TELKOMNIKA, Vol.12, No.2, June 2014, pp. 447~454
ISSN: 1693-6930, accredited A by DIKTI, Decree No: 58/DIKTI/Kep/2013
DOI: 10.12928/TELKOMNIKA.v12i2.2033

Searching and Visualization of References in Research Documents

Firnas Nadirman[1], Ahmad Ridha*[2], Annisa[2]
[1] Agency for the Assessment and Application of Technology
Jl. M.H. Thamrin No. 8 Jakarta, 10340, Indonesia
[2,] Department of Computer Science, Bogor Agricultural University
Kampus IPB Darmaga, Jl. Meranti Wing 20 Level 5 - 6, Bogor, 16680, Indonesia
*Corresponding author, e-mail: ridha@apps.ipb.ac.id

Abstract

This research aims to develop a module for information retrieval that can trace references from bibliography entries of research documents, specifically those based on Bogor Agricultural University (IPB)'s writing guidelines. A total of 242 research documents in PDF from the Department of Computer Science IPB were used to generate parsing patterns to extract the bibliography entries. With modified ParaTools, automatic extraction of bibliography entries was performed on text files generated from the PDF files. The entries are stored in a database that is used to visualize author relationship as graphs. This module is supplemented by an information retrieval system based on Sphinx search system and also provides information of authors' publications and citations. Evaluation showed that (1) bibliography entry extraction missed only 5.37% bibliography entries caused by incorrect bibliography formatting, (2) 91.54% bibliography entry attributes could be identified correctly, and (3) 90.31% entries were successfully connected to other documents.

Keywords: *research documents search, bibliography entries extraction, author relationship visualization, ParaTools*

1. Introduction

Bibliography is an important part of a research document as it lists references cited in the document. The list is useful for readers, usually fellow scientists, to locate other related documents and to know other scientists working on the topic. An important document in a field would be more likely to be cited, so it is also desirable to know the number of citations that a document has.

Numerous studies on searching research documents have been proposed [1]-[5]. A system called Bibliometric Information Retrieval System (BIRS) [1] was developed as a web-based information retrieval system for research documents. BIRS connected three types of search engines: search engine on the internet, libraries, and online databases. Another web-based system [4], DBL Browser Framework, was built to track research documents by dividing the system into three modules: GUI layer, Visualization layer, and Data layer. The search results were visualized in the form of text and graphics. The system displayed the relationship between the journals with the goal of getting journals referred by other journals. The study assumed that a highly referred journal became the basis of the development of knowledge in a particular field.

Also, research [6]-[15] has been conducted on bibliography extraction, and algorithms have been developed to recognize patterns of bibliography. One of them [8] built a small collection of functions based on Perl programming language. It used templates to extract metadata from bibliography entries. A method to extract the research documents [10] was developed using a combination of regular expressions based on heuristics and knowledge system to seek bibliography entries. Another document extraction method [15] has also been developed using the Basic Local Alignment Search Tool (BLAST). BLAST is a sequence alignment tool to find the most similar template to a protein sequence from a template database previously constructed. A database is used to store the bibliography templates. A citation transforms the templates into protein form, and BLAST is used to search for the most similar sequence in the template database.

Received February 25, 2014; Revised April 12, 2014; Accepted April 29, 2014

Research on visualization of search result [16]-[17] has also been conducted. A prototype visualization system [16] was created to enhance author searching. The system was based on author co-citation analysis and algorithms such as Kohonen's feature maps and Pathfinder networks.

Google has Google Scholar (http://scholar.google.com), a search engine to retrieve research documents. It obtains bibliography entries in online documents to measure some metrics and provides authors with a publication profile. Microsoft has also created a search engine called Microsoft Academic Search (http://academic.research.microsoft.com) with similar features. In addition, it can display Co-Author Graph, Co-Author Path, Citation Graph, and Genealogy Graph to visualize the relationship between authors.

