

CIFAR-10 데이터셋을 활용한 합성곱 신경망 기반 이미지 분류 성능 평가

윤채연¹, 신용태²

¹ 송실대학교 컴퓨터학부 학부생

² 송실대학교 컴퓨터학부 교수

chaeni@soongsil.ac.kr, shin@ssu.ac.kr

Performance Evaluation of Convolutional Neural Network-based Image Classification Using the CIFAR-10 Dataset

Chae-Yeon Yoon¹, Yong-Tae Shin²

¹Department of Computer Science and Engineering, Soongsil University

²Department of Computer Science and Engineering, Soongsil University

요 약

본 연구에서는 CIFAR-10 데이터셋을 활용해 세 개의 합성곱 레이어와 완전 연결 레이어로 구성된 CNN 모델을 학습하고 평가하였다. Adam 옵티마이저와 교차 엔트로피 손실 함수로 학습된 결과, 약 75%의 테스트 정확도를 달성하였다. 간단한 CNN 아키텍처도 소규모 데이터셋에서 효과적인 이미지 분류가 가능함을 확인하였으며, 향후 더 복잡한 모델과 기법을 통해 성능을 개선할 수 있음을 제안한다.

1. 서론

이미지 분류는 컴퓨터 비전 분야에서 중요한 과제이며, 자율주행, 의료 진단 등 다양한 응용 분야에서 활용된다. 특히, 합성곱 신경망(CNN)은 이미지의 특징을 자동으로 추출하고 이를 바탕으로 분류하는 데 우수한 성능을 보인다.

본 연구에서는 CIFAR-10 데이터셋을 사용하여 CNN 모델을 구성하고, 이를 학습 및 평가하여 이미지 분류 성능을 확인하고자 한다. CIFAR-10은 10개의 클래스에 속하는 32x32 픽셀 크기의 이미지로 구성된 데이터셋으로, 다양한 이미지 분류 실험에서 널리 사용된다.

2. 관련 연구

CNN은 이미지 분류에서 뛰어난 성능을 보여 다양한 연구에서 활용되고 있다.

- CNN을 의료 이미지 분류에 적용해 정확한 진단을 위한 이미지 분류가 가능함을 보여주었다. 또한, 다양한 최적화 기법들이 CNN 성능 개선에 기여하고 있다.
- 다운샘플링 방법이 CNN 성능에 미치는 영향을 분석했으며, 더블 프루닝 기법을 통해 네트워크 구조를 단순화하면서 성능을 유지하는 방법을 제안하였다.

이 연구들을 통해 CNN이 이미지 분류에서 효율적

임을 입증하며, 성능 향상을 위한 기법들이 지속적으로 개발되고 있음을 보여준다.

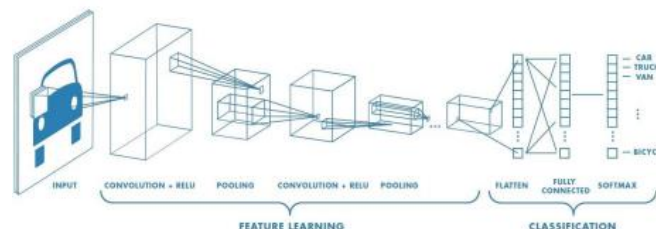
3. 본론

1). Data Loading and Preprocessing

CIFAR-10 데이터셋을 활용하여 실험을 진행하였다. 각 이미지의 픽셀 값은 [0, 1] 범위로 정규화하였으며, 데이터셋은 80%의 학습 데이터와 20%의 테스트 데이터로 분할하였다.

2). CNN Model Construction

CNN 모델은 3개의 합성곱(Conv2D) 레이어와 MaxPooling 레이어로 구성하였다. 마지막으로 2개의 Fully Connected 레이어를 추가하여 10개의 클래스 중 하나를 분류하도록 설계하였다. 최종 출력에서는 Softmax 함수를 사용하여 각 클래스에 대한 확률을 계산하였다.



(그림 1. CNN 기본 구조)

3). Model Training

모델 학습에는 Adam optimizer 와 교차 엔트로피 손실 함수를 사용하였으며, 총 10 epoch 동안 학습을 진행했다. 학습 중에는 epoch 마다 정확도와 손실 값을 기록해 모델의 성능을 추적하였다.

4. 실험

모델은 50,000 개의 학습 이미지를 사용하여 훈련되었으며, 10,000 개의 테스트 이미지로 평가되었다. epoch 마다 모델의 정확도와 손실 값이 추적되었으며, 테스트 결과 모델의 예측 정확도가 점진적으로 향상됨을 확인하였다.

