Personalizing Bibliographic Recommendation under Semantic Web Perspective[1]

Giseli Rabello Lopes, Maria Aparecida Martins Souto,
Leandro Krug Wives, José Palazzo Moreira de Oliveira

Instituto de Informática – Universidade Federal do Rio Grande do Sul (UFRGS)
Caixa Postal 15.064 – 91.501-970 – Porto Alegre – RS – Brasil
{grlopes, souto, wives, palazzo}@inf.ufrgs.br

Abstract. This paper describes a Recommender System for scientific articles in digital libraries for the Computer Science researchers' community. The system employs the Dublin Core metadata standard for the documents description, the XML standard for describing user profile, which is based on the user's *Curriculum*, and on service and data providers to generate recommendations. The main contribution of this work is to provide a recommendation mechanism based on the user academic curriculum reducing the human effort spent on the profile generation. In addition, this article presents and discusses some experiments that are based on quantitative and qualitative evaluations.

Keywords: Recommender System, User profile, Digital library, Semantic Web.

1 Introduction

Today, the scientific publications can be electronically accessed as soon as they are published on the Web. The main advantage of open publications is the minimization of the time and the space barriers inherent to the traditional publication process. In this context, Digital Libraries (DLs) have emerged as the main repositories of digital documents, links and associated metadata. This is a change in the publication process and has encouraged the development of automatic systems to rapidly explore and obtain required information. EPrints [10], DSpace [22], Kepler [16], CITIDEL [4] and BDBComp [12] are examples, among others. Usually, users with different knowledge levels, experiences and interests receive the same information as the answer to their queries. Aiming to avoid these problems, Recommender Systems in DLs have been proposed and developed (e.g., ARIADNE, ResearchIndex, CyberStacks and ARP).

The Recommender Systems involve information personalization. The personalization is related to the ways in which information and services can be tailored to match the specific needs of a user or a community [3]. The human-centered demand specification is not an easy task. One experiences this difficulty when trying to find scientific papers in a good indexing and retrieval system such Scholar Google.

[1] This work was partially supported by the project Pronex FAPERGS, grant 0408933, and Project PerXML CNPq, grant 475743/2004-0. The first and the last authors are partially supported by CNPq.

The query formulation is complex and the fine tuning of the user requirements is a time-consuming task. Few researchers have enough time to spend some hours a week searching for, eventually, new papers in theirs specific research area. This functionality, the query specification, may be reached by the analysis of the user activities, history, information demands, etc.

This article presents a Recommender System to Computer Science researchers and academics. The information and service provided by the system are based on the *Lattes Curriculum Vitae* (*Lattes* CV) [13], a system that registers all the researcher's academic activities and publications with a XML output. The main contribution of this work is to provide a recommendation mechanism based on the user academic curriculum reducing the human effort spent on the profile generation.

The article is organized as follows. We start giving an overview of the background literature and concepts, then the recommender system and detail its architecture and techniques. Finally, we present some quantitative and qualitative experiments to evaluate and validate our system and discuss the results and conclusions of our work.

2 Background

The semantic Web technologies promote an efficient and intelligent access to the digital documents on the Web. The standards based on metadata to describe information objects have two main advantages: computational efficiency during the information harvesting process and interoperability among DLs. The first is a consequence of the increasing use of Dublin Core (DC) metadata standard [8]; the latter has been obtained as a result of the OAI initiative (Open Archives Initiative) [17]. DC metadata standard was conceived with the objective of defining a minimal metadata set that could be used to describe the available resources of a DL. This standard defines a set of 15 metadata (Dublin Core Metadata Element Set - DCMES). Table 1 shows these elements and their associated descriptions.

Table 1. Dublin Core Metadata Element Set, adapted from [8].

Element Name	Description
dc:title	A name given to the resource.
dc:creator	An entity primarily responsible for making the content of the resource.
dc:subject	A topic of the content of the resource.
dc:description	An account of the content of the resource (e.g., abstract).
dc:publisher	An entity responsible for making the resource available.
dc:contributor	An entity responsible for making contributions to the content of the resource.
dc:date	A date of an event in the lifecycle of the resource (typically, *dc:date* will be associated with the creation or availability of the resource).
dc:type	The nature or genre of the content of the resource.
dc:format	The physical or digital manifestation of the resource.
dc:identifier	An unambiguous reference to the resource within a given context (e.g., URL).
dc:source	A reference to a resource from which the present resource is derived.
dc:language	A language of the intellectual content of the resource.
dc:relation	A reference to a related resource.
dc:coverage	The extent or scope of the content of the resource (typically, *dc:coverage* will include spatial location).
dc:rights	Information about rights held in and over the resource.

