DeepPDF: A Deep Learning Approach to Extracting Text from PDFs

Christopher Stahl, Steven Young, Drahomira Herrmannova, Robert Patton, Jack Wells
Oak Ridge National Laboratory
Oak Ridge, TN, USA
{stahlcg, youngsr, herrmannovad, pattonrm, wellsjc}@ornl.gov

Abstract

Scientific publications contain a plethora of important information, not only for researchers but also for their managers and institutions. Many researchers try to collect and extract this information in large enough quantities that it requires machine automation, but because publications were historically intended for print and not machine consumption, the digital document formats used today (primarily PDF) have created many hurdles for text extraction. Primarily, tools have relied on trying to convert PDF documents to plain text for machine processing by reverse engineering the PDF standard. This in itself is a complex process because once a PDF is created it is more closely related to an image file than a document markup language. However, while a number of tools exist, which can extract the contents of a PDF with acceptable accuracy, correctly labeling and piecing together the extracted data to form blocks of text or even sections is a significantly harder task. In this paper we explore the feasibility of treating these PDF documents as images as opposed to a proprietary markup language. We believe that by using deep learning and image analysis we can create more accurate tools for extracting information from PDF documents than those that currently exist.

Keywords: deep learning, text extraction, information extraction, PDF extraction, scholarly publications

Acknowledgments

This manuscript has been authored by UT-Battelle, LLC and used resources of the Oak Ridge Leadership Computing Facility at the Oak Ridge National Laboratory under Contract No. DE-AC05-00OR22725 with the U.S. Department of Energy. The United States Government retains and the publisher, by accepting the article for publication, acknowledges that the United States Government retains a non-exclusive, paid-up, irrevocable, world-wide license to publish or reproduce the published form of this manuscript, or allow others to do so, for United States Government purposes. The Department of Energy will provide public access to these results of federally sponsored research in accordance with the DOE Public Access Plan[1].

1. Introduction

Scientific publications have a wealth of information that can be useful for research, making management decisions, evaluating impact, etc. But often times this data is currently locked behind the PDF standard. While tools do exist to extract text and other information from PDF documents, the resulting output often falls short of the demands of researches. Captions, figures, tables, header, and footer data are among some of the features that cause problems for traditional PDF to text extraction methods, as well as for tools for extracting information from PDFs built on top of these methods. Because PDF is an image format, not a text format, there are no requirements for how text is to be rendered on a page. This causes traditional PDF to text extraction tools such as PDFBox[2] and PDFMiner[3] to have to best guess the correct order of text during extraction. A number of tools, such as GROBID[4] and ParsCit[5], exist which try to overcome this issue and re-order and correctly classify sections in the output document. The lack of document structure makes it hard for these programs to accurately extract double/triple column papers and other difficult formats. Even when text is extracted properly the resulting plain text file is still often littered with header/footers, tables, figures data, etc. In many cases researchers are now using this noise filled plain text file to train their algorithms and machine learning networks, introducing unwanted noise to the pipeline.

In this publication we explore the possibility of creating PDF extraction tools that treat PDF documents as the images they truly are. We believe that scientific publications have inherent structure that is easy for humans familiar with them to separate. When presented with an image of a fully redacted publication researchers can visually determine the difference between a title, paragraph, reference section, headers, etc. Using the idea that PDF structure is a trainable idea, we theorize that a deep learning network can also be trained to separate the different sections of publications. Being able to separate out the different sections of publications is important because it will allow tools to accurately provide raw text versions of individual sections, as well as other portions of the document, without the noise created by traditional methods. To test this idea, we have utilized a set of 50 publications from PubMed (a subset of the *PMC_Sample_1943* dataset (Constantin et al., 2013)), which we have manually annotated (Section 3.), resulting in 407 labeled pages. We have then trained a Deep Neural Network to identify body text in the input (Section 4.). Our evaluation shows this approach offers high accuracy in correctly identifying body text while correctly rejecting other elements such as footers and captions (Section 5.).

