Batch Distribution Matching를 이용한 데이터셋 응축

Dataset Condensation using Batch Distribution Matching

요 약

데이터셋 응축은 거대한 학습 데이터셋을 작은 데이터셋으로 응축해 모델 학습의 부담을 줄이고자 제안되었다. 원본 데이터셋을 학습할 때 발생하는 가중치 혹은 원본 데이터셋의 분포를 학습하는 연구들이 제안되었지만 응축한 데이터셋이 유사한 특징을 보이거나 학습에 필요한 특징을 충분히 합성하지 못하는 문제를 보인다. 본 연구에서는 학습 샘플의 특징을 기반으로 배치를 구성해 데이터셋 응축을 수행한다. 유사한 특징을 갖는 샘플들로 배치를 구성해 응축 데이터셋이 원본 데이터의 핵심 특징을 학습할 수 있다. 제안한 방법을 이용해 CIFAR10 데이터셋을 부류 당 1장, 10장, 50장으로 응축한 결과를 기울기만을 사용한 모델과 비교 분석한다. 응축 데이터셋을 여러 모델에 적용해 데이터셋으로서의 효용을 분석한다. 본 연구에서 제안한 Batch Distribution Matching이 기존 제안 방법들과 결합되어 응축 데이터셋의 정확도를 향상시킬 것으로 기대한다.

1. 서 론

Deep Learning Networks(DNN)은 다양한 분야에서 뛰어난 성능을 보여주지만 이를 학습하기 위한 데이터셋은 방대해지고 용량 또한 증가하고 있다[1]. 거대한 데이터셋을 저장하고 사용하기 위해 요구되는 컴퓨팅 파워 또한 높아짐에 따라 적은 자원으로 효율적인 학습이 가능한 데이터셋에 대한 필요성이 높아지고 있다.

학습 데이터 효율성을 높이기 위해 데이터셋의 핵심 샘플을 사용하여 작은 학습 데이터셋을 구성하는 코어세트 구축 방법들이 제안되었다[2]. 그러나 코어세트 구축 방법은 주어진 샘플에 한정된 데이터셋만 구성할 수 있었다. 이를 해결하기 위해 학습에 최적화된 샘플을 합성하는 방법이 제안되었다[3]. 데이터셋을 합성하는 방법은 학습 데이터는 주어진 데이터셋에 국한되지 않은 샘플을 합성할 수 있어 DNN 모델에 최적화된 데이터를 생성할 수 있다.

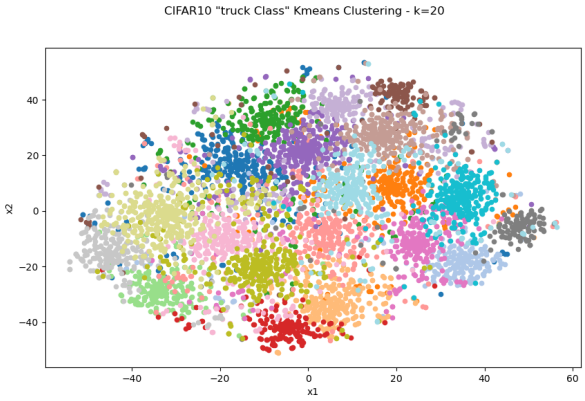
데이터셋 합성을 이용한 데이터셋 응축 방법은 원본 데이터셋으로부터 발생하는 가중치를 학습하거나[3-6] 원본 데이터셋의 분포를 학습한다[7-10]. 가중치를 학습한 데이터셋 응축은 유사한 이미지를 합성하는 반면 분포를 학습한 데이터셋 응축은 학습에 필요한 충분한 특징을 합성하지 못한다. 본 연구는 이런 문제를 해결하고자 학습에 사용되는 배치를 유사한 핵심 특징을 갖는 샘플들로 구성한다. 이를 통해 응축 샘플이 핵심 특징을 충분히 학습하여 합성할 수 있도록 한다.