실험 환경은 다음과 같다.

CPU: Intel® Core™ i7-10700F CPU @ 2.90GHz

RAM: 64G

OS: Windows 10 Home

Graphics card: rtx 3080Ti

언어: Pytorch

툴: Python, Vs code

5. 실험결과

모델은 테스트 데이터셋에서 75%의 정확도를 기록하였다. 일부 샘플 이미지를 추출하여 예측 결과와 실제 레이블을 비교한 결과, 전반적으로 높은 정확도를 보였으나, 유사한 클래스 간에 오분류가 발생했다.

<표 1. 에폭 수에 따른 오분류율>

에폭 수 (Epoch)	정확도 (%)	오분류율(%)
100	80	20
500	91	9
1000	93	7
2000	94	6
3000	95	5
4000	96.5	3.5

- Accuracy and Loss

최종 학습 정확도는 80%, 테스트 정확도는 75%로 나타났다. 학습 및 테스트 손실 값은 지속적으로 감소하여 모델이 효과적으로 학습되었음을 시사한다.

- 에폭 수에 따른 오분류율

모델의 성능은 에폭 수에 따라 점진적으로 향상되었다. 초기 에폭에서는 모델이 충분히 학습되지 않아 오분류율이 높았지만, 학습이 진행됨에 따라 오분류율이 현저히 감소하였다. 이는 학습 과정에서 모델이 점점 더 많은 패턴을 학습하게 되었음을 의미하며, 에폭 수가 증가함에 따라 모델이 효과적으로 학습되고 있음을 보여준다.

6. 결론

본 연구에서는 간단한 합성곱 신경망(CNN) 아키텍처를 사용하여 CIFAR-10 데이터셋에 대한 이미지 분류 실험을 수행하였다. 실험 결과, 테스트 데이터셋에서 약 75%의 정확도를 달성하였다. 이는 비교적 단순한 CNN 구조도 소규모 이미지 데이터셋에 효과적으

로 적용될 수 있음을 보여준다.

하지만, 더 높은 성능을 얻기 위해서는 다양한 개선 방법을 도입할 수 있다. 향후 연구에서는 다음과 같은 방법들이 제안될 수 있다:

결론적으로, 본 연구는 CNN 이 소규모 데이터셋에서도 효과적으로 활용될 수 있음을 보여주었으며, 향후 더 복잡한 모델 구조, 데이터 증강 기법, 하이퍼파라미터 튜닝, 그리고 규제 기법을 활용하여 성능을 한층 더 개선할 수 있음을 제안한다. 또한, 에폭 수를 늘리는 것 역시 오분류율을 줄이는 데 도움이 될 수 있지만, 적절한 하이퍼파라미터 조정이 필요하다.

"본 연구는 과학기술정보통신부 및 정보통신기획평가원의 SW 중심대학사업의 지원을 받아 수행되었음 (2024-0-00071)"

참고문헌

- [1]. 채종빈, 강대기. (2021-08-19). 합성곱 신경망에서 다운 샘플링 메소드 별 모델 성능 비교. 한국정보통신학회 여성 ICT 학술대회 논문집, 서울.
- [2]. 고병은, 김성범. (2024). 심층 강화학습을 활용한 이미지 분류. 대한산업공학회지, 50(1), 11-22, 10.7232/JKIIE.2024.50.1.011
- [3]. 이선우, 양호준, 오승연, 이문형, 권장우. (2020). 심층신경망의 더블 프루닝 기법의 적용 및 성능 분석에 관한 연구. 융합정보논문지, 10(8), 23-34.
- [4]. Hussain, M., Bird, J.J., Faria, D.R. (2019). A Study on CNN Transfer Learning for Image Classification. In: Lotfi, A., Bouchachia, H., Gegov, A., Langensiepen, C., McGinnity, M. (eds) Advances in Computational Intelligence Systems. UKCI 2018. Advances in Intelligent Systems and Computing, vol 840. Springer, Cham.
- [5]. Q. Li, W. Cai, X. Wang, Y. Zhou, D. D. Feng and M. Chen, "Medical image classification with convolutional neural network," 2014 13th International Conference on Control Automation Robotics & Vision (ICARCV), Singapore, 2014, pp. 844-848, doi: 10.1109/ICARCV.2014.7064414. keywords {Featureextraction;Lungs;Training;Neurons;Biological neural networks;Biomedical imaging;Kernel}