Automated document metadata extraction

Bolanle Adefowoke Ojokoh, Olumide Sunday Adewale and Samuel Oluwole Falaki

Department of Computer Science, Federal University of Technology, Nigeria

Abstract.

Web documents are available in various forms, most of which do not carry additional semantics. This paper presents a model for general document metadata extraction. The model, which combines segmentation by keywords and pattern matching techniques, was implemented using PHP, MySQL, JavaScript and HTML. The system was tested with 40 randomly selected PDF documents (mainly theses). An evaluation of the system was done using standard criteria measures namely precision, recall, accuracy and *F*-measure. The results show that the model is relatively effective for the task of metadata extraction, especially for theses and dissertations. A combination of machine learning with these rule-based methods will be explored in the future for better results.

Keywords: keywords; metadata; rules; segmentation; theses

1. Introduction

The availability of large, web accessible, heterogeneous repositories of electronic documents is increasing rapidly [1]. Most of this information is in the form of unstructured text, making the information hard to query [2]. Defining metadata for such documents will be useful for searching, browsing, and filtering. Ideally, metadata is defined by the authors of documents and is then used by various systems. However, people seldom define document metadata by themselves, even when they have convenient metadata definition tools [3]. Hence, automatically extracting metadata from the bodies of documents is an important research issue.

Automated document structure extraction has a number of benefits. Firstly, it makes automated mark-up possible. This helps to preserve information which might be required from the document in the future. In addition, documents with logical structure can be presented differently for different devices. This flexibility in presentation is very useful to handle different devices. For example, a document can be presented differently in a PDA and in a web browser. Keeping document logical structure also allows different users to have different access to a document. Some users may have unlimited access to a document while other users may have some limitations, for example, a textbook in which a professor has access to the answers as well as questions and a

Correspondence to: Bolanle Adefowoke Ojokoh, Department of Computer Science, P.M.B. 704, Akure, Nigeria. Email: bolanleojokoh@yahoo.com

Journal of Information Science, 35 (5) 2009, pp. 563–570 © The Authors, 2009. Reprints and Permissions: http://www.sagepub.co.uk/journalspermissions.nav, DOI: 10.1177/0165551509105195

student can access questions only. Finally, document logical structure helps resource discovery. With logical structure, a document can be searched in other ways instead of full text and metadata searches. For example, a document can be searched in a specific section, such as the introduction. It also allows search by some complex objects such as equations. Building tools for automatic metadata extraction and representation significantly improves the amount of metadata available, the quality of metadata extracted, and the efficiency and speed of the metadata extraction process [4, 5]. This paper focuses on presenting an approach for general metadata extraction from documents, with emphasis on theses. Metadata such as Title, Table of Contents, Abstract, Acknowledgement, Preface, Introduction, Conclusion and References are extracted from documents which could be in PDF, Word or Text formats.

The remainder of this paper is organized as follows: Section 2 presents a number of recent metadata extraction-related researches. An overview of the proposed approach for document metadata extraction is presented in Section 3. Section 4 gives the implementation and evaluation results while Section 5 presents the conclusion and further research directions.

2. Review of related studies

Metadata is, most generally, data that describes other data to enhance their usefulness in content exploration [6]. Several methods have been used for automatic metadata extraction from documents; regular expressions, rule-based parsers and machine learning are the most popular [4]. Liddy et al. [7] and Yilmazel et al. [8] developed rule-based systems founded on natural language processing technologies to extract metadata from educational materials. Mao et al. [9] performed metadata extraction using rule-based methods, particularly using rules based on formatting information which would not be possible with text files. Using a machine learning approach, Hu and colleagues [10] extracted titles from general documents. Han et al. [4] carried out metadata extraction. They viewed the problem as that of classifying the lines in a document into the categories of metadata and proposed using Support Vector Machines (SVM) as the classifier. They mainly used linguistic information as features. They reported high extraction accuracy from research papers in terms of precision and recall. Peng and McCallum [11] also conducted information extraction from research papers. They employed a Conditional Random Fields (CRF) model. Ojokoh et al. [12] used Hidden Markov Models (HMM) to implement the task of metadata extraction from some sets of tagged bibliographic references and particularly contributed to improving the smoothing technique suggested by earlier researchers. Most of these researches, however, focused on extraction from research papers that have most of the metadata to be extracted located on the first page of the paper [10]. Moreover, most of them carried out the task of metadata extraction for just a single metadata, such as author names or titles, only [13, 14]. This research combines the idea of extracting the structure of documents using keywords with regular expressions to extract metadata from documents.