The applications developed by Google and Microsoft can be used freely online, but they cannot be modified to extract the bibliography entries of research documents with a specific format. We also have no control on the scope of documents. This paper is to propose a method to perform extracts bibliographic data entries of research documents from a given collection of documents.

This study aims to create a module that can extract references from the bibliography entries of research documents. A method is created to recognize the bibliography entries from the research documents. Once identified, the bibliography entries are stored into a database. The database is used to build an information retrieval system for searching research documents along with their references and to visualize the relationship between the authors.

2. The Methods

This study began with collecting the research documents as PDF files. Each file was converted into plaintext file and stored in a research document database. The text was extracted and identified to get the bibliography entries. The bibliography entries were stored into the database. The database was used to build an information retrieval system of research documents. A visualization module was created to display the relationships between the authors of the documents from bibliographic entries in the database. The steps of the proposed method are shown in

Figure *1*.

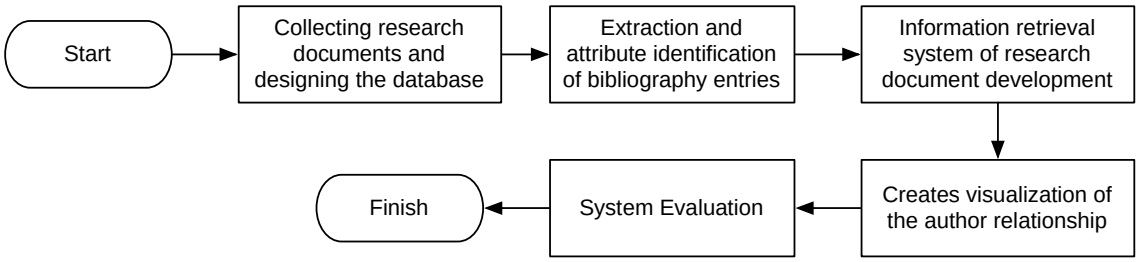


Figure 1. The Proposed Method

2.1. System Evaluation

System evaluation in this study adopts the metrics in information retrieval, namely, recall and precision [18]. This study carries out two kinds of evaluation. The first evaluation is the measurement of the success of extraction and attributes identification of bibliography entries in each document. Assume B_i is a set of bibliography entries in the i-th research document and E_i is a set of bibliography entries that are successfully extracted and identified from the i-th research document by the system, then recall can be calculated by (1) and precision can be calculated by (2).

$$Rb_i = \frac{|B_i \cap E_i|}{|B_i|} \quad (1)$$

$$Pb_i = \frac{|B_i \cap E_i|}{|E_i|} \quad (2)$$

The equations (1) and (2) are used to calculate the percentage of recall with (3) and percentage of precision with (4) from the whole bibliography entries throughout the documents.

$$PRb = \frac{\sum_{i=1}^{n} Rb_i}{n} \times 100\% \quad (3)$$

$$PPb = \frac{\sum_{i=1}^{n} Pb_i}{n} \times 100\% \quad (4)$$

The second evaluation is the measurement of the success of document relationship in the collection with a bibliography entry. Assume C_i is a set of research documents that refer the i-th bibliography entry and F_i is a set of research documents that are determined to refer to the i-th bibliography entry by the system, then recall can be calculated by (5) and precision can be calculated by (6).

$$Rc_i = \frac{|C_i \cap F_i|}{|C_i|} \quad (5)$$

$$Pc_i = \frac{|C_i \cap F_i|}{|F_i|} \quad (6)$$

The equation (5) and (6) are then used to calculate the percentage of recall with (7) and percentage of precision with (8) from the total number of entries that are connected to a document.

$$PRc = \frac{\sum_{i=1}^{n} Rc_i}{n} \times 100\% \quad (7)$$

$$PPc = \frac{\sum_{i=1}^{n} Pc_i}{n} \times 100\% \quad (8)$$

2.2 Collection

Our collection consists of 242 PDF files of Bachelor theses from Computer Science Department, Bogor Agricultural University (IPB), Indonesia, and almost all of them are written in Indonesian language. Therefore, the templates in our bibliographic entries extraction are based on IPB's writing guidelines. The evaluation is performed using this collection.

