

Attention을 이용한 데이터셋 응축

김동훈, 배성호

경희대학교 컴퓨터공학과

{dhkim2810, shbae}@khu.ac.kr

요약

Deep Neural Networks(DNN)이 발전함에 따라 학습시 요구되는 데이터셋의 양도 커져간다. 이는 직접적인 비용 증가로 이어지기 때문에 효율적인 데이터셋 구성의 필요성은 나날이 중요해지고 있다. 본 연구는 효율적인 데이터셋을 구축하기 위해 데이터셋의 중요한 정보만을 필터링하여 응축하는 데이터셋 응축 방법을 제안한다. 해당 방법으로 응축한 데이터셋으로 기존 방법보다 최대 10% 향상된 58.5%의 테스트 정확도를 보인다. 본 연구를 통해 데이터셋 응축을 위한 핵심 특징의 중요성을 강조하고 큰 데이터셋에 대한 제안 방법의 잠재력을 확인한다.

1. 서론

Deep Neural Networks(DNN)은 딥러닝의 눈부신 발전과 방대한 데이터셋에 힘입어 자연어 처리, 컴퓨터 비전, 음성 인식 등 다양한 분야에서 특출난 성능을 보여주고 있다[1-3]. 그러나 DNN을 학습하기 위한 데이터셋이 방대해짐에 따라 저장하거나 처리하기 부담스러워졌고 이를 처리하기 위한 기반 시설의 요구사항도 높아지고 있다[4]. 본 연구 학습 데이터셋을 아주 작은 데이터셋으로 응축시켜 DNN 모델의 성능 저하를 최소화하면서 데이터셋의 효율성을 크게 높이고자 제안되었다.

초기 연구는 데이터셋을 클러스터링하여 최적의 샘플들을 이루어진 코에세트(coreset)를 구축하려는 방법이 제안되었다[5]. 그러나 클러스터 중심에 따라 결과가 크게 바뀌고 충분한 정보를 포함하고 있는 샘플들이 없다면 좋은 성능을 내작기 어려웠다. 최근에는 학습 데이터셋을 학습하여 샘플을 합성하려는 연구가 진행되었다[4]. 이와 같은 방법은 학습 데이터셋의 분포로부터 샘플링하는 것이 아닌 데이터 자체의 특징을 학습하고 이를 토대로 샘플을 생성하는 것이기 때문에 위의 문제를 해결하면서 DNN 모델에 최적화된 데이터를 생성할 수 있다[6].

[6]의 방법의 경우 MNIST[7]와 같이 정보량이 적은 데이터셋을 합성한 경우 높은 데이터셋 효율을 보였다. 그러나 CIFAR10과 복잡도가 높은 데이터셋은 코어세트보다 우수하지만 데이터 응축 효율이 크게 증가하지 않았다. 이는 불필요한 배경 정보가

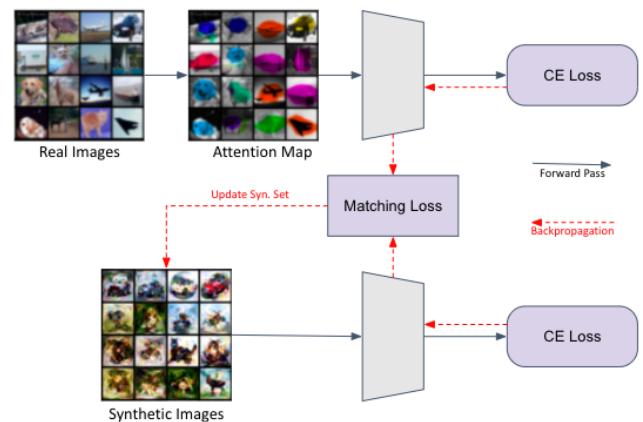


그림 1. 제안 방법은 실제 이미지의 중요한 부분만을 추출하여 모델 학습 시 cross entropy loss를 계산한다. 합성된 이미지로 학습시 발생한 가중치에 대해 마스킹된 이미지로부터 발생한 가중치와 최대한 같아지도록 이미지를 합성한다.

모델에 같이 학습되어 전체 데이터의 복잡도를 높였기 때문이다[8]. 따라서 본 연구는 이런 한계를 극복하기 위해 attention을 사용해 핵심적인 정보를 마스킹하고, 이를 활용한 데이터셋 합성을 제안한다.