3. Document metadata extraction architecture

The developed system is made up of six components, four modules (Converter, Segmentation Engine, Parser and Browser), each carrying out its own function towards the task of metadata extraction, and the Input and Output of the system. Equations (1)–(13) describe these components, their functions and relationship mathematically.

The Document Metadata Extractor, D is a 6-tuple (I, C, S, P, B, O)

$$D = (I, C, S, P, B, O) \tag{1}$$

where I is the Input, C is the Converter, S is the Segmentation Engine, P is the Parser, B is the Browser and O is the Output.

$$\text{(i) } I{:}p \tag{2}$$

where p is the uploaded document needing conversion.

$$\text{(ii) } Cp \to t \tag{3}$$

and

$$t = \{b_1, b_2, \ldots, b_n\} \tag{4}$$

where t is the text document and b_n refers to block n in the document.

$$\text{(iii) } S \text{ is a 2-tuple } (M, G) \tag{5}$$

where M refers to the map function and G refers to the group function.

$$\text{(a) } M\text{: } b_i \to s_k \tag{6}$$

where s_k refers to the set of identified pattern strings

$$\text{(b) } G\text{: } \{b_i \mid s_k = s_{i,\ \ldots,\ n}\} \text{ where } i \neq k \tag{7}$$

Step (b) is repeated until all grouping is done.

$\exists b_i^n$: $t \bullet n$ is the number of groups in which b_i occurs

$$\text{(iv) } P\text{: merge } b_i^{1,\ldots,n} \to b_i \tag{8}$$

then,

$$E\text{: } (b_i * md_i * s_i) \to \{md_i, b_i\} \tag{9}$$

$$md \in \{\text{abstract, tableofcontents, introduction, references}\} \tag{10}$$

where E is the extract function and md refers to the set of extracted metadata.
Title extraction is done with a different method as follows:

$$\text{title} \in \{[\text{A–Z}] * |\text{first 1–100 characters}|\text{1st three lines} \tag{11}$$

$$\text{(vi) } \forall md_i \text{ extracted, } B \text{ displays } O \in \{\text{title}, md_i, b_i\} \tag{12}$$

$$T\text{:}(D\text{:}u, md_{i,\ \ldots 4}, b_{i,\ \ldots n}) \tag{13}$$

where T is the store function and D is the database.

This research work used an approach implementing the model combining segmentation by keyword together with the use of regular expressions, as presented earlier, to extract some set of metadata from documents. Title extraction is another challenging task that may not be easily done using only segmentation by keyword as titles of documents are not usually labelled. In this research some sets of rules were proposed for title extraction. Figure 1 describes the architecture of the document metadata extractor.

Segmentation is the generation of hierarchy of logical divisions from a document. This layout segmentation captures the divisions of the document's logical structure. It can be done in three ways [15]:

- **Segmentation by spacing:** by using spacing information, scanned images are separated into several areas which can be text, figure/table, or formulae areas.
- **Segmentation by style difference:** for each text area, the average size of the characters contained in the area is calculated. The size of a character can be determined by its height. Also boldness can be calculated by the horizontal width of a character. In an area where the styles (bold, italic and size) of lines are obviously different from those of other lines, they are separately segmented.
- **Segmentation by keywords:** in an area, where a special keyword (e.g. abstract, references) comes at the beginning of a line, basically the area is segmented before the line as used in this research. This method is most relevant for this research because there are generally certain keywords specifically used in documents.

