KIPS Trans. Softw. and Data Eng.
Vol.7, No.11 pp.411~418 pISSN: 2287-5905

https://doi.org/10.3745/KTSDE.2018.7.11.411

Automatic Object Extraction from Electronic Documents Using Deep Neural Network

Heejin Jang† · Yeonghun Chae†† · Sangwon Lee††† · Jinyong Jo†

ABSTRACT

With the proliferation of artificial intelligence technology, it is becoming important to obtain, store, and utilize scientific data in research and science sectors. A number of methods for extracting meaningful objects such as graphs and tables from research articles have been proposed to eventually obtain scientific data. Existing extraction methods using heuristic approaches are hardly applicable to electronic documents having heterogeneous manuscript formats because they are designed to work properly for some targeted manuscripts. This paper proposes a prototype of an object extraction system which exploits a recent deep-learning technology so as to overcome the inflexibility of the heuristic approaches. We implemented our trained model, based on the Faster R-CNN algorithm, using the Google TensorFlow Object Detection API and also composed an annotated data set from 100 research articles for training and evaluation. Finally, a performance evaluation shows that the proposed system outperforms a comparator adopting heuristic approaches by 5.2%.

Keywords : Object Extraction, Deep Learning, Tensorflow, PDF Document

심층 신경망을 활용한 전자문서 내 객체의 자동 추출 방법 연구

장 희 진† · 채 영 훈†† · 이 상 원††† · 조 진 용†

요 약

인공지능 기술의 확산으로 인해 과학기술 분야에서도 연구 데이터의 확보, 저장 및 활용이 중요시 되고 있는 상황이다. 연구 데이터를 확보하기 위해 전자문서 형태의 연구논문으로부터 그래프, 표와 같은 유의미한 객체를 추출하는 다양한 방법들이 제안되고 있다. 경험적 방법론을 이용하는 기존의 연구들은 문서의 편집 특성을 일반화하여 객체들을 추출하기 때문에 다수의 이질적인 형태를 갖는 전자문서들을 대상으로 연구결과를 적용하는데는 한계가 있다. 본 논문은 경험적 방법론의 경직성을 극복하고 이질적인 전자문서들로부터 목표 객체들을 효과적으로 추출하기 위해 심층 학습 기반의 객체 추출 시스템을 제안한다. 텐서플로우 객체 탐지 API의 Faster R-CNN 알고리즘을 기반으로 새로운 학습 모델을 생성했으며 심층 학습과 평가를 위해 총 100여 편의 연구논문들을 대상으로 목표 객체들을 데이터화했다. 마지막으로 성능평가를 통해 제안한 시스템이 경험적 방법론을 적용한 비교 대상에 비해 약 5.2% 높은 성능을 보임을 확인하였다.

키워드 : 객체 추출, 심층 학습, 텐서플로우, 전자문서

1. 서 론

인공지능으로 대변되는 4차 산업혁명의 시대가 도래하면서 데이터의 중요성이 강조되고 있다. 과학기술 분야에서는 실험(experiment)과 계산(computation)을 병행한 과학적 발견이 일반화되면서 과학 데이터의 확보 방법에 대한 연구자들의 관심이 높아지고 있다. 특히 논문 등 PDF(Portable Document Format) 형태의 전자문서에서 과학기술 데이터를 추출하기 위한 다양한 연구들이 제안되고 있다[1-6]. 또한 다수의 정보추출 도구들이 공개 및 유료 소프트웨어로 개발되거나 활용되고 있다는 점에서 과학기술 데이터의 추출에 대한 연구자들의 관심을 간접적으로 유추할 수 있다[7-11].

연구논문은 다양한 과학적 결과를 함축하는 데이터의 집합체로써 해마다 250만 건 이상의 새로운 논문들이 출판되는 것으로 추정된다[12]. 다수의 연구논문들로부터 그래프, 표 등 데이터로써 유의미한 객체를 추출하고 디지털화, 데이터

※ 본 논문은 한국과학기술정보연구원의 지원으로 수행된 연구임.
† 정 회 원 : 한국과학기술정보연구원 과학기술연구망센터 선임연구원
†† 비 회 원 : 한국과학기술정보연구원 과학기술연구망센터 연구원
††† 비 회 원 : 한국과학기술원 생명화학공학과 박사과정
Manuscript Received : March 27, 2018
Accepted : June 7, 2018
* Corresponding Author : Jinyong Jo(jiny92@kisti.re.kr)

베이스화함으로써 다양한 과학기술 응용연구자들은 축적된 대규모의 연구 데이터를 효과적으로 활용할 수 있을 것으로 예상된다. 일례로, 미국 화학회의 화학정보 분과에서 발표되는 학술자료[13]들을 살펴보면 PubChem[14], OSDR(Open Science Data Repository[15]), Open Chemistry[16]와 같은 다수의 프로젝트들은 신소재의 화학적 속성을 예측하거나 화학 반응을 분석하기 위해 대규모 연구데이터를 지속적으로 축적해 활용하고 있다.

전자문서형태(PDF)의 연구논문으로부터 객체 추출을 자동화하기 위한 경험적(heuristic) 방법론이 제안되고 있다[1]. 경험적 방법론에서는 논문의 편집 방식(예, 1단 또는 2단 편집), 캡션(caption)의 위치와 사용된 예약어(예, 그림은 Fig.나 Figure 등의 예약어를 이용하고 객체 하단에 배치), 폰트크기나 줄간격(예, 폰트크기가 12 이상일 경우에는 제목) 등 연구논문에 포함된 객체의 편집 특성을 일반화한 후 추출 규칙(rule)으로 활용한다. 정의된 추출 규칙을 포함하는 연구논문으로부터 특정 객체(예, 절(section) 제목 등)를 추출할 경우에는 90% 이상의 높은 정확도(precision)를 갖는 것으로 보고된다[1]. 하지만 정의된 추출 규칙과 상이한 형태를 갖는 연구논문의 경우에는 정확도가 크게 낮아질 것으로 예상된다.

