겹친 문자 이미지 분류를 위한 합성곱 신경망 모델의 정확도 분석

이상민, 김남기
경기대학교 소프트웨어경영대학 AI컴퓨터공학부
e-mail : d9249@kyonggi.ac.kr, ngkim@kyonggi.ac.kr

Accuracy analysis of convolutional neural network models for overlapped character image classification.

Sangmin Lee, Namgi Kim
Department of AI Computer Engineering
at Kyonggi University Software Management University.

요 약

본 논문에서는 필기체 분류의 문제가 아니라 그보다 더 어렵다고 볼 수 있는 필기체로 겹쳐 쓴 문자를 분류하는 문제에 기존의 딥러닝 모델을 적용하여 그 성능을 분석해 본다. 이를 위해 그동안 연구되었던 합성곱 신경망(convolutional neural network)모델이 구성되어있는 tensorflow를 활용하여 합성곱 신경망 모델들에 대한 학습을 진행했으며, 합성곱 신경망의 구조에 따른 성능 차이를 볼 수 있었다.

1. 서 론

MNIST(Modified National Institute of Standards and Technology database)의 이미지 분류 대회의 필기체 분류 문제 부분에서 현재를 기준으로 사람의 인식률은 약 95%라고 알 수 있으며, 딥러닝을 통해 학습된 모델 중 가장 높은 정확도는 99.87%로 사람보다 약 4.8% 더 높은 정확도를 보이므로 딥러닝을 통한 학습이 사람의 판단보다 더 정확하다고 볼 수 있다. [1] 하지만 앞서 설명한 숫자 분류의 문제는 단순히 필기체를 인식해 분류해내는 문제에서 그쳤기에, 본 연구에서는 필기체 분류의 문제가 아니라 그보다 더 어렵다고 볼 수 있는 필기체로 겹쳐 쓴 문자를 분류하는 문제에 기존의 저명한 딥러닝 모델을 적용하여 성능을 파악한다.

2. 관련 연구

그동안 연구되었던 합성곱 신경망(convolutional neural network) 모델들 중 VGGNet [2]은 옥스포드 대학의 연구팀 VGG에 의해 개발된 모델로써, 2014년 이미지넷 이미지 인식 대회 [3]에서 준우승을 한 모델이다. VGGNet이 준우승을 하긴 했지만, 구조의 간결함과 사용의 편이성으로 인해 GoogLeNet(Inception) [4]보다 더 각광받았다. VGGNet은 16개(VGG16) 또는 19개(VGG19)의 층으로 모델을 구성한다.

2015년 ILSVRC에서 우승을 차지한 ResNet [5]은 마이크로소프트에서 개발한 알고리즘이고, 북경연구소의 중국인 연구진이 개발한 알고리즘이다. 50층(ResNet50), 101층(ResNet101), 152층(ResNet152)의 ResNet으로 구분되며, 깊은 구조일수록 성능이 좋아서 152층의 ResNet이 가장 성능이 뛰어나다. Pre-Activation ResNet(ResNetV2) [6]는 기존의 ResNet의 성능을 개선하기 위해 여러 실험을 수행한 뒤 후속 모델로 ResNet처럼 50층(ResNet50V2), 101층(ResNet101V2), 152층(ResNet152V2)의 높이로 구성된 모델들이 탄생하였다.

2014년 이미지넷 이미지 인식 대회에서 우승한 GoogLeNet(Inception)은 Inception-v2, Inception-v3, Ince-ption-v4가 제안되었으며, Inception-v3는 Inception-v2의 architecture는 그대로 가져가고, 여러 학습 방법을 적용한 버전이다.

이의 후속 연구로 Inception에 ResNet의 아이디어를 적용하여서 Inception-ResNet-v1과 Inception-ResNet-v2를 제안하였으며, Inception-ResNet-v1은 Inception-v3과 연산량이 거의 유사한 모델 Inception-ResNet-v2는 Inception-v4와 연산량이 거의 유사하면서 정확도가 더 좋은 모델이다. [7]

DenseNet [8]은 ResNet과 Pre-Activation ResNet보다 적은 파라미터 수로 더 높은 성능을 가진 모델이다. ResNet은 feature map 끼리 더하기를 해주는 방식이었다면 DenseNet은 feature map끼리 병합(Concatenation)을 시키는 것이 가장 큰 차이점을 보이며, 이를 통해서 Vanishing Gradient 개선, Feature Propagation 강화, Feature Reuse, Parameter 수 절약의 이점을 갖는다.

EfficientNet [9]은 이미지 분류 문제에 대해서 기존보다 훨씬 적은 파라미터수로 더욱 좋은 성능을 내서 최첨단(State-Of-The-Art(SOTA))을 달성한 모델이다. 기존의 모델들이 높이를 늘리거나, 채널 넓이를 넓히거나, 이미지의 해상도를 높이는 방법으로 모델의 성능을 높였다면 EfficientNet은 세 가지 방법에 대한 최적의 조합을 AutoML을 통해 찾은 모델이다.

3. 문제 정의 및 실험 방법

본 논문에서는 그림 1에서 볼 수 있듯 Digit과 Letter를 합쳐서 만들어진 이미지에서 Letter의 범위를 넘어서는

Digit 부분의 Pixel 값을 0으로 낮추어 겹치지 않는 부분을 제거하여 제일 오른쪽의 이미지로 만들어진 이미지를 학습 데이터로 사용합니다. 연두색의 영역은 겹쳐진 숫자의 영역이고, 연녹색의 영역은 Letter의 영역이며 해당 부분에는 감추어진 숫자가 없다는 것을 의미한다. 이 때 이미지가 컬러로 표현된 것은 보다 쉽게 보기 위함이고 실제 학습에서는 grayscale의 이미지가 사용된다.

