

정규화 및 교차검증 횟수 감소를 위한 무작위 풀링 연산 선택에 관한 연구

류서현
국방기술품질원

A Study on Random Selection of Pooling Operations for Regularization and Reduction of Cross Validation

Seo-Hyeon Ryu

Defense Agency for Technology and Quality

요약 본 논문에서는 컨볼루션 신경망 구조(Convolution Neural Network)에서 정규화 및 교차검증 횟수 감소를 위한 무작위 풀링 연산을 선택하는 방법에 대해 설명한다. 컨볼루션 신경망 구조에서 풀링 연산은 피쳐맵(Feature Map) 크기 감소 및 이동 불변(Shift Invariant)을 위해 사용된다. 기존의 풀링 방법은 각 풀링 계층에서 하나의 풀링 연산이 적용된다. 이러한 방법은 학습 간 신경망 구조의 변화가 없기 때문에, 학습 자료에 과도하게 맞추는 과 적합(Overfitting) 문제를 가지고 있다. 또한 최적의 풀링 연산 조합을 찾기 위해서는, 각 풀링 연산 조합에 대해 교차검증을 하여 최고의 성능을 내는 조합을 찾아야 한다. 이러한 문제를 해결하기 위해, 풀링 계층에 확률적인 개념을 도입한 무작위 풀링 연산 선택 방법을 제안한다. 제안한 방법은 풀링 계층에 하나의 풀링 연산을 적용하지 않는다. 학습기간 동안 각 풀링 영역에서 여러 풀링 연산 중 하나를 무작위로 선택한다. 그리고 시험 시에는 각 풀링 영역에서 사용된 풀링 연산의 평균을 적용한다. 이러한 방법은 풀링 영역에서 서로 다른 풀링 조합을 사용한 구조의 평균을 한 것으로 볼 수 있다. 따라서, 컨볼루션 신경망 구조가 학습데이터에 과도하게 맞추어지는 과적합 문제를 피할 수 있으며, 또한 각 풀링 계층에서 특정 풀링 연산을 선택할 필요가 없기 때문에 교차 검증 횟수를 감소시킬 수 있다. 실험을 통해, 제안한 방법은 정규화 성능을 향상시킬 뿐만 아니라 및 교차 검증 횟수를 줄일 수 있다는 것을 검증하였다.

Abstract In this paper, we propose a method for the random selection of pooling operations for the regularization and reduction of cross validation in convolutional neural networks. The pooling operation in convolutional neural networks is used to reduce the size of the feature map and for its shift invariant properties. In the existing pooling method, one pooling operation is applied in each pooling layer. Because this method fixes the convolution network, the network suffers from overfitting, which means that it excessively fits the models to the training samples. In addition, to find the best combination of pooling operations to maximize the performance, cross validation must be performed. To solve these problems, we introduce the probability concept into the pooling layers. The proposed method does not select one pooling operation in each pooling layer. Instead, we randomly select one pooling operation among multiple pooling operations in each pooling region during training, and for testing purposes, we use probabilistic weighting to produce the expected output. The proposed method can be seen as a technique in which many networks are approximately averaged using a different pooling operation in each pooling region. Therefore, this method avoids the overfitting problem, as well as reducing the amount of cross validation. The experimental results show that the proposed method can achieve better generalization performance and reduce the need for cross validation.

Keywords : Convolutional neural networks, Cross validation, Deep learning, Overfitting, Pooling

*Corresponding Author : Seo-Hyeon Ryu(Defense Agency for Technology and Quality)

Tel: +82-54-469-6527 email: rsh87@dtq.re.kr

Received January 4, 2018

Revised (1st February 26, 2018, 2nd March 29, 2018)