본 논문은 경험적 방법론의 경직성(inflexibility)을 극복하기 위해 심층 학습 기반의 객체 추출 시스템을 제안한다. 제안된 객체 추출 시스템은 텐서플로우 객체 탐지 API(Tensorflow object detection application programming interface)의 Faster R-CNN(Region based Convolution Neural Network) 알고리즘[17]을 이용했다. 심층 학습과 평가를 위해 총 100여 편의 전자문서에 포함된 객체들을 데이터화했으며 객체 추출을 위한 새로운 학습 모델을 생성했다. 또한 성능평가를 통해 제안한 시스템이 경험적 방법론을 적용한 비교 시스템보다 약 5.2%의 높은 성능을 보임을 확인했다. 본 논문은 전자문서에서 객체를 자동으로 추출하기 위해 최초로 심층 신경망 기술을 적용한 연구라는 점에서 의의를 갖는다.

본 논문의 예비 결과(preliminary result)는 실제 적용 가능성 측면에서 다음과 같은 추가 연구를 필요로 한다. 첫째, 오탐(false-positive)과 미탐(false-negative)을 줄이기 위해 대규모 학습 집단(training set)에 대한 심층 학습이 요구되며 객체에 대한 분류 방법(예, 그림과 캡션을 개별 객체로 분리 등)이 성능에 미치는 영향이 분석되어야 한다. 둘째, 오탐으로 인한 성능 저하를 방지하기 위해 객체를 탐지한 후, 분절된 객체들 간의 군집화(clustering) 등 후처리 과정이 연구되어야 한다. 또한 그림 파일로 얻어진 캡션 객체에 대해서 텍스트화할 수 있는 방법이 고안되어야 한다. 마지막으로, 경험적 방법론과 심층 신경망이 갖는 각각의 장점을 이용함으로써 성능을 극대화하기 위해 하이브리드 모델에 대한 연구도 필요할 것으로 판단된다.

본 논문의 구성은 다음과 같다. 제 2장에서는 연구논문에 포함된 객체 정보를 추출하기 위한 기존의 연구들을 살펴본다. 제 3장에서는 본 논문에서 제안하는 심층 신경망 기반의 객체 추출 방법에 대해 기술한다. 제 4장에서는 제시된 방법의 성능을 평가하고 마지막으로 제 5장에서 논문을 마무리한다.

2. 문헌정보 추출 관련 연구

본 장은 연구논문으로부터 그림과 표 등 대상 객체를 추출하기 위한 기존 연구들을 살펴본다. 일반적으로 연구논문은 제목, 저자, 키워드, 인용과 같은 다양한 서지 정보를 포함한다. 서지 정보를 추출하기 위해 문서에서 편집 규칙을 분석하고 기계학습을 사용하기 위한 시도가 있었다[5, 6]. GROBID는 의사결정트리, SVM(Support Vector Machine) 등의 기계학습 방법을 도입하여 다양한 문헌으로부터 제목, 저자, 키워드 등 텍스트 형태의 서지 정보를 추출할 수 있다[18, 19].

일반적으로 개별 추출도구들은 특정 서지정보(예, 그림, 제목 등)에 한정해서 최적의 성능을 낼 수 있도록 개발되어 왔다. PDFMEF는 GROBID, ParsCit[20]와 같은 개별 추출도구들을 하나의 프레임워크에 통합하고 특정 서지정보의 추출에 최적의 성능을 내는 도구들을 선택적으로 활용함으로써 전체 문헌에서 다양한 정보를 추출할 수 있다[2]. PDFMEF도 텍스트 형태의 서지정보만 추출하기 때문에 연구논문에 포함된 표와 그림의 추출을 위해서 이용되기는 어렵다.

PDFFigures는 그림과 표를 추출하기 위해 경험적 방법론을 활용했다[1]. 전자문서의 메타데이터를 해석한 후, 예약어(예, Fig., Table 등), 폰트크기, 줄 간격, 정렬방식 등 사전에 정의된 추출 규칙을 활용하여 그림과 표 객체를 추출한다. 추출 규칙은 사람이 그림과 표, 캡션(caption), 본문 등을 인지하는 방식과 유사하게 정의된다(예, 문헌의 앞부분에 위치하고 본문 폰트보다 크게 작성된 텍스트는 제목으로 인식). 추출 규칙에 부합하는 연구논문의 경우에는 90% 이상의 높은 정확도을 보이지만 논문 형태가 추출 규칙과 상이하면 성능이 크게 낮아질 수 있다.

학술문서에서 벡터(vector) 형태의 그림을 추출하기 위해 기계학습을 적용하려는 연구[3]도 진행되고 있다. 벡터 그래픽은 PDF 파일과 동일한 형태로 저장되기 때문에 그림의 추출이 어렵다. 해당 연구는, 곡선 등 벡터 이미지를 구성하는 세부 객체 및 객체들의 위치를 기계학습으로 파악하고 군집화를 통해 특정 이미지의 영역을 탐지한다. 정확도가 80% 이상인 것으로 보고되었지만 연구논문이 이미지일 경우에는 객체 모델로 표현되지 않기 때문에 추출이 불가능한 문제가 있다.