The goal of information extraction from scientific docu-

[1] http://energy.gov/downloads/doe-public-access-plan

[2] https://pdfbox.apache.org/

[3] http://www.unixuser.org/~euske/python/pdfminer/

[4] https://github.com/kermitt2/grobid

[5] https://github.com/knmnyn/parscit

ments is, very broadly, to produce structured output from the input document. This is somewhat different from simple PDF to text conversion (such as using PDFBox or OCR methods), which requires accurate extraction of characters found in the PDF, but not identification of the structure of the text performed by ParsCit, GROBID, and others (such as recognition of titles, paragraphs, captions, etc. as such). PDF to text conversion is a necessary step in information extraction from scientific documents and the two tasks can be done in any order. For example, (Ramakrishnan et al., 2012) first identify sections visually in the input document before extracting relevant sections. On the other hand (Lopez, 2009) first performs PDF to text extraction before processing the text output to identify sections. In this work we focus specifically on the task of identifying sections in the input PDF, methods for correctly extracting content of these sections are out of the scope of this paper.

Furthermore, different extraction methods have focused on different sections. For example, (Councill et al., 2008) focus on extraction of references and citation contexts, (Tkaczyk et al., 2014) mainly on header information (title, authors, affiliations, etc.), and (Siegel et al., 2018) on images and captions. While in this paper we focus on extracting the text content of scientific articles (i.e. the body of the article containing the description of the study rather than header information, references, or other parts of the document), our method can be easily trained to extract any or all sections (including tables and images) from scientific documents.

2. Related Work

In this section, we review previous literature relevant to our study, which we categorize according to the method used. First, we discuss methods for automated extraction of information from research articles which use traditional machine learning models such as Conditional Random Fields. Next, we discuss methods which have leveraged information about document layout and other stylistic information (e.g. font sizes) to support the extraction. Finally, we focus on methods which have utilized Deep Learning for this task.

2.1. Traditional methods

The analysis of the structure of documents has been studied for a number of years ((Mao et al., 2003) have provided a survey of some earlier approaches) and a number of freely available tools currently exist which can be used to extract information from scientific documents. These include ParsCit[6] (Councill et al., 2008), GROBID[7] (Lopez, 2009), CERMINE[8] (Tkaczyk et al., 2014), and most recently OCR++[9] (Singh et al., 2016). While previous approaches utilized models such as Hidden Markov Models (HMM) and Support Vector Machines (SVM) (Peng and McCallum, 2004), most of the current tools, such as ParsCit (Councill et al., 2008) and GROBID (Lopez, 2009), utilize Conditional Random Fields (CRF). CRFs are undirected graphical models trained to maximize a conditional probability (Peng and McCallum, 2004) which can be used to segment and label sequence data (Lafferty et al., 2001).

For example, ParsCit uses a CRF model to process reference strings to identify parts such as author, title, and venue information (Councill et al., 2008). The authors have used several heuristics to identify the reference section and to split the section into separate references. The CRF model is then applied to the separate reference strings. The authors also use heuristics and regular expressions to extract the context in which each reference was mentioned in the text. Following the approach of (Peng and McCallum, 2004), GROBID used CRFs for both header and reference string parsing (Lopez, 2009). CERMINE combines several models, mainly SVM which is used to identify zones (header, body, references, other) in the input text, and CRF which is used for parsing reference strings (Tkaczyk et al., 2014). OCR++ also uses CRFs (Singh et al., 2016). The tool uses several separate CRF models and combines them with handwritten heuristics.

While a number of the existing tools report good performance in various tasks, this does not always reflect performance under real-life conditions (such as when working with older documents). For example, (Beel et al., 2013) have reported that while many tools claim around 90% accuracy on title extraction, their experiments with existing tools have resulted in accuracies between 50% to 70%. Furthermore, (Lipinski et al., 2013) have conducted a comparative evaluation of several tools including GROBID, which has revealed poorer performance on abstract extraction. While they haven't analyzed body text extraction performance, this tends to be complicated due to a number of factors such as difficulties in detecting correct flow of text in two column layouts, page numbers, journal information, and footers being incorrectly incorporated into the body text, etc. As we are particularly interested in the body text extraction, in this paper we focus on methods for correctly extracting body text from scientific articles.