Accepted April 6, 2018

Published April 30, 2018

1. 서론

컨볼루션 신경망 구조(Convolutional Neural Networks)는 영상 분류[1], 얼굴인식[2], 보행자 탐지[3], 음성 인식[4]등 다양한 분야에서 사용되고 있으며, 기존의 다양한 알고리즘 대비 매우 우수한 성능을 보이고 있다. 하지만 컨볼루션 신경망 구조가 매우 좋은 성능을 보임에도 불구하고, 매우 많은 학습 가능한 파라미터를 가지고 있기 때문에 과적합(Overfitting) 문제를 겪고 있다. 과적합 문제를 피하기 위해 다양한 기술이 사용되고 있다. 기존에 기계학습 분야에서 많이 사용되어 온 L2 정규화, 학습 시 빠른 중단이 있다. 또한 최근에 컨볼루션 신경망 구조가 발전함에 따라 dropout[5], dropconnect[6], batch normalization[7], stochastic depth[8] 등이 개발되었다. 이 방법들의 기본 원리는 학습 간 무작위 요소가 포함되어 서로 다른 모델이 학습하는 효과를 얻으며, 시험 간 평균을 구함으로써, 모델 앙상블 효과를 가져 온다. 이러한 많은 정규화 기술에도 불구하고 컨볼루션 신경망 구조는 아직도 과적합 문제를 가지고 있다. 또한 컨볼루션 신경망 구조에는 교차검증을 통해 사용자가 직접 정해야 하는 파라미터들이 많이 있다. 이러한 파라미터들 중 본 논문에서는 풀링 연산에 초점을 두었다. 기존의 컨볼루션 신경망 구조에서는 각 풀링 계층에서 하나의 풀링 연산이 사용되었으며, 최고의 성능을 위한 풀링 연산 조합을 찾기 위해 교차 검증을 수행하여 풀링 연산 조합을 선택하였다. 만약 총 L개의 풀링 계층, P개의 풀링 연산이 있다면, 총 P^L 번의 교차 검증을 하여야한다.

본 논문에서는 과적합 문제 해결과 교차검증 횟수를 줄이기 위해, 무작위 풀링 연산 선택 방법을 제안한다. Dropout과 dropconnect처럼 학습 간 확률개념이 포함되면, 모델 앙상블 효과를 볼 수 있다는 것을 알 수 있다. 제안한 방법은 이러한 방법을 풀링 연산에도 적용하여, 학습 간 풀링 계층에 하나의 풀링 연산을 사용하는 것이 아니라, 각 풀링 영역에 무작위로 풀링 연산을 선택하여 학습을 하고, 시험 간 평균을 취하여 모델 앙상블 효과를 얻음으로써 과적합 문제를 해결한다. 또한 풀링 연산을 무작위로 선택함으로써, 최고의 풀링 연산 조합을 찾기 위한 교차검증을 피할 수 있다. 즉 무작위 풀링 연산 선택 방법을 사용하면 과적합 문제뿐만 아니라 교차검증 횟수를 줄일 수 있다.

본 논문은 다음과 같이 구성된다. 2장에서는 풀링 연

산 및 본 제안 방법의 동기가 된 dropout, dropconnect에 대해 설명한다. 3장에서는 무작위 풀링 연산 선택에 대해 설명한다. 4장에서는 제안한 방법을 검증하기 위한 실험방법 및 실험결과를 제시한다. 5장에서는 본 논문의 결론을 제시한다.

2. 관련연구

2.1 풀링 연산

컨볼루션 신경망 구조에서는 피쳐맵(feature map) 크기 감소 및 이동 불변(Shift invariant) 특성을 위해 풀링 계층이 포함된다. 풀링 계층에서 주로 사용되는 연산은 최대값 풀링(Max pooling)과 평균값 풀링(Average pooling)이 있다.

2.1.1 최대값 풀링(Max pooling)

최대값 풀링은 풀링 영역에서 가장 큰 값을 선택하는 것으로 아래의 수식과 같다.

$$x_{p,i,j}^{(l)} = \max_{(u,v) \in R_{i,j}} x_{p,u,v}^{(l-1)} \quad (1)$$

여기에서 $x_{p,i,j}^{(l)}$ 는 l번째 계층의 p번째 피쳐맵의 ij좌표의 피쳐를 의미하며, $R_{i,j}$ 는 풀링 영역을 의미한다.