The main goal of OAI is to create a standard communication way, allowing DLs around the world to interoperate as a federation [21]. The DL metadata harvesting process is accomplished by the OAI-PMH protocol (Open Archives Initiative Protocol for Metadata Harvesting) [18], which defines how the metadata transference between two entities, *data* and *service providers*, is performed. The *data provider* acts by searching the metadata in databases and making them available to a *service provider*, which uses the gathered data to provide a specific service.

Considering that a Recommender System concerns with information personalization, it is essential that it copes with user profile. In our work, the user profile is obtained from the user's *curriculum vitae*, i.e., *Lattes* CV. The *Lattes* CV is a Brazilian Research Council (CNPq) initiative and offers a standard database of researchers and academics curricula. The platform is used: (i) to evaluate the competency of researchers and academics for grant concession; (ii) to select committees' members, consulting people and counselors; and (iii) to assist the evaluation processes of research and post-graduate courses. Thus, all the research personnel must have an updated CV in order to submit research projects or to receive any kind of support from the agencies. It is the main instrument to support the researcher evaluation, as the CV is publicly accessible at the CNPq site the data may be verified by the research community. As a consequence this is the best source for the user profile creation.

Table 2. Lattes CV Metadata Element Subset, adapted from [13].

Metadata Category	Description
Personal information	This category contains general information about the user. Some metadata are: - *cv:name* - *cv:personal-address* - *cv:professional-address*
University degrees	This category contains user's information about his/her academic degrees. Some metadata are: - *cv:graduation-level* (Undergraduate, Master graduate, and PhD. graduate) - *cv:graduation-year* - *cv:monograph-title* - *cv:monograph-keywords* - *cv:monograph-area* - *cv:monograph-advisor*
Language proficiency	This category contains information about the languages that user has any proficiency. Some metadata are: - *cv:language* - *cv:language-skill* (reading, writing, speaking, comprehension) - *cv:language-skill-level* (good, reasonable or little)
Bibliographic production	This category provides user's information about his/her bibliographic publications in proceedings, journals, book chapters, etc. Some metadata are: - *cv:article-title* - *cv:article-keywords* - *cv:article-language* - *cv:article-year*

Table 2 shows a *Lattes* CV metadata elements subset. It presents the categories used in this work to support the recommendation process and their associated descriptions. To better comprehension, the prefix "cv:" is used in this work to reference the metadata elements.

According to [11], there are three different methodologies used in Recommender Systems to perform recommendation: (i) *content-based*, which recommends items classified accordingly to the user profile and early choices; (ii) *collaborative filtering*, which deals with similarities among users' interests; and (iii) *hybrid approach*, which combines the two to take advantage of their benefits. In our work, the *content-based* approach is used, once the information about the user is taken from the *Lattes* CV and is matched with the DC metadata that best describes the articles of a DL.

The recommendation process can be perceived as an information retrieval process, in which user's relevant documents should be retrieved and recommended. Thus, to perform recommendations, we can use the classical information retrieval models such as the Boolean Model, the Vector Space Model (VSM) or the Probabilistic Model [20, 1, 9]. In this work, the VSM was selected since it provides satisfactory results with a convenient computational effort. In this model, documents and queries are represented by terms vectors. The terms are words or expressions extracted from the documents and from queries that can be used for content identification and representation. Each term has a weight associated to it to provide distinctions among them according to their importance. According to [19] the weights can vary continuously between 0 and 1. Values near to 1 are more important while values near to 0 are irrelevant.