2.2. Layout-aware methods

A number of studies have made use of stylistic information, such as font size and position on the page. While different publishers employ different formatting styles, many elements withing scientific publications are formatted similarly. The information about formatting of different elements can be utilized to identify the element type. For example, publications typically begin with a title, which is also usually presented with the largest font. These two simple formatting rules have been used to produce Docear's PDF Inspector [10] (Beel et al., 2013), and have been shown to identify titles with fairly high accuracy (>70%). (Clark and Divvala, 2016) have utilized spacing between elements in a publication to first identify separate blocks (figures, tables, blocks of text, etc.) and then used heuristics about empty spaces to decide which piece of text is a figure caption. (Ramakrishnan et al., 2012) have used a similar method to first identify continuous text blocks which are

[6] http://parscit.comp.nus.edu.sg/

[7] cloud.science-miner.com/grobid/

[8] cermine.ceon.pl/

[9] www.cnergres.iitkgp.ac.in/OCR++/home/

[10] github.com/Docear/PDF-Inspector

Tag	RGB Color Code		
Abstract	255	128	0
Acknowledgement	179	179	179
Author Information	153	102	51
Copyright	128	128	0
Figure	255	0	255
Figure Caption	128	0	128
Header	255	0	0
Keywords	255	255	0
Paper-meta Data	128	0	0
Paragraphs	0	255	255
Reference Section	0	255	0
Title	0	0	255

Table 1: Content types and their corresponding RGB color code.

then labeled and stitched together to form sections. In contrast to (Ramakrishnan et al., 2012), (Klampfl et al., 2014) first utilize a PDF to text extraction library and then use information about layout, font, and reading order to label extracted blocks. These approaches have demonstrated the benefits of utilizing layout information. However, in many cases, the layout information was incorporated into the extraction though hand-written heuristics. Because we treat each page in a document as an image and utilize deep learning to perform segmentation of the image, we are able to leverage layout information without having to hand-craft features.

2.3. Deep Learning methods

In recent years, neural networks have become the state-of-the-art in a variety of computer vision tasks (LeCun et al., 2015). These networks consist of neurons arranged in series of layers, which learn to recognize successively higher-level representations. This can be advantageous in processing scholarly documents where the layout of the document and other stylistic information have been shown to support and improve extraction (Section 2.2.). To the best of our knowledge, only one previous study has treated PDF documents as images and leveraged Deep Learning for this task (Siegel et al., 2018). They have utilized a modified version of the *ResNet-101* network to extract figures and captions from scientific documents. In contrast to this work, we focus on body text identification.

3. Data Collection

The *PMC_Sample_1943* dataset compiled by Alexandru Constantine was selected for this project[11] (Constantin et al., 2013). This dataset consists of 1943 publications selected from 1943 different journals in the Pubmed repository. For the initial testing we selected a random sample of 50 documents, giving us a total of 407 pages of publication data. Each section of the publication was assigned an RGB color code as described in Table 1.

