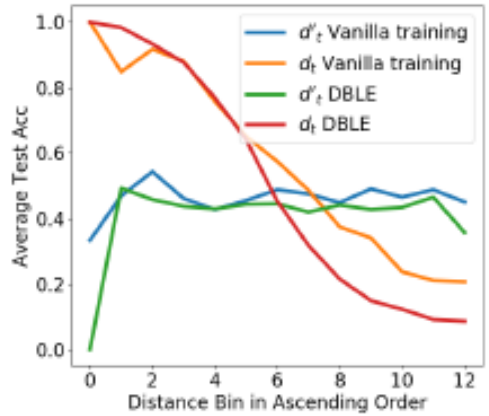
  → CIFAR-100에서 ground-truth class 중심점과의 거리가 5인 sample들의 test acc 평균은 약 0.9로 해석함

- d_t는 test sample x_i의 ground-truth 중심점과의 거리를 뜻하며 $\acute{d}_t$는 예측된 class 중심점과의 거리를 뜻함
- 도표에서 확인할 수 있듯이 ground-truth 중심점과의 거리가 멀어질 수록 모델의 성능이 낮아지며 예측된 class 중심점을 사용한 것도 그러함

  → 다만 예측된 class의 중심점을 사용할 경우 ground-truth를 사용한 것 만큼 정확하진 않음



(a) CIFAR-100



(b) Tiny-ImageNet

2 Proposed Approach

Confidence modeling by learning form errors

- 위 방법의 가장 큰 문제점은 distance를 측정할 때 test sample에 대하여 ground-truth label이 필요하다는 것
  → 그러나 우리가 실제로 풀어야하는 문제는 test sample은 label이 없는 경우임

- 따라서 논문의 저자들은 classification을 학습함과 동시에 confidence를 측정할 수 있는 모델을 동시에 학습함(joint training)
- Confidence model은 g_ϕ로 표현되며 저자들이 제안하는 방법론은 다음과 같음

  1) 먼저 오분류된 sample들의 mapping 값 μ_s와 σ_s를 parameter로 삼는 gaussian distribution에서 sample z_s를 추출함

  2) sample z_s가 올바르게 분류되도록 confidence model을 update함 (초기의 σ_s는 작은 값을 갖지만 update가 반복될 수록 σ_s는 커짐)

- 즉 오분류된 sample이면서 ground-truth class 중심점과 거리가 멀수록 σ_s는 큰 값을 갖게 됨

(a) Before updating ϕ (b) After updating ϕ

2 SIL Proposed Approach

Confidence modeling by learning form errors

- confidence model은 classification model에서 오분류된 sample만을 사용하여 학습함
  - 오분류된 sample만을 활용하는 이유는 neural net의 capacity가 커지면서 training sample은 웬만하면 다 적합이 가능함
  - 즉 ground-truth와의 distance가 작은 sample들이 다수가 되면서 confidence model이 소수의 오분류 sample을 학습하기가 어렵게 됨
  - if all data is used, training of g_ϕ would be dominated by the small distances of the correctly-classified samples which would make it harder for g_ϕ capture the larger distances for the minor mis-classified samples.
- confidence model g_ϕ는 오분류된 sample의 mapping값 μ_s를 받아서 σ_s를 출력하는 model임 $\sigma_s = g_\phi(\mu_s)$
- 앞서 말했듯, μ_s와 σ_s를 파라미터로 삼는 guassian distribution에서 sample z_s를 추출함 $\mathbf{z}_s \sim \mathcal{N}(\mu_s, \mathrm{diag}(\sigma_s \odot \sigma_s))$
- 마지막으로 sample z_s와 ground-truth class의 중심점을 이용하여 prototypical loss를 optimization

$$p(y_s|\mathbf{x}_s;\phi) = \frac{\exp(-d(\mathbf{z}_s, \mathbf{p}_{y_s}))}{\sum_k \exp(-d(\mathbf{z}_s, \mathbf{p}_k))}$$

- 최적화 과정에서 μ_s는 고정된 parameter기 때문에 z_s와 중심점 간의 거리가 멀수록 큰 값의 σ_s 뱉도록 g_ϕ가 update 됨

2 SIL Proposed Approach

Confidence modeling by learning form errors

- 앞서 말한 방식으로 학습이 완료된 g_ϕ는 inference 시 다음과 같은 식에 의해서 confidence를 내뱉음

$$\hat{p}(y'_t|\mathbf{x}_t;\phi) = \frac{1}{U}\sum_{u=1}^{U}\frac{\exp(-d(\mathbf{z}_t^u, \mathbf{p}_{y'_t}))}{\sum_k \exp(-d(\mathbf{z}_t^u, \mathbf{p}_k))}$$