본 논문은 객체 추출의 유연성(flexibility) 확보를 위해 텐서플로우 객체 탐지 API[21]를 활용했다. 텐서플로우 객체 탐지 API는 심층 신경망을 이용하는 오픈소스 프레임워크로써 이미지에 포함된 다수의 객체를 탐지하기 위해 고안되었다. 다양한 서식의 연구논문들을 이미지 형태로 변환한 후 추출하고자 하는 객체에 주석(annotation)을 달아 심층 신경망을 학습시킬 수 있다. 텐서플로우 객체 탐지 API는 복잡한 심층 신경망을 직접 구현하지 않더라도 다양한 객체 탐지 분야에 적용될 수 있도록 총 5 개의 학습 모델을 제공하고 있다.

3. 심층 신경망 기반의 객체 탐지 및 추출

3.1 객체 추출 시스템의 개요

본 논문은 전자문서로부터 그림(figure), 표(table), 캡션

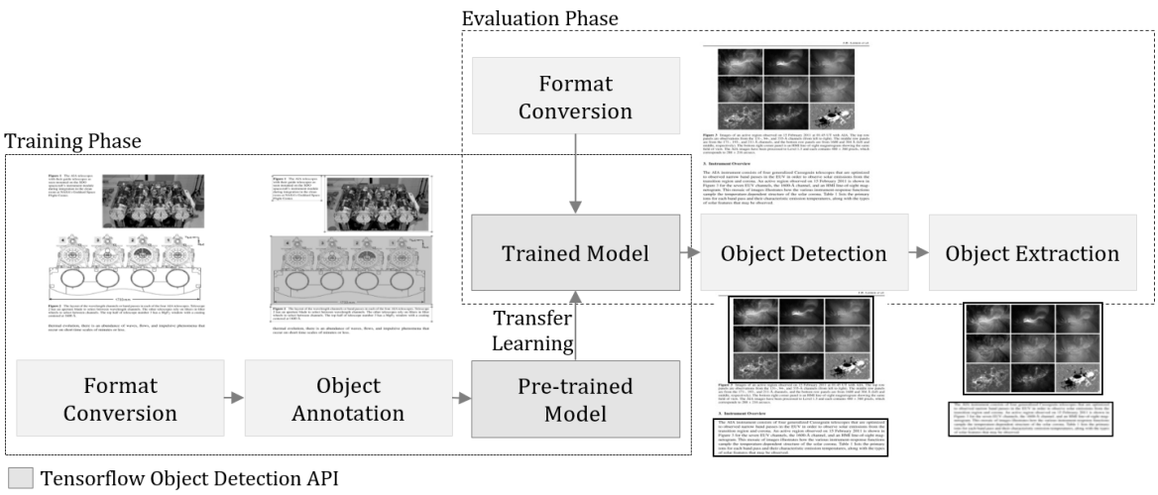


Fig. 1. Overview of the Proposed System

(caption)의 영역을 탐지하고 탐지된 객체를 자동으로 저장하기 위한 심층 학습 기반의 객체 추출 시스템을 제안한다. 제안된 시스템은 텐서플로우 객체 탐지 API를 이용해 심층 신경망을 학습한 후, 입력된 전자문서로부터 개별 객체의 종류와 위치 정보를 추출한다.

심층 학습과정을 통해 전자문서로부터 목표 객체(그림, 표, 캡션)를 추출하는 과정은 Fig. 1과 같다. 추출 과정은 학습 단계(training phase)와 추론 단계(evaluation phase)로 구분된다. 학습 단계에서는 전자문서의 형태 변환(format conversion), 주석 처리(object annotation) 등과 같은 전처리 과정을 통해 학습 집단(training set)을 생성하고 심층 학습을 수행함으로써 학습 모델(trained model)을 완성한다. 추론 단계에서는 생성된 학습 모델을 이용해 평가 집단(evaluation set)에 속한 목표 객체를 탐지하고 개별 객체들의 상세 정보를 추출한다. 형태 변환 단계에서는 전자문서의 개별 페이지들을 PNG(Portable Network Graphics) 형태의 이미지 파일로 변환한다. 제안하는 시스템은 이미지 인식 기술을 활용하기 때문에 이미지 파일을 시스템의 입력으로 받는다. 주석 처리 단계에서는 생성된 이미지에 포함된 그림, 표, 캡션 객체를 분류(labeling)하고 주석 정보를 생성한다. 가중치 학습을 위해 이미지 원본(전자문서의 개별 페이지)과 주석 정보, 텐서플로우 객체 탐지 API를 이용한다.

본 논문에서는 학습 모델의 생성을 위해 ResNet-101[22] 구조의 Faster R-CNN을 이용하며 새로운 학습 모델은 전이 학습(transfer learning)을 통해 생성된다. 전이 학습은 사전에 학습된 모델(pre-trained model)의 최종 가중치 값을 이용해 새로운 모델의 초기 가중치 값을 설정한 후, 추가적인 학습 집단을 대상으로 재학습을 수행함으로써 새로운 학습 모델을 생성하는 방법이다. 전이 학습을 이용하면 성능이 검증된 기존 모델의 설정값을 재활용함으로써 한정된 학습 집단으로부터 효율적으로 새로운 모델을 생성할 수 있다. 본 논문에서는 전이 학습을 기반으로 객체 추출을 위한 새로운 학습 모델(trained model)을 제시한다. 추론 단계에서는 대상이 되는 전자문서를 이미지 형태로 변환한 후, 심층 학습을 통해 생성된 학습 모델을 활용하여 목표 객체를 탐지(object detection)한다. 최종적으로, 객체 탐지를 통해 획득한 개별 객체의 종류와 영역정보를 기반으로 해당 객체를 추출(object extraction)한 후, 그림 파일 형태로 저장한다.

