International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 07 Issue: 05 | May - 2023 Impact Factor: 8.176 ISSN: 2582-3930

PDF to Text Converter using Natural Language Processing Approach

Nishu Kumari, Akanksha Singh

Computer Science and Engineering, Chandigarh University

Abstract - Extracting text from PDF documents is a challenging task that requires high accuracy and consistency, particularly when dealing with academic literature. While numerous systems are available, many fail to deliver satisfactory performance. This is a significant problem for researchers involved in text mining and project analysis, as they require a reliable tool for converting PDFs to text. In this paper, we propose a Natural Language Processing-based PDF-to-text (NLPDF) conversion system that consists of two major steps: extracting content from the PDF and reconstructing the text. We evaluate the performance of our system using four metrics: Precision, Recall, F-Measure (AF), and standard deviation. We compare our system with eight other benchmarked systems currently available in the market. The experimental results clearly demonstrate the effectiveness of our proposed system.

Key Words: PDF-to-Text Conversion, Natural Language Processing, Edit Distance.

1. INTRODUCTION

PDF files are widely used in digital libraries and publishing firms for storing research publications and related documents. The unique feature of PDF files is their inability to be easily altered, making them a suitable format for sensitive correspondence. However, extracting text from PDF documents is a challenging task because the file format stores text as images. This poses a dilemma for researchers in the text mining and analysis fields who must decide between reliable but costly PDF text extraction tools or settling for freeware with lower extraction performance.

To address this issue, we propose NLPDF, a PDF-to-Text extraction tool that leverages natural language processing techniques to enhance performance compared to publicly available freeware. The main contribution of our paper is the use of a natural language processing approach to identify words commonly misconverted by existing PDF conversion systems in the market.

In this paper, we begin by reviewing related studies and existing systems available in the market for PDF-to-text conversion. We then explain the methods employed to implement our proposed system, including a block diagram that illustrates its potential. Subsequently, we discuss the preliminary performance results of our system compared to other popular freeware using evaluation metrics such as precision, recall, and F-measure. Finally, we conclude by outlining potential future improvements for our system.

2. Related Works

Several freeware tools are available in the market for PDF-to-Text conversion, each with varying performance depending on the intended use. In our case, we are particularly interested in evaluating how well these tools perform text extraction from PDF files for research purposes. We have selected several systems for benchmarking against our NLPDF tool.

Among the systems, iTextSharp and NLTK are NLP toolkits commonly used in research, but only Online-Convert.com is able to extract all pages without any missing text segments. NLTK and iTextSharp's modified location-based extraction strategy (iTextSharp-Location) are the only ones capable of extracting text segments in a logical sequence. Zamzar stands out as the only system that avoids the misconversion of words due to certain font types and doesn't have words sticking together in the results.

TextSharp-Location initially had some issues with word detection sensitivity, but it was resolved by adjusting the divisor value. Pdf-To-Text performs better in extracting equations present in the documents, thanks to their improved Unicode conversion support.

It is worth noting that none of these systems are specifically designed to prepare text data for facilitating further research work, except for MSL. MSL can extract both textual and graphical information, such as pictures, diagrams, and tables, with the option for manual specification of the extraction zone. However, it requires significant user intervention, which can be time-consuming when dealing with a large number of documents.

To address these limitations and provide additional functionality such as section detection based on headers and numbering systems, we propose NLPDF. NLPDF utilizes NLP techniques and offers advantages such as fast processing, user-friendliness, the ability to infer solutions, and the capability to converse in the English language.

NLPDF aims to address the issues mentioned earlier and improve the arrangement and logical flow of extracted text. Additionally, it can recognize spaced-out words as header names, which is important as some research articles have headers with spaced-out characters. The other three systems compared in this study are Pdf-To-Text, Online2PDF, and Online-Convert.com.

3. Methodology

Figure 1 illustrates the block diagram of the NLPDF proposed system's functionalities. Basically, NLPDF has two major processes; read PDF text and reconstruct the text, each of which has four and five sub-processes respectively.

