

딥 러닝 기반 테이블 검출 기법*

최진서[○] 강동현
창원대학교 컴퓨터공학과

jinseo@gs.cwnu.ac.kr, donghyun@changwon.ac.kr

Table Detection Scheme based on Deep Learning

Jinseo Choi[○] Donghyun Kang
Dept. of Computer Engineering, Changwon National University

요 약

최근, 컴퓨팅 성능 향상으로 인해, 딥 러닝 기반의 표 탐지 자동화에 대한 연구들이 활발하게 진행되고 있다. 이는 전산 업무나 산업 현장에서 표의 데이터를 수동으로 처리해야 하는 단순 작업을 경감시킴으로써 업무 효율을 향상시킬 수 있다. 다만, 지금까지 문서에서 표 영역과 구조 탐지 알고리즘에 대한 정밀도가 충분히 높지 않다는 문제가 있다. 이에 본 논문에서는 딥러닝 기반 표 탐지 알고리즘의 정밀도를 높이기 위해 기존 알고리즘의 사전학습 모델을 교체하고 구조를 변경하여 정밀도를 측정하는 실험을 진행했다. 실험 결과, 기존 알고리즘의 정밀도와 비교하여 구조 변경 시, 최대 64.9%만큼 정밀도가 향상하는 결과를 확인할 수 있었다.

1. 서 론

일반적으로 표(Table)는 문서에서 전달하고자 하는 정보를 요약하여 표현하거나 수치 데이터를 제공하기 위한 서식으로 많이 사용된다. 특히 자동차, 조선과 같은 제조 산업에서는 제품 설계에 필요한 수치 데이터나 실험 데이터와 같은 핵심 정보를 알아보기 쉽게 전달하기 위해서 표 형태로 제시된다. 따라서 데이터를 검증하기 위해서는 문서에서 표를 찾아 내용을 추출하는 과정이 요구된다. 이 과정을 수동으로 수행할 경우, 표를 포함하고 있는 문서의 양에 따라 단순 작업 반복이 증가함으로 인해 작업자의 피로도가 증가하고 업무 효율이 감소할 수 있다. 따라서 이러한 문제를 해결하기 위해 문서에서 자동으로 표를 탐지하여 내용을 추출하는 연구들이 진행되었다. 하지만 표 서식은 표준화되어 있지 않고 문서의 양식에 따라 모양이 매우 다양하다. 따라서 다양한 문서에서 표 서식 일반화에 어려움이 있었기 때문에 표 탐지에 대한 성능이 충분히 높지 않았다. 최근에는 컴퓨팅 성능의 향상과 머신러닝(Machine learning) 기술이 발전함에 따라 딥 러닝(Deep learning)을 활용한 표 탐지 자동화에 기법들이 높은 성능을 보이면서 주목받고 있다.

딥 러닝은 입력 데이터를 통해 학습 가능한 가중치를 가진 인공신경망이 깊은 레이어(Layer)를 형성한 심층 신경망을 사용하여 학습하는 방법이다[1]. 딥 러닝은 특히 합성곱(Convolution) 연산을 통해 이미지 데이터로부터 특징(Feature)을 추출하는 방법이 제안된 이후, 합성곱신경망(Convolutional neural network)이 이미지 분류 문제에서 높은 성능을 보이고 있다[2][3].

TableNet은 합성곱 신경망을 사용하여 문서 이미지에서 표와 열(Column) 영역을 추론하는 딥러닝 기반 알고리즘으로 문서 이미지의 특징을 추출하는 인코더(Encoder)와

표와 열을 추론하는 디코더(Decoder)로 구성되어 있다[4]. 이를 통해 문서 이미지에서 표의 위치와 구조를 추론한다. 다만, 문서 이미지에서 그래프나 그림 영역이 존재하는 경우, 해당 영역을 표로 추론하는 것과 같이 여전히 정밀도가 떨어지는 문제가 여전히 존재한다. 이에, 본 논문에서는 TableNet 알고리즘을 분석하고 인코더로 사용되는 학습모델과 인코더의 구조를 변경하여 실제 문서 이미지에 대한 추론 정밀도를 측정하는 실험을 진행했다. 그 결과 기존 대비 정밀도가 최대 64.9%만큼 개선하는 결과를 보였다.

본 논문의 구조는 다음과 같다. 2장에서는 TableNet의 구조에 대해 살펴보고 3장에서는 정밀도를 개선하는 방안을 제안한다. 4장에서 실험을 통해 결과를 분석하고 5장에서 결론을 맺는다.

