Inferring Document Structure

Weiyen Lin
Department of Computing
Macquarie University
Sydney, Australia
will@gmail.com

Abstract

PDF documents form a rich resource repository of knowledge on the Internet both for academia and for business. The lack of logical structure information of PDF documents, however, limits the possibility of automatic information retrieval in many ways. In order to enhance its usability, the logical components of a PDF document, such as title, heading, paragraph, or reference for academic articles, need to be detected. This project builds on previous work that extracts the physical layout information of conference papers from PDF files, and aims to detect the logical structure from their physical layouts. We designed and implemented two algorithms for homogeneous block aggregation and for logical structure detection in software using objected-oriented technology. They will be evaluated on an unseen test set of conference articles from the Association for Computational Linguistics (ACL) Anthology.

1 Introduction

With the use of the Internet spreading fast in the last decade, the demand of transforming document images into machine-readable documents with logical structures is rapidly increasing. Those document images available on the internet mostly come from early-age archives in the libraries, such as out-of-date newspapers, magazines, etc., as well as the great volume of PDF documents in later digital age since 90's, such as academic articles, technical instructions, and business advertisements. Most of document image are generated for the aims of reducing storage space or facilitating printing and representation. However, since they do not provide information about their logical structure, such as titles, authors, section headings, figures and tables, this greatly limits the possibility of automatic information processing in many ways, such as retrieval, modification, and transformation. For example, we cannot perform specific queries toward those academic documents in PDF with certain search criteria, such as "Author = 'John Smith'" or "Title LIKE '%Mobile Commerce%'", since this logical information is not specified in those documents. This may result in low accuracy of searching and a waste of search time for internet search engines when matching the search keywords in the full text, instead of matching with title, abstract, or heading only. On the other hand, if logical structure information is provided, re-formatting can be easily done in an automated manner. Readers can choose preset style sheets or even design their own to reformat the document according to their preference of reading without need to change the original document. In many academic areas, researchers can do more in-depth analysis. For example, linguistics researchers can analyze the writing style of an author.

Due to the many advantages of documents being annotated with logical meaning, there have been a large number of related studies on detection of logical structure for document images for the last decades. This research builds on previous work that has extracted the physical layout information from conference papers from PDF into an XML format. Our aim is to detect the logical structure of the articles from these XML files. We developed two algorithms, one for homogeneous block aggregation and one for logical structure detection, and implemented them in an extensible object-oriented framework. They will be evaluated on a test set of unseen conference articles from the Association for Computational Linguistics Anthology.

This paper first examines related literature both on physical layout analysis and logical

structure analysis. Some of the methodologies in this literature are similar to the ones adopted by this research. Section 3 outlines the layout of our source material. In Section 4, we introduce a two-phase processing strategy for logical structure detection and describe our algorithms for aggregating homogeneous blocks and detecting logical structure. Following this are the evaluation and future work in Sections 5 and 6, respectively.

2 Related Work

Namboodiri (2003) referred to digital documents containing both text and graphics as document images, which are digitally generated from scanner or digital cameras. When reading a document image, a human reader can use various clues from its formatting layout, such as the title, headings, tables, and figures, easily to gain the main ideas of the document or quickly to retrieve certain contents he or she is interested. In fact, those formatting features reveal the logical structure of a document with more meaningful instructions for human's processing. However, even though document images are digitally stored and can be viewed by human readers from computer screens, they are not accessible by machines to retrieve document contents in the same manner, let alone to understand them.

The study dealing with document image processing is recently referred as "document analysis and understanding" with an aim of transforming a document image into more meaningful or logical manner. It involves two major processes: physical layout analysis and logical structure analysis, which are described as below.

2.1 Physical Layout Analysis

Namboodiri (2003) defined physical layout analysis as a process of "decomposing a document image into a hierarchy of maximally homogeneous regions, where each region is repeatedly segmented into maximal sub-regions of specific type." Those homogeneous regions or physical blocks include of figures, background, text block, text lines, words, and characters, etc. There are two major strategies for extracting those physical blocks from a document image: top-down and bottom-up methods.

Top-down methods process the extraction by splitting a document image into smaller regions using horizontal and vertical projection profiles. The X-Y Cut algorithm is one of typical top-down methods (Namboodiri, 2003). It starts dividing a document image based on valleys in their projection profiles. The algorithm repeats to project the regions of the current segment both on the horizontal and vertical axes until a stop criterion that determines the minimal unit of a region is reached.