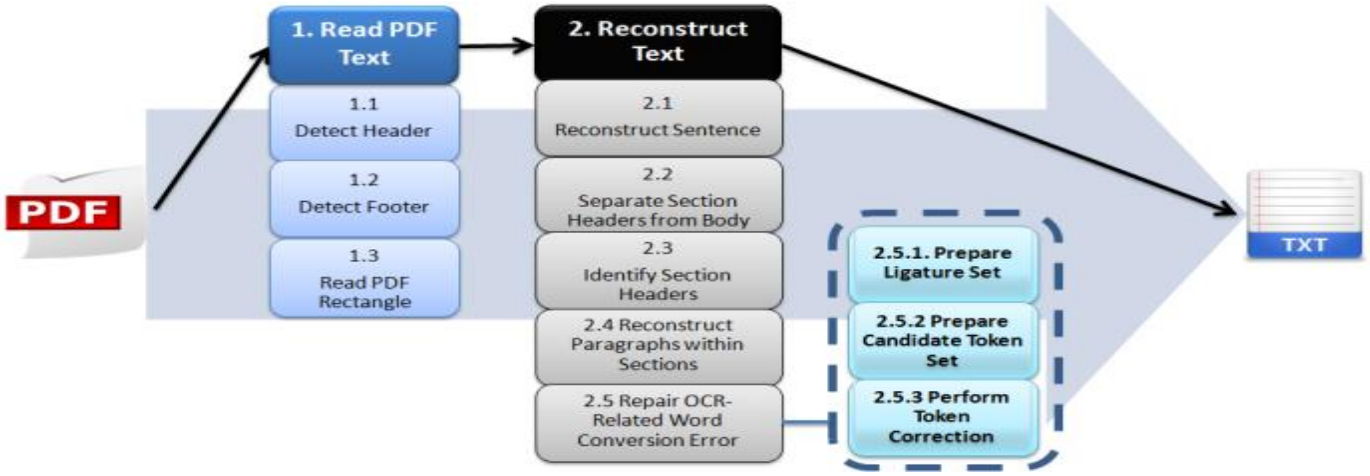

Fig.1. Block Diagram Showing Steps in NLPDF

A. Read PDF Text

Before extracting the content from a PDF file, several pre-processing steps are necessary to ensure that only the relevant content is extracted, excluding elements such as headers, footers, and page numbers. It's important to consider that not all research papers are formatted in a single-column layout. Many academic institutions now require articles to be formatted in two columns to make the content more compact and reduce the number of pages needed for publication.

The process of reading the PDF document involves three sub-processes: detecting the header and footer, identifying the coordinates of the header and footer, and setting the PDF rectangle zone to be read.

B. Detect Header of PDF Documents

A header in a document is the information placed at the top of each page, typically including page numbers. However, when extracting plain text for analysis or mining purposes, the header information can introduce noise. To address this, a header scanning technique is proposed to identify the position of headers in a PDF document.

The PDF document is typically in A4 paper size, with a maximum width of 595 points and a maximum height of 842 points. The text (and images) in the document is stored within rectangular zones defined by coordinates (x1, y1) to (x2, y2).

While most PDF-to-text converters extract all the text on a page, NLPDF avoids reading the headers and footers, which often contain repetitive text and page numbers. In the header detection process, the objective is to find the optimal y2 point that avoids all identified headers on the pages. Mathematically, this can be summarized as minimizing the page number times the gap size from the maximum height (max-y). A gap size of c=10 points is used for the effective detection of all text lines. The highest position is not considered to ensure that all headers are successfully covered during the elimination stage. By subtracting the gap size from y2, a new position is determined to bypass the headers safely.

The detection of the footer follows a similar approach to the header detection. Starting from the y1 position, the process gradually gaps up by the gap size (c) until the text is detected or y2 is reached (if the page is blank or it reaches the detected header). The goal is to identify the highest point of y1 that bypasses all footers in the document. Equation 2 represents this process, where the maximum x value is added to the product of the page number times the gap size (c). The lowest detected position is not considered to ensure full coverage of the footer elimination area.