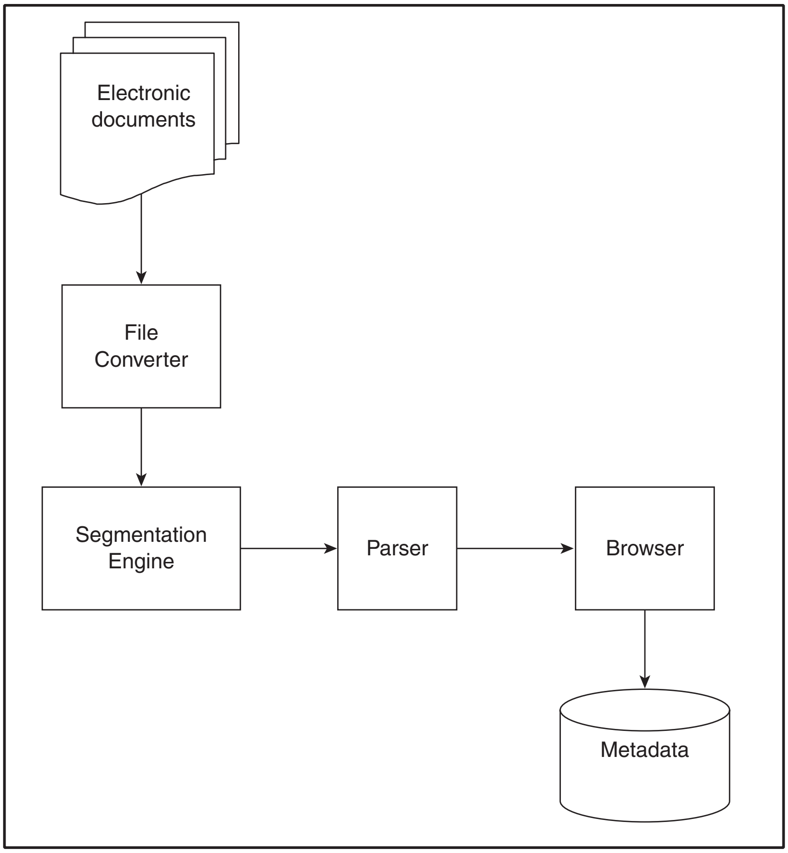

Fig. 1. Document metadata extractor architecture.

The segmentation algorithm proposed by Summers [1] was used for the purpose of logical structure extraction. This research names the algorithm used here as segmentation by keyword. It adopts some relevant techniques from Summers's algorithm which is embedded in the entire algorithm used for document metadata extraction as follows:

- Algorithm 1: Creating document input and converting uploaded file by typing document URL or locating from folder and converting document to text format
- Algorithm 2: Segmenting the document by keywords.

Divide the text document into blocks.

Represent each block by an appropriate string of indentation alphabet characters.

Find sets of blocks that form repeating patterns.

Find runs of isolated blocks that conform to patterns found elsewhere.

Group together the blocks in each element of each pattern and group the isolated block forming the next level of the tree surrounding blocks.

Repeat (*) in order until no changes are generated.

Group together the elements of each pattern forming another tree level.

Repeat until no new changes are generated.

- Algorithm 3: Extracting document metadata.

Locate the keywords associated with metadata.

Locate the block of document corresponding to the keyword(s) identified.

Extract keyword found in the document alongside the block of document.

- Algorithm 4: Displaying and storing result of extraction.

Display metadata and extracted block of document in the browser window.

Store the extracted metadata and block in the database.

The keywords used in the grouping include: Table of Contents, Abstract, Preface, Chapter 1(One), Introduction, Conclusion, References, Bibliography, Citation and Appendix (A/1). They were used with their respective regular expressions used to match them.

4. System implementation and evaluation

The document metadata extractor receives an uploaded document and converts it to text. On clicking on any of the metadata listed on the left side of the browser window as hyperlinks, the extractor scans through the document and displays the corresponding content of the document.

The document metadata extractor was tested over a set of randomly selected theses, dissertations and a few technical reports downloaded from the web. Theses were, however, the main focus of the experiments (about 80% of the sample).

The evaluation of the system was done using the following criteria: recall, precision, accuracy, and *F*-measure [as defined in equations (14)–(17)].

An exact performance comparison may not be possible, because of differences in the documents used for testing the different systems [16]. However, attempts are made to refer to related work and how the results compare with theirs. The results of the experimental evaluation of the reference metadata extractor are presented as follows.

$$\text{Precision} = \frac{A}{A+C} \tag{14}$$

$$\text{Recall} = \frac{A}{A+B} \tag{15}$$

$$\text{Accuracy} = \frac{A+D}{A+B+C+D} \tag{16}$$

$$F\text{−measure} = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \tag{17}$$

where A is the number of correctly extracted fields, B is the number of fields with existing, but not extracted data, C is the number of fields with wrongly extracted data, while D is the number of fields with not existing and not extracted data.

