AUTOMATIC COLLECTION OF AUTHORSHIP INFORMATION FOR WEB PUBLICATIONS

Daniel Lichtnow[1,2], Ana Marilza Pernas[1,3], Edimar Manica[1], Fahad Kalil[1]
José Palazzo M. de Oliveira[1] and Valderi Reis Quietinho Leithardt[1]
[1]*Instituto de Informática, UFRGS Universidade Federal do Rio Grande do Sul, Porto Alegre, RS, Brazil*
[2]*Centro Politécnico, UCPel Universidade Católica de Pelotas, Pelotas, RS, Brazil*
[3]*Departamento de Informática, UFPel Universidade Federal de Pelotas, Pelotas, RS, Brazil*

Keywords: Information quality, Expertise identification, Information extraction.

Abstract: The authorship is an important criteria to evaluate content quality. Frequently, Web users have to spend a lot of time in Web searchers to find information about author's expertise. This paper presents an approach to help Web users in this task. The approach consists of: a set of techniques to extract information about authors from Web and an architecture of an extraction tool. An application scenario is presented, in which the user can read details about a specific author of a Web page when reading the document.

1 INTRODUCTION

Although some mechanisms have been created to identify the quality of Web pages, normally related to search engines, as the *PageRank* algorithm (Brin and Page, 1998), the final quality evaluation is a task that Web users must perform individually.

One common content quality criteria is the authorship. An experienced user will try to discover some information about the authors to validate the Web page quality. Unfortunately, many Web pages don't include author's information. In that case, the users have to look for author's information elsewhere. This task causes great loss of time by the user.

The present work proposes an approach to help Web users in this task, presenting a set of information extraction techniques to identify authors' information on the Web. To easily understand the functionality of our proposal, an architecture of a tool was defined. Our objective is to show an author's curriculum with information easily to be understood by Web users without further explanation (e.g. number of citations, subject related to author). Besides, the proposal is to present this information to Web users while they are reading the content of a Web page.

The paper is organized as follows: section 2 presents some related works; section 3 describes the model that contains relevant author's information; section 4 presents an overview of the tool's architecture the process of extraction and techniques used. Section 5 presents an application scenario. Finally, section 6 presents the conclusion and future works.

2 RELATED WORK

This work focuses in identify information about authors' expertise or authors' qualification using available information in the Web. The related works includes aspects of quality criteria for Web sites/pages, information extraction and expertise identification.

2.1 Quality Criteria for Web Sites or Pages

Ahead of criteria based on pages reputation, like *PageRank*, there are some organizations that award quality seals for Web sites following some politics (e.g. *HONCode*[1]). However, in general a user (even an inexperience user) must to do the final quality evaluation of the content quality. This task is time-consuming and with a limited Web coverage.

Considering this problem, Tim Berners-Lee proposed the *"Oh, yeah?"* button. This functionality presents in Web browsers a button that explains to a Web user *"a list of assumptions on which the trust is*

Lichtnow D., Marilza Pernas A., Manica E., Kalil F., Palazzo M. de Oliveira J. and Reis Quietinho Leithardt V.
AUTOMATIC COLLECTION OF AUTHORSHIP INFORMATION FOR WEB PUBLICATIONS.
DOI: 10.5220/0002773603390344
In *Proceedings of the 6th International Conference on Web Information Systems and Technology (WEBIST 2010)*, page
ISBN: 978-989-674-025-2

based" (Berners-Lee, 1997). The problem in this proposition is the difficulty of implementation, with the lack of semantic representation in Web content.

In Bizer and Cyganiak (2009) this functionality was implemented but aspects related with how to extract quality information from Web isn't emphasized - the work uses functionalities developed by Huynh, Mazzocchi and Karger (2006). In both works, aspects related with authorship are not emphasized.

2.2 Information Extraction

The aim of Information Extraction is to reduce the information present in a document to a tabular form (Kayed and Shaalan, 2006). Some techniques are trying to do information extraction using small sets of domain-independent patterns.

One example is the *KnowItAll* system, which for a specific relation (defined in advance) tries to identify instances. *KnowItAll* uses patterns like "*X is a Y*" to find a set of possible instances. For example, for a class 'Author' the phrase "*X is an author*" in a Web page indicating that *X* is a possible author of the content (Etzioni et. al., 2008).

