SideNoter: Scholarly Paper Browsing System based on PDF Restructuring and Text Annotation

Takeshi Abekawa and **Akiko Aizawa**
National Institute of Informatics
Tokyo, Japan
{abekawa,aizawa}@nii.ac.jp

Abstract

In this paper, we discuss our ongoing efforts to construct a scientific paper browsing system that helps users to read and understand advanced technical content distributed in PDF. Since PDF is a format specifically designed for printing, layout and logical structures of documents are indistinguishably embedded in the file. It requires much effort to extract natural language text from PDF files, and reversely, display semantic annotations produced by NLP tools on the original page layout. In our browsing system, we tackle these issues caused by the gap between printable document and plain text. Our system provides ways to extract natural language sentences from PDF files together with their logical structures, and also to map arbitrary textual spans to their corresponding regions on page images. We setup a demonstration system using papers published in ACL anthology and demonstrate the enhanced search and refined recommendation functions which we plan to make widely available to NLP researchers.

1 Introduction

In recent years, there has been significant progress in the digitization of scientific papers; it has become common to distribute papers in electronic format, from paper submissions to the hand of readers without passing through print media.

Major academic publishers have defined their own XML format and utilize a corresponding publishing process called single-source multi-use, in which conversion from an XML file to paper print or electronic formats such as PDF, HTML, and EPUB is realized. However, in many scholarly publishing arenas, no XML editing process is available yet − after publication, only the corresponding PDF files are stored by the publisher. PDF format was established with the objective of maintaining the same page layout on printed paper as on a computer screen. Consequently, PDF does not contain any information indicating the logical structure within the file format. This logical structure is very important for understanding the document, and humans do it effortlessly and intuitively. To replicate that, the difficult mechanical extraction process will necessarily involve heuristics.

We have developed a paper browsing system called SideNoter that runs in a web browser. Because most existing papers are distributed in PDF, the challenge lies in how to handle the file format of the fixed layout. In our system, the constraint of the fixed layout is utilized in a converse manner — the paper itself is displayed in the image and overlapping supplementary information obtained from the full-text is displayed on the page layout. We designed a workflow to structurally parse documents in PDF. Based on this, SideNoter provides several advanced search functions, including figures and tables search, related section search, and per-page information recommendation. We also implemented tools that associate the layout with logical and semantic structures of documents. This enables us to incorporate semantic annotations produced by NLP tools into the visualized document image shown in the browser. Currently, we are investigating the usability of the system under development using papers published in the ACL anthology. Figure 1 illustrates the overall flow of our proposed system.

Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: System Demonstrations, pages 136–140, Osaka, Japan, December 11-17 2016.