다음 절부터 객체 추출을 위한 심층 학습 과정의 세부 절차를 상세히 설명한다.

3.2 학습 집단의 확보 및 전처리

심층 학습을 위한 학습 집단은 논문 서식과 연구 분야, 출판 연도 등을 고려해 선택했다. 학습 집단에 포함된 객체의 통계는 Table 1과 같다.

Table 1. The Number and Type of Objects in a Training Set

Phase	Pages	Objects		
		Figure	Table	Caption
Training	647	351	111	462
Validation	72	43	12	55
Total	719	394	123	517

전자문서로 저장된 49 개의 연구논문과 1 개의 연구서적이 학습(training) 및 검증(validation)의 대상이다. 선택한 50 개의 전자문서는 총 719 페이지로 이루어져 있다. 제 4장에서 이용된 평가 집단(evaluation set)과 본 절의 학습 집단은 별개의 데이터 집단이다. 학습 집단에 포함된 검증용 부집단(subset)은 생성된 학습 모델의 정확도를 검증하기 위해 이용된다. 검증용 부집단은 학습 집단의 10%로 설정했다.

학습 집단에 속한 전자문서(PDF)들의 각 페이지는 개별 PNG 파일로 저장된다. 전자문서를 이미지 파일로 변경하기 위해 공개 소프트웨어인 pdftoppm을 이용하였다. pdftoppm은 전자문서들의 각 페이지를 개별 레스터(raster) 이미지로 변환해 저장할 수 있다. 변환된 이미지 파일에 대한 주석은 공개 소프트웨어인 LabelImg를 이용해 수동으로 생성했다.

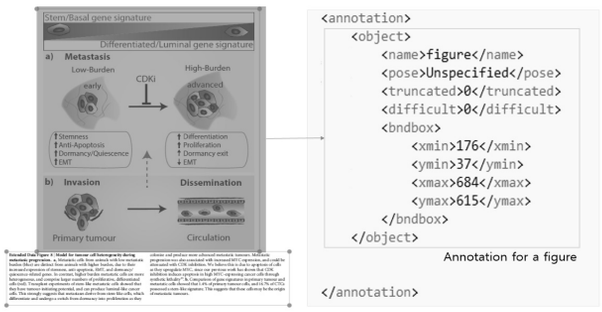


Fig. 2. Example annotation (Reproduced from Lawson et al. Nature 2015;526(7571):131-5, with Permission of Springer Nature[23])

Fig. 2와 같이 GUI(Graphical User Interface) 상에서 목표 객체에 대해 테두리 상자(bounding box)를 설정하고 해당 객체를, 예를 들어, 그림으로 분류하면, 객체의 영역 정보와 종류가 XML 파일에 저장된다. 본 논문에서는 그림, 표, 캡션을 각각 독립된 객체로 주석 처리한다. 테두리 상자의 설정 범위(예, 그림과 캡션의 분리 또는 통합 등) 또는 목표 객체의 추가(예, 본문, 수식, 참고문헌 등) 등이 제안하는 시스템의 성능에 영향을 줄 수 있다. 추가 연구를 통해 주석처리와 추출 성능 간의 상관관계를 분석할 예정이다.

마지막으로, 학습 집단은 학습용 부집단과 검증용 부집단으로 구분된다. 텐서플로우가 심층 학습을 할 수 있도록 개별 부집단에 대한 각각의 TFRecord 파일을 생성해야 한다. TFRecord는 텐서플로우의 심층 학습을 위한 이진 데이터로써 인코딩된 이미지, 주석 내용(객체의 위치 및 종류), CRC (Cyclic Redundancy Check)를 포함한다. TFRecord 파일을 생성하기 위한 의사 코드는 Fig. 3과 같다.

```
CREATE_TF_CONTENT(image, annotation)
{
  CONVERT image TO encoded-image;
  PARSE annotation;

  FOREACH object
    BUILD bounding-box(xmin, ymin, xmax, ymax);
      ADD bounding-box TO bounding-box list;
      ADD object type TO class list;
  BUILD tf_content WITH encoded-image,
    bounding-box list, class list;
  RETURN tf_content;
}

GENERATE_TFRECORD(images, annotations)
{
  FOREACH image
    BUILD tf_content = CREATE_TF_CONTENT(image,
                       annotation);
    WRITE tf_content;
}
```

Fig. 3. Pseudo Code to Generate a TFRecord File

3.3 심층 학습 모델 및 학습 환경

심층 학습을 위한 새로운 알고리즘의 제안은 본 논문의 연구 범위를 넘어선다. 본 논문은 적용 가능한 학습 모델을 이용해 전자문서로부터 목표 객체를 추출하는데 있으며 Faster R-CNN의 전이 학습을 통해 새로운 학습 모델을 생성한다. 객체 탐지의 정확도와 높은 학습 속도[24]를 보장하기 위해 Faster R-CNN을 활용했다. Faster R-CNN은 R-CNN에서 이미지 분류와 영역 조율을 각각 담당하는 SVM과 선형 회귀(linear regression) 모델을 CNN 네트워크에 수용하고, 영역 제안(region proposal)을 위한 선택적 탐색(selective search [25]) 알고리즘을 영역 제안 네트워크(region proposal network)로 대체함으로써 R-CNN의 성능을 크게 개선한 심층 학습 알고리즘이다.

Table 2. Deep Learning Environment for Object Detection

Component	Description
CPU/MEM	2 x Intel Xeon E5-2637 v4 3.5GHz/128GB
GPU	NVIDIA GeForce GTX 1080 TI
Deep learning framework	Tensorflow-gpu-1.4.0

심층 학습을 위한 서버 환경은 표 2와 같다. Nvidia™ GTX 1080 Ti GPU(Graphical Processing Unit) 카드가 설치된 우분투 리눅스 상에서 텐서플로우 1.4.0 버전을 이용한다.