Some works extracts quality indicators from Web pages. In Stamatakis (2007) is presented a tool to assist members of organizations that give quality seals to Web sites. The work identifies ads on Web pages about health. According to some organizations, ads might compromise the impartiality. Wang and Liu (2007) try to extract various indicators defined by organizations like *HONCode*. These works does not identify the author's expertise and just consider information present in the Web page that is being evaluated.

2.3 Expertise Identification

An Expert Search is a system that tries to identify persons with expertise in some specific area. This class of system looks for evidence of expertise in documents (written or read), e-mails, curriculum vitae, etc (MacDonald and Ounis, 2006).

In Serdyukov, Ali and Hiemstra (2008), for example, the authors consider that experts are popular not only locally in the organizational context, but in other Web spaces (e.g. news, blogs, and academic libraries). Thus, they extract expertise evidence from search engines using specific queries for each expert candidate (one example of information considered is the number of inlinks to Web pages related to the expert candidate).

Jiang, Han and Lu in (Balog, 2008) uses Web information to identify experts and discuss how to build Web search queries to search for information relevant about experts. Also Serdyukov and Hiemstra in (Balog, 2008) discuss how to use Web as an evidence of expertise and presents some ways to extract evidence from distinct Web repositories (blogs, news, academic information, intranet, Web search). These authors discuss how to create a global rank considering evidences from different sources.

2.4 Our Approach

Our proposal is to use a set of information extraction techniques to obtain information (see section 3) about a person in/from Web. We believe that an application that uses Web data to indicate some reliable information have to initially looks for information located in reliable repositories (well known repositories). Examples are *DBLP* and *CiteSeer* (computer science) and *PubMed*[2] (health). After obtain initial data, following tendencies presented in Section 2.3, the information is complemented with information from other Web sources.

It's important to explain that the extraction process (that will be more explained in the section 4) is performed automatically, just in time, when user needs information about an author.

Our proposal includes a tool that shows author's information to Web users while he/she is accessing the Web page. Take in care issues related to costs in acquiring information and the fact of a Web user does not have much time to think and take decisions about quality while browsing process, quality indicators must to be produced quickly and information overload must to be avoided.

3 AUTHOR'S MODEL

The author's model contains useful information to evaluate author's expertise. In our work, the idea is to combine some vocabularies (*Dublin Core*[3] and *FOAF*[4]) to describe aspects related to author expertise, like in Aleman-Meza (2007). The model (Figure 1) respects the vocabulary specification with some exceptions:

- *hindex*. Contains the author's *h-index* (Hirsch,2005);
- *belongsToOrganization.* Contains the organization name where the author works/study;
- *numberOfReferencesToOrganization.* Contains the number of Web pages in which the organization is found;

- *numberOfCitations*. Contains the number of citations/references to a specific publication founded in papers;
- *positionScholar*. Contains the paper position, considering Google Scholar ranking. This ranking considers just publications related to the same area;
- *avgCitations*. Contains the average of citations considering the top 1.000 publications (1.000 is a Google Scholar limit). This ranking considers just publications related to the same area.

```
<foaf:Person rdf:about="">
    <foaf:name/>
    <foaf:title/>
    <foaf:firstname/>
    <foaf:surname/>
    <foaf:mbox />
    <foaf:workplaceHomepage />
    <foaf:homepage />
    <foaf:phone/>
    <belongsToOrganization />
    <numberOfReferencesToOrganization/>
    <hindex/>
</foaf:Person>
<foaf:publications>
    <foaf:Document rdf:about="../pub/p11">
            <dc:identifier/>
            <dc:title />
            <dc:date  />
            <dc:creator "co-author1" />
            <dc:creator "co-author2" />
            <dc:language />
            <dc:publisher/>
            <dc:subject/>
            <dc:source/>
            <numberOfCitations />
            <positionScholar/>
            <avgCitationsArea/>
            </foaf:Document>
    </foaf:publications>
```

Figure 1: Author's Model.

For each data extracted, is stored its provenance - the origins of data and the process by which it were retrieved (Figure 2) – which could be required by user (section 5) if the user want. This kind of information is important since its give more confidence to Web user. It's also important to note that some errors could occur in the extraction process (see section 6) and provenance information must be used to indicate that to Web user (Hartig, 2008). Information about access method (*HTTP-based, API-based*, etc), source (*URL*) and data provider are stored. This model is based in Hartig (2008). Figure 2 shows provenance information about *dc:title*.

```
<provenance_information>
    <access_method API_based/>
    <dc:source www.ncbi.nlm.nih.gov/pubmed/>
    <dc:publisher PubMed/>
</provenance_information>
```

Figure 2: Provenance Information.