2.1.2 평균값 풀링(Average pooling)

평균값 풀링은 풀링 영역의 평균을 취하는 것으로 아래의 수식과 같다.

$$x_{p,i,j}^{(l)} = \frac{1}{|R_{i,j}|} \sum_{(u,v) \in R_{i,j}} x_{p,u,v}^{(l-1)} \quad (2)$$

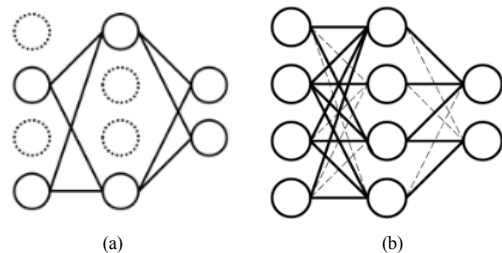


Fig. 1. Dropout and dropconnect
(a) dropout, (b) dropconnect

2.2 정규화

매우 많은 학습 가능한 파라미터를 가지는 컨볼루션 신경망 구조는 높은 성능을 보여준다. 하지만 학습 자료에 비해 매우 많은 파라미터 때문에 과적합 문제를 가지고 있다. 이를 피하기 위해 L2 정규화, dropout, dropconnect 등의 방법이 사용된다. 이 중 제안된 방법의 동기가 된 dropout 및 dropconnect에 대해 간략하게 설명한다.

2.2.1 Dropout

Dropout은 딥러닝에서 과적합 문제를 해결하기 위해 제안되었다. 핵심개념은 그림 1의 (a)와 같이 학습 동안 무작위로 노드를 0으로 만드는 것이다. 이는 이웃 노드 간 상호작용을 예방하며, 동시에 많은 컨볼루션 신경망 구조에 대해 학습함으로써 평균을 취하는 효과를 볼 수 있다. 시험 시에는 전체 구조가 사용되며 출력은 평균값이 된다.

2.2.2 Dropconnect

Dropconnect는 dropout을 일반화 시킨 것으로 학습 기간 동안 무작위로 노드를 0으로 만드는 것이 아닌 노드 간 연결을 0으로 만드는 것으로 그림 1의 (b)와 같다. 노드보다는 노드 간 연결이 훨씬 많기 때문에 더 많은 구조의 평균을 취하는 효과를 가져 온다.

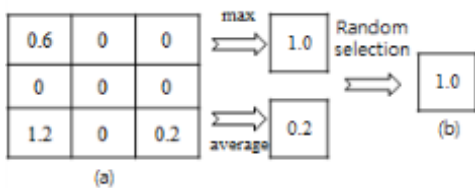


Fig. 2. Random selection of pooling operation during training
 (a) Activations, (b) Pooled response in the case of max pool

3. 무작위 풀링 연산 선택

컨볼루션 신경망 구조는 교차검증을 통해 사용자가 수동으로 정해야 하는 많은 파라미터를 가지고 있다. 파라미터로는 풀링 연산, 필터 크기, 풀링 크기, 풀링 이동 거리, 계층 수, 피쳐맵 수 등이 있다. 깊은 컨볼루션 신경

망 구조를 학습하는 것은 매우 많은 연산량이 필요하기 때문에 교차검증을 통해 최고의 파라미터를 찾는 것은 매우 오랜 시간이 걸린다. 또한 컨볼루션 신경망 구조는 매우 많은 학습 가능한 파라미터를 가지고 있기 때문에 과적합 문제를 가지고 있다. 따라서 이 두 가지의 문제를 해결하지 위해 무작위 풀링 연산 선택 방법을 제안한다. 기존의 컨볼루션 신경망 구조는 각 풀링 계층에 하나의 풀링 연산만 선택되었으며, 최고의 풀링 연산 조합을 찾기 위해 교차검증을 실시하였다. 만약 L개의 풀링 계층과 P개의 풀링 연산이 있다면, 총 P^L 번의 교차검증을 통해 최고의 풀링 연산 조합을 찾아야 한다. 만약 풀링 연산 선택이 없어진다면 교차검증 횟수를 줄일 수 있을 것이다. 그리고 dropout처럼 학습 간 확률개념이 도입하여 여러 구조의 평균을 취하는 효과를 얻으면 과적합 문제를 해결하는데 도움이 된다는 것을 알 수 있다. 위의 개념과 바탕으로 본 논문에서는 무작위 풀링 연산 선택을 제안하였다. 이것은 각 풀링 계층에 하나의 풀링 연산을 정하는 것이 아니라, 학습 간 각 풀링 영역에 무작위로 풀링 연산이 선택되도록 하는 것이다. 이로써 사용자가 각 풀링 계층에 하나의 풀링 연산을 정하지 않음으로써 교차검증 횟수를 줄일 수 있으며, 또한 여러 구조의 평균을 취하는 효과를 얻어 과적합 문제를 해결할 수 있다.