2. 연구배경

TableNet은 표와 열 영역을 추론하기 위해 기본적으로 FCN(Fully Convolution network)구조를 사용한다. FCN은 이미지 내 객체 탐지를 위해 객체별 영역을 분류하는 시맨틱 세그멘테이션(Semantic segmentation)을 수행하기 위해 고안된 구조이다. 기존 CNN모델에서는 입력된 이미지 데이터가 합성곱 연산에 의해 추출된 값이 완전연결신경망을 통과하면서 클래스별로 분류된다. 이 과정에서 모델의 최종 출력값은 이미지 내 객체의 위치정보가 소실되고 클래스에 대한 정보만 남게 된다. 따라서 시맨틱 세그멘테이션을 수행하기 위해서는 완전연결신경망 레이어를 제거하고 합성곱 레이어로 대체하여 위치정보를 유지한 출력값을 가진 구조가 제안되었다[5].

2.1 TableNet

TableNet은 FCN 구조를 따르는 1개의 인코더와 2개의 디코더로 구성되어 있다. 인코더는 입력받은 이미지 형식의 문서 데이터의 특징을 추론한다. 이때 인코더는 이

* 이 논문은 2021년도 정부(교육부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임(No. NRF-2021R111A3047006)

이미지넷(ImageNet) 데이터셋으로 미리 학습된 이미지 분류 모델인 VGG-19를 사용한다[6]. VGG-19는 합성곱 레이어의 개수를 추가하여 구성된 깊은 네트워크로 이미지 분류 문제에서 높은 정밀도를 보인 모델이다[7]. TableNet에서는 완전연결신경망을 제거하여 인코더 모델로 사용한다. 이는 추론 값이 입력 이미지의 위치정보를 포함하기 위함이다. 다만, 이 값은 대략적인 위치정보만 가지고 있다. 이는 합성곱 연산의 특성 상, 특징을 추출하면서 이미지 해상도가 줄어들기 때문이다. 따라서 원본 크기 이미지에서 표 위치 탐지에 어려움이 존재한다. 이러한 문제를 해결하기 위해 각 디코더는 인코더의 출력값과 인코더 모델에서 표와 열의 위치정보가 비교적 명확한 중간 레이어의 출력값을 가져온다. 이를 통해 문서 이미지 내에서 표와 열의 범위를 추론할 수 있다.

2.2 기울기 소실

일반적으로 심층 신경망의 레이어가 깊을수록 가중치 개수가 증가하면서 보다 복잡하고 정확한 예측이 가능하다[6]. 그러나 학습모델의 네트워크가 너무 깊어지면 가중치 수가 증가하면서 학습모델의 연산량이 증가한다. 또한 학습 시 기울기 소실(Gradient vanishing)[8]로 인해 예측 정밀도가 떨어지는 문제가 발생한다. 기울기 소실은 모델 학습 시 가중치를 업데이트할 정도를 결정하는 기울기 값이 깊은 네트워크를 통과하면서 소실되는 것을 의미한다. 그 결과, 가중치 학습이 제대로 이루어지지 않아 학습모델의 추론 성능이 감소하게 된다. ResNet[8]은 이전 레이어의 출력값을 다음 레이어 출력값에 더해 줌으로써 깊은 네트워크에서 가중치 소실 문제를 해결했다. 이를 통해 더 깊은 네트워크를 구성하여 정밀도를 향상시키는 결과를 보였다.

3. 제안기법

본 논문에서는 TableNet의 정밀도를 개선하기 위해 인코더 모델과 디코더의 구조를 변경하는 기법을 제안한다. 제안하는 기법은 기존에 인코더로 사용하는 VGG-19 보다 정밀도를 높은 ResNet50V2를 사용하고 디코더는 VGG-19의 중간레이어 출력값을 입력으로 사용하도록 변경한다. ResNet50V2는 ResNet구조를 따르는 50개의 합성곱 레이어를 사용하여 구성된 모델이다. 인코더로 사용할 ResNet50V2는 추론 값이 이미지 내 표의 위치정보를 유지하기 위해 완전연결신경망 레이어를 제거하여 사용한다. 또한 기존 방식과 동일하게 이미지넷 데이터셋으로 미리 학습된 가중치를 그대로 사용한다. 또한 인코더의 가중치 학습으로 인해 훈련 데이터셋에 과적합(Overfit)되어 성능이 감소하는 것을 방지하기 위해 모든 레이어의 가중치 업데이트는 수행하지 않도록 구성한다. 디코더는 인코더의 출력값과 VGG-19 중간 레이어의 추론값을 입력으로 받아 가중치를 학습하는 합성곱 레이어로 구성한다. 이는 인코더 모델인 ResNet50V2의 연산방식에 의해 이전 레이어의 출력값이 재사용되기 때문에 중간 레이어에서 이미지 데이터 위치정보에서 노이즈가 발생할 수 있기 때문이다. 따라서 이미지 내 표와 열 위치정보를 VGG-19의 추론값으로 보정한다. 또한 이를 통해 인코더와 디코더가 서로 다른 학습모델을 사용함으로