The VSM uses an *n*-dimensional space to represent the terms, where *n* corresponds to the number of distinct terms. For each document or query represented, the weights represent the vector's coordinates in the corresponding dimension. The VSM principle is based on the inverse correlation between the distance (angle) among term vectors in the space and the similarity between the documents that they represent. To calculate the similarity score, the cosine (Equation 1) can be used. The resultant value indicates the relevance degree between a query (Q) and a document (D), where w represents the weights of the terms contained in Q and D, and t represents the number of terms (size of the vector). This equation provides ranked retrieval output based on decreasing order of the ranked retrieval similarity values [19].

$$Similarity(Q, D) = \frac{\sum_{k=1}^{t} w_{qk} \cdot w_{dk}}{\sqrt{\sum_{k=1}^{t} (w_{qk})^2 \cdot \sum_{k=1}^{t} (w_{dk})^2}} \quad \textbf{(1)}$$

The same equation is widely used to compare the similarity among documents, and similarly, in our case, Q represents the user profile and D the documents descriptors that are harvested in the DL (see Section 3.2 for details). The term weighting scheme is very important to guarantee an effective retrieval process.

The results depend crucially of the term weighting system chosen. In addition, the query terms selection is fundamental to obtain a recommendation according to the user necessities. Our research is focused in the query terms selection and weighting.

Any person that experienced a bibliographical retrieval may evaluate the process complexity and the difficulty to find the adequate articles. The central idea is to develop an automated retrieval and recommendation system where the price for the user is limited to the submission of an already existing *Lattes* XML CV at subscription time. For a researcher from a country without a similar CV system it will be necessary to substitute the XML CV upload for a Web extracting module that will try to recover the needed metadata from Web pages and, perhaps, from the Scholar Google or other equivalent systems.

3 The Recommender System

Our system focuses on the recommendation of scientific articles to the Computer Science community. The information source to perform recommendations is the Brazilian Computer Science Digital Library (BDBComp) [2], while the user profile is obtained from a Lattes CV subset. However, any DL repository providing DC metadata and supporting the OAI-PMH protocol can be used as a source. An alternative to the user profile generation is under development. This alternative approach is composed by an information retrieval system to gather data from personal homepages and other data sources in order to replace the Lattes CV where the Lattes personal data is not be available.

A DL repository stores digital documents or its localization (web or physical), and the respective metadata. A DL *data provider* allows an agent to harvest documents metadata through the OAI-PMH protocol. Our system handles the documents metadata described with XML in DC standard. The *Lattes* CV and the DC metadata are described as an XML standard document according to the W3C XML Schema, which can be found in [15] for the Lattes CV and in [7] for the DC standard.

3.1 The Recommender System Architecture

In this section we present the architecture elements of our system and its functionalities (Fig. 1). To start the process, the users must supply their *Lattes* CV in the XML version to the system. Whenever a user makes its registration in the system and sends his *Lattes* CV (1), the ***XML Lattes to Local DB*** module is activated and the information about the user's interests is stored in the local database named *User Profile* (2)**.** Then the ***Metadata Harvesting*** module is activated to update the local database *Articles Metadata*. This module makes a request to a DL *data provider* to harvest specific document metadata. It receives an XML document as response (3) and the ***XML DC to local DB*** module is activated (4). This module extracts the relevant metadata to perform the recommendations from the XML document and stores it in the local database named *Articles Metadata* (5). Once the user profile and the articles metadata are available in the local database, the ***Recommendation*** module can be activated (6). The focus is to retrieve articles of a DL that best matches the user profile described through the *Lattes* CV (7).

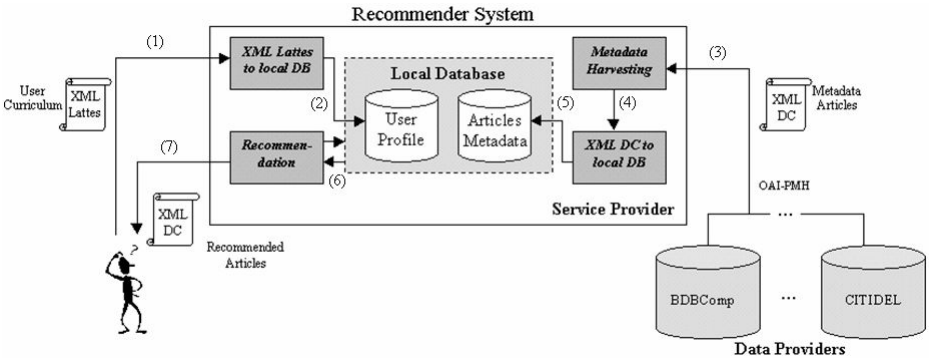

Fig. 1. The recommender system architecture.