3.1 학습 시 무작위 풀링 연산 선택

무작위 풀링 연산 선택은 학습 동안 각 풀링 영역에서 무작위로 풀링 연산을 선택하게 된다. 학습 간 무작위 풀링 연산 선택 방법은 그림 2와 같다. 우선 풀링 연산을 선택하기 위해 다항분포(Multinomial distribution)를 이용하여 하나의 풀링 연산을 선택한 후, 선택된 풀링 연산을 수행하게 되며 아래의 수식과 같다.

$$r \sim \text{Multinomial}(p_1, p_2, \dots, p_K) \quad (3)$$

$$y_j = \text{pool}_r(x_i), \forall i \in R_j \quad (4)$$

여기서 p_k 는 k번째 풀링 연산이 선택될 확률, K는 풀링 연산 수, $\text{pool}_r()$ 은 r번째 풀링 연산, R_j 는 j번째 풀링 영역, x_i 는 풀링 영역 내에서 i번째 노드, y_j 는 풀링 연산 결과이다. 풀링 연산이 선택될 확률은 사용자가 정해야 하는 파라미터이지만, 교차검증 횟수를 감소시키기 위한 목적으로 동일한 확률로 설정하였다. 학습을 위한 역 전파는 선택된 풀링 연산에 따라 수행된다. 무작위 풀

링 연산 선택은 각 풀링 영역에 다른 풀링 연산을 사용한 다양한 구조의 평균을 취한 것으로 볼 수 있다. 예를 들어, P개의 풀링 연산, R개의 풀링 영역이 있다면, 총 P^R 개의 구조에 대해 평균을 취하는 것으로 볼 수 있다.

3.2 시험 시 추론

각 풀링 영역에 다른 풀링 연산을 사용하여 만들 수 있는 구조는 매우 많다. 따라서 이 모든 가능한 구조에 대해 추론을 하여 평균을 구하는 것은 실질적으로 불가능하다. 따라서 각 풀링 영역마다 평균값이 나올 수 있도록 확률적 가중치 합을 구하였다. 그림 3은 풀링 연산으로 최대값 풀링 및 평균값 풀링을 사용했을 경우 무작위 풀링 연산 선택의 추론 방법에 대해 나타낸다. 각 풀링 영역에서 학습 동안 사용된 모든 풀링 연산을 수행한다. 그리고 난 후 학습 시 사용된 확률을 이용하여 확률적 가중치 합을 구하며, 수식은 아래와 같다.

$$y_j = \sum_{k=1}^K p_k \text{pool}_k(x_i), \forall i \in R_j \quad (5)$$

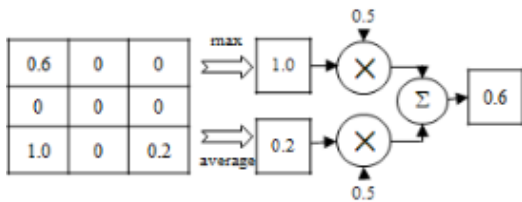


Fig. 3. Random selection of pooling operation at test time

Table 1. Network for the first experiment

layer name	output size (h x w x c)	layer
conv1	32x32x128	conv 5x5, 128
pool1	16x16x128	pool 3x3 / 2
conv2	16x16x128	conv 5x5, 128
pool2	8x8x128	pool 3x3 / 2
conv3	8x8x128	conv 5x5, 128
pool3	4x4x128	pool 3x3 / 2
fc	10(or 100)	fully connected 10(or 100)

4. 실험

4.1 실험 개론

무작위 풀링 연산 선택을 검증하기 위해 총 2가지 실험

으로 구성하였다. 첫 번째 기존의 컨볼루션 신경망 구조처럼 하나의 풀링 계층에 하나의 풀링 연산을 선택하는 방법으로 만들어진 모든 풀링 연산 조합과 무작위 풀링 연산 선택에 대해 비교 실험하였다. 예를 들어, 두 개의 풀링 계층과 최대값, 평균값 두 개의 풀링 연산이 있다면, 최대값-최대값, 최대값-평균값, 평균값-최대값, 평균값-평균값 총 4개의 풀링 연산의 조합으로 이루어진 구조를 만들 수 있다. 두 번째 실험으로는 풀링 계층을 사용하고 비교적 복잡한 구조에 무작위 풀링 연산 선택을 적용하여 비교실험을 하였다.