2. 관련 연구

2.1 데이터셋 합성 기법

큰 데이터셋으로부터 작은 학습 데이터를 합성하는 방법은 데이터셋 증류(Dataset Distillation, DD)[6]에서 처음으로 제안되었다. 데이터셋 증류는 딥러닝 모델을 고정시킨 채 실제 샘플에 대한 합성 데이터셋의 손실을 최소화하는 방식으로 합성 데이터셋을 구축한다. 합성된 샘플은 큰 데이터셋

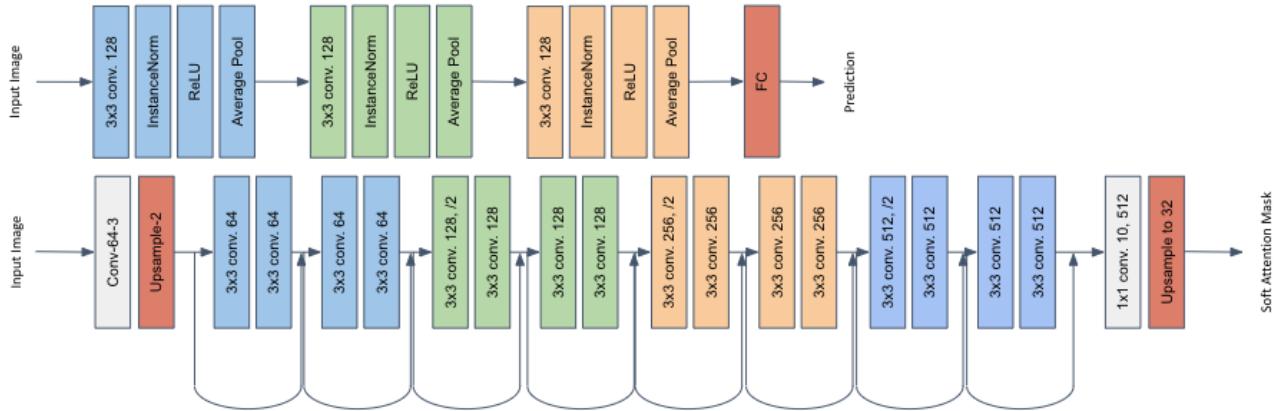


그림 2. 합성된 데이터셋으로 학습한 Convolutional Neural Network(CNN)의 구조 (위).
CIFAR-10 데이터셋의 attention 지도를 얻기 위한 Fully Convolutional Networks(FCN)의 구조 (아래)

의 분포에서 얻은 것은 아니지만 모델 학습에 필요한 중요한 특징들을 추출할 수 있다. 데이터셋 응축(Dataset Condensation, DC)[4]은 데이터셋 종류와 유사하지만 학습 손실을 사용해 데이터를 합성하는 대신 실제 샘플로 학습할 때 발생하는 기울기(gradients)를 사용한다. 실제 샘플과 합성 샘플로 모델을 학습할 때 발생하는 모델의 기울기가 일치하도록 데이터셋을 합성한다. 나아가 실제 샘플과 합성 샘플에 동일한 데이터 증강 기법을 적용하여 데이터셋 응축 기법보다 더 많은 특징을 학습할 수 있도록 개선한 연구도 제안되었다[9]. 이 때, 미분 가능한 데이터 증강 기법[10]을 사용하여 입력 이미지에 대한 기울기도 계산할 수 있도록 했다. 그러나 앞선 연구들은 이미지의 백그라운드 정보를 같이 사용했기 때문에 중요도가 떨어지는 정보들을 함께 사용한다. 본 연구는 전면(foreground) 정보에 집중하여 데이터를 학습하고 작은 데이터셋을 합성한다.

2.2 Attention

인간의 인지 과정에서 attention은 중요한 역할을 수행한다[11]. 즉, 인간은 들어오는 시각 정보를 통으로 해석하지 않는다. 대신 여러번으로 나누어 중요한 부분에 집중하여 시각 정보를 받아들이며, 이러한 메커니즘을 attention 메커니즘이라고 한다. 이런 인지적 특징을 Convolutional Neural Networks(CNN)에 적용한 여러 연구가 제안되었다. Residual Attention Network[12]는 인코더-디코더로 이루어진 attention 모듈을 사용해 CNN의 특징 맵에 attention mask를 써워 이미지 인식 성능을 개선했다. GAIN(Guided Attention Inference Network)[13]은 이미지로부터 얻은 attention 지도와 약한 레이블을

사용해 더 정확한 attention 지도를 추출하고, 이를 이미지 분석에 사용한다. 본 연구는 GAIN에서 사용한 추가적인 약한 레이블 없이 attention 지도를 추출하고 이를 데이터셋 합성에 사용한다.

3. 제안 방법 및 실험

3.1 제안 방법

샘플 이미지 $x \in X \subset \mathbb{R}^d$ 와 레이블 $y \in \{0, \dots, C-1\}$ 로 이루어진 크기 $|T|$ 의 큰 데이터셋 $T = \{(\mathbf{x}_i, y_i)\}_{i=1}^{|T|}$ 이 있다고 하자. X 는 d 차원의 입력 공간이고 C 는 클래스 개수이다. 이 때, 이전에 보지 못한 이미지에 대해 올바르게 예측하는 미분 가능한 함수 ϕ 와 최적해 θ 를 학습하는 것이 목표이다. 여기서 T 가 크기 때문에 $|S| \ll |T|$ 인 학습 데이터셋 $S = \{(\mathbf{s}_i, y_i)\}_{i=1}^{|S|}$ 을 효율적으로 구축하는 것을 목표로 한다. 합성된 이미지 $s \in \mathbb{R}^d$ 와 레이블 $y \in \mathbb{Y}$ 은 T 와 유사한 학습 정확도를 갖도록 합성된다.