써 2가지 학습모델의 추론 값을 모두 고려하여 학습하는 앙상블(Ensemble) 학습 효과로 정밀도를 향상시킬 수 있다[10]. 열 영역을 탐지하는 디코더는 표 탐지보다 상대적으로 더 정교한 추론이 요구된다. 따라서 정밀한 추론을 위해 특징 추출을 위한 합성곱 레이어 개수의 표 탐지 디코더의 2배로 구성하여 깊이를 늘렸다. 이후 각 디코더는 입력 데이터로부터 추론한 값을 업스케일(Up-scale)하여 해상도를 높인다. 마지막으로 표와 열이 마스킹된 이미지를 생성한다.

4. 실험 및 평가

논문에서 제안하는 방법의 정확도를 평가하기 위해 본 장에서는 인코더 모델별로 다양한 비교실험을 진행하였다. 실험환경은 표1과 같다.

표 1 실험환경

운영체제	Ubuntu 20.04 LTS
프로세서	Intel® Xeon® Gold 5215 @2.50GHz
GPU	Nvidia® Tesla® T4 16GB
메모리(RAM)	64GB
저장장치	Samsung 980 pro NVMe M.2 SSD 1TB

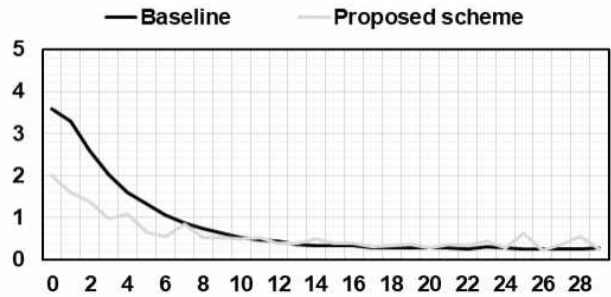


그림 1. 에폭 증가에 따른 검증 오차 비교

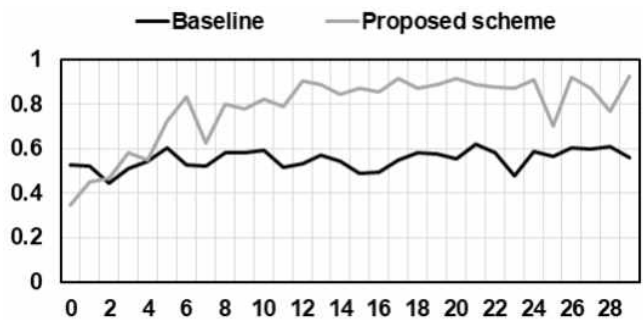


그림 2. 에폭 증가에 따른 F1-score 비교

비교대상으로는 기존 TableNet에서 사용하는 VGG-19를 Baseline으로, 제안 기법인 ResNet50V2를 인코더 모델로 변경하여 실험을 진행하였다. 모델 구현은 Tensorflow2.8[11], Keras2.8[12], CUDA11.4, cuDNN8.4을 사용하였으며 인코더 모델은 케라스(Keras)에서 API 형태로 제공하는 이미지넷 데이터 셋으로 미리 학습된 모델을 사용했다. 전체모델 학습 데이터는 Marmot[13] 데이터 셋에서 표의 위치와 메타데이터 정보가 명시된 xml 파일과 표가 포함된 영문서 이미지 500개를 사용하였다. 또한 표와 열 위치를 학습하기 위해 메타데이터를 참조

하여 문서 이미지에서 각각 표의 위치와 열 위치를 마스킹 한 라벨(Label) 이미지 데이터를 생성하였다. 따라서 원본 문서 이미지가 훈련 데이터, 마스킹 된 이미지가 라벨 데이터로 모델이 입력된다. 전처리가 완료된 전체 데이터 셋에서 80%를 훈련 데이터, 20%를 검증 데이터로 분할하여 훈련을 진행하였다.

모델의 정밀도를 비교하기 위해 추론 데이터와 라벨 데이터와의 오차 Loss와 F1-score를 사용하였다. F1-score는 학습모델의 정답률인 정밀도와 예측값과 실제 정답과 가까운 정도를 나타내는 재현율을 모두 고려하여 조화평균으로 나타낸 지표이다. 수식은 식(1)과 같이 계산된다.

$$F1 - score = 2 \frac{RP}{R+P} \quad (1)$$

R: 재현율(Recall), P: 정밀도(Precision)