[11] `https://grobid.s3.amazonaws.com/PMC_sample_1943.zip`

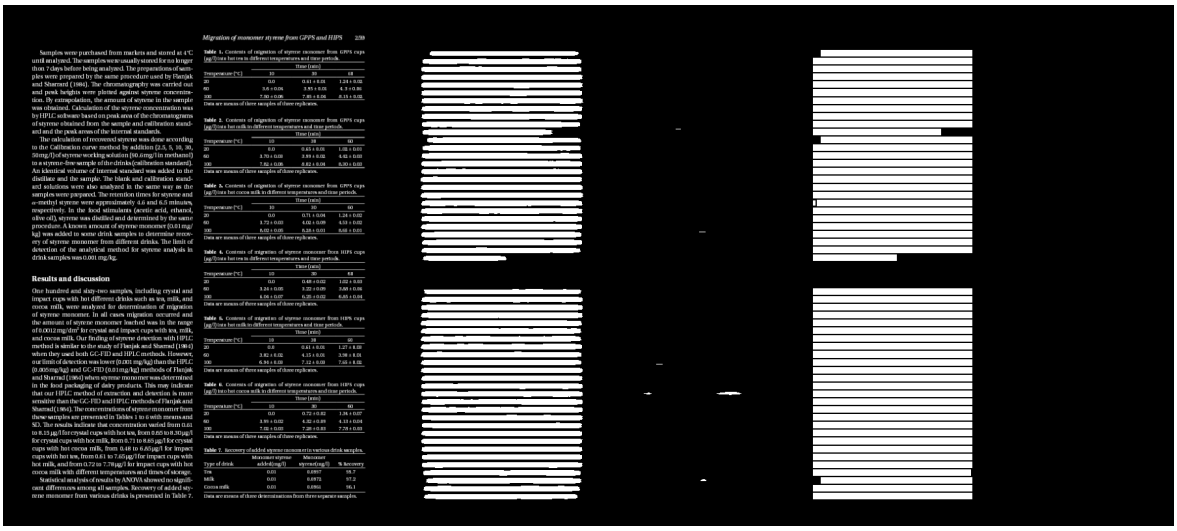
Figure 1: Example of redacted document where redaction color indicates the type of content for each pixel.

Using Adobe Acrobat's redaction tool, a researcher manually applied redactions to each section of the PDF documents as shown in Figure 1. A copy of the PDF was then saved giving us a redacted or masked version of the document and the original document itself. The original document is converted to a grayscale PNG, while the masked document is converted to an RGB PNG. While this process for redaction is highly accurate, it is manually intensive. We believe that while there are many different journals and publication venues, scientific publications cluster into a much smaller number of visual differences. Because of this we believe that a smaller dataset can be used for proof of concept and that results can be achieved with a much smaller data set than was is generally required for image analysis problems.

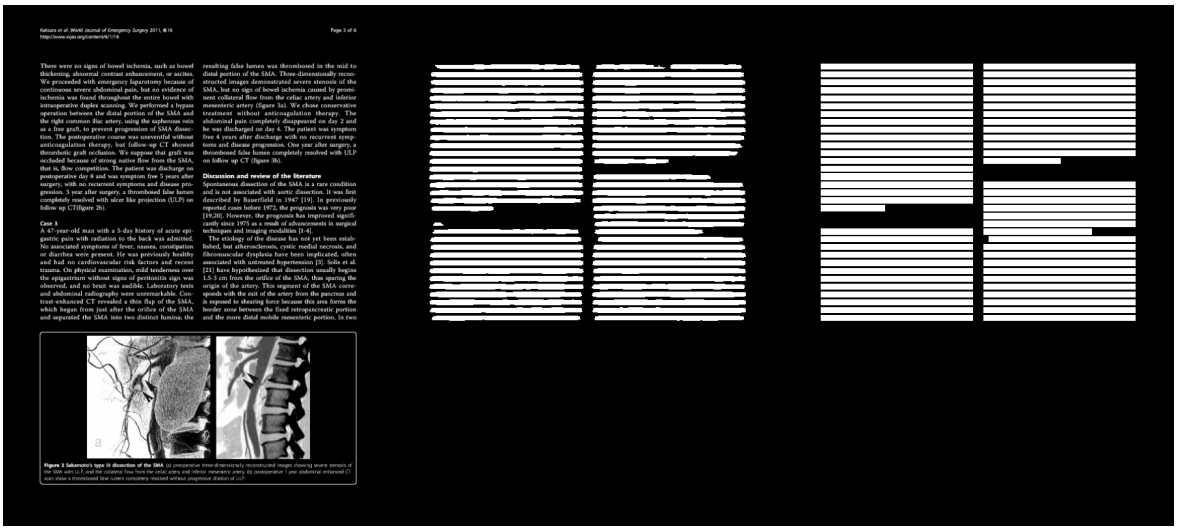
4. Methods

As the data has been processed such that we have an image of each page of each PDF and a pixel-wise label for the type of content corresponding to each pixel, the problem is naturally set up for semantic segmentation. Semantic segmentation is the process of assigning a label to each pixel of an image. A popular network for semantic segmentation tasks is *U-Net* (Ronneberger et al., 2015), and this is the network architecture we chose to utilize in this work. This network was chosen as it typically provides good performance with relatively few training examples.