3. Results

3.1. Data Characteristics

The bibliography in our collection has several characteristics, i.e., (i) they use two columns; (ii) new chapter does not necessitate page break; and (iii) the bibliography is indicated with a title of 'DAFTAR PUSTAKA' or 'REFERENCES', and located before the appendices.

3.2. Database Design

Database design starts with identifying entities in information retrieval system of research documents. The main entity of the system is a document that has at least one author and has a bibliography. The results of the identification of entities in the database design are used to obtain a conceptual design, logical, and physical designs illustrated in

Figure *2*.

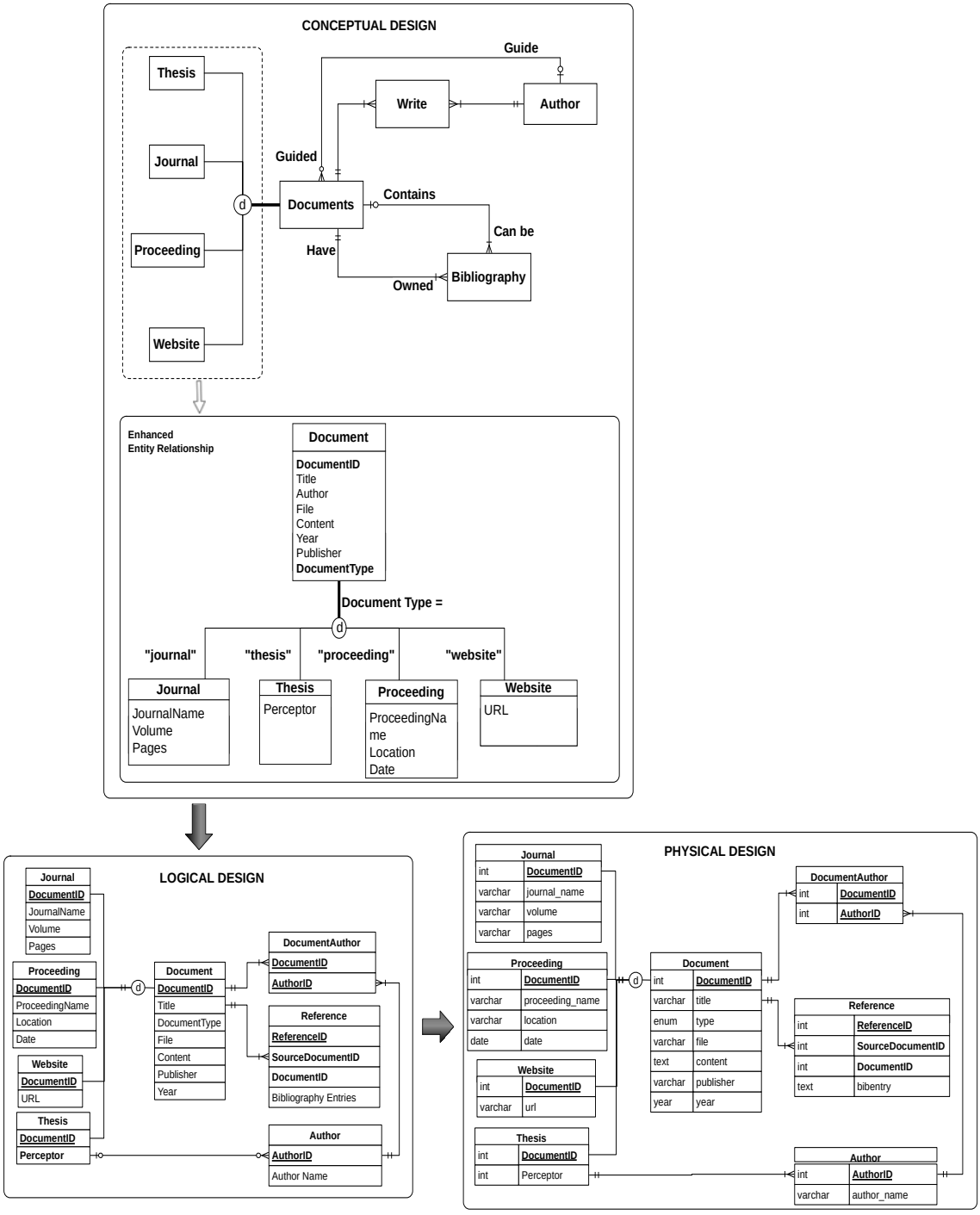


Figure 2. Database Design

3.3. Data Processing

Data processing involves three steps. The first is conversion of the research document from PDF files into text files, the second is extraction of the