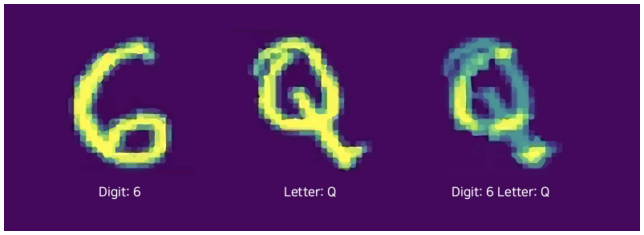

그림 1 문제 설명을 위한 이미지 [10]

4. 실험

본 논문에서는 표 1과 같이 앞서 설명한 모델들 중 총 20개의 모델에 대해 학습을 진행하였다. keras documentation [11]을 참조한 표 1에서 깊이(Depth)란 네트워크의 토폴로지 깊이로 활성화 레이어, 배치 정규화 레이어 등을 포함한다. Input size는 대상 모델의 최적 입력 이미지 크기를 말하며, Result Private Accuracy는 문제에 대한 정확도 수치를 의미한다.

(표 1) 각 모델에 대한 설명 및 실험 결과

Model	Parameter	깊이	Input Size	Result Private Accuracy
VGG16	138,357,544	23	224	0.86037
VGG19	143,667,240	26	224	0.88991
ResNet50	25,636,712	-	224	0.90816
ResNet101	44,707,176	-	224	0.90377
ResNet152	60,419,944	-	224	0.89568
ResNet50V2	25,613,800	-	224	0.90076
ResNet101V2	44,675,560	-	224	0.91512
ResNet152V2	60,380,648	-	224	0.89647
InceptionV3	23,851,784	159	299	0.82831
Inception ResNetV2	55,873,736	572	299	0.74758
DenseNet121	8,062,504	121	224	0.91689
DenseNet169	14,307,880	169	224	0.91285
DenseNet201	20,242,984	201	224	0.90940
Xception	22,910,480	126	299	0.91009
EfficientNetB0	5.3M	-	224	0.89830
EfficientNetB1	7.8M	-	240	0.90032
EfficientNetB2	9.2M	-	260	0.90930
EfficientNetB3	12M	-	300	0.90693
EfficientNetB4	19M	-	380	Learning faild
EfficientNetB5	30M	-	456	Learning faild
EfficientNetB6	43M	-	528	Learning faild
EfficientNetB7	66M	-	600	Learning faild

VGG16, VGG19 모델을 학습하는 과정에서 학습이 진행되지 않아 원인을 찾아보니 FC layer에서 4,096개에서 10개로 줄이는 과정에서 과적합 발생으로 인하여 학습이 제대로 이루어지지 않았으나 FC layer를 추가하여서 학습을 진행하여 약 88%의 정확도를 보였지만, 다른 모델과의 학습은 추가 Layer 없이 학습이 진행되었기 때문에 같은 과정의 학습을 진행하였다고는 보기 어렵다.

모델의 최적화된 input size와 학습의 사용한 image의 input size가 같은 경우 가장 높은 정확도를 보인 모델은 DensetNet121 모델임을 알 수 있고, 최적 image size와 input image size가 다른 경우의 가장 정확도가 높은 모델은 Xception임을 볼 수 있다.

5. 결론

더욱 정확도를 높이기 위해서는 학습 데이터의 숫자가 있는 영역을 한 장의 이미지로 만들어본 결과 하나의 숫자를 사용하였다고 추론을 할 수 있었는데 숫자가 무조건 없어야 하는 영역을 전처리를 진행하여 학습데이터의 숫자 영역을 모두 합친 이미지 10개 중 하나의 이미지로 예측하는 학습을 진행하는 방법과 model ensamble, Validation K-fold, Parameter optimization 등의 방식을 사용한다면, 더 높은 정확도를 높일 수 있을 것으로 판단된다 따라서 향후에는 이러한 방향의 연구를 추가적으로 수행해 볼 계획이다.

6. acknowledgement

이 논문은 2020년도 정부(교육부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임
(No. 2020R1A6A1A03040583)

참 고 문 헌

[1]https://paperswithcode.com/sota/image-classification-on-mnist

[2] Karen Simonyan, Andrew Zisserman, “Very Deep Convolutional Networks for Large-Scale Image Recognition”, ICLR, 2015

[3] https://image-net.org/challenges/LSVRC/

[4] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, Andrew Rabinovich, “Going Deeper with Convolutions”, CVPR, 2015

[5] Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun, “Deep Residual Learning for Image Recognition”, CVPR, 2016

[6] Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun, “Identity Mappings in Deep Residual Networks”, ECCV, 2016

[7] Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, Alex Alemi, “Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning”, ICLR, 2016

[8] Gao Huang, Zhuang Liu, Laurens van der Maaten, Kilian Q. Weinberger, “Densely Connected Convolutional Networks”, CVPR, 2017

[9] Mingxing Tan, Quoc V. Le, “EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks”, ICML, 2019

[10] https://dacon.io/competitions/official/235626/overview

[11] https://keras.io/ko/applications