Knowledge-Based Metadata Extraction from PostScript Files

Giovanni Giuffrida, Eddie C. Shek, and Jihoon Yang

HRL Laboratories,LLC
3011 Malibu Canyon Road
Malibu, CA 90265
giovanni@cs.ucla.edu, shek@gowarehouse.com, yang@wins.hrl.com

ABSTRACT

The automatic document metadata extraction process is an important task in a world where thousands of documents are just one "click" away. Thus, powerful indices are necessary to support effective retrieval. The upcoming XML standard represents an important step in this direction as its *semistructured* representation conveys document metadata together with the text of the document. For example, retrieval of scientific papers by authors or affiliations would be a straightforward tasks if papers were stored in XML. Unfortunately, today, the largest majority of documents on the web are available in forms that do not carry additional semantics. Converting existing documents to a semistructured representation is time consuming and no automatic process can be easily applied. In this paper we discuss a system, based on a novel spatial/visual knowledge principle, for extracting metadata from scientific papers stored as PostScript files. Our system embeds the general knowledge about the graphical layout of a scientific paper to guide the metadata extraction process. Our system can effectively assist the automatic index creation for digital libraries.

INTRODUCTION

The recent proliferation of computers and communication networks made it possible for individuals around the world to quick access a wide variety of information sources through the Internet. Such sources are fairly diverse and range from simple text (e.g., journal articles, conference papers, broadcast news) to complex multimedia (e.g., video, audio, image) data. Thus, the design of automated tools for accessing these information sources and extracting relevant knowledge is of great interest. Among the variety of information sources, we are interested in text data, in particular, papers published in the scientific literature.

In many circumstances it is fundamental to disaggregate a scientific paper into its basic components [2]. Today, a large number of scientific papers are freely available for download in PostScript format. This format is ready to either be printed on high quality printers or shown on the screen. In this paper, we present a spatial rule based approach for the automatic extraction of metadata, such as title, authors, affiliations, and table of contents, from scientific papers stored in PostScript files. Metadata are extracted using specific *spatial cues*. For example, title extraction can be based on the spatial cue: "title is on the first page of the paper and it is printed using the largest font on the page."

In our system, we first translate the input PostScript file to a set of strings of text annotated with spatial/visual properties (such as xy position, page number, and font metrics). Such set of annotated strings are the "facts" for our knowledge base. Then, a set of rules *reason* upon those facts. Metadata are produced as results of firing such rules. The spatial knowledge to extract metadata is encoded within the set of rules.

Our system can be effectively used in a variety of situations such as: (1) automatic creation of indices for digital libraries, (2) conversion of documents to semantically richer representation and, (3) metadata mining.

SPATIAL RULE-BASED METADATA EXTRACTION

In the past, various systems have been presented to disaggregate text based documents. They broadly fall into

one of the following two categories:

- *Context-free grammar parsing.* In such an approach a somewhat rigid syntactical structure of the document is necessary. The text is composed of set of *tokens* and a set of *syntactical rules* to express legal relationships among tokens. This is the *de facto* approach for computer languages interpreters and compilers. This approach requires a well defined syntax and it is too rigid to parse free text.
- *Domain semantics based parsing.* This is an extension of the previous approach. Here a parser that embeds specific domain knowledge is used. Such a parser recognizes keywords and structural relationships for a well defined domain of the document at hand. (A successful example of this approach for medical domains is discussed in [8].) In this class of text interpretation, the parser is highly trained to work on a specific domain and its application to another domain requires radical changes to the parser itself.

We propose a *spatial knowledge based* approach to document disaggregation. Our approach can be easily integrated with any of the two methods mentioned above for an improved extraction accuracy. Our approach is based on exploiting the visual/spatial knowledge humans make use of when reading a document. In general, within a document category, a certain visual layout can be identified for all documents within that category. Some spatial properties on the layout of a scientific paper are listed in Fig. 1.

- The title is located on the *upper* portion of the *first page* and it uses the *largest font* on the *first page*;
- Authors are listed *immediately under* the title in a certain order;
- Affiliations *follow* the authors' list;
- If *only one* affiliation appear then *all authors* are associated with it;
- The *same font* is used for all authors and, similarly, for all affiliations;
- The first level headers use a *larger* font than the second level;

Figure 1: Some general *spatial* properties about a scientific paper layout.

Our systems embeds this type of knowledge to disaggregate the input document. (We used italic words in the above statements to denote relationships—spatial and of other types—among document metadata.)