On the other hand, bottom-up approaches, such as the run length smoothing algorithm (RLSA), first define the basic unit in order to start the grouping process. The distance between two adjacent units is calculated and compared with a threshold, either a horizontal threshold or vertical one. If the distance is less than the threshold, then two units are joined together.

The work, which this paper builds on, extracted all text roughly in the order of reading from conference papers in PDF and annotated it with a rich set of physical information, such as font size, font style, and position (Brett, 2009). The structure of the resulting XML files is described in Section 3. In order to enhance the accuracy of logical structure detection, we first apply a bottom-up algorithm similar to the RLSA to aggregate the text in the XML files into homogeneous physical blocks. Section 4.1 describes the details of aggregation process.

2.2 Logical Structure Analysis

Namboodiri (2003) defined the logical structure analysis as the process of 'assigning the logical layout labels to physical regions identified during physical layout analysis. Simply speaking, the logical structure is a mapping from the physical blocks in the document to their logical entities. Researchers dealing with logical structure analysis diverse in their selection of names for the methods they adopt, even when some of them tried to classify those adopted methods. For example, Lee et al. (2003) divided related works for logical structure analysis into the syntactic methods and the model-matching methods. Mao et al. (2003) stated that document logical structures are represented by models derived either from a set of rules or from formal grammars. Stehno and Retti (2003) categorized those models representing the logical structure into three: rule-based models, grammar-based models, and models using statistical or probabilistic methods.

Although researchers did not give the definition for each method, by inspecting those works under each classification, there are two major streams of methods adopted by researchers: syntactic and rule-based methods. Syntactic methods or grammatical methods regard the docu-

ment as a sequence of repeated objects. Lee et al. (2003) view a document as a sequence of headers and bodies, while Conway (1993) regard a document as a string or sentence to be parsed. They both create a grammar to describe the logical structure in terms of sequences of neighboring blocks. By applying a certain parsing algorithm repeatedly, the logical structure is identified either in a top-down or a bottom-up manner.

On the other hand, rule-based methods or model-matching methods do not create a syntax or grammar to represent the logical structure. Instead, they encode the knowledge about mapping each physical block with the most likely logical entity and by applying the preset rules toward each physical block, each block can be specified with a logical entity label. For example, a rule would say "If a block is of type 'large text' and located at the beginning of document, then it is a title". By apply those rules to predict each physical block to be a logical entity to build up the logical structure of the whole document. They also apply rules to regulate the process of logical structure detection.

In this research, we opt for a rule-based approach, matching each physical block to a preset logical entity model for conference articles in both top-down and bottom-up manner, when detecting logical structure from those physical blocks due to the availability of heuristic rules. A detection algorithm is described in Section 4.2.

3 Material

This research is based on an XML source of the ACL Anthology corpus which was derived by running Brett's software over a large part of the online PDF version of the ACL Anthology corpus, and slightly different from document images generated by scanning mentioned earlier. This XML source provides a rich set of physical features for each text roughly in the order of reading on a page, one page after another. Figure 1 is an example of the XML source, noting the height and width of a document page as well as the font, font size, and the position of each text under that page. The physical meanings of figures in the XML source are illustrated in Figure 2, and the extraction sequence is roughly in the order of reading, from top to bottom, left to right, and left column to right column, shown in Figure 3. The XML source is merely a raw file extracting each text sequentially from a PDF document and represented in a hierarchical structure using XML format, which is easier for later processing. This is referred as XML source by text in following paragraphs for differentiation from other XML formats we use.

```
<?xml version="1.0" encoding="UTF-8" ?>
- <document>
- <page index="1" height="841.890015" width="595.276001">
  - <text start="0" end="12" x1="104.952003" x2="149.371552" y="809.973022" h="4.971869" font="NimbusRomNo9L-ReguItal"
    fontsize="8.966400">
      <![CDATA[ Proceedings   ]]>
    </text>
  - <text start="12" end="15" x1="151.613159" x2="158.589020" y="809.973022" h="4.971869" font="NimbusRomNo9L-ReguItal"
    fontsize="8.966400">
      <![CDATA[ of   ]]>
    </text>
  - <text start="15" end="19" x1="160.830627" x2="171.787567" y="809.973022" h="4.971869" font="NimbusRomNo9L-ReguItal"
    fontsize="8.966400">
```