2. 관 련 연 구

2. 1. 데이터셋 응축

데이터셋을 합성하는 방법은 데이터셋 증류(Dataset Distillation, DD)[3]에서 처음으로 제안되었다. 데이터셋 응축(Dataset Condensation, DC)[4]은 합성 샘플이 원본 이미지와 같은 기울기를 갖도록 학습한다. DSA(Differentiable Siamese Augmentation)는 원본 샘플과 합성 샘플에 동일한 데이터 증강 기법을 적용하여 DC를 개선하였다[5]. 이와 유사하게 Kim et al.(2022)[6]는 하나의 응축 샘플에 서로 다른 데이터셋 증강 기법을 적용한 패치를 만들어 많은 샘플로 데이터셋 응축을 수행하는 효과를 보였다.

DM(Distribution Matching)[7]은 원본 샘플과 합성 샘플을 임베딩하여 같은 데이터 공간에서 두 분포를 최소화하도록 학습해 적은 연산으로 DC와 유사한 결과를 보여줬다. 이와 유사하게 Lee et al.(2022)[8]는 code와 decoder를 이용해 응축 샘플의 다양성을 늘리며 DC에서 사용하는 이중 최적화 문제보다 적은 비용으로 데이터셋을 응축했다. CAFE[9]는 원본 샘플의 분포를 유지할 수 있도록 원본 샘플의 특징맵을 학습하여 데이터셋을 응축해 뛰어난 결과를 보였다. 본 연구는 데이터셋 응축에 사용하는 배치를 유사한 특징을 갖는 샘플로 구성해 합성 샘플을 합성한다.



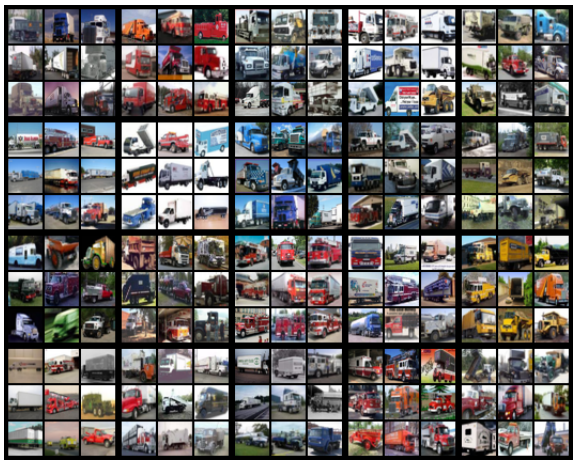


그림 1. CIFAR10 Truck 클래스에 대해 클러스터링을 수행한 결과이다. 이미지 특징에 따라 클러스터링한 결과이다. 클러스터링 결과와 이미지를 비교할 때 이미지 특징에 따라 샘플들이 묶인 것을 확인할 수 있다.(L=1, k=20)

3. 제 안 방 법

본 연구에서는 이미지의 특징을 기반으로 배치를 구성하는 Batch Feature Matching 방법을 제안한다. 원본 데이터셋 $\mathcal{T} = \{(x_i, y_i)\}|_{i=1}^{|\mathcal{T}|}$의 샘플 x_i의 특징을 추출하기 위해 ImageNet으로 모델 $\phi_\theta(\cdot)$을 사용한다. 클래스 c의 샘플들 $\mathcal{T}_c$의 레이어 L의 특징맵 $F_L^c = [f_{x_1}^c, f_{x_2}^c, \cdots, f_{x_n}^c] = \phi_\theta(\mathcal{T}_c)$을 추출한다. 이 때, $n = |\mathcal{T}_c|$이다.

$$\min_V \sum_{i=1}^{n} \min_{h=1,\cdots,k} \left(\frac{1}{2}\|x_i - V_h\|_2^2\right). \qquad (1)$$

클래스 c의 샘플을 클러스터링 하기 위해 F_L^c을 이용하여 클러스터 중심 $V = [V_1, V_2, \cdots, V_k]$를 식(1)을 이용하여 구한다. 이후 각 x_i에 대해 가장 가까운 V_h를 데이터셋에 추가하여 원본 데이터셋 $\mathcal{T} = \{(x_i, y_i, V_h)\}|_{i=1}^{|\mathcal{T}|}$을 구성한다. 데이터셋 응축은 같은 클러스터 중심의 샘플들로 학습 배치를 구성하고 Zhao et al.(2021)[4]과 같은 방법으로 데이터셋 응축을 수행한다.