Table 3. Deep Learning Parameters

Hyper-parameter	Value	Hyper-parameter	Value
epoch/batch size	200,000/1	Momentum optimizer	0.9
Initial learning rate	0.0003	Moving average	false

Table 3은 심층 학습을 위한 개별 매개변수들의 설정 값을 보여준다. 전체 학습 집단에 대해서 200,000번 반복 학습(epoch/batch size)하며 학습 속도의 초기값은 0.0003이다. 모멘텀 최적화기(Momentum optimizer[26])는 0.9로 설정했으며 이동 평균(moving average)은 이용하지 않았다. 매개변수들의 최적화 문제는 추가 연구를 통해 해결할 예정이다. 마지막으로 심층 학습은 총 975분 동안 수행되었다.

학습된 모델의 성능 지표로써 겹침 공간(IoU, Intersection over Union)에 대한 평균 정확도(average precision)를 이용한다. 겹침 공간은 학습을 위해 사전에 주석 처리된 특정 객체의 영역과 학습 모델을 통해 추출한 해당 객체의 영역이 겹치는 비율이다. 임계값이 t 이면, 겹침 공간의 비율이 t 이상일 때만 객체 추출에 성공한 것으로 판단한다.

Fig. 4는 검증용 부집단을 대상으로 측정한 평균 정확도를 보여준다. 앞 절에서 기술했듯이 학습 집단의 10%(검증용 부집단)가 학습 결과를 검증하기 위해 활용되었다. 겹침 공간의 임계값이 0.7인 경우, 모든 목표 객체에 대한 평균 정확도가 92.02%를 보였다. 측정된 높은 정확도를 통해 효과적인 객체 추출이 가능할 것으로 유추할 수 있다.

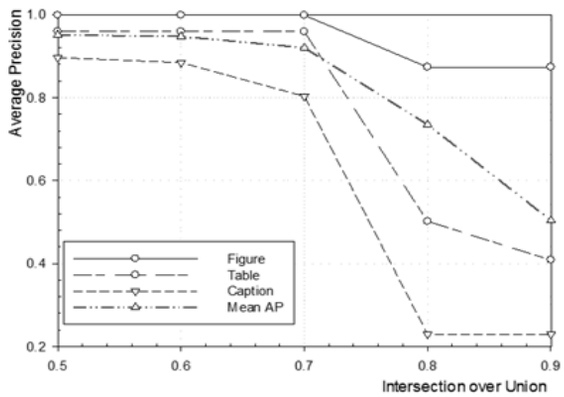


Fig. 4. Average Precision for a Validation Set

겹침 공간의 임계값이 0.8 이상일 경우, 그림에 비해 표와 캡션의 평균 정확도가 낮아지는 것을 확인할 수 있다. 일반적으로 표와 캡션 객체의 크기가 그림에 비해 상대적으로 작기 때문에 낮은 정확도를 나타내는 것으로 판단된다. 예를 들어, 주석 처리된 목표 객체의 영역과 추출된 객체의 영역 간에 절대치 오차가 작더라도 주석 처리된 목표 객체의 영역이 작으면 겹침 공간의 비율과 평균 정확도가 상대적으로 낮아진다.

4. 성능 평가

본 장에서는 평가 집단과 성능평가 지표에 대해서 살펴본 후 제안된 객체 추출 시스템과 경험적 방법론이 적용된 PDFFigures의 성능을 비교 분석한다.

4.1 평가 집단 및 평가 방법

객체 추출 시스템과 PDFFigures의 성능평가를 위해 학습 집단에 포함되지 않은 새로운 50개의 연구논문을 수집함으로써 평가 집단(evaluation set)을 구성하였다. 연구논문의 발행 시기, 연구 분야, 작성 언어, 파일 내 이미지의 표현 방식 등 정보 추출 성능에 영향을 줄 수 있는 다양한 요소들을 고려하여 총 39개 출판사가 발행한 전자저널로부터 전자문서 형태의 연구논문을 선택하였다.

Table 4. The Number and Type of Target Objects in the Evaluation Set

Papers	Pages	Objects		
		Figure	Table	Caption
50	593	349	125	474

평가 집단에 포함된 객체 별 총 개수는 Table 4와 같다. 선택된 전자문서의 92%는 영문이며 8%가 한국어 또는 프랑스어로 작성되어 있다.

논문의 발행 시기와 전자문서에 포함된 이미지의 표현 형태는 Table 5와 같다. 1990년대 논문은 전체 평가 집단 중 10%에 해당되며 2000년대와 2010년대에 발행된 문헌이 90%로 다수를 차지하고 있다. 평가 집단의 약 52%에 해당하는 전자문서들이 레스터 이미지만 포함하고 있으며 약 12%의 전자문서에는 벡터 이미지만 삽입되어 있다. 동일한 전자문서에서 벡터(vector)와 레스터(raster) 이미지를 모두 가지고 있는 경우는 24%이다. 경험적 방법론을 이용할 경우, 전자문서에 독립된 객체로 삽입되는 레스터 이미지는 전자문서와 동일한 형태로 저장되는 벡터 이미지보다 쉽게 추출할 수 있다[3]. 평가 집단 중 약 12%의 전자문서에서는 사용된 이미지 형태를 확인할 수 없었다.