Figure 1. An example of XML source by text

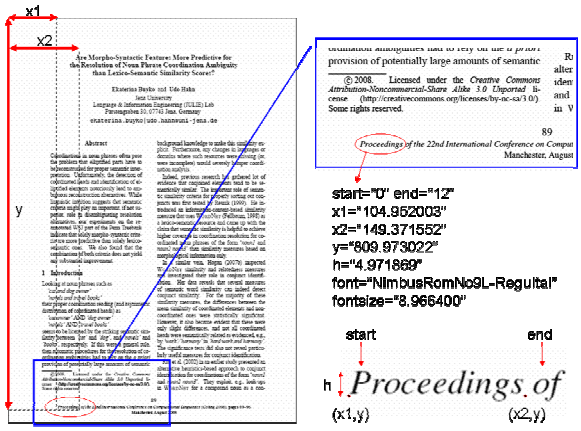

Figure 2. Physical feature of a document

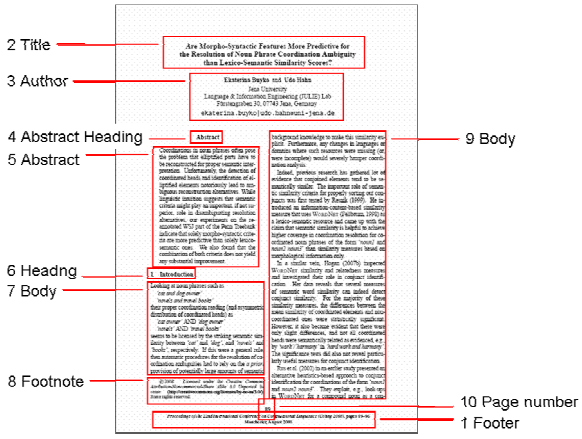

Figure 3. Extraction sequence of XML source

4 Methodology

This section describes the methodology we adopt for logical structure detection of academic articles. A two-phase detection strategy is introduced, followed by the development of algorithms for aggregating homogeneous blocks from XML source by text and for annotating logical label for each block.

Figure 4 illustrates the process of detection, consisting of two phases. In Phase I, the XML source by text is read-in and texts with same y-position are grouped into a line (as shown 1a in Figure 3) and the process outputs a new XML file, referred as XML source by line, with a tag <line> appended under <page> tag in Figure 4 and containing those <text> tags with the same y-position. Those lines in the XML source by line are further aggregated into homogeneous block according to their physical features and outputs another new XML file, XML source by block (as shown 1b in Figure 4), which is detailed in Section 4.1 shortly. Similarly, a tag <block> is appended under the <page> tag in

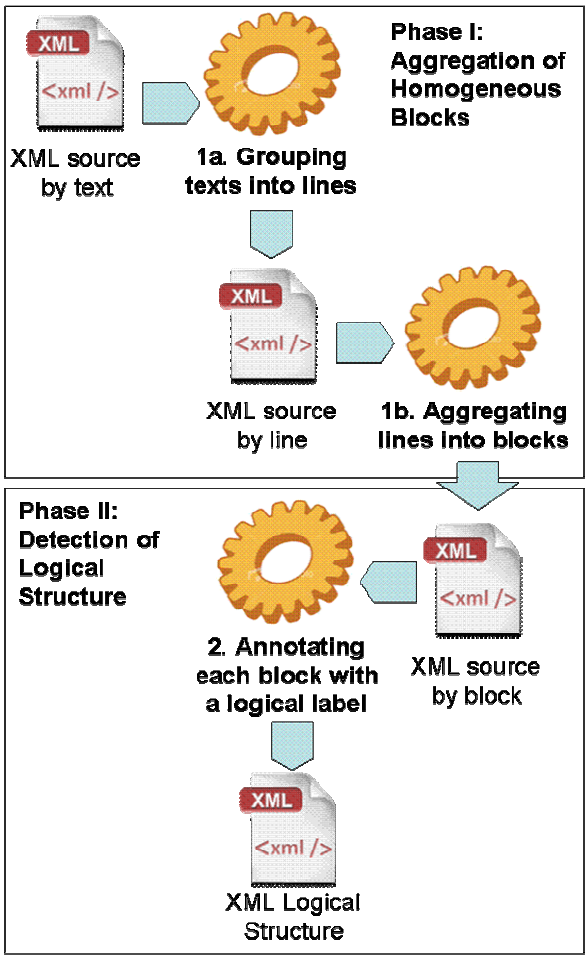

Figure 4. Process of logical structure detection

XML source by line, containing those <line> tags of homogeneous physical feature. With this XML source by block, the physical structure of the document can be shown in HTML format.

In Phase II, an algorithm is applied to the XML source by block to predict the logical meaning for each block according to preset rules based on the knowledge of the format of academic articles in ACL Anthology corpus. The process annotates each block with a logical label such as title, author, heading, or paragraph. Section 4.2 describes some rules for the detection.