4 APPLICATION

The following sections give information about tool's architecture and information extraction techniques.

4.1 Architecture

The proposal architecture consists of an extension of Web Browser that, when invoked by a Web user, extracts author's names and shows information about one specific selected author. This extension is going to be implemented following Bizer and Cyganiak (2009). The basic architecture has 3 modules (see Figure 3):

- The module 1 identifies and extracts the author's names from a Web page;
- The module 2 extracts author´s information from Web and generate author's model;
- The module 3 shows information to users.

Details about the implementation of these modules (especially module 1) are beyond of the scope of this paper. In the case of module 1, an author name could be identified as in Etzioni (2008).

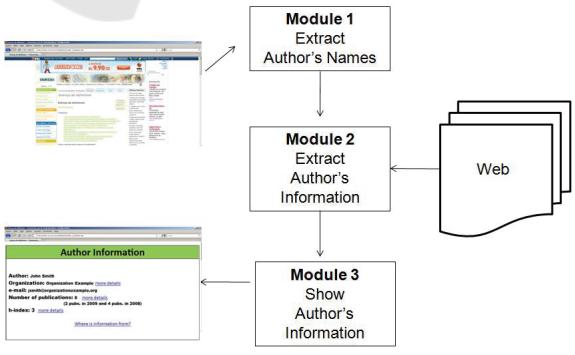

Figure 3: Tools' Architecture.

Details about the extraction techniques (module 2 in the Figure 3), which are more related to present work, are given in section 4.2. Relating with the module 3, section 5 presents a scenario of applica-

tion where is possible to understand what the module must do and how.

4.2 Extraction Process

The application starts looking for author's information in a specific digital library (in our work, *PubMed*), which provide information pre-evaluated, needed as a start point. After, a specialized search engine is used (in our work, Google Scholar) and, after that, a generic search engine is used (in that case, Google). In each activity, some support tools are used. The *SAXON*[5] is used to generate the RDF/XML file with author's model. Concerning the activities related with data extracting from specialized and generic search engine, is used the tool *WebHarvest*[6].

In the case of *PubMed*, after informs an author name, it is possible to obtain a XML file with the results founded for this author. In resulting XML file, there is information related to author's publications: title, co-authors of the paper, author's organization and keywords. To present this informative data, is employed the *MeSH* (*Medical Subject Headings*) vocabulary[7].

After using the title of the publication, the tool retrieves the number of citations from Google Scholar. To retrieve any information that is not present in *PubMed* and in Google Scholar, the tool utilizes Google as a generic Web-search engine. The strategy used differs according with the data retrieved. For example, in the case of e-mail - when this e-mail is not present in the publication-, the strategy consists in the following steps.

At first, a search is done with a Web search engine (Google), using: the name of the author; a set of keywords related to his/her publications (the keywords are *MeSH* descriptors - the 3 more frequently); author's institution name and indicatives of e-mail presence (string like *e-mail*, *contact*, etc). This approach increases the precision, reducing homonymous problems.

After using strings like *e-mail* and *contact*, the author's e-mail is showed in the Web Page resume generated by Google, so it is possible to extract an e-mail from a Web page with Google's results without access and process the Web pages with this information. This strategy represents a performance gain. After, from each page, are extracted strings that represent e-mails (author@xxx.xx).

Finally, using each e-mail founded, a new search is made using Google. Basically, the tool retrieves the number of pages that contain e-mails and the number of Web pages that contains e-mail and author's name. These values are used to calculate a rate (1).

$$r = nea*100/ne \quad (1)$$

Where *ne* is the number of times where an e-mail was found for the search engine and *nea* is the number of times where an e-mail was found with author's name. The e-mail with higher rate is considered the author's e-mail. Thus, it is considered that an e-mail with author's name has higher probability to be the real e-mail's author.

To discover the author´s home page, is used a query with author's name, author's organization and a set of keywords related to his/her publications. The process is similar to e-mail retrieval. In the future, this process could be improved considering others techniques as in Xi and Fox (2002).

One important point is that an inexperience user cannot evaluate some information about an author, as the number of citations, for example, because the user does not know if a specific number of citations is high or low. Thus, the idea is show to users some information to facilitate the evaluation process. To give this information, some strategies were defined. The real convenience of these strategies needs to be evaluated in a near future, especially in terms of computational cost.