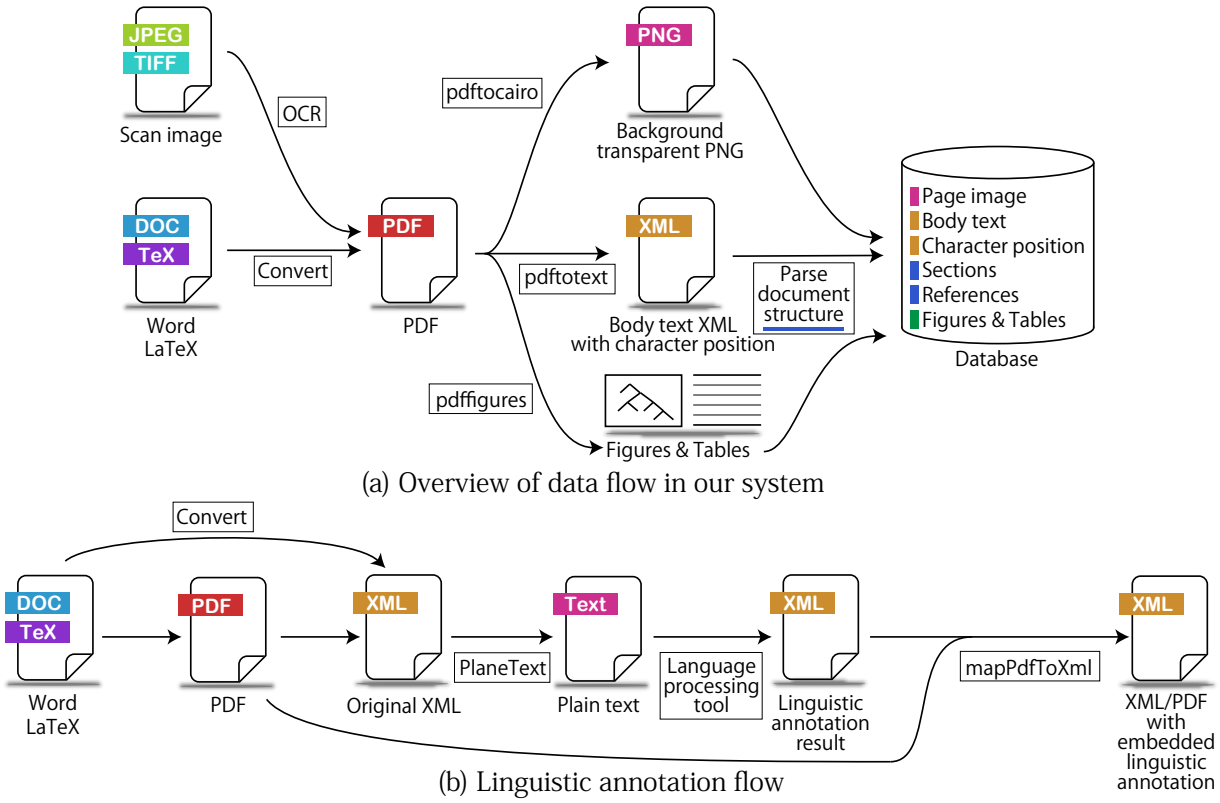

Figure 1: Overall workflow of the proposed system.

2 Related works

There are many systems for searching for papers. In the field of NLP, web services such as CiteSeerX[1] and ACL searchbench (Schäfer et al., 2011) are typical examples. To perform a flexible paper search in a specific field, it is necessary to extract the logical structure of the paper and its bibliographic information. Tools such as ParsCit (Councill et al., 2008), LA-PDFText (Ramakrishnan et al., 2012), PDFX (Constantin et al., 2013), and GROBID (Lopez, 2009) can be used to analyze the logical structure of papers. In addition, many frameworks that enable knowledge extraction from scholarly documents have been proposed, such as PDFMEF (Wu et al., 2015) and Dr. Inventor Project (Ronzano and Saggion, 2015). If the knowledge acquired from a paper could be displayed at the same time the paper is being read, readers' understanding of the paper would improve significantly. However, because a special viewer is often used to read PDF, it is difficult for other systems to add information to the same location as the PDF page.

3 Document Processing Work-flow

3.1 Document structure analysis flow

The system performs structural analysis of body text with the position coordinates obtained in the text extraction process, and obtains the logical structure of the paper. At this point, our system requires that the following three functions be realized: (a) font size and font name, (b) page coordinates of word units, and (c) in languages with a lack of space between words, such as CJK, the page coordinates of character units. We examined various open-source tools but could not find any that is sufficiently satisfactory. As a result, we decided to apply our own patch to *pdftotext* that is included in the Poppler library[2]. Because our objective is to adapt to other languages and other domains without having to prepare training data,

[1] http://citeseerx.ist.psu.edu/
[2] http://poppler.freedesktop.org/

we created our own rule-based structure analysis tool.

In this system, instead of displaying the PDF in a web browser, the image files converted from the PDF are displayed. To facilitate changes to the background color of the page, we utilize a transparent PNG background image format. In addition, our system uses the PDFFigures tool (Clark and Divvala, 2015) to extract figures and tables from papers. This tool can also recognize the corresponding caption text.

3.2 Linguistic annotation flow

We have developed a workflow to visualize and easily verify the annotation information generated by NLP tools on the page layout. First, the XML file of a paper is converted to plain text using our PlaneText framework[3] (Hara et al., 2014). PlaneText facilitates application of any NLP tools to target real-world documents containing structured text. Currently, a tool is also being developed to convert XML-tagged text into plain text sequences that can be directly inputted to NLP tools.