In summary, the header and footer detection techniques in NLPDF involve finding the optimal positions (y2 for the header and y1 for the footer) to avoid including noise in the extracted text while maintaining coverage of the elimination areas.
Afterward, we need to filter the header and footer positions and then read a PDF Triangle. And text reconstruction takes place. Once this is done, then separate section headers from the body text.

Table 1: F-Measure Score Results

F-Measure Score	Article #1	Article #2	Article #3	Article #4	Article #5	Article #6	Article #7	Article #8	Article #9	Article #10	Average	Min	Max	Stdev
iTextSharp-Simple	0.904	0.928	0.767	0.986	0.962	0.874	0.713	0.899	0.868	0.996	0.890	0.713	0.996	9.10%
iTextSharp-Location	0.911	0.930	0.769	0.982	0.966	0.638	0.735	0.896	0.816	0.996	0.864	0.638	0.996	11.94%
MSL	0.912	0.932	0.771	0.967	0.921	0.587	0.736	0.902	0.858	0.918	0.850	0.587	0.967	11.79%
NLTK	0.960	0.928	0.935	0.989	0.940	0.212	0.882	0.941	0.881	0.990	0.866	0.212	0.990	23.27%
Online2PDF	0.964	0.952	0.938	1.000	0.916	0.634	0.932	0.943	0.877	0.859	0.902	0.634	1.000	10.24%
Online-Convert.com	0.880	0.854	0.877	0.921	0.841	0.846	0.809	0.842	0.725	0.992	0.859	0.725	0.992	6.96%
PdfToText	0.972	0.967	0.915	0.989	0.933	0.982	0.906	0.923	0.822	0.990	0.940	0.822	0.990	5.21%
Zamzar	0.952	0.935	0.955	0.989	0.931	0.922	0.883	0.919	0.794	0.990	0.927	0.794	0.990	5.68%
NLPDF	0.976	0.965	0.930	0.983	0.987	0.938	0.906	0.926	0.845	0.997	0.945	0.845	0.997	4.65%

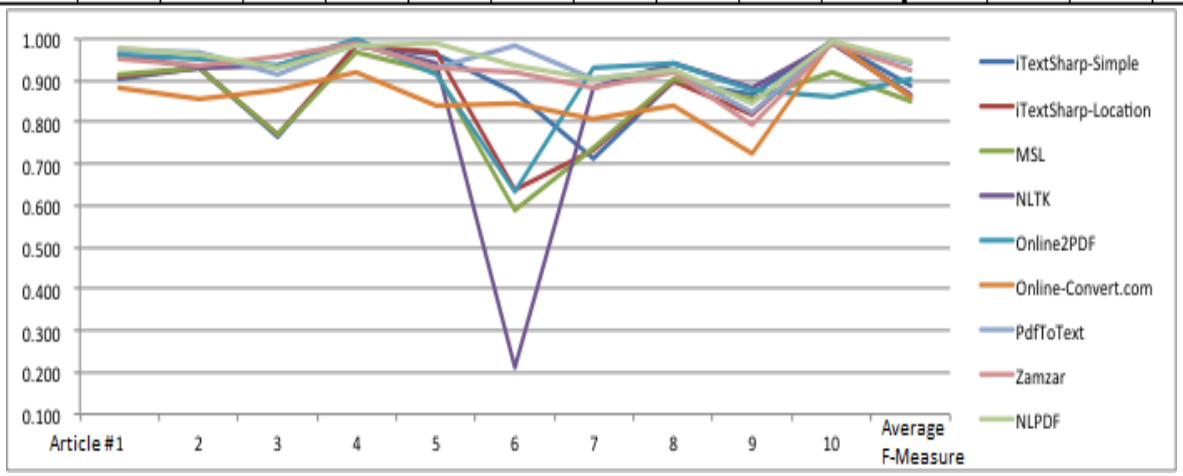

Fig. 4. Chart that Summarizes the F-Measure Scores

4. Results And Discussions

NLPDF achieved remarkable results in terms of performance, as summarized in Table 1 and Figure 4. Notably, it obtained the highest AF (94.5%), along with consistent AP (94.1%) and AR (95.2%) scores. The AP metric reflects the extraction of relevant terms while minimizing the inclusion of noisy elements such as headers, footers, punctuation, numbers, and Unicode errors. On the other hand, AR emphasizes the extraction of relevant terms. Although NLPDF did not achieve the highest AR score (ARmax = ZamzarAR = 95.9%), it still performed strongly in this aspect. Zamzar's exceptional AR performance can be attributed to its ability to preserve the presentation of equations, thereby maintaining the integrity of the equation structure.