The test for automatic extraction correctness was based on a manual verification of the correctness of the extracted metadata against the original document. Due to such manual verification, which was lengthy and tedious, the sample size was limited to 40 documents.

Tables 1–4 summarize the overall precision, recall, accuracy and *F*-measure results respectively.

Table 1
Metadata extraction precision over 40 theses and related documents

	Precision
Title	0.75
Table of Contents	0.73
Preface	0.86
Abstract	0.77
Acknowledgment	0.64
Introduction	0.68
Conclusion	0.90
References	1.0

Table 2
Metadata extraction recall over 40 theses and related documents

	Recall
Title	0.75
Table of Contents	0.81
Preface	1.0
Abstract	0.92
Acknowledgment	0.90
Introduction	0.68
Conclusion	0.68
References	0.91

Table 3
Metadata extraction accuracy over 40 theses and related documents

	Accuracy
Title	0.75
Table of Contents	0.68
Preface	0.98
Abstract	0.78
Acknowledgment	0.70
Introduction	0.60
Conclusion	0.70
References	0.93

Table 4
Metadata extraction *F*-measure over 40 theses and related documents

	F-measure
Title	0.75
Table of Contents	0.77
Preface	0.92
Abstract	0.84
Acknowledgment	0.75
Introduction	0.68
Conclusion	0.77
References	0.95

From the results, ‘References’ was extracted with the highest precision. Its extraction is also relatively high compared to others in terms of recall, accuracy and *F*-measure. This could be largely due to the fact that they appear, in most cases, at the end of the document.

Most of the ‘Preface’ extraction cases fall under the not existing and not extracted cases because very few of the documents (only six) contained a preface. As a result, recall and accuracy were highest for ‘Preface’.

‘Introduction’ was extracted generally with the least values in terms of the evaluation criteria (0.68 for precision, recall and *F*-measure and 0.60 for accuracy). ‘Conclusion’ extraction was low too in terms of accuracy, although relatively high in precision (0.90). Reasons for these could be attributed to the fact that in many cases introductions and conclusions exist in some theses (sometimes in every chapter), and sometimes too, other keywords that might not be recognized by the system are attached to them. In addition, some patterns not embedded in the rule-based system might be encountered leading to incorrect extraction.

‘Abstract’ extraction was performed with relatively high recall and *F*-measure. The precision and accuracy of its extraction was mostly affected negatively because in a few theses, the abstract part was not specifically labelled ‘Abstract’ or was not labelled at all.

‘Table of Contents’ was extracted with relatively high recall, but with fair precision and *F*-measure. The accuracy with which it was extracted was a bit low, because in some cases, the system extracted some other parts, which were not actually part of the table of contents, with it.

‘Title’ extraction was particularly challenging as a result of the fact that titles of documents are not usually labelled as such. Nevertheless, the task was done with consistent precision, recall, accuracy and *F*-measure showing the effectiveness of the rule-based system. During the evaluation, it was also discovered that, for most of the wrongly extracted cases for titles, the information extracted was still relevant, for instance when the authors’ names or institution of study were extracted.

Some of the studies with presented results include the work of Hu et al. [10] who extracted title from Word documents with precision and recall of 0.810 and 0.837, respectively, and precision and recall of 0.875 and 0.895 from PowerPoint documents, respectively in an experiment on intranet data using machine learning technique.

Some systems whose work could be fairly compared to this system include that of Berkowitz and Elkhadiri [17] who extracted authors and titles from documents; they reported recall of 25.96% for exact extraction of author names; for 24.99% of the papers they managed to extract either part of the author name(s), or the name(s) plus extra text. All these focused on extraction of a single metadata. Giuffrida and colleagues [14] extracted title, author, affiliation, author-to-affiliation mapping and table of contents from Postscript files using a knowledge-based approach obtaining 92%, 87%, 75%, 71% and 76% accuracy respectively. Hence, the methods proposed by this research for general documents metadata extraction with particular emphasis on theses can handle the task with relatively high efficiency. Compared to existing systems, it extracts more metadata, and with relatively comparable effectiveness.