There are many advantages of separating the detection into two phases. First, by separating the implementation of both algorithms, it can prevent the confusion of error roots resulted from physical layout analysis or logical structure detection. Second, both algorithms can be further refined and extended according to the results of evaluation. The more precise the block aggregation is, the more accurate the detection of the logical structure will be, which can greatly enhance the accuracy of detection. Third, with the modularization using object-oriented technology, the software can behave as a shell detecting different types of document images by accommodating different layout knowledge for certain type of documents.

4.1 Aggregation of Homogeneous Physical Blocks

The adoption of aggregating homogeneous physical block before directly detecting the logical structure of a document is mimicking the processing of human's vision. When humans read a document, the attention is first drawn by the physical features of that document, instead of logical features. In other words, human readers first identify the homogeneous blocks of lines or texts according to their physical attributes, such as position, dominant font size or font style, and spacing between those blocks. Then they start to have a closer look at the those blocks according to the message that each block transmits to them; for example, read the most upper block with the biggest font size on the first page of the document, which is identified as the title right after the block aggregation, and confirmed as the title after reading.

This research applies the same strategy as human's vision and aggregates those lines with homogeneous physical features into a block. The method we use is similar to the run length smoothing algorithm (RLSA), adopted in physical layout analysis. The algorithm, as shown in

Figure 5, assumes those texts under a <line> tag in XML source by line belong to the same block, and reads-in three lines at a time to determine which lines, with their texts, could further belong to the same block. The algorithm only considers the dominant font size of each line and the spacing between lines.

First consider the dominant font size, which is the most frequent font size appeared in those texts of a line. In Figure 5, if the dominant font sizes of three lines are not identical, as the cases AAB, ABB, A1BA2, and ABC, lines with the same dominant font size are aggregated into one block, and the reset, the other. If the dominant font sizes of three lines are identical, then the y-spacing between lines is further considered. Lines with smaller spacing are aggregated into one block, and the rest, the other. If the spacing is the same, three lines are aggregated into one block. For next iteration of reading, the last line of previous iteration is read in again as the first line of three in order to continue the aggregation without disconnection.

By applying this methodology, a fairly good aggregation outcome is obtained; not considering the noises contained in the XML source by text.

4.2 Detection of Logical Structure

Our research aims to detect the most important items of the logical structure, including the title, authors and affiliation, abstract heading, abstract, section headings, and body text by applying heuristic rules. These rules are based on the unique characteristics of each logical entity and compared with the statistics of the characteristic of the whole document. For example, the rule to detect the title is as follows:

"*IF the block is [in the upper half on the first page] AND [the dominant font size is the biggest font size of the whole document], THEN the block could be a title.*"

And for detecting section headings including reference and acknowledge, the rule is:

"IF the block's dominant font size is larger than [the most frequent font size of the whole document] AND [the starting text is a number OR in {"reference", "acknowledgement", "introduction", "conclusion"}], THEN the block could be a section-heading."

When choosing rules to be applied, we try to be as general as possible to accommodate the majority of document we observe from the corpus. However, it is impossible to find a single rule that applies to all the variations in the format of the conference papers.

5 Evaluation

5.1 Results

This research utilises the ACL Anthology corpus both for development and testing. Around 10 percent of the corpus was selected as the development set, accounting for 572 academic papers roughly evenly distributed over 13 conferences and 1 journal since 2000. Another 10 percent of unseen corpus is selected as the test set for a final evaluation of detection accuracy. Evaluation will be done by comparing the logical structure output in HTML with the original PDF document manually.

Table 1 summarizes the detection results for title, author and affiliation, abstract heading, abstract, section-heading, and page-number from 40 documents randomly selected from the development set. For the time being, the author and affiliation are detected as one block due to their large variation in format.

From the summary, we can see the system obtains fairly high accuracy when detecting title (97.5%), abstract heading (90%), and abstract (90%). The accuracies for authors-affiliation, page numbers, and section headings are lower. Generally speaking, the accuracy of detection is satisfactory considering the limited implementation time.

5.2 Error Analysis

When observing the details of detection results and looking up the original XML sources and PDF documents, we found several causes for the detection errors which can be solved in the near future as well as some defects due to the nature of format variation.

For example, the failure to detect section heading or sub-section headings can be improved by considering the length of lines spacing before and after the block. Detecting page numbers can also be improved by calculating their positions and examining the total number of pages. Furthermore, one abstract heading was detected with its abstract texts, which came from the incorrect aggregation in Phase I. It can be solved by refining the aggregation algorithm to separate them as different homogeneous blocks.