We used a rule-based language to encode the visual knowledge in our system. As already mentioned, different "types" of documents require different domain knowledge—e.g., an article on a weekly magazine has a quite different underlying structure than a scientific paper. The knowledge base we developed so far copes with scientific papers appearing on conference proceedings and specialized journals.

At this time, the artificial reasoning in our system is solely based on the spatial text properties extracted from the input file. We did not exploit any domain semantics expertise at this time—e.g., humans know that "University of California" is not any of the authors, regardless of any spatial relationship it is involved into. Domain semantics would greatly help the disaggregation process and is in our future work agenda.

Architecture

Our system is developed on top of anss *expert system shell* called GCLIPS [6, 7]. GCLIPS is oriented to support development of *graphical* expert systems; it is built on top of CLIPS [3].

The input for our system is a PostScript file. As previously stated, at this time we have only implemented knowledge for scientific papers appearing in conference proceedings and/or journals; thus, the PostScript file given in input needs to contain that type of document.

Figure 2 shows the overall architecture of our system. The knowledge engineer (KE) provides a set of *rules* that embodies the *expertise* to extract the metadata of interest from the input document. pstogclips reads the input PostScript file and produces a set of *facts* for GCLIPS. Each fact contains a piece of information—text and spatial data—about the input PostScript document. Rules provided by KE *reason* on the extracted facts to identify (and extract) relevant metadata from the input documents.

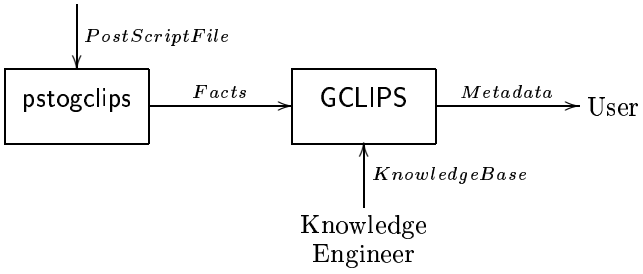


Figure 2: System Architecture

pstogclips: From PostScript to GCLIPS

Converting PostScript to other formats is not a trivial task [9] due to the nature of PostScript itself. PostScript is not a *bitmap* format such as GIF or JPEG, it is a programming language specifically tailored to model page layouts. Figure 3 shows a small piece of a PostScript file, fragments of words are embedded within parenthesis and interleaved with PostScript keywords.

As a programming language, there are many different ways to produce the same output, that is, there are many different PostScript programs that can be written to produce the same results. Thus, printing a PostScript file means *interpret-and-execute* the PostScript program contained in that file. This is the task accomplished by any printer PostScript driver or PostScript viewer such as "Ghostview."

We developed pstogclips by extending pstotext [11]. This latter is a PostScript-to-text translator developed at Digital as part of the *Virtual Paper* project [12]. pstotext reads a PostScript file and extracts all textual information contained in it. In a PostScript file, the typical word boundary based structure of a document is broken down into fragments (of words); no specific information is encoded in the file on how to recover the original structure from such a collection of fragments. pstotext—as well as other PostScript-to-text translators [10]—follows some heuristics to rebuild the word based structure of the original document. Furthermore, pstotext performs a set of other smart tasks such as merging hyphenated lines. However, it overlooks all spatial and font-related data contained in the input PostScript file—it simply does not need those to accomplish its task. Conversely, pstogclips retains all such additional data: its output consists of a set of *augmented* strings of text. These additional data are summarized in the following:

- Page of the document where the specified string appears;
- Absolute line counter order for each generated string;
- xy location of the lower left corner of the string bounding box (in paper-dot coordinate systems);
- xy location of the upper right corner of the string bounding box (in paper-dot coordinate systems);
- Font metrics (bounding-box extensions) used to represent the given string of text.

The Knowledge Base. Our knowledge base reasons on the facts extracted by pstogclips. We encoded the knowledge base by means of GCLIPS rules. We designed the rule set to extract the following information from the PostScript file: title, author(s), affiliation(s), mapping(s) author–affiliation, and table of contents. At this time the knowledge base is composed of 77 rules for a total of 788 lines of GCLIPS source code. Table 1 shows the GCLIPS rule usage distribution for the different extraction purposes:

Table 1: Rule Usage Distribution.