Table 5. Issued Year of Articles and Graphic Format of PDF Files

Issued date	Percentage	Image format	Percentage
1990s	10	Raster	52
2000s	42	Vector	12
2010s	48	Both	24

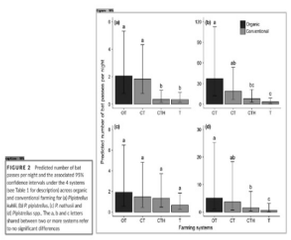
제안한 객체 추출 시스템과 PDFFigures의 성능을 비교하기 위해 검출률(recall), 정확도(precision) 및 $F1$ 지수를 평가 지표로 선정했다. 겹침 공간의 비율이 75% 이상이면 정상적으로 추출한 것으로 처리한다. 수동으로 주석 처리된 객체의 영역은 해당 객체가 차지하고 있는 실제 영역보다 크기 때문에 겹침 공간의 비율이 상대적으로 낮아도(즉, 75%) 정상적인 객체 추출이 가능하다. 또한 간단한 후처리 작업을 통해서 최종 추출 객체의 겹침 공간 비율을 더 높일 수 있을 것으로 판단한다.

검출률(R)은 전자문서에 포함된 목표 객체의 수에 대비해서 실제로 추출된 목표 객체의 비율이다. 정확도(P)는 추출된 객체의 수와 비교해 추출된 목표 객체 수의 비율로 정의한다. 추출된 객체가 목표 객체가 아닐 수 있기 때문에 정확도는 오탐을 포함한다. 검출률과 정확도를 포함하는 종합적인 성능 지표로 $F1$ 지수가 이용되었다. $F1$ 지수는 검출율과 정확도의 조화 평균(harmonic mean)이며 다음과 같이 정의된다.

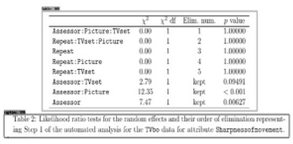
$$F1 = \frac{2RP}{R+P} \tag{1}$$

4.2 평가 결과 및 분석

PDFFigures는 평가 집단 중 1 건의 전자문서를 판독하는데 실패했으며 4 건의 전자문서에 대해서는 목표 객체를 전혀 추출하지 못했다. 결과적으로 평가 집단에 속한 전자문서, 총 593 페이지 중 88.6%의 페이지에 대해서만 하나 이상의 목표 객체를 추출할 수 있었다. 하지만 본 논문에서 제안한 객체 추출 시스템은 평가 집단의 모든 전자문서에서 하나 이상의 목표 객체를 추출할 수 있었다. 제안한 시스템이 전자문서의 저장형태, 작성언어, 전자문서에 포함된 객체의 그래픽 형태 등에 제한을 받지 않기 때문에 경험적 방법들에 비해 보다 유연하게 적용될 수 있음을 알 수 있다.



(a) Correctly extracted Figure and Caption



(b) Correctly extracted Table and Caption

Fig. 5. Correctly Extracted Figure, Table, and Caption ((a) Reproduced from Barre et al. Ecol Evol 2018;8(3): 1496-1501[27], (b) Kuznetsova et al. J Stat Softw 2017;82(13):1-26[28])

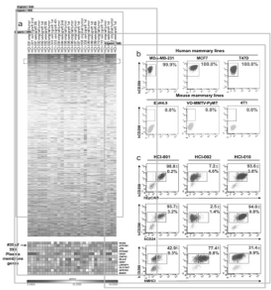
Fig. 5는 제안된 시스템을 이용해 그림, 표, 캡션 객체를 정상적으로 추출한 예시이다. 탐지된 목표 객체 중 그림은 녹색, 표는 청색, 캡션은 적색으로 표시되며 각 객체의 종류, 영역, 신뢰도를 확인할 수 있다. 신뢰도는 탐지된 객체가 그림, 표 등 특정 종류에 속할 확률이다. 일반적으로 그림에 대한 캡션은 그림의 하단에, 표의 캡션은 상단에 표시된다. Fig. 5를 통해, 제안한 시스템이 일반적인 편집 형태를 갖지 않는 전자문서를 대상으로도 목표 객체들을 성공적으로 탐지할 수 있음을 알 수 있다.

Table 6은 PDFFigures와 제안 시스템의 검출률, 정확도 및 $F1$ 지수를 각각 보여준다. 제안된 시스템은 목표 객체에 대해 평균 84.0%의 검출률을 보임으로써 56.8%의 검출률을 갖는 PDFFigures에 비해서 약 23.2% 높은 성능을 나타냈다. 제안된 시스템이 경험적 방법론이 적용되기 힘든 다양한 형태의 전자저널들을 대상으로 효과적인 객체 추출이 가능함을 의미한다.

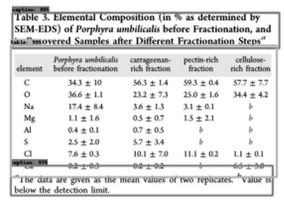
Table 6. Performance Comparison

Method	Index	Figure	Table	Caption	Average
Proposed system	Recall	0.877	0.879	0.765	0.840
	Precision	0.789	0.522	0.761	0.690
	F1	0.830	0.655	0.763	0.749
PDFFigures	Recall	0.590	0.516	0.598	0.568
	Precision	0.924	0.865	0.921	0.903
	F1	0.720	0.646	0.725	0.697

정확도는 PDFFigures가 약 21.3% 높은 성능을 보여주었다. 즉 PDFFigures는 목표 객체에 대한 검출률은 낮지만 일단 검출된 객체는 목표 객체일 가능성이 높다는 것을 알 수 있다. 제안하는 시스템의 정확도가 낮은 이유는 Fig. 6과 같이 오탐이 많이 발생하기 때문이다. 특히 표의 경우에는 논문 개요의 테두리 처리, 직선으로 교차하는 무늬가 포함된 그림, 참조 문헌이나 행렬 연산식 등 반복적 패턴이 격자 형태를 이루는 경우에는 다수의 오탐이 발생했다. 학습 집단에 포함된 표 객체의 수가 그림이나 캡션에 비해 상대적으로 적었기 때문에 표에 대해서는 충분한 학습이 이뤄지지 않은 것으로 판단된다.