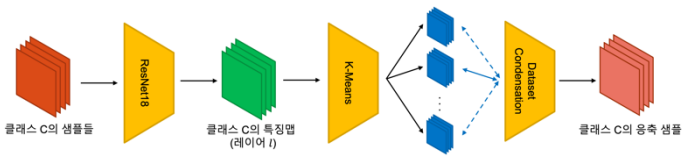


그림 2. 본 연구에서 제안하는 데이터셋 응축 방법이다. 이미지 샘플의 특징을 바탕으로 클러스터링을 수행하고 각 클러스터로만 배치를 구성해 데이터셋 응축을 수행한다.

4. 실 험 및 분 석

실험은 CIFAR10 데이터셋으로 수행한다. K-Means 클러스터링에 사용되는 특징 추출을 위해 ImageNet으로 사전학습한 ResNet18을 사용한다. 본 실험에서는 $L = 1, k = 20$을 사용한다. 클러스터링 실험 결과 분석한 결과 클래스 내 샘플들을 형태, 형상, 혹은 배경 색에 따라 분류한 것을 볼 수 있다. 이 외에도 '새' 클래스, '배' 클래스 등에서도 좋은 클러스터링 결과를 보이는 것을 확인할 수 있었다.

향후 실험으로는 먼저 데이터셋 응축은 클래스 당 1장, 10장, 그리고 50장으로 응축한 결과를 분석한다. 데이터셋 응축에 사용한 모델은 128 채널의 Conv, ReLU, BatchNorm으로 이루어진 블록을 3층으로 쌓은 Convolutional Neural Network(ConvNet)이다. 1000 epoch 동안 SGD optimizer를 사용하여 모델과 응축 샘플을 학습한다. Learning rate는 0.1(합성 샘플), 0.01(모델)이다. 제안 방법과 DC, DSA, DM을 비교한다. 또한 응축 데이터셋의 효용을 분석하기 위해 합성한 데이터를 이용해 MLP, ConvNet, ResNet18을 학습한다.

5. 기 대 효 과

본 연구에서 제안하는 Batch Feature Matching을 통해 효과적으로 원본 데이터셋의 특징을 학습하고 응축 데이터셋을 합성한다. 기존 방법보다 다양한 샘플을 만들어 정확도 향상을 이룬다. 또한 기존 제안된 다양한 방법들과 함께 사용되었을 때 성능 향상을 보인다. 원본 데이터셋과 유사한 학습 성능을 가진 응축 데이터셋은 적은 에너지와 비용으로 효과적인 학습이 가능하다. 따라서 평생학습(LifeLong Learning)과 NAS(Neural Architecture Search) 분야 등에서 응축 데이터셋을 활용할 수 있을 것이다.

참 고 문 헌

[1] Krizhevsky, Alex, Ilya Sutskever, and Geffrey E Hinton. "Imagenet classification with deep neural convolutional neural networks." Advances in neural information processing systems 25 (2012): 1097-1105.

[2] 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.

[3] Tongzhou Wang, Jun-Yan Zhu, Antonio Torralba, and Alexei A Efros. "Dataset distillation." arXiv preprint arXiv:1811.10959 (2018).

[4] Zhao, Bo, Konda Reddy Mopuri, and Hakan Bilen. "Dataset condensation with gradient matching." arXiv preprint arXiv:2006.05929 (2020).

[5] Zhao, Bo, and Hakan Bilen. "Dataset Condensation with Differentiable Siamese Augmentation." arxiv preprint arXiv:2102.08259 (2021).

[6] Kim, Jang-Hyun, Jinuk Kim, Seong Joon Oh, Sangdoo Yun, Hwanjun Song, Joonhyun Jeong, Jung-Woo Ha, and Hyun Oh Song. "Dataset Condensation via Efficient Synthetic-Data Parameterization." arXiv preprint arXiv:2205.14959 (2022).

[7] Zhao, Bo, Konda Reddy Mopuri, and Hakan Bilen. "Dataset condensation with gradient matching." arXiv preprint arXiv:2006.05929 (2020).

[8] Lee, H. B., Lee, D. B., & Hwang, S. J. (2022). Dataset Condensation with Latent Space Knowledge Factorization and Sharing. arXiv preprint arXiv:2208.10494.

[9] Wang, Kai, Bo Zhao, Xiangyu Peng, Zheng Zhu, Shuo Yang, Shuo Wang, Guan Huang, Hakan Bilen, Xinchao Wang, and Yang You. "Cafe: Learning to condense dataset by aligning features." In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 12196-12205. 2022.