Most other systems, including NLPDF, struggled with extracting equations, leading to the introduction of unnecessary noise and subsequently lowering the AR score. However, NLPDF managed to compensate for this weakness by focusing on and improving its word-based extraction approach, utilizing natural language processing techniques to predict words that might have been treated as misconversions. These misconversions were primarily caused by specific font types and sizes.

Additionally, the successful reduction of noise through strategies such as removing headers/footers and detecting section headers contributed to NLPDF's overall success.

5. Conclusion

Our study involved proposing and comparing multiple PDF-to-text conversion systems with a focus on extracting tokens accurately from both one- and two-column research paper formats. The results clearly demonstrated that our proposed NLPDF system outperformed the other eight systems in terms of reliability, consistency, and overall accuracy when extracting text from scientific and non-scientific PDF files. One notable contribution of NLPDF, as evidenced by the experimental results, is its ability to extract word-based tokens effectively. However, this comes at the expense of speed, which increases significantly with larger file sizes.

While our work has showcased the strengths of NLPDF, there are still two key issues that should be addressed in future research, providing numerous avenues for further study.

Firstly, improving the extraction of mathematical notations and operators, which are represented by Unicode characters, could enhance both the speed and accuracy of the system.

In conclusion, by combining the word-based extraction strategy of NLPDF with Zamzar's robust equation extraction using Unicode models, we can create an exceptional PDF-to-Text converter. Such a tool would automate the process of effortlessly generating thousands of reliable plain text documents, facilitating more researchers to engage in text information mining research.

ACKNOWLEDGEMENT

We would like to express our sincere gratitude to all individuals and organizations who have supported and contributed to this research. Their valuable insights, guidance, and assistance have been crucial in the successful completion of this work.

REFERENCES

1. R. Mohemad, A. R. H. Z. A. Othman, N. M. M. Noor. Automatic document structure analysis of structured PDF files. International Journal of New Computer Architectures and their Applications (IJNCAA), 1(2)(2011), 404-411.
2. R. A. Amsler. The structure of the merriam-webster pocket dictionary. (1980). Doctoral Dissertation.
3. Printernational. ISO Paper sizes. (2017). Available: http://www.printernational.org/iso-paper-sizes.php
4. Cambridge Computer Laboratory. (2017). Available: https://www.cl.cam.ac.uk/~mgk25/iso-paper-ps.txt
5. X. Xueya, Z. Yanmei. The research and application of the creation PDF document based on the iTextSharp. Proceedings of the IEEE Symposium on Robotics and Applications (ISRA), Kuala Lumpur, Malaysia, (2012).
6. S. Bird. NLTK: the natural language toolkit. Proceedings of the COLING/ACL on Interactive presentation, Sydney, Australia, (2006).
7. A. Kamble, A. Jaiswal, N. Dekate, S. Haridas, K. Pendke. Integrated System for Reading Multiple Files. International Journal of Computer Science and Mobile Computing, 3(2)(2014), 45-52.
8. R. Milton. Extracting Data from PDFs: Clean Air in Schools. (2011). Available: http://genesis.blogs.casa.ucl.ac.uk/2011/07/18/ extracting-data-from-pdfs-clean-air-in-schools/.
9. V. Gaikar. Zamzar: Convert Your Files to Any Format by Email. Tricks Machine, (2012). Available: http://www.tricksmachine.com/2012/03/zamzar-convert-your-files-by-mail.html.
10. Z. Ahmed, T. Dandekar. MSL: Facilitating automatic and physical analysis of published scientific literature in PDF format. F1000Research, 4, (2015).
11. Mediafox Marketing s.r.o. PDF to Text. (2017). Available: http://pdftotext.com/
12. Online2PDF.com. Online2PDF.com 8.0. (2017). Available: https://online2pdf.com/