Concerning citations number, the proposal is to give an average of citations related to the same area. To obtain this information, we utilize the search engine Google Scholar. The strategy consists on retrieves documents from Scholar using the keywords related to a specific author's publication. After that, the author's publications are located on this set (*positionScholar*, section 3). At the same time, the average citations of this set are calculated (*avgCitations*, section 3).

In the same sense, using Google Scholar's information is possible to show information about author's h-index. However, this information must to be explained to users. In this sense, one possibility is to compare the author's h-index with others authors (e.g. authors who the user have been looked up before).

5 SCENARIO

This scenario shows how information extracted from Web can be used on partial implementation of the *"Oh, yeah?"* button (see section 2.1).

Initially, a user accesses a Web page about Alzheimer's disease, which has the site author's name. The user, who is interested in an evaluation of the

page quality, request information about this author. This request is made through a button implemented as an extension of a Web browser. The author's name is extracted from the Web page. More than one author may be identified, so the Web user receives a list of authors (section 4.1).

Using the author's name, the application accesses *PubMed* and automatically retrieves author's publications. The author's affiliation is retrieved too. For each publication the application extracts the number of citations from Google Scholar.

Another search engine (Google) retrieves more information about the author (e.g. author's home page). The process of extraction and filtering is finished and the information about the author is presented to user. Initially, just a resume is displayed (Figure 4).

Figure 4: Author's Information.

The user may request more detailed information. If a user asks for more details about related publications, the application will show the content presented in Figure 5 (for each publication).

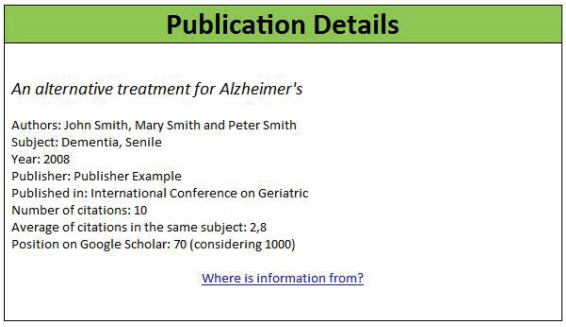

Figure 5: Information about publication.

A user can request details about provenance. This could be important because data quality involves the analysis of provenance (Hartig, 2008). In our scenario, the application will show the information according with Figure 6. The same occurs with organization and h-index.

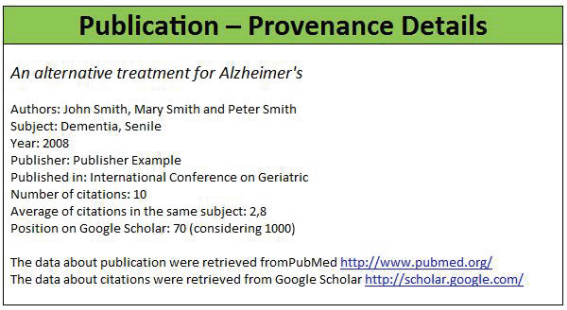

Figure 6: Information about publication – provenance.

6 CONCLUSIONS

One important indicator of quality content is the authorship. In this sense, our work defines:

- A model with data about author's and provenance's information, using known vocabularies;
- An architecture of a extraction tool with a set of extraction techniques to populate the model;
- An application scenario that shows how could be possible to shows relevant information to Web users.

The contribution of our work is related to a definition of a functionality similar to the "*Oh, yeah?"* button, using information extraction techniques. This functionality was implemented in Bizer and Cyganiak (2009), but the issues related to information extraction are not emphasizes. In future works some of patterns defined in Bizer and Cyganiak (2009) should be considered. These patterns include the use of *WIQA-PL Information Quality Assessment Framework - Policy Language*. These features will facilitate the process of explanation according with is described in section 5.

Another important point is that the work follows an actual tendency to search for expertise evidence on Web (section 2.3). In future works, some issues must to be deeply considered:

- The same entity may appear with a variety of names. In authors case, sometimes the complete name is used, sometimes just part or initials;
- The same string may refer to distinct entities;
- There is incorrect information on Web;
- There is contradictory information (e.g. number of citations in ACM and in Scholar);
- There are multiple opinions present on Web.

