드랍필터가 적용된 인셉션 모듈기반의 희소 컨볼루션 신경망을 활용한 한글 필기체 인식

강우영○ 장병탁
서울대학교 컴퓨터공학부
{wykang, btzhang}@bi.snu.ac.kr

Hangul Handwriting Recognition using Sparse Convolutional Neural Networks Based on Inception Modules with Dropfilter

Woo-Young Kang ○, Byoung-Tak Zhang
School of Computer Science and Engineering, Seoul National University

요 약

본 논문에서는 드랍필터 기법을 인셉션 모듈 기반의 컨볼루션 신경망 (CNN)에 적용하여 한글 필기체에 대한 인식을 수행하였다. 인셉션 모듈은 뛰어난 성능을 보이면서 동시에 CNN의 연산량을 획기적으로 감소시킨 GoogLeNet의 핵심 모듈이다. 또한 드랍필터는 일반화 오류를 줄이기 위해 CNN의 마스크필터에 적용되는 Regularization 기법의 일종으로 CNN의 마스크필터 자체를 임의의 확률로 드랍시키는 방법이다. 실험은 드랍필터가 적용된 인셉션 모듈 기반의 CNN을 사용하여 520클래스, 260,000글자로 구성된 한글 필기체 이미지 DB에 대한 인식 실험을 수행하였다. 실험 결과 제안하는 모델이 드랍필터를 적용하지 않은 기본적인 LeNet구조의 CNN에 비해 약 3배 적은 학습 변수로 3.275% 더 높은 인식률을 달성할 수 있음을 확인하였다.

1. 서 론

자동화된 우편번호 인식, 문서의 주제분류 등에서 필기체 인식에 대한 수요가 꾸준히 증가하면서, 1990년대부터 필기체 인식기 대한 연구가 활발히 수행되기 시작하였다. 과거에는 주로 지지벡터 머신 (SVM)[1]이나 확률 그래프 기반의 모델[2]이 필기체 인식에 주로 사용되었다. 최근에는 컨볼루션 신경망(CNN)이 여러 이미지 인식 분야에서 뛰어난 성능을 보임에 따라 이를 기반으로 한 필기체 인식 연구 또한 수행되었다[3]. 한글 필기체의 경우 그림 1에서 볼 수 있듯 숫자나 영어 알파벳보다 더 복잡한 구조를 가지고 있다. 또한, 글자의 종류 역시 2,000가지 이상으로 매우 많아서 과거에 사용되었던 단순한 선형 모델들로는 높은 인식률을 달성하기 어렵다. 따라서 CNN을 기반으로 한 모델을 사용하여 한글 필기체를 인식하는 것은 적절한 선택이라고 볼 수 있으며, 관련된 연구 역시 수행된 바 있다[4].

필기체 인식이 응용되는 분야는 주로 스마트폰이나 태블릿 PC와 같이 자원이 제한된 환경이므로 인식기에 사용되는 모델을 디자인 할 때에는 인식률과 이용 자원을 동시에 고려해야 한다. GoogLeNet은 최근 Large Scale Visual Recognition Challenge에서 적은 파라미터로도 인상적인 인식률을 보인 바 있다[5]. 이러한 장점을 사용하여 GoogLeNet의 핵심 구성요소인 인셉션 모듈을 한글 필기체 인식 문제에 적용한 사례 역시 보고되었다[6]. 하지만 딥 러닝 모델의 경우 일반적으로 데이터를 쉽게 과 적합 (over fitting) 시키는 경향이 있다. 따라서 낮은 일반화 오류를 달성하기 위해 적절한 regularization 기법을 적용시키는 것 역시 중요하다.

드랍아웃은 과 적합 문제를 완화시키기 위해 주로 사용되는 기법이다[7]. 드랍아웃은 일반적으로 완전 연결된 (Fully-Connected) 다층 퍼셉트론 (Multi Layer Perceptron)에 주로 사용되었으며, 각 은닉 층의 은닉 노드들을 임의의 확률로 제거하는 방법이다. 이를 통해 노드들 간의 co-adaptation을 완화시키고 앙상블 학습의 효과를 얻을 수 있어 일반화 오류를 낮추는 데 도움이 된다고 알려져 있다. 본 논문에서는 드랍아웃 기법을 CNN에 적합한 형태로 바꾼 spatial dropout 기법[8]의 일종인 dropfilter기법을 인셉션 모듈에 적용하여 한글 필기체에 대한 인식을 수행하였다. 실험은 드랍필터를 인셉션 모듈에 적용하여 CNN 모델을 구성하였으며, 이를 500클래스, 260,000글자로 구성된 한글 필기체 인식에 적용하였다. 실험을 통해 제안한 모델이 드랍필터가 적용되지 않은 기본적인 LeNet[9] 구조의 CNN보다 약 3