bibliography entries, and the third is identification of the attributes of bibliography entries. The process of converting PDF into text is using Xpdf module called pdftotext. Each PDF file is converted into two text files, i.e., raw text and layout text. Conversion into raw text format provides the text in sequential layout, while the layout text has a similar layout to the PDF file (see Figure 3).

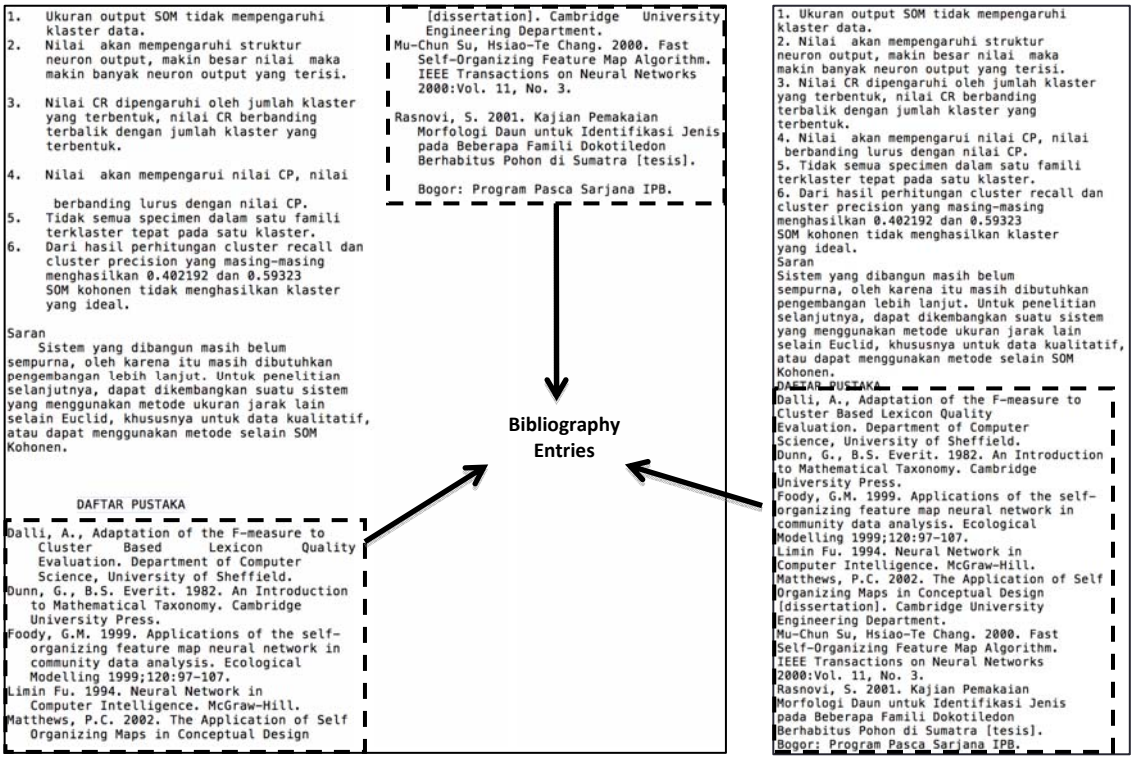


Figure 3. The Result of Conversion of PDF Files into Text

The next step is extracting the bibliography entries. Raw text is used to separate the bibliography from other parts of the document. The bibliography entries are separated using a ParaTools module, Biblio Document Parser. The module checks each line from the raw text to extract bibliographic entries. The bibliographic entries are then parsed to obtain single bibliography entry by comparing each line with the text position in the layout text.