Extrac. Purpose	*# of Rules Involved*
Title	9
Author(s)	12
Affiliation(s)	10
Auth.-Affil.	10
Table of Cont.	8
Print results.	19
Init and misc.	9

A fundamental component of our knowledge base is the implicit *fuzziness* involved in the visual/spatial based metadata recognition process. For instance, with reference to the list of properties of Fig. 1 we may notice the following exceptions:

- not always the title is printed by using the largest font on the first page,
- not all papers use numbered section headers and not always use different fonts for different section levels,
- sometimes authors are all listed on the same line next to each other while other times they are scattered across different lines,
- when authors belong to different affiliations different methods are employed to specify their correspondence. Two of the most popular are: (1) superscripts on authors and affiliations; (2) each author is spatially closer to his/her affiliation. Still, many other different cases exist such as reporting affiliations as footnotes or listing authors vertically with respective affiliation on the right on the same line.

Exceptions like the ones just listed, are the hardest part of the artificial visual recognition process. We coded the GCLIPS rules in our knowledge base in order to be tolerant against many of such exceptions. We will discuss the accuracy of our findings in Section . In some cases we develop a sort of general fuzzy strategy for certain metadata, whereas other cases have been treated as *special cases.* In the next section we further discuss it by means of real examples.

Discussion and Examples

In this section we first present an example of execution of our system, and discuss a couple of cases where a metadata extraction non based on visual/spatial properties would hardly succeed, then we turn to the accuracy of

```
1 0 bop 389 428 a Fr(Kno)n(wledg)q(e\255Based)43 b(Metadata)f
(Extraction)g(fr)m(om)1391 594 y(P)-6 b(ostScript)41
b(Files)988 861 y Fq(Giovanni)31 b(Giu\013rida,)h(Eddie)f(C.)g(Shek,)g
(and)f(Jiho)l(on)g(Y)-6 b(ang)1623 1055 y Fp(HRL)28 b(Lab)r(oratories)
1297 1155 y Fo(f)p Fp(gio)n(v)-5 b(anni)p Fo(j)p Fp(eddie)p
Fo(j)p Fp(y)n(ang)p Fo(g)p Fp(@wins.hrl.com)-150 1530
y Fn(ABSTRA)m(CT)-150 1640 y Fp(The)35 b(automatic)g(do)r(cumen)n(t)g
(metadata)f(extraction)g(pro)r(cess)-150 1740 y(is)25
b(an)h(imp)r(ortan)n(t)f(task)g(in)h(man)n(y)e(di\013eren)n(t)i
(domains.)36 b(This)25 b(is)-150 1839 y(esp)r(ecially)c(true)g(in)g(a)g
(w)n(orld)f(where)g(thousands)h(of)g(do)r(cumen)n(ts)-150
```

Figure 3: A fragment of a PostScript file

our system.

Example of Execution. Consider the (portion of) paper of Figure 4. Once pstogclips has extracted all necessary facts from the PostScript file, we can process them from GCLIPS. The output of the GCLIPS screen for this paper is the following:

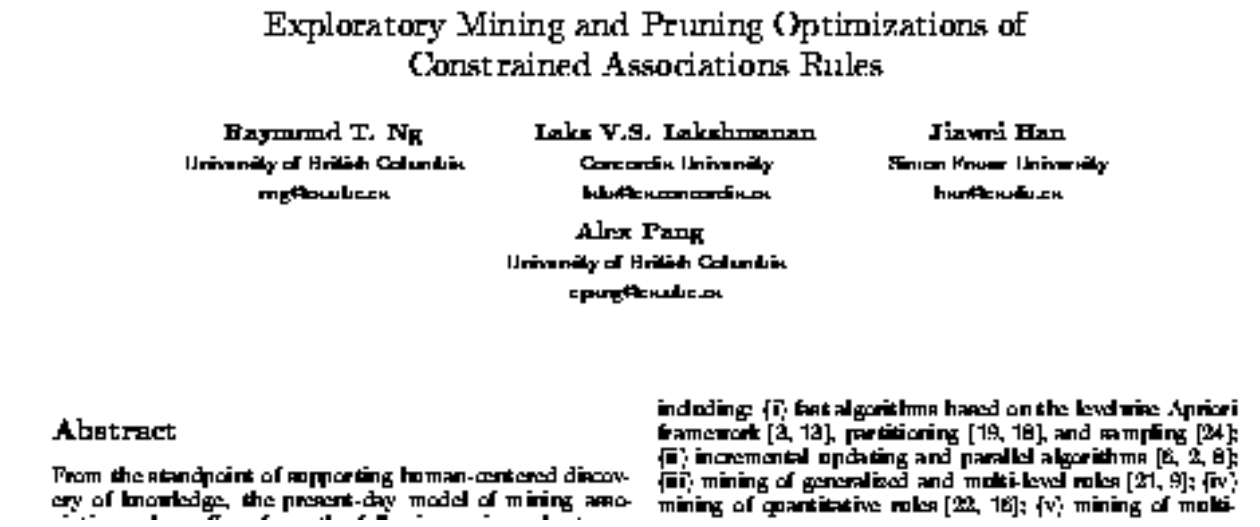
```
FILE: sigmod98
TITLE: Exploratory Mining and Pruning Optimizations of Constrained
Associations Rules
AUTHOR: Laks V.S. Lakshmanan (7)
AUTHOR: Jiawei Han (10)
AUTHOR: Raymond T. Ng (4)
AUTHOR: Alex Pang (4)
AFFILIATION 4: University of British Columbia
AFFILIATION 7: Concordia University
AFFILIATION 10: Simon Fraser University
---Table of Contents---
1  Introduction
3  Constrained Association Queries
4  Optimization Using Anti-Monotone
5  Optimization Using Succinct
6  Algorithms for Computing
6.1  Algorithm Apriori +
6.2  Algorithm Hybrid(m)
6.3  Algorithm CAP
8  Conclusions and Future Work
```

The title has been assembled from two lines into a single one. Authors have been correctly identified and linked to each respective affiliation (the index following each author's name links him/her to the affiliation). Notice that the system reported only once the affiliation "University of British Columbia" of two distinct authors. The table of contents misses some entries.

Caveat 1: Title Extraction. At a first thought, one may think that the title of a scientific paper is contained in the first line of text (or couple of lines for longer titles) of the paper; therefore, a text based extraction from a PostScript file could be easily applied. Unfortunately, this is not the case when, for instance, authors report information on the proceedings containing the paper as shown in Figure 6. In such cases, a straight text based approach will have hard time in extracting the wanted information.

We encoded the following two hints in our knowledge base when extracting titles: (1) titles appear on the first page of the paper and (2) very often they are printed using the largest font on the first page. However, we have soon found out that not always titles are printed by using the largest font as, for instance, at times section headers use a larger (or same size) font of the title. In such a case we rely on the word "Abstract" and extract the lines printed by using the largest font among all the lines above that word. We now discuss this particular case in more details. Figure 5 shows the GCLIPS rules used to extract the title from the paper when the word "Abstract" was found on the first page as a stand-alone string. The first rule, `CandidateTitleLines`, asserts all lines above the one containing the word "Abstract" as candidates for the title—these will include all authors, affiliations, etc. At the same time it extracts the font size of each text line (the font size is specified in the slot `bbh` of the fact `metrics`). Subsequently, the rule `GetLargestFontForCandidateTitle` extracts the largest font among all candidate title lines. The rule `GetTitle1` get the first line of the title, that is, the one that has the largest font *and* does not have any other line above it with the same font. The last rule, `GetTitleNextLines`, fires for multi-line titles, it merges successive title lines having the same font.

However, sometime this may still not be sufficient, for example, when author names are printed using the same title font: they both appear above the abstract. Thus, we further reinforced our knowledge base by relying on the line-space (measured along the y coordinate) of title lines and authors' line.



Exploratory Mining and Pruning Optimizations of Constrained Associations Rules

Raymond T. Ng
University of British Columbia
rng@cs.ubc.ca

Laks V.S. Lakshmanan
Concordia University
laks@cs.concordia.ca

Jiawei Han
Simon Fraser University
han@cs.sfu.ca

Alex Pang
University of British Columbia
apang@cs.ubc.ca

Abstract

From the standpoint of supporting human-centered discovery of knowledge, the present-day model of mining associ-

including: (i) fast algorithms based on the levelwise Apriori framework [3, 13], partitioning [19, 18], and sampling [24]; (ii) incremental updating and parallel algorithms [6, 2, 8]; (iii) mining of generalized and multi-level rules [21, 9]; (iv) mining of quantitative rules [22, 16]; (v) mining of multi-