그림 1. 한글 필기체 데이터의 예시

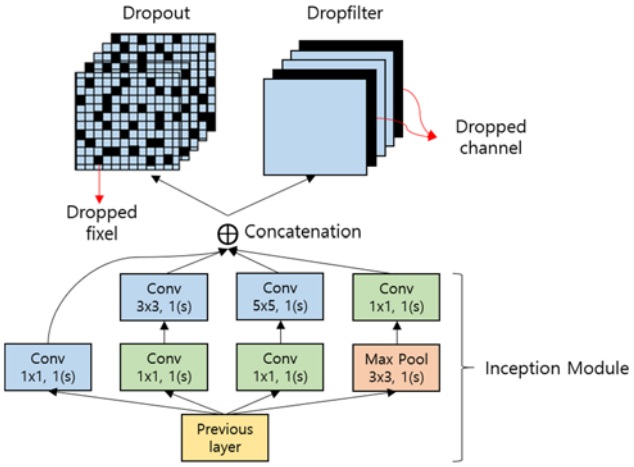

그림 2. 드랍아웃과 드랍필터 적용된 인셉션 모듈의 차이. 검은색 부분은 드랍된 픽셀 값 혹은 채널을 의미한다.

배 적은 변수로도 3.275% 높은 인식률을 달성할 수 있음을 관찰하였다.

2. 드랍아웃과 드랍필터

드랍아웃 기법은 MLP의 각 은닉 층에서 임의의 확률로 각 노드의 값을 0으로 만드는 기법이다[7]. 이를 통해 전체 네트워크는 학습 시 희소한 성질을 가지며 일반화 오류를 낮추는데 도움이 된다고 알려져 있다. Tompson 등은 드랍아웃을 CNN에 그대로 적용시킬 경우 과 적합 문제를 완화시키지 못한다고 하였다[8]. 이는 자연 이미지의 경우 인접 픽셀들 간에 연관성이 강하기 때문이다. 따라서 개개의 픽셀 값을 드랍시키는 대신 인접한 픽셀을 같이 드랍시키는 spatial dropout 기법을 제안하였다. 본 논문에서는 spatial dropout을 서브 네트워크들의 앙상블 학습의 효과 측면에서 재해석하였다. 즉 CNN에서 핵심이 되는 학습 변수는 컨볼루션 마스크들이므로 픽셀 값 각각이 아닌 각각의 마스크단위로 드랍을 시켜야 서브 네트워크들의 앙상블 학습 효과를 얻을 수 있다고 보았다. 이를 수식으로 표현하면 다음과 같다.

$$r^{(l)} \sim Bernoulli(p), \quad (1)$$

$$y^{(l+1)} = \mathcal{F}(b^{(l)} + z^{(l)} * \mathcal{M}^{(l)} \otimes r^{(l)}) \quad (2)$$

수식에서 $r^{(l)} \in \{0,1\}$는 베르누이 분포로부터 임의의 확률 p로 추출된 드랍아웃 마스크를 나타내며 드랍아웃 마스크의 크기는 마스크 필터 $\mathcal{M}^{(l)}$의 갯수와 동일하다. $b^{(l)}$와 $z^{(l)}$, $y^{(l+1)}$은 각각 l층의 바이어스, 입력 특징 맵, 출력 특징 맵을 의미하며 $\mathcal{F}(\cdot)$은 비 선형 활성화 함수를 의미한다. 또한, $*$ 과 $\otimes$ 은 각각 컨볼루션 연산과 element-wise 곱셈을 의미한다. 이후 이를 인셉션 모듈에 적용하면 그림 2와 같으며, 그림 2에서는 또한 드랍아웃과 드랍필터에 차이에 대해 개념적으로 보여주고 있다. CNN에서 드랍된 마스크 필터로 컨볼루션을 수행하면 출력 특징 맵에서 드랍된 마스크 필터에 대응되는 채널 자체가 통째로 드랍되는 것과 결과적으로 같게 되므로 그림 2에서 비교의 편의상 채널이 드랍된 형태로 표현 되었다.