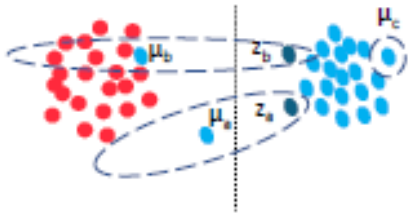
- 먼저 각각의 test sample x_t에 대해서 mappin값 μ_t와 예측값 $\hat{y}_t$를 구하고 confidence model을 통해서 σ_t를 구함
- 이후 $\mathcal{N}(\mu_t, \mathrm{diag}(\sigma_t \odot \sigma_t))$ 를 따르는 z_t를 U번 sampling하여 각각 prototypical loss의 평균으로 test sample의 confidence를 측정함
- 만약 σ_t가 큰 값을 갖는다면 U번의 prototypical loss의 차이가 클 것이며, 이를 평균하면 결과적으로 모든 class에 대해서 비슷한 값을 갖음
  → 즉 오분류된 sample일 수록 특정 class에 대한 confidence가 낮게 측정됨

z_t별 softmax output

0.8	0.05	0.05	0.1
0.1	0.1	0.05	0.75
0.2	0.2	0.3	0.3

Softmax output의 평균

0.36	0.11	0.13	0.38



z_t별 softmax output

0.8	0.05	0.05	0.1
0.8	0.1	0.05	0.05
0.8	0.05	0.1	0.05

Softmax output의 평균

0.8	0.06	0.06	0.06

3 Experiments

Experiments setup

- Baselines : vanilla training, MC-Dropout, Temperature Scaling, Mixup, Label smoothing, TrustScore
- Datasets and network architectures
  - MLP on MNIST
  - VGG-11on CIFAR-10
  - ResNet-50 on CIFAR-100
  - ResNet-50 on Tiny-ImageNet
- Evaluation metrics
  - ECE와 negative log-likelihood(=cross entropty)를 사용함
  - ECE란 accuracy와 confidence의 차이의 기대값을 뜻함

$$\text{ECE} = \sum_{l=1}^{L} \frac{|\mathcal{I}_l|}{|\mathcal{D}_{test}|} \left| \sum_{x_t \in I_l} p(y_t'|x_t) - \sum_{x_t \in I_l} \mathbf{1}(y_t' = y_t) \right|$$

$$\text{NLL} = \frac{1}{|\mathcal{D}_{test}|} \sum_{(x_t, y_t) \in \mathcal{D}_{test}} -\log(p(y_t|x_t))$$

3 SIL Experiments

Results

Method	MNIST-MLP			CIFAR10-VGG11		
	Accuracy%	ECE%	NLL	Accuracy%	ECE%	NLL
Vanilla Training	98.32	1.73	0.29	90.48	6.3	0.43
MC-Dropout	98.32	1.71	0.34	90.48	3.9	0.47
Temperature Scaling	95.14	1.32	0.17	89.83	3.1	0.33
Label Smoothing	98.77	1.68	0.30	90.71	2.7	0.38
Mixup	**98.83**	1.74	0.24	90.59	3.3	0.37
TrustScore	98.32	2.14	0.26	90.48	5.3	0.40
DBLE	98.69	**0.97**	**0.12**	**90.92**	**1.5**	**0.29**
Deep Ensemble-4 networks	99.36	0.99	0.08	92.4	1.8	0.26

Method	CIFAR100-ResNet50			Tiny-ImageNet-ResNet50		
	Accuracy%	ECE%	NLL	Accuracy%	ECE%	NLL
Vanilla Training	71.57	19.1	1.58	46.71	25.2	2.95
MC-Dropout	71.57	9.7	1.48	46.72	17.4	3.17
Temperature Scaling	69.84	2.5	1.23	45.03	4.8	2.59
Label Smoothing	**71.92**	3.3	1.39	**47.19**	5.6	2.93
Mixup	71.85	2.9	1.44	46.89	6.8	2.66
TrustScore	71.57	10.9	1.43	46.71	19.2	2.75
DBLE	71.03	**1.1**	**1.09**	46.45	**3.6**	**2.38**
Deep Ensemble-4 networks	73.58	1.3	0.82	51.28	2.4	1.81

3 Experiments

Results

- Cofidence model을 training 할 때 모든 sample을 사용하는 것 보다 오분류된 sample만 사용하는 것의 성능이 월등히 좋음

Method	CIFAR100		Tiny-ImageNet	
	ECE%	NLL	ECE%	NLL
Vanilla Training	19.1	1.58	25.2	2.95
Learning with errors in vanilla training	18.3	1.43	20.9	2.61
DBLE with calibration learning using all samples	18.9	1.54	24.8	2.87
DBLE	**1.1**	**1.09**	**3.6**	**2.38**

Reference

[1] Guo, C., Pleiss, G., Sun, Y., & Weinberger, K. Q. (2017). On calibration of modern neural networks. arXiv preprint arXiv:1706.04599.

[1-1] https://3months.tistory.com/490

[1-2] https://3months.tistory.com/465?category=756964

[2] Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (2016). Rethinking the inception architecture for computer vision. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 2818-2826).

[3] Zhang, H., Cisse, M., Dauphin, Y. N., & Lopez-Paz, D. (2017). mixup: Beyond empirical risk minimization. arXiv preprint arXiv:1710.09412.

Q&A