3.2 The Recommendation Model

As stated before, the recommendation is based on the VSM model. The *query vector* is built with the terms parsed from: (i) the *cv:monograph-title* and *cv:monograph-keywords* of the user university degrees; and (ii) *cv:article-title* and *cv:article-keywords* of the bibliographic productions in *Lattes* CV (table 2). The parser ignores stop-words [5] (a list of common or general terms that are not used in the information retrieval process, e.g., prepositions, conjunctions and articles). The parser considers each term of the *cv:monograph-title* and *cv:article-title* as a single word. On the other hand, in both *cv: monograph-keywords* and *cv:article-keywords*, the terms are taken integrally, as single expressions.

The *query vector* terms weights are build up according to the Equation 2. This equation considers the type of the term (keyword or title), the language and the year of the publication (monograph or article). Keyword terms are considered more important than the titles and have higher weights assigned. Publications written in a language in which the user has more reading proficiency are more valorized (higher weight), and the terms obtained from the most recent university degree and productions are assigned a more important weight than the less recent ones.

$$w_t = w_{KeywordOrTitle} . w_{Language} . w_{Year} \quad \textbf{(2)}$$

The weights $W_{KeywordOrTitle}$, $W_{Language}$ and W_{Year} are calculated with Equation 3.

$$w_i = 1 - (i - 1) \left(\frac{1 - w_{\min}}{n - 1} \right) \quad \textbf{(3)}$$

In this equation, W_i varies according to the type of weight we want to compute. To illustrate, in the experimental evaluation (Section 4), for $W_{KeywordOrTitle}$, W_{min} was 0.95, and *i* is 1 for keywords and 2 for title terms. For $W_{Language}$, W_{min} was 0.60 and *i* is 1 if the language-skill-level is "good", 2 for "reasonable" and 3 for "few". For W_{Year}, W_{min} was 0.55 and *i* vary from 1 to *n*, where *n* is the interval of years considered, being 1 the highest and *n* the lowest. In the experimental evaluation it was considered

the interval between 2006 and 2003. However, if the interval is omitted, it will be considered as between the present year and the less recent year (the smallest between *cv:graduation-year* and *cv:article-year*).

If w_{min} is not informed, the default value will be used (presented in Equation 4). In this situation, Equation 3 is reduced to Equation 5.

$$w_{\min\, default} = \frac{1}{n} \quad \textbf{(4)}$$

$$w_i = \frac{n - i + 1}{n} \quad \textbf{(5)}$$

Once the *query vector* is build, the *documents vector* terms and the respective weights must be defined. The adopted approach was (*tf x idf*), i.e., the product of the term frequency and the inverse document frequency [19]. This approach allows automatic term weights assignment for the documents retrieval. The *term frequency* (*tf*) corresponds to the number of occurrences of a term in the document. The *inverse document frequency* (*idf*) is a factor that varies inversely with the number of the documents *n* to which a term is assigned in a collection of *N* documents (typically computed as log (N/n)).

The best terms for content identification are those able to distinguish individuals ones from the remainder of the collection [19]. Thus, the best terms correspond to the ones with high term frequencies (*tf*) and low overall collection frequencies (high *idf*). To compute *tf x idf*, the system uses the DC metadata *dc:title* and *dc:description* to represent the documents content. Moreover, as our system deals with different languages, the total number of documents will vary accordingly. After building the *query* and *documents* vectors, the system is able to compute the similarities values among the documents and the query according to Equation 1.

4 Experimental Evaluation

In order to evaluate the recommender system, we have asked for the *Lattes* CV from a group of individuals of our Institution entailed to different research teams of different Computer Science research areas, such as Information Systems and Theory of Computation. As response, a group of 14 people send us their *Lattes* CV, whose information were loaded in the User Profile local database. The Articles Metadata local database was loaded with metadata of all digital documents stored in BDBComp Digital Library up to June of 2006, totalizing 3,978 articles from 113 conferences editions.

After, 20 recommendations were generated by the system for each participant, considering individual's university degrees and bibliographic production information present in the Lattes CV. This information corresponded just to the last three years (i.e., 2003 to 2006). Each recommendation had the following attributes extracted: title (*dc:title*), authors (*dc:creator*), URL (*dc:identifier*), idiom (*dc:language*), publication year (*dc:date*), conference (*dc:source*) and abstract (*dc: description*).