We used a *U-Net* implementation available at `https://github.com/shreyaspadhy/UNet-Zoo` and the network is trained using softmax cross entropy loss. For this paper, we are only exploring a two class problem where one class is "paragraphs" and the other is "not paragraphs." The former is defined as the main text of the paper, and the latter includes titles, authors, author information, blank space, figures, tables, references, abstracts, etc. We target



(a) Example of properly rejecting tables.



(b) Example of properly rejecting figure and caption.

Figure 2: (Left) Input image. (Center) Network output. (Right) Ground truth.

this problem since existing PDF text extraction tools often fail at separating the main text from these other text within the document, making the result challenging to read (Section 1.).

5. Results

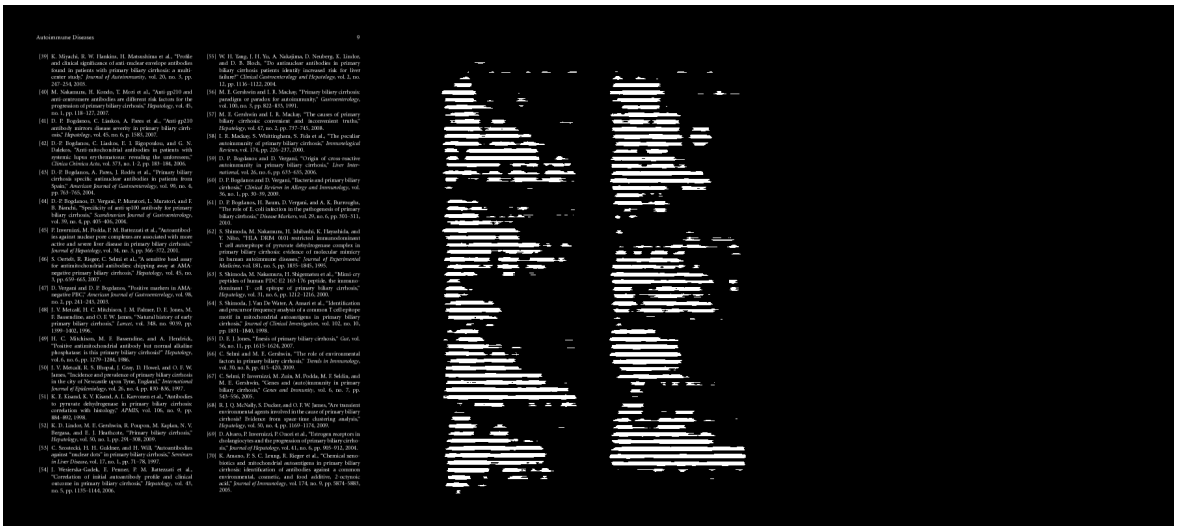
In order to evaluate this method, we split the 407 pages of publications in to 366 training examples and 41 validation examples. Figures 2 and 3 provide validation examples of an input provided to the network, examples of network output, and ground truth target output. These results demonstrate impressive results with such a small dataset. In particular, the network is able to reject header and footer text extremely reliably. The network rejects most abstracts, figure captions and references, confusing only some where the text formatting is extremely similar to typical paragraph text. The per pixel classification accuracy on the validation set was 94.32%, compared to a baseline of classifying each pixel as "not paragraph" which would provide 79.67% accuracy.

6. Conclusion and Future Work

In this paper we demonstrated that deep learning-based image analysis can be used to identify sections of scientific publications. Given the results from our current experiments, we feel that deep learning can be successfully used to enhance current PDF extraction methods, and based on our findings we plan to continue collecting data in order to further increase our networks results, as we feel many of the misclassified portions of text are due to insufficient training data that does not currently characterize features such as reference sections and abstracts sufficiently.