5. Conclusions and further research directions

This paper describes a method for general metadata extraction from general documents using a combination of the segmentation by keyword algorithm and regular expressions. Extraction of title is done using a different set of rules, implemented before segmentation by keyword is carried out because title extraction is often not affected by keywords. The rules used for title extraction proved consistently effective for theses and dissertations. The proposed model for extracting other metadata was tested on some randomly downloaded theses and related documents and used to extract Abstract, Preface, Table of Contents, Introduction, Conclusion, References and Acknowledgments. The experimental results show that the extraction was done with a relatively high degree of precision, recall and accuracy except for ‘Introduction’ and ‘Conclusion’ with low recall because of the usual existence of multiples of them in theses documents. Compared to existing systems, this system extracts more metadata, and with relatively comparable effectiveness. However, since it is rule based, it needs refinement over time as more keywords are discovered and metadata extraction is to be done for some different types of documents.

References

[1] K.M. Summers, Automatic discovery of logical document structure (PhD dissertation, Cornell University, Ithaca, NY, 1998).

[2] A. Arasu and H. Garcia-Molina, Extracting structured data from web pages, *Proceedings of the 2003 ACM SIGMOD International Conference on Management of Data (SIGMOD)* (June 2003) 337–348.

[3] A. Crystal and P. Land, Metadata and search, *Global Corporate Circle DCMI 2003 Workshop* (2003). Available from http://dublincore.org/groups/corporate/Seattle/

[4] H. Han, C.L. Giles, E. Manavoglu, H. Zha, Z. Zhang and E.A. Fox, Automatic document metadata extraction using support vector machine, *Joint Conference on Digital Libraries (JCDL'03)* (Houston, Texas USA, 2003).

[5] B.A. Ojokoh, S.O. Falaki and O.S. Adewale, Automated information extraction system for heterogeneous digital library documents, *Proceedings of the ACM/IEEE Joint Conference on Digital Libraries (JCDL 2007) Doctoral Consortium* (Vancouver, British Columbia, Canada, June 18–23, 2007).

[6] A. Kawtrakul and C. Yingsaeree, *A Unified Framework for Automatic Metadata Extraction from Electronic Document* (2004). Available at: http://iadlc.nul.nagoya-u.ac.jp/archives/IADLC2005/kawrtrakul.pdf

[7] E.D. Liddy, S. Sutton, E. Allen, S. Harwell, S. Corieri and O. Yilmazel, Automatic metadata generation and evaluation, *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (Tampere, Finland, 2002) 401–402.

[8] O. Yilmazel, C.M. Finneran and E.D. Liddy, MetaExtract: an NLP system to automatically assign metadata, *Proceedings of the 2004 Joint ACM/IEEE Conference on Digital Libraries* (Tuscan, AZ, USA, 2004) 241–242.

[9] S. Mao, J.W. Kim and G.R. Thoma, A dynamic feature generation system for automated metadata extraction in preservation of digital materials, *Proceedings of the First International Workshop on Document Image Analysis for Libraries* (Palo Alto, CA, USA, 2004) 225–232.

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

[11] F. Peng and A. McCallum, Accurate information extraction from research papers using conditional random fields, *Proceedings of the Human Language Technology Conference/North American Chapter of the Association for Computational Linguistics Annual Meeting* (New York, NY, USA, 2004) 329–336.

[12] B.A. Ojokoh, O.S. Adewale and S.O. Falaki, Improving on the smoothing technique for obtaining emission probabilities in hidden Markov models, *Oriental Journal of Computer Science and Technology* 1(1) (2008) 15–24.

[13] K. Seymore, A. McCallum and R. Rosenfeld, Learning hidden Markov model structure for information extraction. In: *AAAI Workshop on Machine Learning for Information Extraction* (1999).

[14] G. Giuffrida, E.C Shek and J. Yang, Knowledge-based metadata extraction from postscript files, *ACM Digital Libraries* (2000) 77–84.

[15] K. Nakagawa, A. Nomura and M. Suzuki, *Extraction of Logical Structure from Articles in Mathematics* (Springer, Berlin, 2004).

[16] M. Krämer, H. Kaprykowsky, D. Keysers and T. Breuel, Bibliographic metadata extraction using probabilistic finite state transducers, *Ninth International Conference on Document Analysis and Recognition* (2007) 609–613.

[17] E.G. Berkowitz and M.R. Elkhadiri, *Creation of a Style Independent Intelligent Autonomous Citation Indexer to Support Academic Research* (MAICS 2004) 68–73.