On the other hand, some erroneous detections of section-headings or page numbers mainly resulted from the *noises*, such as incomplete table contents and mathematic formula containing

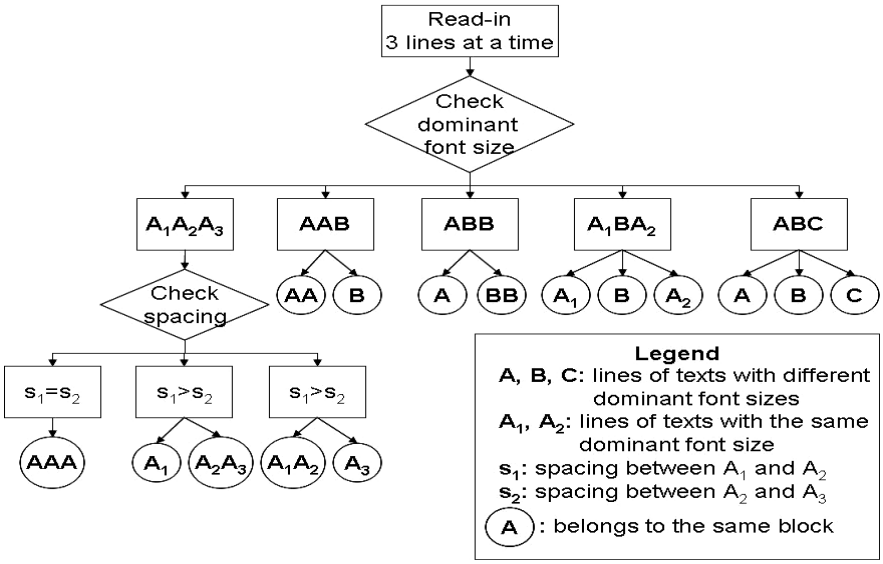

Figure 5. Algorithm for aggregating blocks

Error Type	Error Found	Accuracy of Detection
Incorrect title or missing title	1	97.5% (39/40)
Incorrect Abstract heading or Missing Abstract heading	4	90.0% (36/40)
Incorrect Abstract or Missing Abstract	4	90.0% (36/40)
Incorrect Affiliation(s) or Missing Affiliation(s)	11	72.5% (29/40)
Missing >50% of Page number(s) or Erroneous Page number(s) found	15	62.5% (25/40)
Missing >50% Section heading(s) or Erroneous Section heading(s) found	11	72.5% (29/40)

Table 1. Summary of detection results out of 40 randomly selected documents

numbers and random characters. There should be certain rules dealing with noises in order to obtain higher accuracy. However, this can also be resolved by improving the extraction process from original PDF documents. At this stage, we regarded those erroneous detections due to the noises as the inevitable loss of accuracy

6 Future Work

Both algorithms for physical block aggregation and for logical structure detection need to be further refined until they obtain as high detection accuracy as possible for the 572 documents of the development set.

In the near future, the separation of author and affiliation, more accurate detections of section-headings, sub-section heading, and paragraph texts need to be achieved as mentioned in Section 5.2. Following this, noises such as table contents and mathematical formula should also be removed.

References

B Powley, R Dale, and I Anisimoff, 2009. Enriching a Document Collection by Integrating Information Extraction and PDF Annotation. In *Proceedings of Document Recognition and Retrieval*, XVI, San Jose, California, USA.

Conway, A., 1993, Page grammars and page parsing: A syntactic approach to document layout recognition, In *Proceedings of the Second International Conference on Document Analysis and Recognition*, 761-764, Tsukuba Science City, Japan.

Lee, K., Choy Y. and Cho S. 2003, Logical Structure Analysis and Generation for Structured Documents: A Syntactic Approach, *Knowledge and Data Engineering, IEEE Transactions*, 15(5): 1277-1294.

Mao, S.,Rosenfeld, A. and Kanungo, T. 2003, *Document Structure Analysis Algorithms: A Literature Survey*, IBM Almaden Research Center, San Jose, USA.

Namboodiri A. and Jain A., 2007, Document Structure and Layout Analysis, in *Digital Document Processing: Major Directions and Recent Advances*, Springer-Verlag, London, 29-48.

Stehno, B. and Retti, G. 2003. Modeling the logical structure of books and journals using augmented transition network grammars, *Journal of Documentation*, 59(2): 69-83.