Figure 4: Example of a scientific paper

```
(defrule CandidateTitleLines
  (declare (salience 9100))
  (abstract-word-found ?la)
  (doc (page 1) (font ?f \$?)
       (absline ?n&:(< ?n ?la)) (text ?s))
  (metrics (page 1) (font ?f) (bbh ?hi))
  =>
  (assert (candidate-title-line ?n ?hi ?f ?s)))

(defrule GetLargestFontForCandidateTitle
  (declare (salience 9090))
  (abstract-word-found ?la)
  (candidate-title-line ?n ?hi ?f ?)
  (not (candidate-title-line ?
                             ?h2&:(> ?h2 ?hi)
                             ? ?))
  =>
  (assert (ltf ?f)))

(defrule GetTitle1
  (declare (salience 9000))
  (abstract-word-found ?la)
  (ltf ?f)
  (candidate-title-line ?n ?hi ?f ?s)
  (not (candidate-title-line ?n2&:(< ?n2 ?n)
                             ? ?f ?))
  =>
  (assert (paper-title ?n ?s)))

(defrule GetTitleNextLines
  (declare (salience 9000))
  (abstract-word-found ?la)
  (ltf ?f)
  ?indx <- (paper-title ?n ?s)
  (candidate-title-line ?n2&:(= (+ 1 ?n) ?n2)
                        ? ?f ?t)
  =>
  (retract ?indx)
  (bind ?s (str-cat ?s " " ?t))
  (assert (paper-title ?n2 ?s)))
```

Figure 5: Rules to Extract Title when the word "Abstract" was found

Caveat 2: Spatial Based Mapping Authors to Affiliation. An important metadata of a scientific paper is the affiliation of each author. Our rule base first extracts both authors and affiliations then tries to link them. There are many different cases to be considered since this is a n–to–m mapping. The simplest case is the n–to–1, in this case all n authors are affiliated to the same institute; one simple GCLIPS rules takes care of that. Another case is when the number of authors is different from the number of affiliations and there is more than one affiliation. In such a case a common practice is to use superscripts over authors and affiliations. We exploit a text-based parsing to resolve the associations in this case.

The case we now discuss is the n–to–n as of Figure 4—notice however that one affiliation appear twice. In this case we perform a spatial reasoning to link each author to his/her affiliation. This task is accomplished by the GCLIPS rules shown in Figure 7. In particular, the rule `XY-AffiliationLocation` asserts the xy location (in paper dot coordinates) of the center of the string bounding box of each affiliation (the slot `xc` of the fact `doc` contains that location). Similarly, the rule `XY-AuthorLocation` asserts the bounding box center xy location of each author. In turn, the rule `SpatialLink-1` computes the Euclidean distance among each possible pair author-affiliation and asserts each of such possible combinations using the fact `link-distance`. Eventually the rule `SpatialLink-2` associates each author to

Digital Libraries'98 - Third ACM Conference on Digital Libraries, Edited by I. Witten, R. Akscyn, F. Shipman III, ACM Press, New York, pp 89-98, 1998

CiteSeer: An Automatic Citation Indexing System

C. Lee Giles, Kurt D. Bollacker, Steve Lawrence
NEC Research Institute, 4 Independence Way, Princeton, NJ 08540
{giles,kurt,lawrence}@research.nj.nec.com

ABSTRACT
We present *CiteSeer*: an autonomous citation indexing system which indexes academic literature in electronic format the advantages of traditional (manually constructed) citation indexes (e.g. the ISI citation indexes [10]), including: literature retrieval by following citation links (e.g. by providing a

Figure 6: Title is not the first text string to appear

```
(defrule XY-AffiliationLocation
  (declare (salience 5800))
  (paper-affiliations ?n ?t)
  (doc (page 1) (absline ?n) (xc ?xc) (y ?y))
  =>
  (assert (xy-affiliation ?n ?xc ?y)))

(defrule XY-AuthorLocation
  (declare (salience 5800))
  (paper-authors ?n ?t)
  (doc (page 1) (absline ?n) (xc ?xc) (y ?y))
  =>
  (assert (xy-author ?n ?xc ?y)))

(defrule SpatialLink-1
  (declare (salience 5800))
  (xy-author ?n ?xp ?yp)
  (xy-affiliation ?m ?xa ?ya)
  =>
  (assert (link-distance ?n ?m
            =(sqrt (+ (* (- ?xp ?xa) (- ?xp ?xa))
              (* (- ?yp ?ya) (- ?yp ?ya)))))))

(defrule SpatialLink-2
  (declare (salience 5800))
  (n-affiliations ?n ?)
  (n-authors ?n ?)
  (paper-authors ?na ?t)
  (not (link ?t ?))
  (link-distance ?na ?m ?d1)
  (paper-affiliations ?m ?tt)
  (not (link-distance ?na ? ?d2&:(< ?d2 ?d1)))
  =>
  (assert (link ?t ?tt)))
```

Figure 7: Spatial Based Mapping Authors to Affiliations

the (spatially) closest affiliation and asserts this by using the fact `link`.

