Docear's PDF Inspector: Title Extraction from PDF files

Joeran Beel
OvGU, Magdeburg
Germany

Stefan Langer
Docear, Magdeburg
Germany

Marcel Genzmehr
Docear, Magdeburg
Germany

Christoph Müller
Docear, Magdeburg
Germany

beel@ovgu.de   langer@docear.org   genzmehr@docear.org   mueller@docear.org

ABSTRACT

In this demo-paper we present *Docear's PDF Inspector* (DPI). DPI extracts titles from academic PDF files by applying a simple heuristic: the largest text on the first page of a PDF is assumed to be the title. This simple heuristic achieves accuracies around 70% and outperforms the tool ParsCit which uses machine learning (accuracy between 36-50%). In addition, DPI is around 40 times faster than ParsCit, released under the free open source license GPL 2+, written in JAVA and runs on any major operating system.

Categories and Subject Descriptors

H.3.1 [Information Storage and Retrieval]: Content Analysis and Indexing – *Indexing methods*.

General Terms

Management, Documentation

Keywords

title extraction, pdf processing, style information, heuristic

1. INTRODUCTION

Several applications in the field of Academia require extracting titles from PDF files. For instance, academic search engines identify PDFs found on the Web, and reference managers such as Mendeley and Zotero extract titles (and other metadata) from PDFs to help users creating bibliographies. In the ideal case, a PDF's title is stored in the PDF's metadata and can easily be retrieved with standard PDF libraries (e.g. PDFBox, jPod, or iText). However, often a title is not available via the PDF's metadata. To retrieve a title anyway, the full-text of a PDF must be analyzed.

In the past years, several tools used machine learning to identify titles from PDFs [3–6], some of them being open source. However, the recently developed "SciPlore Xtract" [2] showed that a simple heuristic outperformed machine learning approaches. SciPlore Xtract extracts the largest font from the first page of a PDF and assumes this to be the title. Although researchers often claim accuracies of around 90% for title extraction [4–6], we recently showed that under "real-life" conditions, accuracies are rather between 50% to 70% [2].

All solutions have some shortcomings. Either they are proprietary solutions being not freely available (Mendeley), have problems in processing PDF files that do not comply 100% to the PDF standard (SciPlore Xtract), don't process PDFs at all and require third party tools (ParsCit), are rather slow and achieve low accuracies (ParsCit), are not available for all operating systems, or are available only as stand-alone tools which cannot be easily integrated into other applications.

2. DOCEAR'S PDF INSPECTOR

We developed "*Docear's PDF Inspector*" which identifies titles from (academic) PDF files and does not suffer from the aforementioned shortcomings. Namely, Docear's PDF Inspector (a) achieves good accuracies with excellent run times (see next section for details) (b) can be used as library by other JAVA applications which means other tools can easily integrate Docear's PDF Inspector (c) can be used as a stand-alone application that returns a PDF's title on the command line or stores the data into a CSV file (Figure 1) (d) can process several PDFs in a batch (e) can process all PDF files of all PDF versions, including those with minor deviations from the PDF standard. In the rare cases that a PDF cannot be parsed the title from a PDFs metadata is returned (if available) (f) is written 100% in JAVA 1.6 which means Docear's PDF Inspector runs on any major operating system, including Windows, Linux, and MacOS, without any other tools required (besides the JAVA runtime environment, of course) (g) is released under the GNU General Public License (GPL) 2 or later, which means it is completely free to use and its source code can be downloaded and modified by anyone. Both source code and compiled library can be found at *http://www.docear.org*.
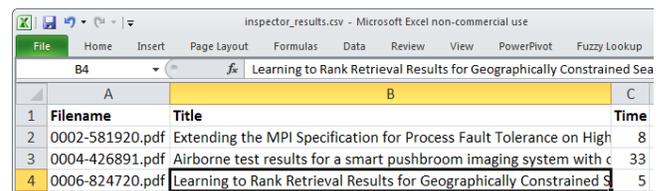
Figure 1: Output CSV opened in Microsoft Excel