The last step is identifying the attributes of each extracted bibliography entry by comparing it with bibliography entry templates. ParaTools Biblio Citation Parser module is used to implement this process. The results are bibliographic attributes composed of two types, i.e., general attributes and specific attributes (see Table 1).

Table 1. Bibliography Entries Attributes

Type of Attributes	Attributes Name
General	Authors Name, Title, Publisher, Year
Specific	Journal Name, Volume, Number, Proceedings Name (Publication), Location, URL

Each template of bibliography entry contains certain words to identify bibliography entry attributes. The templates can be adjusted according to the bibliography entries format. This study makes several bibliography entry templates based on IPB's writing guidelines because the documents in our testing collection are from IPB. In addition, more templates are also generated to enable extracting malformed bibliographic entries in the documents.

After identifying the attributes of a bibliography entry, all entries are stored in the database. Before storing an entry into the database, the system will run a bibliography similarity examination. The process checks the similarity of the author's name, title, and year of the extracted entries by using *Levenshtein Distance* function on all bibliography entries that are already stored in the database. Two bibliography entries are considered the same if a condition shown in Table *2* in satisfied.

Table 2. Condition of two bibliography entries attributes are considered the same

Attribute	Condition
Year	$year_1 = year_2$
Title	$\left(\frac{levenshtein\ (title_1, title_2)}{length\ (title_1)} \times 100\%\right) \leq 10\%$
Authors Name	$levenshtein\ (author_1, author_2) \leq 2$

3.4. Information Retrieval of Research Document

An information retrieval system for research documents is built to implement the modules that have been made. The system consists of three components, i.e., the backend (document entry), search engine, and user interface.

The search engine component in this study uses an information retrieval engine called Sphinx [19], which provides basic tasks such as creating index, assigning weights to the index, and searching the collection. Sphinx is configured to connect to research document database as a data source and create an index from a given table name or SQL query.

The interface of the information retrieval system consists of a form that contains a field to enter search terms and a button to initiate the search (see Figure 4). Users can enter search terms, then the system uses Sphinx search module to retrieve the relevant documents from the collection. The retrieved documents are shown in the search results sorted based on descending relevance provided by Sphinx search module (see Figure 5

Figure *5*).