일반적으로, 데이터셋 응축은 T 로 학습한 모델의 최적해 θ^T 와 S 로 학습한 모델의 최적해 θ^S 가 가장 유사하도록 S 를 학습한다. 나아가 식[1]과 같이 데이터셋 S 를 합성할 때 T 와 유사한 경로로 학습될 수 있도록 한다. 이 때, loss 함수 $l(\cdot, \cdot)$, $L^S(\theta) = \frac{1}{|S|} \sum_{(\mathbf{s}, y) \in S} l(\phi_\theta(s), y)$

$$L^T(\theta) = \frac{1}{|T|} \sum_{(\mathbf{x}, y) \in T} l(\phi_\theta(x), y) \text{이다.}$$

$$\theta_{t+1}^S(S) = \text{opt} - \text{alg}_\theta(L^S(\theta_t^S), \xi^S) \quad \text{and} \\ \theta_{t+1}^T = \text{opt} - \text{alg}_\theta(L^T(\theta_t^T), \xi^T) \quad (1)$$

표 1. CIFAR10 데이터셋을 응축한 Baseline 모델들의 정확도를 비교한 결과.
IPC는 각 클래스 당 합성한 샘플의 숫자이다 (Images Per Sample, IPC).

IPC	DD[6]	DC[4]	Ours	전체 데이터
1	-	24.2±0.9	24.5±0.6	83.1±0.2
10	36.8±1.2	39.1±1.2	40.1±1.6	

표 2. CIFAR10 데이터셋을 AlexNet을 이용해 응축하고 여러 AlexNet 모델들에 학습시킨 결과이다. DSA와 마스킹을 함께 사용하는 것이 가장 우수하다

IPC	DC[4]	DSA[9]	DSA + Ours	전체 데이터
1	24.5±0.6	25.3±1.0	25.4±0.8	83.1±0.2
10	39.1±1.2	44.5±1.1	44.9±1.0	
50	47.4±0.8	56.8±0.7	58.5±0.6	

3.2 Attention 마스킹

이미지 $x \in \mathbb{R}^{C \times W \times H}$ 를 입력으로 받고 $y \in \mathbb{R}^{N \times W \times H}$ 를 출력하는 모델 ϕ 가 있다. 이 때, C 는 이미지의 채널(예, RGB 이미지의 경우 $C=3$) 수이고, N 은 클래스 개수이다. 모델은 식[2]를 최소화하는 최적해 θ 를 가진다.

$$L(\theta) = BCELoss(Avg(\phi_\theta(x)), y) \quad (2)$$

따라서, 입력 이미지 x 에 대한 attention 지도 $A_x \in \mathbb{R}^{W \times H}$ 은 식[3]으로 구할 수 있다. 이때, 활성 함수로 sigmoid 함수를 사용한다.

$$A_x = \sigma(\log \sum_{j=1}^C \exp(\phi_\theta(x))) \quad (3)$$

A_x 는 ϕ_θ 가 중요하다고 여기는 정보들의 위치가 존재하기 때문에 식[4]를 사용해 마스킹된 이미지 x_m 을 데이터셋 응축에 사용한다.

$$x_m = x \odot A_x, \quad x_m \in \mathbb{R}^{C \times W \times H} \quad (4)$$

3.3 실험

CIFAR 10 데이터셋 응축 실험은 AlexNet[1]와 간단한 convolutional neural network(CNN)을 사용했다. CNN의 구조는 [그림 2.1]과 같다. CIFAR-10 데이터셋의 attention 지도를 그리기 위해 Fully Convolutional Networks(FCN)을 사용했고 구조는 [그림 2.2]와 같다.

기존 데이터셋 응축 알고리즘[4][6]과의 성능 비교를 위한 실험을 진행했고 결과는 [표 1]과 같다. 마스킹을 사용했을 때 성능 개선을 확인할 수 있다. 불필요한 정보를 제거하여 학습한 데이터셋이기 때문에 보다 보다 효율적인 데이터 합성이 가능하다. [그림 3]에서 확인할 수 있듯이 마스킹 비율이 늘어남에 따라 불필요한 정보가 사라져 정확도 향상과 오류 편차의 감소로 이어지는 것을 알 수 있다. 이미지의 크기가 큰 데이터셋(예, ImageNet)은 배경 정보가 많기 때문에 본 연구에서 제안하는 방법이 효과적일 것으로 예상된다.