Via command line, Docear's PDF Inspector is started with `java –jar PdfInspector.jar [OPTION][FILE]` and both options and files can be specified multiple times. Available options are `'header'` which includes a PDF's header in the output, `'name'` which includes the file name, `'time'` includes the time required for processing the PDF, `'out <arg>'` specifies the file to write to, `'outappend'` appends the output to an existing file instead of overwriting it, and `'delimiter'` specifies how fields are separated in the CSV file. The title extraction is performed in the same way as *SciPlore Xtract* does [2]. Namely, the largest font on the first page that is not exceeding eight lines is assumed to be the title. Docear's PDF Inspector uses the PDF library jPod for processing PDF files.

3. METHODOLOGY

To evaluate the performance of Docear's PDF Inspector we created a test collection of 500 PDF files. To have a PDF collection that contains various formats of academic articles we send 500 search queries to Google Scholar and from the result pages (each with 100 entries) we randomly downloaded one paper. 57 PDFs were removed from the collection because they had no title or were no academic articles at all, i.e. 443 articles remained for the evaluation. The search queries were randomly generated from words contained in the mind maps of the users of our literature management software

Docear [1]. We did not conduct a detailed analysis of the downloaded papers but it appeared to us that most papers were written in English, and some in German, French and Spanish. Papers were from various disciplines (computer science, psychology, biology, social sciences, business, etc.) and there was a very high variety of different formats of the articles. The collection of 500 PDFs is available upon request, so other researchers can use this PDF collection for their research and making their results comparable to ours. We also publish our research data, i.e. the extracted titles and charts we created, on http://labs.docear.org.
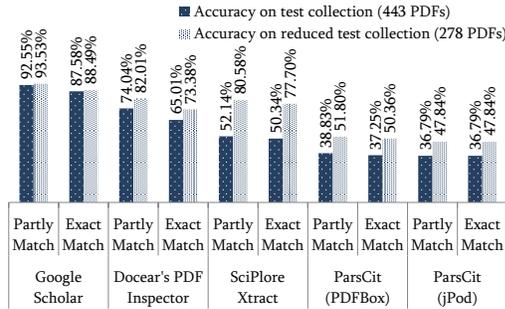

Figure 2: Accuracies of the tools on the two test collections

We evaluated Docear's PDF Inspector against SciPlore Xtract and ParsCit to have a comparison of how good the achieved results are. Because ParsCit cannot process PDF files by its own, we converted PDFs to plain text with PDFBox and jPod and run ParsCit on both text sets. If an extracted title was identical to the actual title, we classified the result as "exact match". If the extracted title was a substring of the actual title we classified the result as "partly match". Such a partly match occurred, for instance, when a tool failed to extract a PDF's sub-title. For both, exact and partly match comparisons, we ignored spaces and special characters.

Some PDFs caused parsing errors probably because they did not comply 100% with the PDF standard. For SciPlore Xtract and PDFBox (and hence ParsCit) this problem was most apparent: 35.21% (SciPlore) and 20.77% (PDFBox) of the 443 PDFs could not be parsed at all, for jPod the error was only 5.19%. While we consider the original test collection to be representative for a real-life scenario that applications such as academic search engines or reference managers face, we also wanted to have a test collection that could be processed by all tools, to evaluate the effectiveness of the title extraction algorithms (ignoring any PDF parsing problems). Therefore, we inferred a 'reduced test collection' by removing all PDFs from the original test collection which couldn't be processed by at least one of the tools. This resulted in a subset of 278 PDFs.

4. RESULTS

The results we present in this section also show how often titles from Google Scholar were accurate. We need to emphasize that accuracies from Google Scholar are not comparable with results from the other tools evaluated because Google Scholar often receives metadata directly from the publishers. That means, Google Scholar does not always extract metadata from PDFs. We provided these results only to show that even Google Scholar seems to have problems with extracting titles in some cases.