(a) Multiple detection for a single object

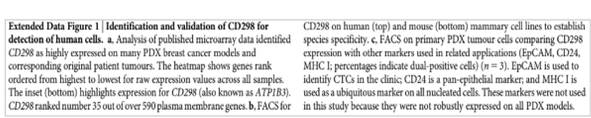


(b) Mispresentation of caption as Caption

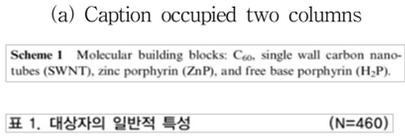
Fig. 6. Incorrect Object Extraction of the Proposed System ((a) Reproduced from Lawson et al. Nature 2015;526(7571): 131-5, with Permission of Springer Nature[23], (b) Wahlstrom et al. Ind Eng Chem Res 2017;57(1):42-53[29])

검출률과 정확도를 고려한 $F1$ 지수는 74.9%와 69.7%로써 제안한 시스템의 성능이 PDFFigures에 비해 약 5.2% 높았다. 상대적으로 낮은 정확도를 개선하기 위해서 탐지된 객체들 간의 군집화, 이진 분류의 적용 등 탐지된 객체의 후처리 방법들에 대해서 추가적으로 연구할 예정이다. 예를 들어 하나의 객체 영역이 중복으로 탐지되는 경우에는 후처리 과정에서 가장 높은 신뢰도를 갖는 객체만 추출하게 함으로써 정확도를 높일 수 있다.

Fig. 7은 PDFFigures가 객체 추출에 실패한 사례를 보여준다. PDFFigures는 추출규칙으로 사용된 예약어가 전자문서의 메타데이터에 존재하지 않으면 객체 추출에 실패했다. Fig. 7의 사례들은 그림과 표를 지칭하는 키워드(Extended Data Figure 1, Scheme, 표)가 추출규칙의 예약어(예, Fig., Table)와 다르기 때문에 객체 추출에 실패한 경우이다. 연구 논문은 연구 분야와 작성언어에 따라서 그림과 표를 지칭하는 키워드가 다를 수 있다. 키워드가 추출규칙에서 벗어난 경우에는 PDFFigures를 이용한 객체 추출이 어렵다는 것을 알 수 있다.



(a) Caption occupied two columns



(b) No use of the keyword “Fig.” or “Table”

Fig. 7. Incorrect object extraction of the PDFFigures ((a) Reproduced from Lawson et al. Nature 2015;526(7571):131-5, with permission of Springer Nature[23])

결론적으로 제안한 객체 추출 시스템은 경험적 방법론을 적용한 PDFFigures와는 다르게 다양한 형태의 전자문서로부터 목표 객체의 추출이 가능했다. 또한 $F1$ 지수를 통해 객체

추출 시스템의 성능이 PDFFigures 보다 뛰어남을 확인했다. 제안 시스템의 정확도를 높이기 위해 오탐과 미탐을 방지할 수 있는 전처리 및 후처리 방법에 대한 추가 연구가 필요할 것으로 판단된다.

5. 결 론

본 논문은 전자문서로부터 그림, 표, 캡션과 같은 목표 객체들을 효과적으로 추출하기 위해서 심층 학습망 기반의 객체 추출 시스템을 제안하고 경험적 방법론을 활용한 PDFFigures를 대상으로 비교평가를 수행하였다. 제안한 시스템은 PDFFigures에 비해서 약 5.2% 높은 성능을 보였으며 전자문서의 편집 형태나 객체의 문서 내 삽입 형태 등에 구애받지 않고 유연하게 목표 객체를 추출할 수 있었다. 하지만 오탐으로 인한 낮은 정확도의 개선은 추가 연구가 필요할 것으로 판단된다. 향후 학습 집단의 확대와 후처리 방법에 대한 연구를 통해 정확도를 높여나갈 예정이다.

References

[1] C. Clark and S. Divvala, “PDFFigures 2.0: Mining figures from research papers,” in *Proceedings of IEEE/ACM Joint Conference on Digital Libraries (JCDL)*, pp.143-152, 2016.

[2] J. Wu et al., “Pdfmef: A multi-entity knowledge extraction framework for scholarly documents and semantic search,” in *Proceedings of the 8th International Conference on Knowledge Capture*, Article No.13, 2015.

[3] S. Ray Choudhury, P. Mitra, and C. L. Giles, “Automatic extraction of figures from scholarly documents,” in *Proceedings of the 2015 ACM Symposium on Document Engineering*, pp.47-50, 2015.

[4] S. J. Chalk, “ChemExtractor: Enhanced Rule-Based Capture and Identification of PDF Based Property Data,” 253rd American Chemistry Society (ACS) National Meeting, 2017.

[5] S. Klampfl and R. Kern, “Machine learning techniques for automatically extracting contextual information from scientific publications,” *Semantic Web Evaluation Challenge*, Springer, pp.105-116, 2015.

[6] P. Lopez, “GROBID: Combining automatic bibliographic data recognition and term extraction for scholarship publications,” in *Proceedings of International Conference on Theory and Practice of Digital Libraries*, pp.473-474, 2009.

[7] M. Aristaran, Extract Tables from PDFs [Internet], http://tabula.technology.

[8] Y. Shinyama, PDFMiner: Python PDF Parser and Analyser [Internet], http://www.unixuser.org/~euske/python/pdfminer/.