Caveat 3: Extraction of Table of Contents. When extracting table of contents we distinguish two basic cases: (1) when section headers are numbered and (2) when they are not. We use different sets of rules according to the style adopted by the paper at hand. Thus, the first thing the rule base does is to find out whether or not section headers are numbered.

Section header numbering is a fundamental hint for a text-based extraction of table of contents; this is because the numbering is expected to follow a certain order throughout the paper and the numbers always appear at the beginning of the line. However, not infrequently, headers are not numbered, therefore an extraction based on text-parsing becomes hardly applicable. In our system we exploit the visual properties of section headers, that is, they have (1) a larger font than the text before and after and (2) a different line-space compared to the average line-space of the entire document.

Furthermore, we initially look for common header names such as "Introduction," "Overview," "Motivation," or "References" to find an initial hint for the font size of the first level headers.

Performance

We tested our system over a set of 100 scientific papers downloaded from the web (in PostScript format). We chose our papers among conference proceedings, journal papers, and few technical reports. (We did not include thesis, dissertations, or any reports with unusual layouts—we simply did not implement any expertise to read them.) Due to the more standardized layouts, journal papers are easier to process than conference papers. Therefore, conference papers were the main focus of our experiments (about 70% of the sample). In the following table we report some of the conferences and journals where papers in our sample appeared:

Conferences	Journals
AAAI94, AAAI96, AAAI97, PKDD97, PAAM96, EDBT96, EDBT97, ECML98, ICDE99, ILP98, IJCAI97, KDD97, KDD99, ICML95, SIGIR98, DL98, AA98, ...	Machine Learning, IEEE TKDE, Theor. Computer Science, ACM Trans. DAES, Future Generation of Compt. Systems, IEEE Trans. CAD, JASIS90, ...

We designed the rule set to extract the following metadata from the PostScript file: title, author(s), affiliation(s), author-to-affiliation mapping(s), and table of contents.

For each paper in our sample we tested whether the automatically extracted metadata were correct. Such test was based on a manual verification of the correctness of the extracted metadata against the original paper. Due to such manual verification, this process was lenghty and tedious, thus we had to limit our sample size to 100 papers. Table 2 summarizes the overall accuracy results.

Table 2: Metadata extraction accuracy over a total of 100 scientific papers.

	Accuracy
Title	92%
Author(s)	87%
Affiliation(s)	75%
Auth.-Affil.	71%
Table of Cont.	76%

Partially correct results were considered as wrong, for instance, in some cases not all authors were properly identified which yielded to a negative score in our performance estimation.

"Titles" and "authors" are the most accurate findings due to their simple structural description. Mappings between authors and affiliations are difficult due to the very many different ways they are encoded. Furthermore, notice that most of the papers used in our test were never seen before, that is, we may not even have considered that specific situation during the design of our knowledge base. At this time we are already reinforcing our knowledge base to deal with them. In any knowledge based system (either natural or artificial), a knowledge refinement process takes place over time as a results of things like growing experience or trial-and-errors. We follow a similar path by reinforcing our rules over time to handle previously unseen cases.

Some errors are also due to unconventional ways of coding the PostScript file. Different drivers use different ways of coding their PostScript output. The different representations may sometime confuse the low-level PostScript feature extraction process (which may even fail its task). We need further investigation in this direction.

RELATED WORK

Bishop [2] discusses the importance of document disaggregation for more effective ways to support the work practice of researchers. She discusses how the components of a paper are identified and used efficiently through the web interface DeLIver [4]. DeLIver allows the user to search for terms in specific parts of the document, e.g., "University of California" in affiliation. Documents in DeLIver are stored in SGML format.