The annotation information is then applied to the resulting plain text using any of the NLP tools. In this case, the resulting generated file format is set to XML. Finally, the PDF layout information is embedded into the XML file using the mapPdfToXml[4] tool we are currently developing. This tool generates a new XML document by combining an original XML document and a PDF document that is converted from the original XML. The elements in the generated XML will have layout information that is extracted from the PDF: page number, position in the page, width, height, font name, font size, and color. Because SideNoter displays the page layout as an image, the annotation information generated by this workflow can be overlaid directly onto the paper's image.

4 Demonstration System: SideNoter for acl_anthology

4.1 XML-like advanced search functions

In this paper, search page used as the entrance to the system, a common search function is provided that facilitates full-text and metadata search such as paper title, author, conference name, and publication year. Search results display a facet list of the year of publication and the authors next to the paper list. The search results can also be narrowed to year of publication and author. In addition, the search can be limited to the text in the caption of figures and tables.

On clicking the paper title obtained in a search result, the outline of the paper is displayed onscreen. On the screen, thumbnails of each page, extracted figures and their captions, extracted section headings, and reference list are displayed. Relevant papers are listed by similarity with a vector space model weighted by TF-IDF on the right side of the screen. Clicking on sections in the section headings results in sections of other papers associated with the selected sections being listed.

4.2 Section-based retrieval

A click on the SideNoter icon in the search results or in outline view results in the screen transiting to paper browsing view. The system can display auxiliary information associated with the paper to facilitate reading comprehension as side-note columns on the left and right of the page. The system can also highlight specific terms or areas in the body text and draw an auxiliary line from a side-note column to the body text using an overlay over the image. The current system performs entity linking to Wikipedia articles, and displays explanatory text and images obtained from Wikipedia in a side-note column. It differs from other wikification systems in that it displays an image file that users simply look at to understand the meaning of a corresponding term. Thus, if a term is linked to the wrong entity, the user knows immediately that an error has occurred. Improving the accuracy of wikification is part of our future work.

In addition, the system utilizes a search API for terms and can dynamically display the search results. The current system searches for terms on video and slide-sharing sites, and displays the top results as side-notes.