Our current results show that a deep learning network can successfully distinguish and learn the difference between the body text and other portions of a PDF document. The next step is to extend the approach to identifying each type of text (title, author, abstract, body text, etc.) rather than simply body text versus other. Additionally, we plan to increase the accuracy of our network by adding more data and to create an extraction tool that leverages the output of the deep learning network to extract text. While we are currently evaluating accuracy based on a per pixel count of

(a) Example of properly rejecting references.



(b) Example of not properly rejecting references.

Figure 3: (Left) Input image. (Center) Network output. (Right) Ground truth.

estimated versus redacted image, an improved test of accuracy would be to leverage such an extraction tool to identify the per character accuracy of this text extraction approach.

7. Bibliographical References

Beel, J., Langer, S., Genzmehr, M., and Müller, C. (2013). Docear's PDF Inspector: Title Extraction from PDF Files. In *Proceedings of the 13th ACM/IEEE-CS joint conference on Digital libraries*, pages 443–444. ACM.

Clark, C. and Divvala, S. (2016). PDFFigures 2.0: Mining figures from research papers. In *Proceedings of the 16th ACM/IEEE-CS on Joint Conference on Digital Libraries - JCDL '16*. ACM Press.

Constantin, A., Pettifer, S., and Voronkov, A. (2013). Pdfx: fully-automated pdf-to-xml conversion of scientific literature. In *Proceedings of the 2013 ACM symposium on Document engineering*, pages 177–180. ACM.

Councill, I. G., Giles, C. L., and Kan, M.-Y. (2008). ParsCit: an Open-source CRF Reference String Parsing Package. In *LREC*, volume 8, pages 661–667.

Klampfl, S., Granitzer, M., Jack, K., and Kern, R. (2014). Unsupervised document structure analysis of digital scientific articles. *International Journal on Digital Libraries*, 14(3-4):83–99, jun.

Lafferty, J. D., McCallum, A., and Pereira, F. C. N. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *ICML '01 Proceedings of the Eighteenth International Conference on Machine Learning*, pages 282–289.

LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *nature*, 521(7553):436.

Lipinski, M., Yao, K., Breitinger, C., Beel, J., and Gipp, B. (2013). 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 - JCDL '13*. ACM Press.

Lopez, P. (2009). GROBID: Combining Automatic Bibliographic Data Recognition and Term Extraction for Scholarship Publications. In *International Conference on Theory and Practice of Digital Libraries*, pages 473–474. Springer.

Mao, S., Rosenfeld, A., and Kanungo, T. (2003). Document structure analysis algorithms: a literature survey. In Tapas Kanungo, et al., editors, *Document Recognition and Retrieval X*. SPIE, jan.

Peng, F. and McCallum, A. (2004). Accurate information extraction from research papers using conditional random fields. In *HLT-NAACL 2004: Human Language Technology Conference of the North America Chapter of the Association for Computational Linguistics, Proceedings of the Main Conference*, pages 329–336.

Ramakrishnan, C., Patnia, A., Hovy, E., and Burns, G. A. (2012). Layout-aware Text Extraction from Full-text PDF of Scientific Articles. *Source code for biology and medicine*, 7(1):7.

Ronneberger, O., Fischer, P., and Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer.

Siegel, N., Lourie, N., Power, R., and Ammar, W. (2018). Extracting Scientific Figures with Distantly Supervised Neural Networks. In *To appear in ACM/IEEE Joint Conference on Digital Libraries in 2018 (JCDL 2018)*. ACM/IEEE.

Singh, M., Barua, B., Palod, P., Garg, M., Satapathy, S., Bushi, S., Ayush, K., Rohith, K. S., Gamidi, T., Goyal, P., and Mukherjee, A. (2016). OCR++: A Robust Framework For Information Extraction from Scholarly Articles. *International Conference on Computational Linguistics (COLING)*, pages 3390–3400.

Tkaczyk, D., Szostek, P., Dendek, P. J., Fedoryszak, M., and Bolikowski, L. (2014). Cermine – Automatic Extraction of Metadata and References from Scientific Literature. In *Document Analysis Systems (DAS), 2014 11th IAPR International Workshop on*, pages 217–221. IEEE.