|  |  |  |  |
| --- | --- | --- | --- |
| **연 구 논 문 의 검 토** | | | |
|  | | | |
| **논문제목** | Going Deeper with Convolutions | | |
| **저 자** | Christian Szegedy | **출 처** |  |
| **키 워 드** | **Convolution, Sparse, Inception Module** | | |
| **연 구 동 기 및 목 표** | | | |
| * 높은 성능을 내는 모델은 모델의 크기를 키우면 된다. (depth와 width 둘 다) 하지만 이렇게 모델의 크기를 키울 경우 overfitting이 발생하고 연산이 늘어나기 때문에 비용이 증가한다. * 위 두 문제는 sparsity로 해결할 수 있다. 하지만 sparsity를 모델에 부여하는 것에도 문제가 있다. 그것은 컴퓨터 연산에는 비효율 적이고 현실에서는 sparsity보다 density가 더 많이 쓰이기 때문이다. * 따라서 완벽한 sparse구조는 아니지만 dense구조와 유사한 구조를 가진 Inception 모델을 고안하게 되었다. | | | |
| **데 이 터 셋** | | | |
| * ILSVRC 2014 | | | |
| **Optimization** | | | |
| * 0.9 momenturm(SGD) * learning rate는 8epoch마다 4%씩 감소 * 모든 필터 이후 ReLU 사용 | | | |
| **모 델(아 키 텍 처)** | | | |
| * 한 layer에 여러 개의 필터 결과를 적용하여 결과를 종합한다. 모델은 낮은 지점(input과 가까운 지점)에서는 작은 필터에 가중치가 높아 지협적인 부분을 잘 탐지하지만, 높은 지점(output과 가까운 지점)에서는 큰 필터가 높은 가중치를 가져 점점 이미지 전체를 아우르게 된다.      * 초기 Inception Module : 위에서 이야기한 대로 한 layer에 여러 개의 필터를 적용한 후 결과를 종합한다. 초기 모듈의 문제점은 5x5 필터의 비용과 pooling에 있다. 5x5는 비용이 굉장히 비싼데(연산량이 많음) 5x5 이후 pooling을 적용한다. 그런데 pooling은 이전 layer의 결과와 pooling의 결과로 나오는 필터의 수가 같다. 이는 비싼 5x5 필터를 한 층에서만 두 번을 쓰는 것과 동일한 비용이 든다는 것이다.(pooling은 필수적이라고 함)      * Inception Module : 이런 비용적인 측면을 해결하기 위해 각 필터 앞에, pooling 뒤에 1x1 필터를 적용한다. 1x1 필터를 적용하는 주된 이유는 1X1 필터를 통과하고 나면 이미지의 특성은 보존한 채 차원을 원하는 대로 조절할 수 있어 연산량을 줄일 수 있고, 1x1 필터 이후에 ReLU 활성화 함수를 적용하기 때문에 비선형성이 증가하기 때문이다. 따라서 1x1 필터를 사용하므로 파라미터는 증가시키면서 연산량은 그렇게 크게 증가하지 않게 되었다. | | | |
| **성 능** | | | |
|  | | | |
| **한 계** | | | |
|  | | | |
| **결 론** | | | |
| * Inception구조는 sparse구조를 dense구조로 근사화 했고, 성능 개선이 있었다.(기존 CNN에서 사용한 방식과 다름) * 성능은 크게 상승하지만 연산량은 그렇게 크게 증가하지 않는다. | | | |

|  |
| --- |
| **노 트** |
| * Hebbian Learning   연결된 뉴런들 사이에서 한 뉴런이 다른 뉴런을 활성화하는데 충분한 자극을 지속적으로 제공하면, 뇌는 그에 맞게 구조를 변화한다. ex) 파블로프의 개   * sparse data와 dense data   sparse data : 대부분의 값이 0인 행렬  예 0 0 0 0 1  0 2 0 6 0  0 0 0 2 2  dense data : 대부분의 값이 0이 아닌 행렬  예 1 2 4 6 1  1 2 9 6 0  6 7 0 2 2   * 왜 1X1 필터를 통과하면 차원이 줄어들까?([참고 링크](https://gaussian37.github.io/dl-dlai-network_in_network/))   예를 들어 input이 28x28x192라고 하고, 원하는 output이 28x28x32가 되게 하려면 어떻게 해야 할까? 1x1x192의 필터를 32개 사용하면 된다. 여기서 중요한 것은 **1x1 필터의 채널 개수가 input 채널 개수와 동일하다는 것이다**. Input을 한 장의 feature map으로 바꾼 feature map을 32장 겹딘다는 것 같은데,, 이렇게 하면 데이터의 특징을 잘 반영할 수 있을까..???   * sparse구조, dense구조가 직관적으로 떠오르지 않음. 이 둘은 무엇일까? |

참고한 사이트

* <https://phil-baek.tistory.com/entry/3-GoogLeNet-Going-deeper-with-convolutions-%EB%85%BC%EB%AC%B8-%EB%A6%AC%EB%B7%B0>
* <https://euneestella.github.io/research/2021-10-14-why-we-use-1x1-convolution-at-deep-learning/>