[3] http://kmcs.nii.ac.jp/planetext/en/

[4] https://github.com/KMCS-NII/mapPdfToXml

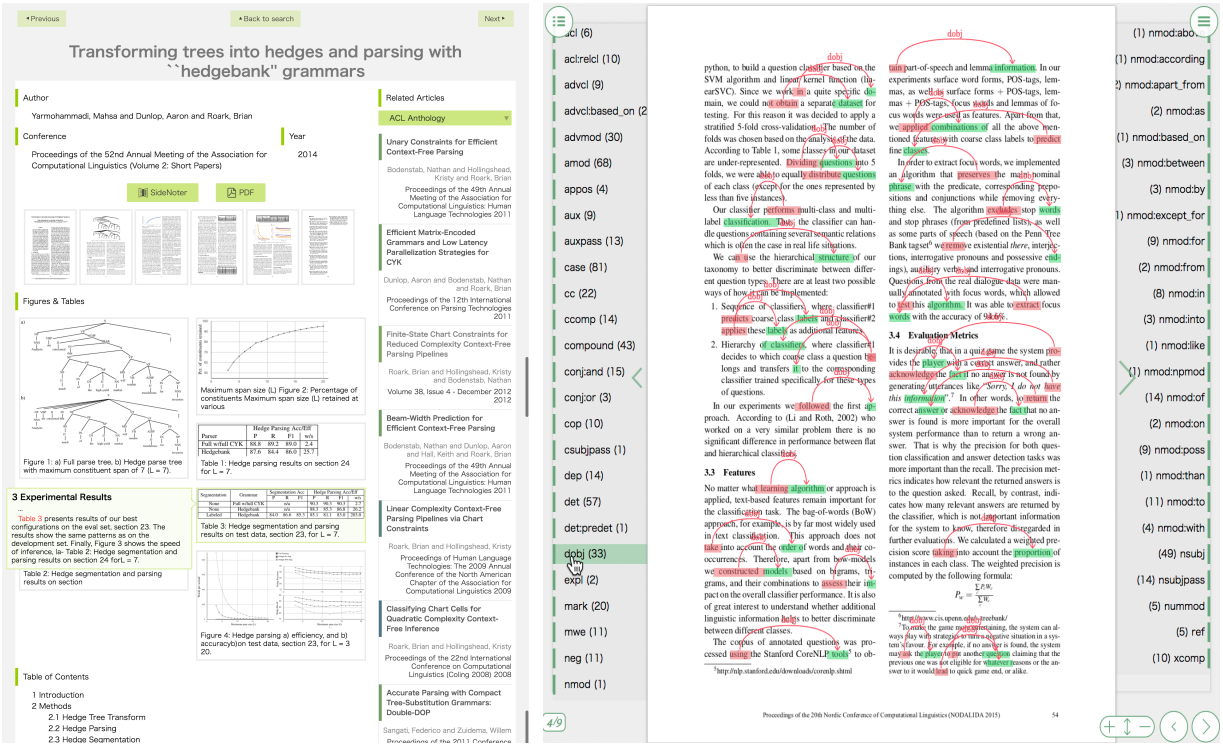

Figure 2: The screen shot of paper outline view (left) and dependency relation display (right).

4.3 Ability to seamlessly display linguistic annotation

As an example of this workflow, we annotated the dependency structure in the body text of a paper. In this workflow, the XML file annotated with dependency information was generated using the Stanford dependency parser (Chen and Manning, 2014) as the NLP tool. An example of the dependency relation displayed on SideNoter is shown in Figure 2. Conventionally, the dependency information has been discussed only in terms of one-sentence units. However, by viewing the overall relationship in the whole document, each of the dependencies occurring in the document and the density of the relationships can be understood.

5 Conclusion

In this paper, we presented a paper browsing system that displays a variety of information obtained from the body text of papers in side-note columns. In addition, we have developed a framework that displays annotation information obtained using an NLP tool on the paper layout. We believe that this platform will form a part of various useful NLP tools. This system will be published if a license can be obtained from ACL Anthology.

Acknowledgements

This work was supported by CREST, Japan Science and Technology Agency.

References

Danqi Chen and Christopher D Manning. 2014. A fast and accurate dependency parser using neural networks. In *the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 740–750.

Christopher Clark and Santosh Divvala. 2015. Looking beyond text: Extracting figures, tables, and captions from computer science papers. In *AAAI 2015, Workshop on Scholarly Big Data*.

Alexandru Constantin, Steve Pettifer, and Andrei Voronkov. 2013. Pdfx: Fully-automated pdf-to-xml conversion of scientific literature. In *the 2013 ACM symposium on Document engineering (DocEng2013)*, pages 177–180.

Isaac G. Councill, C. Lee Giles, and Min-Yen Kan. 2008. Parscit: An open-source crf reference string parsing package. In *the Language Resources and Evaluation Conference (LREC2008)*.

Tadayoshi Hara, Goran Topic, Yusuke Miyao, and Akiko Aizawa. 2014. Significance of bridging real-world documents and nlp technologies. In *Workshop on Open Infrastructures and Analysis Frameworks for HLT (OIAF4HLT)*, pages 44–52.

Patrice Lopez. 2009. Grobid: Combining automatic bibliographic data recognition and term extraction for scholarship publications. In *Proceedings of the 13th European Conference on Research and Advanced Technology for Digital Libraries*, ECDL'09, pages 473–474, Berlin, Heidelberg. Springer-Verlag.

Cartic Ramakrishnan, Abhishek Patnia, Eduard Hovy, and Gully APC Burns. 2012. Layout-aware text extraction from full-text pdf of scientific articles. *Source code for biology and medicine*, 7(1):1–10.

Francesco Ronzano and Horacio Saggion. 2015. Dr. inventor framework: Extracting structured information from scientific publications. In *Discovery Science*, volume 9356, pages 209–220. Springer International Publishing.

Ulrich Schäfer, Bernd Kiefer, Christian Spurk, Jörg Steffen, and Rui Wang. 2011. The acl anthology searchbench. In *the ACL-HLT 2011 System Demonstrations*, pages 7–13.

Jian Wu, Jason Killian, Huaiyu Yang, Kyle Williams, Sagnik Ray Choudhury, Suppawong Tuarob, Cornelia Caragea, and C. Lee Giles. 2015. Pdfmef: A multi-entity knowledge extraction framework for scholarly documents and semantic search. In *the 8th International Conference on Knowledge Capture (K-CAP2015)*.