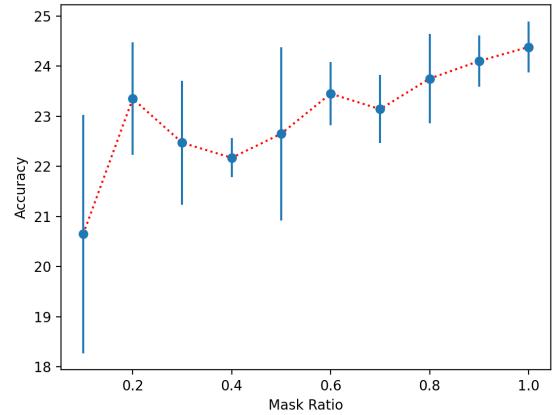


그림 3. 마스킹 비율을 달리하며 CIFAR10 데이터셋을 AlexNet으로 클래스 당 1장으로 합성한 결과이다. 마스킹 비율이 높아지면서 정확도 향상과 오류 편차 감소가 나타나는 것을 볼 수 있다.



그림 4. AlexNet으로 CIFAR10 데이터셋을 각 클래스 당 10장으로 노이즈부터 합성한 결과이다. 각 클래스대부분을 구별할 수 있고 위 데이터셋으로 AlexNet을 학습했을 때 전체 데이터셋 대비 48%의 정확도를 보인다.

Attention 지도로 마스킹한 데이터셋은 데이터 증강 기법과 함께 사용하면 많게는 10%의 정확도 향상을 보여준다 [표 2]. 모든 실험은 AlexNet에서 수행되었고, 합성 데이터셋은 1000번동안 합성되었다. 필터링된 중요한 정보들만을 사용해 데이터 증강을 적용하기 때문에 다양하게 변경되는 배경에 대해서 학습하지 않고 핵심 특징에 집중하여 데이터를 합성한다.

4. 결론

본 연구에서는 이미지의 핵심 부분을 추출하는 attention 지도로 입력 신호를 필터링해 데이터셋을 응축한다. 기존 방법보다 핵심 특징을 추출하고 데이터셋 합성에 사용하는 것이 더 높은 효율을 보이는 것을 확인했다. 배경 정보가 많지 않은 CIFAR10에서 유의미한 정확도 향상을 얻었기 때문에 배경 정보가 많은 ImageNet 등 큰 데이터셋에 활용할 가치가 충분해보인다. 향후 연구에서는 attention에 대한 가중치를 실제 이미지 가중치와 함께 사용하는 방법으로 확장할 수 있다.

감사의 글

“본 연구는 과학기술정보통신부 및 정보통신기획평가원의 SW중심대학 사업의 연구결과로 수행되었음” (2017-0-00093)

참고문헌

- [1] Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E Hinton. "Imagenet classification with deep neural convolutional neural networks." *Advances in neural information processing systems* 25 (2012): 1097-1105.
- [2] Devlin, Jacob, et al. "Bert: Pre-training of deep bidirectional transformers for language understanding." arXiv preprint arXiv:1810.04805(2018).
- [3] Dosovitskiy, Alexey, et al. "An image is worth 16x16 words: Transformers for image recognition at scale." arXiv preprint arXiv:2010.11929 (2020).
- [4] Zhao, Bo, Konda Reddy Mopuri, and Hakan Bilen. "Dataset condensation with gradient matching." arXiv preprint arXiv:2006.05929 (2020).
- [5] Feldman, Dan, Melanie Schmidt, and Christian Sohler. "Turing big data into tiny data: Constant-size coresets for k-means, pca and projective clustering." *SIAM Journal on Computing* 49.3 (2020): 601-657.
- [6] Tongzhou Wang, Jun-Yan Zhu, Antonio Torralba, and Alexei A Efros. "Dataset distillation." arXiv preprint arXiv:1811.10959 (2018).
- [7] LeCun, Yann, et al. "Gradient-based learning applied to document recognition." *Proceedings of the IEEE* 86.11 (1998): 2278-2324.
- [8] Tian, Maoqing, et al. "Eliminating background-bias for robust person re-identification." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018

- [9] Zhao, Bo, and Hakan Bilen. "Dataset Condensation with Differentiable Siamese Augmentation." arxiv preprint arXiv:2102.08259 (2021).
- [10] Zhao, Shengyu, et al. "Differentiable augmentation for data-efficient gan training." arXiv preprint arXiv:2006.1-738 (2020).
- [11] Rensink, Ronald A. "The dynamic representation of scenes." *Visual cognition* 7.1-3 (2000): 17-42.
- [12] Wang, Fei, et al. "Residual attention networks for image classification." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017.
- [13] Li, Kunpeng, et al. "Tell me where to look: Guided attention inference network." *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018.