Two evaluations were performed. The first was based on the hypothesis that the best articles to describe the profile of a researcher should be those produced by the researcher himself. Since we had information about the articles written by each author (from the curriculum), we can match the items recommended to those that were actually written by them. This evaluation was accomplished by the *recall* and *precision* metrics that is a standard evaluation strategy for information retrieval systems [20, 1]. The *recall* is used to measure the percentage of relevant documents retrieved in relation to the amount that should have been retrieved. In the case of document categorization, the *recall* metric is used to measure the percentage of documents that are correctly classified in relation to the number of documents that should be classified. *Precision* is used to measure the percentage of documents correctly recovered, i.e., the number of documents correctly retrieved divided by the number of documents retrieved.

As the profiles can be seen as classes and the articles as items to be classified in these profiles, we can verify the amount of items from the author that are correctly identified (i.e., classified) by the user profile. As we have many users (i.e., many classes), it is necessary to combine the results. The *macroaverage* presented in Equation 6 was designed by *D. Lewis* [14] to perform this specific combination ("*the unweighted mean of effectiveness across all categories*"), and was applied by him in the evaluation of classification algorithms and techniques.

$$macroaverage = \frac{\sum_{i=1}^{n} X_i}{n} \quad \textbf{(6)}$$

In this formula, X_i is the *recall* or the *precision*, depending on the metric we want to evaluate, of each individual class (user in our case) and *n* is the number of classes (users). Thus, the *macroaverage recall* is the arithmetic average of the recalls obtained for each individual, and the *macroaverage precision* is the arithmetic average of the *precisions* obtained for each individual.

Given that the users are not interested in its own articles as recommendations, we performed another evaluation that takes in to account only the items from others authors. Then, 15 recommendations were presented to each individual ranked on the relative grade of relevance generated by the system In this rank, the article with the highest grade of similarity with the user profile was set as 100% relevant and the others were adjusted to a value relative to it. In this case, each author was requested to evaluate the recommendations generated to them assigning one of the following concepts (following the bipolar five-point Likert scale): "Inadequate", "Bad", "Average", "Good", and "Excellent", and were also asked to comment the results. The following section presents the results obtained.

5 Analysis of the experiments

The first experiment was designed to evaluate the capability of the system to correctly identify the user profile (i.e., to represent its research interests), since we believe that the best articles to describe the user profile are those written by themselves, as stated before. To perform such evaluation, we identified the number of articles that each

author had at BDBComp. After that, we employed the *recall* metric to evaluate the number of articles recovered for each author and combined them with the *macroaverage* equation explained before.

We have found a macroaverage recall of 43.25%. It is important to state that each author received 20 recommendations. This is an acceptable value as the query construction was made automatically without human intervention. It happened to be lower than it should be if we have used more than the last three years of information stored in the *Lattes* CV. Thus, articles related to the previous research interest areas were not recommended as the objective of the system resumed on the recommendation of articles associated to recent research interest areas of the users. Other important consideration is that the recommendation ranking was generated with a depreciation degree that was dependent on the publication year and on the user language proficiency, as explained in the previous section. As the time-slice considered corresponds to a small part of the full conference period stored in the BDBComp, not all articles are good recommendations since the research profile changes along the time.

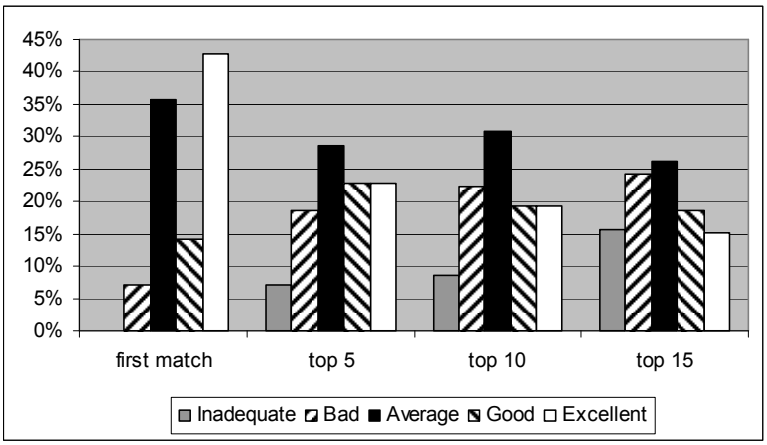

Fig. 2. Users' evaluations of the recommendations.