Docear's PDF Inspector achieves the highest accuracies (Figure 2). For our standard test collection Docear's PDF Inspector outperforms the second best tool (SciPlore Xtract) notably. Docear extracts 65.01% of the titles exactly, i.e. without any errors, while

SciPlore Xtract extracts only 50.34% titles accurately. ParsCit performs worst with an accuracy of 37.25% (PDFBox) and 36.79% (jPod). Also measured by 'partly matches' Docear performs best with an accuracy of 74.04% compared to SciPlore with 52.14% and ParsCit with 38.83% and 36.79%.

Looking at the reduced test collection the picture slightly changes. Now, Docear and SciPlore perform about the same. Docear extracts 73.38% of the titles flawlessly, SciPlore 77.70%. Based on 'partly matches' Docear extracts 82.01% of the titles correctly, SciPlore 80.58% (the differences are statistically not significant). ParsCit still performs far worse with accuracies around 50%.

Docear's PDF Inspector also performs best in terms of runtime. On average (mean), Docear's PDF Inspector needs 50ms to extract a title from a PDF while SciPlore Xtract needs 428ms and ParsCit 2965ms with the PDFBox library and 1786ms with jPod (Table 1). The comparison is not completely fair because ParsCit does not extract only the title (as Docear does) but also other metadata such as authors. However, for those users being only interested in the title, Docear's PDF Inspector identifies a title definitely fastest.

Table 1: Runtimes (in milliseconds) per PDF

	Docear	SciPlore	ParsCit (PDFBox)	ParsCit (jPod)
Mean	50	428	2965	1786
Std. Dev.	61	611	1383	1332
Median	23	352	2706	1394
Max	475	17667	15131	17585

Summarized, from a user perspective, Docear's PDF Inspector is notably the most effective tool. It is about 50% more effective than SciPlore Xtract and almost twice as effective as ParsCit for a PDF collection that we consider representative for real-life scenarios. In addition, Docear's PDF Inspector is around 40 to 100 times faster than ParsCit and eight times as fast as SciPlore Xtract which uses basically the same heuristic. From a research perspective (i.e. on the reduced data set), the simple heuristic applied by Docear and SciPlore is around 50% more effective than the machine learning approach applied by ParsCit.

5. REFERENCES

[1]   Beel, J., Gipp, B., Langer, S. and Genzmehr, M. 2011. Docear: An Academic Literature Suite for Searching, Organizing and Creating Academic Literature. *Proceedings of the 11th annual international ACM/IEEE joint conference on Digital libraries* (2011), 465–466.

[2]   Beel, J., Gipp, B., Shaker, A. and Friedrich, N. 2010. SciPlore Xtract: Extracting Titles from Scientific PDF Documents by Analyzing Style Information (Font Size). *Research and Advanced Technology for Digital Libraries, Proceedings of the 14th European Conference on Digital Libraries (ECDL'10)* (Glasgow (UK), Sep. 2010), 413–416.

[3]   Councill, I.G., Giles, C.L. and Kan, M.Y. 2008. ParsCit: An open-source CRF reference string parsing package. *Proceedings of LREC* (2008), 661–667.

[4]   Han, H., Giles, C.L., Manavoglu, E., Zha, H., Zhang, Z. and Fox, E.A. 2003. Automatic document metadata extraction using support vector machines. *Proceedings of the 3rd ACM/IEEE-CS Joint Conference on Digital libraries* (2003), 37–48.

[5]   Hu, Y., Li, H., Cao, Y., Teng, L., Meyerzon, D. and Zheng, Q. 2006. Automatic extraction of titles from general documents using machine learning. *Information Processing and Management*. 42, (2006), 1276–1293.

[6]   Peng, F. and McCallum, A. 2004. Accurate information extraction from research papers using conditional random fields. *HLT-NAACL04* (2004), 329–336.