Figure 4. The Interface of Information Retrieval System

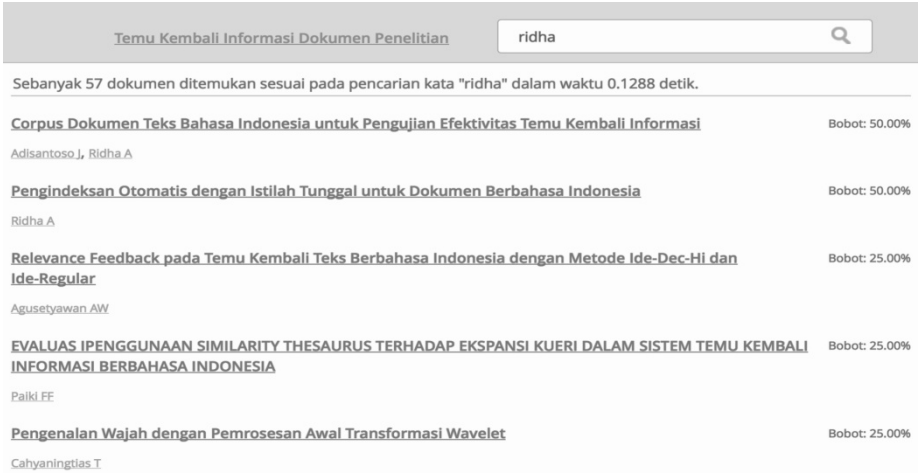
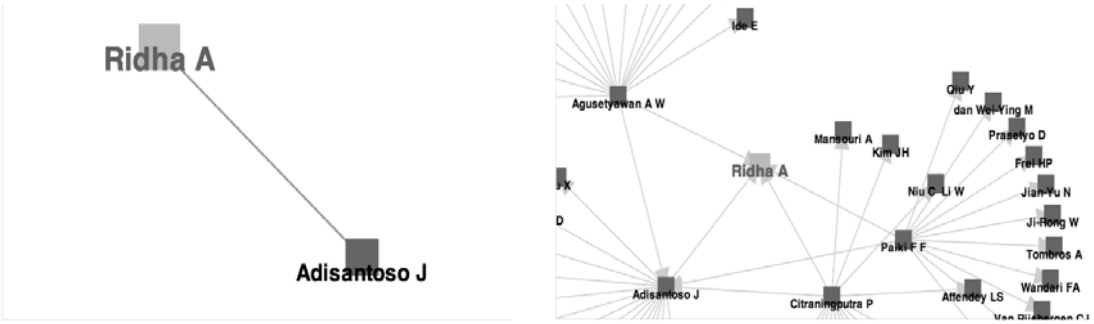


Figure 5. Search Result Interface

3.5. The Authors Relationship Visualization

The bibliography entries stored in the database can be used to visualize author relationship (see

Figure *6*). From the database, there are two types of author relationship, i.e., co-author relationship and citation relationship. The visualization is created by using JavaScript InfoVis Toolkit that utilizes the HTML 5 Canvas.



Co-Author Graph Citation Graph

Figure 6. The Authors Relationship Visualization

3.6. System Evaluation

The document processing consists of converting the PDF files into text files, extracting the bibliography entries, identifying the attributes, and storing the bibliography entries into the database. A document can take 10 seconds up to 3 hours with an average of 1 minute 20 seconds. The varied duration is due to the large number of templates used for identifying bibliography entries.

There are errors in the process of attributes identification due to malformed entries. The errors include (i) wrong placement of year or unstated year; (ii) wrong placement of publisher name; (iii) wrong format for internet addresses (not preceded with the protocol format such as http://, https://, or ftp://); and (iv) wrong format for author name.

Measurement of bibliography extraction and bibliography attributes identification in the documents is conducted by using the equation (3) and (4). It is carried out by counting the number of bibliography entries successfully extracted by the system from each document and the number of bibliography entries actually contained in the documents. The extraction process produces 94.63% bibliography entries successfully and 98.92% of the bibliography entries are extracted correctly by the system. In the attributes identification, 91.54% bibliography entry attributes are identified correctly, and 99.84% bibliography entry attributes are successfully identified.

Evaluation for relationships of bibliography entries with all the documents in the collection is carried out to measure the number of the bibliography entry referred by the documents by the system. Evaluation is performed on 50 authors most referred documents and calculated with the equation (7) and (8). The results show that 90.31% of the bibliography entries are successfully referred to the documents, and 95.19% of the bibliography entries correctly referred to documents.

4. Conclusion

This paper proposes a system to extract bibliography entries in research documents automatically. Extracted bibliography entries can be used to create two visualizations, the co-author graph and the citation relationship graph.

The process of identifying the attributes of a bibliography entry depends on the extraction process. Errors during the extraction will affect the result of attributes identification. It

is shown by our evaluation results. Around 5% of bibliography entries could not be extracted accurately. As a result, almost 9% of the attributes were not correctly identified.

Acknowledgement

This research was supported by the Computer Science Department, Bogor Agricultural University (IPB), Indonesia; and the Agency for the Assessment and Application of Technology (BPPT), Indonesia.