표 1. LeNet 구조의 한글인식기의 구조

Layers	# of feature maps / stride	# of parameters
Conv1	5x5x64 + 64 / 1	1,664
Poo1	2x2 / 2	–
Conv2	5x5x64x128 + 128 / 1	204,948
Pool2	2x2 / 2	–
Conv3	4x4x128x256 + 256 / 1	524,544
Pool3	2x2 / 2	–
Conv4	4x4x256x512 + 512 / 1	2,098,664
Pool4	2x2 / 1	–
FC1	512x384 + 384	196,992
FC2	384x520 + 520	200,200
Total	–	3,226,012

3. 실 험

실험은 우선 드랍필터가 CNN에 적용되었을 때 기존의 드랍아웃보다 우수함을 검증하기 위해 이미지 분류 문제를 수행하였다. 검증 실험에 사용된 데이터는 10개 클래스, 60,000장의 자연 이미지로 구성된 CIFAR-10 [10]을 사용하였다. 실험에 사용된 모델은 표 1과 같다. 표 3에서 No Drop은 드랍아웃이나 드랍필터를 적용시키

표 3. 드랍필터의 효과 검증

Model	Accuracy %		
	Training	Validation	Test
No Drop	99.892	80.38	80.15
Dropout	91.073	77.6	76.54
Dropfilter	98.86	85.32	84.52

표 2. 인셉션 모듈 기반의 CNN을 사용한 한글인식기의 구조

Layers	Output	1x1	3x3 reduce	3x3	5x5 reduce	5x5	Pool proj	parameters
Incep1	30x30x112	1x32	1x48+48	9x48x32	1x16+16	25x16x16	1x32	20,416
Incep2	15x15x240	1x112x64	1x112x64+64	9x64x64	1x56x8+8	25x16x48	1x112x64	79,440
Incep3	8x8x448	1x240x64	1x240x64+64	9x64x128	1x240x32+32	25x32x128	1x120x128	245,343
Incep4	1x1x512	1x448x128	1x448x96+96	9x96x128	1x448x48+48	25x48x128	1x448x128	443,536
FC1	512x384+384	–	–	–	–	–	–	196,992
FC2	384x520+520	–	–	–	–	–	–	200,200
Total	–	–	–	–	–	–	–	1,185,927

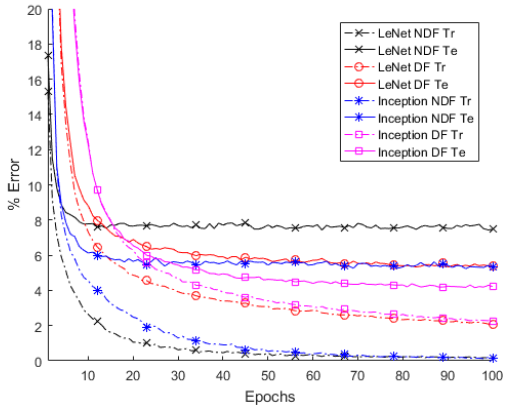

그림 3. 드랍필터를 적용한 LeNet과 인셉션 모듈 기반 CNN에 대한 학습곡선.

지 않은 모델을 의미하며, Dropout, Dropfilter은 각각 드랍아웃과 드랍필터가 적용된 모델을 의미한다. 검증실험 결과 드랍아웃은 모델의 인식률 향상에 도움이 되지 않은 반면 드랍필터는 기본 모델보다 인식률이 테스트 셋 기준 4.37% 향상됨을 확인하였다. 이후 드랍필터를 인셉션 모듈에 적용한 뒤 한글 필기체에 대한 인식을 수행하였다. 한글 필기체 인식에 사용된 데이터는 총 520 클래스, 260,000글자로 이루어져 있다. 실험은 드랍필터와 인셉션 모듈의 효과를 보기 위해 각각을 적용하지 않은 모델과 적용 한 모델로 나누어 실험하였으며, 실험에 사용된 모델의 구조는 표 1과 표 2에 나타나 있다. 실험 결과에 대한 학습 곡선은 그림 3과 같다. 그림 3에서 NDF와 DF는 각각 드랍필터가 적용되지 않은 것과 적용된 것을 의미하며 Tr과 Te는 각각 학습 셋과 테스트 셋에 대한 성능을 의미한다. 실험 결과 LeNet과 인셉션 모듈 기반의 CNN 모두에서 드랍필터를 적용하였을 때 적용하지 않은 경우보다 각각, 2.1%, 1.086% 높은 인식률을 보였다. 또한, 학습 셋과 테스트 셋의 인식률 차이에 해당하는 일반화 오류 역시 드랍필터가 적용되면 더 낮아짐을 볼 수 있었다. 마지막으로, 제안한 모델이 드랍필터를 적용하지 않은 기본적인 LeNet 기반의 CNN에 비해 3배 적은 학습 변수들로 3.275% 높은 인식률을 달성함을 확인할 수 있었다. 결과는 표 4에 정리하였다.