Figure 2 presents the results of the second experiment, which was based on the users' qualitative evaluation of the recommended articles. On this experiment each user received 15 recommendations and evaluated them according to one of the following concepts: "inadequate", "bad", "average", "good" and "excellent". The results were grouped into the categories "first match", "top 5", "top 10", and "top 15", and are presented in Figure 2.

Analyzing these results, it is possible to observe that, if we only consider the first article recommended (the "first match"), the number of items qualified as "excellent" is greater than the others (i.e., 42.86%) and none of them were classified as "inadequate". This strengthens the capability of the system on performing recommendations adjusted to the present user's research interests. We have also grouped the concepts "good" and "excellent" into a category named "positive

recommendation" and the concepts "bad" and "inadequate" into a "negative recommendation" group, so we could obtain a better visualization and comprehension of the results (Fig. 3).

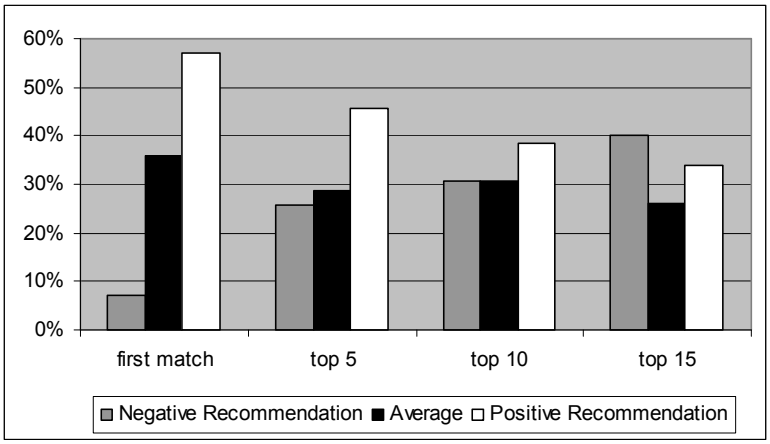

Fig. 3. Grouped users' evaluations.

We could perceive that the positive recommendations, considering only the "first match", are superior (57.14%) in relation to the negative ones (7.14%). The same behavior can be perceived in the "top 5" and "top 10" categories, the recommendations had a negative evaluation only in the "top 15" category, and that probably happened because as the number of recommendations grows, the number of correct recommendations falls. It is clear that the automated procedure here adopted is adequate for an alert recommender system. Our proposal is to add to the BDBComp an automated alert system that periodically sends to the user a list of the most relevant papers recently published in some of the nearly 35 Brazilian computer symposiums and 64 co-organized local events.

It is important to observe that today BDBComp has a limited coverage of the Computer Science area and it may have negatively influenced the quality of the recommendations. This was perceived in the commentaries made by some users, such as *"[...] I suppose that the generation of such results is a very complex task, as I worked with two distinct areas and mixed with even more themes. Besides, the two fields in which I have more publications have a very limited group of people working in this subjects here in Brazil. To conclude, considering such circumstances, the list of recommendations is good*.", and "*I can conclude that: (a) there are not many articles in my research area in BDBComp; or (b) I have not correctly described my articles metadata in Lattes CV*". In a near future all the SBC (Brazilian Computer Society) sponsored conferences will be automatically loaded [6].

Further, in our tests the authors that have changed their research area in the last three years have negatively qualified the recommendations. In the next experiments a variable time threshold and different depreciation values will be employed and the temporal component will be exhaustively analyzed.

6 Conclusion

This article presented a Recommender System to researchers and academics of the Computer Science area. In current days, in which the recovery of relevant digital information on the web is a complex task, such systems are of great value to minimize the problems associated to the information overload phenomena, minimizing the time spent to access the right information.

The main contribution of this research consists on the heavy utilization of automated CV data provider and in the use of a Digital Library (DL) metadata to create the recommendations. The system was evaluated with BDBComp, but it is designed to work with the open digital library protocol OAI-PMH, then it may be easily extended to work with any DL that supports this mechanism. The same occurs with the Curriculum Vitae, the system will be able to receive any XML-base CV data. Presently, the system uses the *Lattes* CV format, but it can be extended to support other formats or to analyze information about the user stored on tools like Scholar Google and DBLP. Alternatively the operational prototype offers the possibility to the user to load the CV data via an electronic form.