실험에 사용된 구조는 다음과 같다. 첫 번째 실험은 비교적 단순한 구조에 대해 실험을 실시하였으며, 구조는 표1과 같다. 여기서 conv 5x5, 128은 필터의 크기가 5x5, 피쳐맵의 개수가 128인 컨볼루션 계층을 의미하며, pool 3x3 / 2는 풀링 영역 크기가 3x3, 스트라이드(stride)가 2인 풀링 계층을 의미한다. 그리고 마지막 풀링 계층 뒤에 노드가 0이 될 확률이 0.5인 dropout을 적용하였다. 두 번째 실험은 vgg16[9] 및 densenet100[10]에 제시된 구조에 대해 무작위 풀링 연산 선택을 적용하여 실험하였다.

학습 자료는 영상분류에서 주로 사용되는 CIFAR10, CIFAR100에 대해 실험하였다. 두 학습 자료 모두 50000장의 학습 자료와 10000장의 시험 자료가 있으며, 크기가 32x32인 RGB 영상이다. 전처리는 RGB 각 채널 별로 평균이 0, 표준편차가 1이 되도록 하였다.

학습 방법은 그래디언트 하강 방식을 적용하였다. 미니배치의 크기는 128, 모멘트는 0.9로 하였으며, 학습률은 초기 0.1로 수행하여 더 이상 학습이 되지 않을 때 마다 0.1배를 하였으며 최종 학습률 0.001까지 진행을 하였다.

마지막으로 풀링 연산은 컨볼루션 신경망 구조에서 많이 사용하는 최대값 풀링 및 평균값 풀링을 사용하였으며, 무작위 풀링 연산 선택의 경우 두 풀링 연산 선택 확률을 동일한 확률로 수행 하였다.

4.2 실험 결과

첫 번째 실험인 풀링 연산의 조합으로 만들어진 모든 구조와 무작위 풀링 연산 선택의 실험 결과는 표2, 3와 같다. 우선 각 풀링 연산의 조합마다 성능이 다르다는 것을 알 수 있다. 이는 직접 학습을 하기 전까지는 알 수 없으므로 총 8번의 교차검증을 통해 최고의 조합을 선택

하였다. 반면 제안한 방법은 풀링 연산을 선택할 필요가 없기 때문에 1번의 학습만 수행하였으며 학습량이 1/8로 감소하였음을 알 수 있다. 또한 무작위 풀링 연산 선택 방법은 모든 풀링 연산 조합보다 성능이 좋음을 볼 수 있다. 따라서 무작위 풀링 연산 선택을 사용하면 최고의 풀링 연산 조합을 찾기 위한 교차검증을 수행하지 않아도 되며, 추가적으로 제안한 방법은 학습 간 풀링 연산이 무작위로 선택됨으로써 학습자료가 하나의 고정된 구조가 아닌 다양한 구조에 학습되어 dropout, dropconnect와 같이 여러 구조에 대해 평균을 취한 효과를 얻어 과적합 문제를 해결하는데 도움을 줌을 알 수 있다.

두 번째 실험은 비교적 복잡한 구조인 vgg16, densenet100에 대해 기존 논문에서 제시한 풀링 연산과 무작위 풀링 연산 선택이 적용되었을 때 성능을 비교하였으며, 실험결과를 표4, 5와 같다. 두 구조 모두 무작위 풀링 연산 선택이 적용되었을 때 더 좋은 성능을 보임을 알 수 있다.

두 실험 모두에서 볼 수 있듯이 동등조건(동일한 학습 가능한 파라미터 수)하에서 제안한 방법이 기존방법대비 분류 성능이 향상되었음을 볼 수 있다.