Some recent works address some of these problems (Balog et al, 2009) (Etzioni et al, 2008).

In our work, some of these problems (e.g. homonym) are solved (partially) using strategies described in section 4.2 (e.g. e-mail extraction).

In function of these situations, we believe that provenance information must contain details about how these situations were solved. In this sense, in (Borges, Galante and Gonçalves, 2008) information about publication are extracted from 3 distinct digital libraries, some differences are detected, and information about provenance conflict resolution are stored.

ACKNOWLEDGEMENTS

This work is partially supported by CNPq, Conselho Nacional de Desenvolvimento Científico e Tecnológico, Brazil and CAPES, Coordenação de Aperfeiçoamento de Pessoal de Nível Superior, Brazil.

REFERENCES

Aleman-Meza, B., Bojārs, U., Boley, H., Breslin, J. G., Mochol, M., Nixon, L. J., Polleres, A., and Zhdanova, A. V. (2007). Combining RDF Vocabularies for Expert Finding. In *Proc. of the 4th European Conference on the Semantic Web: Research and Applications*, pages 235-250, Berlin, Springer-Verlag.

Balog, K. (2008). The SIGIR 2008 workshop on future challenges in expertise retrieval (fCHER). *SIGIR Forum* 42(2) 46-52.

Balog, K., Azzopardi, L. A. and Rijke de M. (2009) Resolving person names in Web people search., in Weaving Services and People on the World Wide Web, pages 301-323 Springer, Berlin, Springer-Verlag.

Berners-Lee, T. (1997) Cleaning up the User Interface, Section—The "Oh, yeah?"-Button, Retrieved May 4, 2009, from http://www.w3.org/DesignIssues/UI.html

Bizer, C. and Cyganiak, R. (2009). Quality-driven information filtering using the WIQA policy framework. *Web Semant.* 7(1).

Borges, E. N., Galante, R. de M., Gonçalves, M. A. (2008). Uma Abordagem Efetiva e Eficiente para Deduplicação de Metadados Bibliográficos de Objetos Digitais. In: *Proc. of the XXIII SBBD,* pages 76-90, São Paulo, Brazil, SBC.

Brin, S. and Page, L. (1998). The anatomy of a large-scale hypertextual Web search engine. *Comput. Netw. ISDN Syst.* 30(1-7), 107-117.

Etzioni, O., Banko, M., Soderland, S., and Weld, D. S. (2008). Open information extraction from the Web. *Commun. ACM* 51(12), 68-74.

Hartig, O. (2009). Provenance Information in the Web of Data, in *Proc. of the Linked Data on the Web Workshop at WWW*

Hirsch, J. E. (2005). An index to quantify an individual's scientific research output. *PNAS* 102 (46), 16569–16572

Huynh, D., Mazzocchi, S., and Karger, D. (2007). Piggy Bank: Experience the Semantic Web inside your Web browser. *Web Semant.* 5(1), 16-27.

Kayed, M. and Shaalan, K. F. (2006). A Survey of Web Information Extraction Systems. *IEEE Trans. on Knowl. and Data Eng.* 18(10), 1411-1428.

Macdonald, C. and Ounis, I. (2006). Voting for candidates: adapting data fusion techniques for an expert search task. In *Proc. of the 15th ACM international Conference on information and Knowledge Management*, pages 387-396 New York, NY, ACM Press.

Stamatakis, K. et al. AQUA, a system assisting labelling experts assess health Web resources. In *Procs. of iSHIMR*, 2007.

Serdyukov, P., Aly, R., Hiemstra, D. University of Twente at the TREC 2008 Enterprise Track: Using the Global Web as an expertise evidence source. In *Procs. of 16th TREC*.

Wang Y., Liu Z. (2007) Automatic detecting indicators for quality of health information on the Web, International Journal of Medical Informatics, 76(8), 575-582.

Xi, W. and Fox, E. A. (2002) Machine Learning Approach for Homepage Finding Task *In Procs. of* 9th International Symposium on String Processing and Information Retrieval, pages 145-159.

[1] *http://www.hon.ch/*
[2] *http://www.ncbi.nlm.nih.gov/pubmed/*
[3] *http://dublincore.org/*
[4] *http://xmlns.com/foaf/spec/*
[5] *http://saxon.sourceforge.net*
[6] *http://web-harvest.sourceforge.net/*
[7] *http://www.nlm.nih.gov/mesh/*