The developed system will have many applications. One of them is the recommendation of articles to support the learning process, especially on eLearning systems. Thus, the student could log into a specific distance or electronic learning environment supported by this system and receive recommendations of articles containing actualized relevant material to complement its current study topic.

References

1. Baeza-Yates, R.; Ribeiro-Neto, B.: Modern Information Retrieval. Addison-Wesley, Wokingham, UK (1999)
2. BDBComp: Biblioteca Digital Brasileira de Computação, http://www.lbd.dcc.ufmg.br/bdbcomp/, Nov. (2006)
3. Callan, Jamie et al.: Personalisation and Recommender Systems in Digital Libraries. Joint NSF-EU DELOS Working Group Report. May (2003)
4. CITIDEL: Computing and Information Technology Interactive Digital Educational Library, http://www.citidel.org/, Nov. (2005)
5. CLEF and Multilingual information retrieval, http://www.unine.ch/info/clef/, Institut interfacultaire d'informatique, University of Neuchatel (2005)
6. Contessa, Diego Fraga; Oliveira, José Palazzo Moreira de: An OAI Data Provider for JEMS. Proceedings of the ACM DocEng 2006 Conference, Amsterdam. Oct. (2006) 218-220
7. DC-OAI: A XML schema for validating Unqualified Dublin Core metadata associated with the reserved oai_dc metadataPrefix, http://www.openarchives.org/OAI/2.0/oai_dc.xsd, Mar. (2005)
8. Dublin Core Metadata Initiative, http://dublincore.org, Sept. (2005)
9. Grossman, David A.: Information retrieval: algorithms and heuristics. 2nd ed. Dordrecht: Springer, 332p. (2004)
10. Gutteridge, C.: GNU EPrints 2 overview, Jan. 01 (2002)
11. Huang, Z. et. al.: A Graph-based Recommender System for Digital Library. In: JCDL'02. Portland, Oregon (2002)

12. Laender, A. H. F.; Gonçalves, M. A.; Roberto, P. A.: BDBComp: Building a Digital Library for the Brazilian Computer Science Community. In: Proceedings of the 4th ACM/IEEE-CS Joint Conference on Digital Libraries, Tucson, AZ, USA (2004) 23-24
13. Lattes-CNPq: Plataforma Lattes - Conselho Nacional de Desenvolvimento Científico e Tecnológico, http://lattes.cnpq.br/, Mar. (2005)
14. Lewis, D. D. : Evaluating text categorization. In Proceedings of Speech and Natural Language Workshop. Defense Advanced Research Projects Agency, Morgan Kaufmann. (1991) 312-318.
15. LPML-CNPq. Padronização XML: Curriculum Vitae, http://lmpl.cnpq.br/lmpl/?go=cv.jsp, Mar. (2005)
16. Maly, K.; Nelson, M.; Zubair, M.; Amrou, A. ; Kothamasa, S.; Wang, L.; Luce, R.: Light-weight communal digital libraries. In Proceedings of JCDL'04, Tucson, AZ (2004) 237-238
17. OAI: Open Archives Initiative, http://openarchives.org, Oct. (2005)
18. OAI-PMH: The Open Archives Initiative Protocol for Metadata Harvesting, http://www.openarchives.org/OAI/2.0/openarchivesprotocol.htm, Nov. (2005)
19. Salton, Gerard; Buckley, Christopher.: Term-Weighting Approaches in Automatic Text Retrieval, Information Processing and Management: an International Journal, v.24, Issue 5, 513-523. (1988)
20. Salton, Gerard; Macgill, Michael J.: Introduction to Modern Information Retrieval. New York: McGRAW-Hill. 448p. (1983)
21. Sompel, H. V. de; Lagoze, C.: The Santa Fe Convention of the Open Archives Initiative. D-Lib Magazine, [S.l.], v.6, n.2, Feb. (2000)
22. Tansley, R.; Bass, M.; Stuve, D.; Branschofsky, M.; Chudnov, D.; McClellan, G.; Smith, M.: DSpace: An institutional digital repository system. In Proceedings of JCDL'03, Houston, TX. (2003) 87-97