5. 결 론

본 논문에서는 드랍필터를 인셉션 모듈에 적용하여 이를 기반으로 한 CNN을 통해 한글 필기체인식 문제에 대해 다루었다. 실험을 통해 제안하는 모델이 기존의 LeNet 구조의 CNN보다 3.275% 높은 인식률을 보였으며, 이때 사용된 학습 변수의 수 역시 3배정도 적게 사용 되었다. 하지만 드랍필터가 매우 깊은 모델의 모든 은닉 층에 적용 될 경우 학습이 매우 느려지거나 심지어는 학습이 안 되는 문제점을 발견하였다. 이는 임의의 확률로 선택되는 서브 네트워크의 조합수가 너무 많아 학습이 어려워 지기 때문인 것으로 분석되었다. 따라서 추후 드랍필터를 깊게 적용하는 방법에 대한 연구를 수행할 것이다. 또한, 좀 더 실용적인 응용을 위해 하나의 모델로 일본어, 중국어 등과 같은 다양한 언어를 분류할 수 있게 하는 것 역시 추후 연구해 볼 예정이다.

표 4. 한글 필기체 인식 결과

Model	Accuracy %		
	Training (a)	Test (b)	(a) - (b)
LeNet NDF	99.847	92.573	7.274
LeNet DF	97.943	94.673	3.270
Inception NDF	99.876	94.762	5.114
Inception DF	97.737	95.848	1.889

감사의 글

본 연구는 미래창조과학부 및 정보통신기술진흥센터의 정보통신·방송 연구개발 사업[2017-0-00162, 고령 사회에 대응하기 위한 실환경 휴먼케어 로봇 기술 개발]과 정보통신·방송 기술 개발 사업[2015-0-00310-SW스타랩, 웨어러블센서기반 실생활 학습 자율지능 인지에이전트 SW]의 일환으로 수행하였음.

참고문헌

[1] Bahlmann, C., et al. “Online handwriting recognition with support vector machines-a kernel approach.” In *Proc. of the IEEE International Workshop on Frontiers in Handwriting Recognition*, 2002.

[2] Cho, S. J., et al. “Bayesian network modeling of hangul characters for online handwriting recognition.” In *Proc. of the IEEE International Conference on Document Analysis and Recognition*, 2003.

[3] Sermanet, P., et al. “Convolutional neural networks applied to house numbers digit classification.” In *Proc. of the IEEE International Conference on Pattern Recognition, 2012*.

[4] Kim, I. J., et al. “Improving discrimination ability of convolutional neural networks by hybrid learning”. *International Journal on Document Analysis and Recognition*, 19(1):1-9. 2016.

[5] Szegedy, C., et al. “Going deeper with convolutions”. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1-9, 2015.

[6] Kang, W. Y., et al. “Hangeul handwriting recognition using deeper convolutional neural networks based on inception modules.” *Korea Computer Congress 2016 (KCC2016)*, pp. 883-885, 2016.

[7] Srivastava, N, et al. "Dropout: a simple way to prevent neural networks from overfitting." *Journal of Machine Learning Research 15.1 (2014)*: 1929-1958.

[8] Tompson, J, et al. "Efficient object localization using convolutional networks." *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2015.

[9] LeCun, Y, et al. "Gradient-based learning applied to document recognition." *Proceedings of the IEEE 86.11 (1998)*: 2278-2324.

[10] Krizhevsky, A, et al. "Learning multiple layers of features from tiny images." (2009).