References

[1] Ding Y, Chowdhury GG, Foo S, Qian W. Bibliometric information retrieval system (BIRS): A web search interface utilizing bibliometric research results. *Journal of The American Society for Information Science.* 2000; 51(13): 1190–1204.
[2] Jacso P. As we may search-Comparison of major features of the Web of Science, Scopus, and Google Scholar citation-based and citation-enhanced databases. *CURRENT SCIENCE-BANGALORE*. 2005; 89(9): 1537.
[3] Alt FL, Kirsch RA. Citation searching and bibliographic coupling with remote on-line computer access. *JOURNAL OF RESEARCH of the Notional Bureau of Standards - B. Mathematical Sciences*. 1968; 72(1): 61-78.
[4] Klink S, Ley M, Rabbidge E, Reuther P, Walter B, Weber A. *Browsing and visualizing digital bibliographic data*. IEEE TCVG Symposium on Visualization. Konstanz: Eurographics Association. 2004.
[5] Stuart DG, Simpson F. Efficient literature searching: a core skill for the practice of evidence-based medicine. *Intensive care medicine*. 2003; 29(12): 2119-2127.
[6] Day MY, Tsai TH, Sung CL, Lee CW, Wu SH, Ong CS, Hsu WL. *A Knowledge-based Approach to Citation Extraction*. IRI-2005 IEEE International Conference. 2005.
[7] Gardfield E. Citation analysis as a tool in journal evaluation. *Science*. 1972; 178(60): 471-479.
[8] Jewell M. ParaTools Reference Parsing Toolkit-Version 1.0 Released. D-lib Magazine. 2003; 9(2).
[9] Hetzner E. *A simple method for citation metadata extraction using hidden markov models*. JCDL '08 Proceedings of the 8th ACM/IEEE-CS joint conference on Digital libraries. New York: ACM. 2008.
[10] Gupta D, Morris B, Catapano T, Sautter G. A new approach towards bibliographic reference identification, parsing and inline citation matching. *IC3 of Communications in Computer and Information Science.* 2009; 40: 93-102.
[11] Ohta M, Daiki A, Atsuhiro T, Jun A. *CRF-based bibliography extraction from reference strings focusing on various token granularities*. 10th IAPR International Workshop on Document Analysis Systems (DAS). 2012: 276-281.
[12] Staelin C, Elad M, Greig D, Shmueli O, Vans M. Biblio: automatic meta-data extraction. *International Journal of Document Analysis and Recognition (IJDAR)*. 2007; 10(2): 113-126.
[13] Peng F, McCallum A. *Accurate Information Extraction from Research Papers using Conditional Random Fields*. Human Language Technology conference / North American chapter of the Association for Computational Linguistics annual meeting (HLT/NAACL) 2004. 2004: 329-336.
[14] Han H, Giles C, Manavoglu E, Zha H, Zhang Z, Fox E. *Automatic Document Meta-data Extraction using Support Vector Machines*. Proceedings of Joint Conference on Digital Libraries. 2003.
[15] Huang IA, Ho JM, Kao HY, Lin SH. Extracting citation metadata from online publication lists using BLAST. *PAKDD of Lecture Notes in Computer Science.* London: Springer. 2004; 3056: 539-548.
[16] Lin X, White HD, Buzydlowski J. Real-time author co-citation mapping for online searching. *Information Processing and Management.* 2003; 39: 689-706.
[17] Anegón FM, Quesada BV, Solana VH, Rodríguez ZC, Álvarez EC, Fernández M, José F. A new technique for building maps of large scientific domains based on the cocitation of classes and categories. *Scientometrics*. Budapest: Kluwer Academic Publisher. 2004; 61:129-145.
[18] Baeza-Yates R, Ribeiro-Neto B. Modern Information Retrieval: The Concepts and Technology Behind Search. New Jersey: Pearson Higher Education. 2011: 135.
[19] Ali A. Sphinx Search Beginner's Guide. Packt Publishing Ltd. 2011.