Digital libraries have been introduced in the Internet to store a variety of documents and to provide services with the documents. These documents include journal articles, conference papers, technical reports, and dissertations. Most of these digital libraries retrieve relevant documents by a keyword-based search in human-generated databases. However, document metadata can be exploited for more efficient retrieval. For instance, a search of "relevant documents" can be performed by following the links through the citations of a paper. CiteSeer [5] provides a framework for literature retrieval by following citation links, evaluation of papers based on the number of citations, and identification of research trends. CiteSeer locates, downloads and parses Postscript files to extract citations from the papers in order to produce the citation index. However, CiteSeer does not extract other useful information such as title, authors, affiliations, and the like.

BIRD (BIbliometric Retrieval of Documents) [1] is a bibliometric query by example search engine. Given a set of pages of interest to the user, it retrieves a set of similar documents by following citation paths that pass through the given documents. It defines and computes a similarity measure, *relatedness*, between related and given set of documents based on the number and nature of citation linkages.

Although our work shares same principles with the research in the Optical Character Recognition (OCR) arena, it is quite different for the following two reasons:

- We do not process bitmaps, in fact the input for our system is a PostScript program;

- We do not try to extract single characters (or words) but higher level concepts such as title, authors, and affiliations.

CONCLUSIONS AND FUTURE WORK

A novel approach to automatic metadata extraction is introduced and its performance on a number of papers is demonstrated. The system retrieves information on the title, authors, affiliations, and table of contents of scientific papers with good accuracy. This system is also incorporated in the architecture of an *active* bibliographical digital libraries that disseminates relevant information to users.

Some avenues for ongoing and future research include: The system often fails to extract metadata from papers that have somewhat complicated structures. We are currently revising our system and making it robust in order to process papers with a variety of spatial cues and to retrieve desired information successfully. In addition, the system can be incremental. In other words, new sets of rules can be defined and added to the system to process certain types of documents (e.g., a specific scientific journal). The current system can also be extended and include additional sets of rules to extract additional information (e.g., figures, tables, references, etc.). Furthermore, we can apply machine learning techniques to complement the rule-based approach, and make the system robust.

The information retrieved from the previously processed papers can help retrieving and verifying information from new papers. For instance, all the metadata extracted from a paper (including title, authors, affiliations of the paper as well as additional information in the references) can be stored into databases and used to verify the metadata to be extracted from new papers. This will boost the accuracy of the system.

A wide variety of document types can also be considered in addition to the Postscript papers. For example, the system can be extended to handle patents, technical reports, dissertations, and broadcast news possibly in different formats (e.g., HTML, PDF).

Mining of the metadata extracted by the system is of significance. The current system can be extended to extract sentences, abstracts, paragraphs, references, and the like. These newly extracted metadata as well as previously extracted metadata (i.e., title, authors, affiliations, table of contents) can be mined in order to discover useful knowledge. In addition, the extracted metadata can be *summarized*. Automatic summarization will yield a concise, precise, well-organized and more comprehensible report for users' interests, and will set users free from the pains in reading through the huge amount of documents.

REFERENCES

1. Bibliometric retrieval of documents. http://ai.iit.nrc.ca/.
2. A.P. Bishop. Digital libraries and knowledge disaggregation: The use of journal article components. In *Digital Libraries 98*, Pittsburgh, PA, USA, 1998. ACM.
3. CLIPS: A tool for building expert systems. http://www.ghg.net/clips/CLIPS.html.
4. Deliver. http://dli.grainger.uiuc.edu/deliver.html.
5. C. Giles, K. Bollacker, and S. Lawrence. Citeseer: An automatic citation indexing system. In *Proceedings of the Third International Conference on Digital Libraries*, 1998.
6. G. Giuffrida. Graphical reasoning with augmented rule-based languages. Master's thesis, Computer Science Department, University of Houston, 1992.
7. G. Giuffrida and M. Salvemini. Loosening the connection between syntax and semantics for spatial data. In *AGILE '99, Rome, Italy*, April 1999.
8. D.B. Johnson, R.K. Taira, A.F. Cardenas, and D.R. Aberle. Extracting information from free text radiology reports. *International J. on Digital Libraries*, 1:297–308, 1997.
9. C.G. Nevill-Manning, T. Reed, and I. H. Witten. Extracting text from postscript. Technical report, Comp. Science Dept., University of Waikato, New Zealand, 1997.
10. Prescript. http://www.nzdl.org/technology.
11. Pstotext. http://www.research.digital.com/SRC/virtualpaper/pstotext.html.
12. Virtual paper. http://www.research.digital.com/SRC/virtualpaper/home.html.