표 2. 최종 오차, F1-score 비교

Model	검증오차	F1-score
Baseline	0.280	0.562
Proposed scheme	0.253	0.927

모델 성능 평가는 전체 데이터셋을 반복하는 횟수인 에폭(Epoch)을 30으로 설정하여 훈련을 수행했다. 그림1은 에폭 증가에 따른 각 모델별 검증 오차를 비교한 결과를 보여준다. Baseline은 기존 TableNet에서 인코더 모델로 사용한 VGG-19를 나타낸다. 에폭 초기, 제안기법이 Baseline보다 낮은 오차를 보이다가 후반에 비슷하게 수렴하는 모습을 보여준다. 그림2는 F1-score의 분포를 비교한 결과로 3 에폭부터 제안기법이 더 높은 성능을 나타낸다. 이는 기울기 소실 문제를 해결하여 더 깊은 네트워크를 구성한 ResNet이 다른 모델보다 인코더 모델에서 데이터의 특징을 더 잘 추출하는 것으로 보인다. 표2는 마지막 에폭에서 각 모델의 검증 오차와 F1-score를 보여준다. 제안기법의 경우 Baseline보다 검증 오차가 9.64% 감소하고 F1-score가 64.9% 증가하는 성능을 보였다.

5. 결 론

본 논문에서는 문서에서 표와 열을 탐지하는 딥러닝 기반 알고리즘은 TableNet을 분석하였다. 또한 인코더 모델로 사용하는 VGG-19의 많은 연산량과 기울기 소실 문제를 해결한 ResNet50V2를 인코더 모델로 사용하고, 디코더에서 VGG-19의 중간레이어 출력값을 사용하는 기법을 제안하여 성능을 비교, 분석하였다. 그 결과, 제안기법이 기존 모델보다 F1-score가 64.9% 증가하는 성능을 보였다. 향후, 가중치 규제, 학습률과 같은 하이퍼파라미터를 최적화하고 TableBank[14]와 같은 대규모 데이터셋을 활용하여 인코더 모델을 표 검출에 적합하도록 훈련하는 미세조정과 같은 기법을 사용하여 정확도를 개선하는 연구를 진행하고자 한다.

참 고 문 헌

[1] 김건희, 엄현상, “고성능 딥 러닝 시스템,” 정보과학회지, Vol. 34, No. 2, pp.57-62, 2016.

[2] LeCun, Y., Bengio, Y., and Hinton, G., “Deep Learning”, Nature, Vol. 521, pp.436-444, 2015.
 [3] 김지원, 표현아, 하정우, 이찬규, 김정희, “다양한 딥러닝 알고리즘과 활용,” 정보과학회지 Vol. 33, No. 8, pp.25-31, 2015.
 [4] Paliwal, S., D, V., Rahul, R., Sharma, M., and Vig, L., “TableNet: Deep Learning Model for End-to-end Table Detection and Tabular Data Extraction from Scanned Document Images,” International Conference on Document Analysis and Recognition, pp.128-133, 2019.
 [5] Long, J., Shelhamer, E., and Darrell, T., “Fully Convolutional Networks for Semantic Segmentation,” Computer Vision and Pattern Recognition, pp.3431-3440, 2015.
 [6] Krizhevsky, A., Sutskever, I. and Hinton, G., “ImageNet Classification with Deep Convolutional Neural Networks,” Advances in Neural Information Processing Systems, Vol. 25, pp.1097-1105, 2012.
 [7] Simonyan, K. and Zissermac, A., “Very Deep Convolutional Networks for Large-scale Image Recognition,” arXiv preprint, 2014.
 [8] He, K., Zhang, X., Ren, S., and Sun, J., “Deep Residual Learning for Image Recognition,” IEEE Conference on Computer Vision and Pattern Recognition, pp.770-778, 2016.
 [9] Chollet, F., “Xception: Deep Learning With Depthwise Separable Convolutions”, IEEE Conference on Computer Vision and Pattern Recognition, pp.1251-1258, 2017.
 [10] Ganaie, M., A., Hu, M., Malik, A., K., Tanveer and M., Suganthan, P., N., “Ensemble Deep Learning: A Review,” arXiv, 2021.
 [11] “TensorFlow Core v2.8.0API Documentation”, lastmodified 22 04, 2022, accessed 05 03, 2022, <https://www.tensorflow.org/>
 [12] “Keras API reference”, Keras. lastmodified 30 04, 2022, accessed 05 03, 2022, <https://keras.io/>
 [13] Jing, F., Xin, T., Zhi, T., Ruiheng, Q. and Ying, L., “Dataset, Ground-Truth and Performance Metrics for Table Detection Evaluation,” International Workshop on Document Analysis Systems, pp.445-449, 2012.
 [14] Li, M., Cui, L., Huang, S., Wei, F., Zhou, M. and Li, Z., “TableBank: A Benchmark Dataset for Table Detection and Recognition,” arXiv, 2019.