[9] Apache PDFBox: A Java PDF Library [Internet], https://pdfbox.apache.org/.

[10] Pdftohtml [Internet], http://pdftohtml.sourceforge.net.

[11] Poppler: a PDF rendering library based on the xpdf-3.0 code base [Internet], https://poppler.freedesktop.org/.

[12] A. E. Jinha, “Article 50 million: an estimate of the number of scholarly articles in existence,” *Learned Publishing*, Vol.23, No.3, pp.258-263, 2010.

[13] 254th American Chemical Society National Meeting and Expo [Internet], http://washingtondc2017.acs.org/t/ 197077-acs-national-meeting-washington-dc-2017.

[14] E. E. Bolton, Y. Wang, P. A. Thiessen, and S. H. Bryant, “PubChem: integrated platform of small molecules and biological activities,” in *Annual reports in computational chemistry*, Elsevier, Vol.4, pp.217-241, 2008.

[15] R. Zakharov, V. Tkacheonko, A. Korotcov, I. Presniakov, and S. Kalmykov, “Open Science Data Repository: The platform for materials research,” 253rd American Chemistry Society (ACS) National Meeting, 2017.

[16] Open Chemistry [Internet], https://www.openchemistry.org/.

[17] S. Ren, K. He, R. Girshick, and J. Sun, “Faster R-CNN: towards real-time object detection with region proposal networks,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol.39, No.6, pp.1137-1149, 2017.

[18] M. Lipinski, K. Yao, C. Breitinger, J. Beel, and B. Gipp, “Evaluation of header metadata extraction approaches and tools for scientific PDF documents,” in *Proceedings of the 13th ACM/IEEE-CS Joint Conference on Digital Libraries*, pp.385-386, 2013.

[19] P. Lopez and L. Romary, “HUMB: Automatic key term extraction from scientific articles in GROBID,” in *Proceedings of the 5th International Workshop on Semantic Evaluation*, pp.248-251, 2010.

[20] I. G. Councill, C. L. Giles, and M.-Y. Kan, “ParsCit: an Open-source CRF Reference String Parsing Package,” in *Proceedings of the Language Resources and Evaluation Conference (LREC 08)*, Vol.8, pp.661-667, 2008.

[21] TensorFlow Object Detection API [Internet], https://research.googleblog.com/2017/06/.

[22] K. He, et al., “Deep residual learning for image recognition,” in *Proceedings of 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp.770-778, 2016.

[23] D. A. Lawson, et al., “Single-cell analysis reveals a stem-cell program in human metastatic breast cancer cells,” *Nature*, Vol.526, No.7571, pp.131-135, 2015.

[24] J. Huang et al., “Speed/accuracy trade-offs for modern convolutional object detectors,” in *Proceedings of 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3296-3305, 2017.

[25] J. R. Uijlings, K. E. Van De Sande, T. Gevers, and A. W. Smeulders, “Selective search for object recognition,”

International Journal of Computer Vision, Vol.104, No. 2, pp.154-171, 2013.

[26] I. Sutskever, J. Martens, G. Dahl, and G. Hinton, "On the importance of initialization and momentum in deep learning," in *Proceedings of International Conference on Machine Learning*, pp.1139-1147, 2013.

[27] K. Barré, et al., "Tillage and herbicide reduction mitigate the gap between conventional and organic farming effects on foraging activity of insectivorous bats," *Ecology and Evolution*, Vol.8, No.3, pp.1496-1506, 2018.

[28] A. Kuznetsova, P. B. Brockhoff, and R. H. Christensen, "lmerTest package: Tests in linear mixed effects models," *Journal of Statistical Software*, Vol.82, No.13, pp.1-26, 2017.

[29] N. Wahlström, et al., "A Strategy for the Sequential Recovery of Biomacromolecules from Red Macroalgae Porphyra umbilicalis Kützing," *Industrial & Engineering Chemistry Research*, Vol.57, No.1, pp.42-52, 2017.

장 희 진

https://orcid.org/0000-0001-9629-9251
e-mail : jhj@kisti.re.kr
2001년 포항공과대학교 컴퓨터공학과(학사)
2003년 포항공과대학교 컴퓨터공학과(석사)
2003년~2009년 삼성종합기술원 전문연구원
2010년~2011년 국가보안기술연구소 연구원
2012년~현 재 한국과학기술정보연구원 과학기술연구망센터 선임연구원
관심분야 : Federated Identity Management, 기계학습

채 영 훈

https://orcid.org/0000-0002-6860-7533
e-mail : proin@kisti.re.kr
2015년 고려대학교 전자및정보공학과(학사)
2017년 과학기술연합대학원대학교(UST) 빅데이터과학(석사)
2012년~현 재 한국과학기술정보연구원 과학기술연구망센터 연구원
관심분야 : Federated Identity Management, 심층학습

이 상 원

https://orcid.org/0000-0002-8283-4572
e-mail : lsw91@kaist.ac.kr
2014년 서울시립대학교 화학공학과(학사)
2016년 서울시립대학교 화학공학과(석사)
2016년~현 재 한국과학기술원 생명화학공학과 박사과정
관심분야 : 계산화학, 기계학습

조 진 용

https://orcid.org/0000-0001-6830-3604
e-mail : jiny92@kisti.re.kr
1999년 전남대학교 컴퓨터공학과(학사)
2002년 광주과학기술원 정보통신학과(석사)
2013년 광주과학기술원 정보통신학과(박사)
2012년~현 재 한국과학기술정보연구원 과학기술연구망센터 선임연구원
관심분야 : Software Defined Networking, Federated Identity Management