Table 2. CIFAR10 classification performance for first experiment. ‘m’ means max pooling and ‘a’ means average pooling.

pool	accuracy(%)	pool	accuracy(%)
m-m-m	85.13	a-m-a	83.20
m-m-a	85.13	a-a-m	82.79
m-a-m	84.82	a-a-a	83.98
a-m-m	82.12	proposed	86.45
m-a-a	85.53		

Table 3. CIFAR100 classification performance for first experiment

pool	accuracy(%)	pool	accuracy(%)
m-m-m	54.34	a-m-a	54.97
m-m-a	57.84	a-a-m	51.99
m-a-m	55.70	a-a-a	55.48
a-m-m	52.17	proposed	60.80
m-a-a	58.56		

Table 4. CIFAR10 classification performance for second experiment

network	accuracy(%)
vgg16	89.48
vgg16 with proposed method	90.91
densenet100	93.30
densenet100 with proposed method	93.63

Table 5. CIFAR100 classification performance for second experiment

network	accuracy(%)
vgg16	64.44
vgg16 with proposed method	67.56
densenet100	71.87
densenet100 with proposed method	72.79

5. 결론

본 논문은 각 풀링 영역마다 풀링 연산을 무작위로 선택하는 무작위 풀링 연산 선택을 제안하였다. 이 방법은 두 가지 이점이 있다. 첫 번째는 최고의 풀링 연산 조합을 찾기 위한 교차검증 횟수의 감소이다. 두 번째는 학습 간 확률 개념을 반영함으로써 과적합 문제를 해결하는데 도움을 준 것이다. 제안된 방법을 검증한 결과 모든 풀링 연산의 조합보다 더 좋은 성능을 보였다. 결론적으로 무작위 풀링 연산 선택을 활용하면 교차검증 횟수를 줄일 수 있으며, 과적합 문제를 해결할 수 있다.

References

- [1] Ciregan, Dan, Ueli Meier, and Jürgen Schmidhuber. "Multi-column deep neural networks for image classification," *Computer vision and pattern recognition (CVPR), 2012 IEEE conference on. IEEE*, 2012. DOI: <https://doi.org/10.1109/CVPR.2012.6248110>
- [2] Parkhi, Omkar M., Andrea Vedaldi, and Andrew Zisserman, "Deep Face Recognition," *BMVC*, vol. 1. no. 3. 2015. DOI: <https://doi.org/10.5244/C.29.41>
- [3] Tian, Yonglong, et al, "Deep learning strong parts for

- pedestrian detection," *Proceedings of the IEEE international conference on computer vision*, 2015.
DOI: <https://doi.org/10.1109/ICCV.2015.221>
- [4] Amodei, Dario, et al, "Deep speech 2: End-to-end speech recognition in english and mandarin," International Conference on Machine Learning. 2016.
- [5] Srivastava, Nitish, et al, "Dropout: a simple way to prevent neural networks from overfitting," *Journal of Machine Learning Research* 15.1 (2014): 1929-1958.
- [6] Wan, Li, et al, "Regularization of neural networks using dropconnect," *Proceedings of the 30th international conference on machine learning (ICML-13)*. 2013.
- [7] Ioffe, Sergey, and Christian Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *arXiv preprint arXiv:1502.03167* (2015).
- [8] Huang, Gao, et al, "Deep networks with stochastic depth," *European Conference on Computer Vision. Springer, Cham*, 2016.
DOI: https://doi.org/10.1007/978-3-319-46493-0_39
- [9] Simonyan, Karen, and Andrew Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556* (2014).
- [10] Iandola, Forrest, et al, "Densenet: Implementing efficient convnet descriptor pyramids," *arXiv preprint arXiv:1404.1869* (2014).

류 서 현(Seo-Hyeon Ryu)

[정회원]



- 2013년 2월 : 경북대학교 전자공학과(전자공학학사)
- 2015년 2월 : 한국과학기술원 공과대학원 전기 및 전자공학과(전기 및 전자공학석사)
- 2014년 12월 ~ 현재 : 국방기술품질원 연구원

<관심분야>

딥러닝, 레이더, 영상처리