Natalia L. Mitzig, Mónica S. Mitzig, Fernando A. Martínez, Ricardo A. Piriz, Víctor M. Ferracutti, María Paula González, Ana G. Maguitman, SciELO suggester: An intelligent support tool for cataloging library resources, Library & Information Science Research, Volume 38, Issue 1, January 2016, Pages 39-51, ISSN 0740-8188, http://dx.doi.org/10.1016/j.lisr.2016.01.001. (http://www.sciencedirect.com/science/article/pii/S0740818816300251)

SciELO Suggester: An intelligent support tool for cataloging library resources

Natalia L. Mitzig[a,b], Mónica S. Mitzig[b], Fernando A. Martinez[a], Ricardo A. Piriz[a], Víctor M. Ferracutti[a,b], María Paula Gonzalez[b,c], Ana G. Maguitman[b,c,*]

[a]Biblioteca Central de la Universidad Nacional del Sur, Avda Alem 1253, 8000 Bahía Blanca, Argentina

[b]Departamento de Ciencias e Ingeniería de la Computación, Universidad Nacional del Sur, Avda Alem 1253, 8000, Bahía Blanca, Argentina

[c]Consejo Nacional de Investigaciones Científicas y Técnicas, CONICET, Argentina

Abstract

Existing cataloging interfaces are designed to reduce the bottleneck of creating, editing, and refining bibliographic records by offering a convenient framework for data entry. However, the cataloger still has to deal with the difficult task of deciding what information to include. The SciELO Suggester system is an innovative tool developed to overcome certain general limitations encountered in current mechanisms for entering descriptions of library records. The proposed tool provides useful suggestions about what information to include in newly created records. Thus, it assists catalogers with their task, as they are typically unfamiliar with the heterogeneous nature of the incoming material. The suggester tool applies case-based reasoning to generate suggestions taken from material previously cataloged in the SciELO scientific electronic library. The system is implemented as a web service and it can be easily used by installing an add-on for the Mozilla Firefox browser. The tool has been evaluated through a human-subject study with catalogers and through an automatic test using a collection consisting of 45,742 training examples and 120 test cases from 12 different subject areas. In both experiments the system has shown very good performance. These evaluations indicate that the use of case-based reasoning provides a powerful alternative to traditional ways of identifying subject areas and keywords in library resources. In addition, a heuristic evaluation of the tool was carried out by taking as a starting point the Sirius heuristic-based framework, resulting in a very good score. Finally, a specially designed cognitive walk was completed with catalogers, providing additional insights into the strengths and weaknesses of the tool.

* Corresponding author: Ana Maguitman (agm@cs.uns.edu.edu)

1 Introduction

Although many standardized resources and well-established practices are commonly used to generate library records, the process of cataloging remains a bottleneck in library management. Organizing resources associated with diverse topics is a difficult and costly task for the cataloger, who is typically unfamiliar with incoming resources due to their heterogeneous nature. A variety of solutions based on information technologies have been proposed to assist in the cataloging process (Buckland, 1992; Levy & Marshall, 1995; Park & Lu, 2009, Sølvberg, 2001).

The SciELO Suggester system is an innovative tool developed to facilitate the process of cataloging resources arriving at a library. The task of cataloging involves associating a set of metadata with incoming resources. For example, a thesis is associated with an author, an advisor, a title, an abstract, one or more subject areas, a small set of keywords (words or short phrases that are used to describe the topic of a resource), and a date of publication. Some of these data, such as the title, author, advisor, abstract, and date are explicitly given in the digital resource itself, while other data, for instance, subject areas and keywords, typically need to be inferred by the cataloger.

The proposed tool applies ideas from case-based reasoning (CBR) to assist catalogers, supplementing traditional cataloging tools by identifying appropriate subject areas and keywords for incoming material. The suggester tool operates as an experience-based system by presenting suggestions taken from material previously cataloged in the SciELO scientific electronic library (http://www.scielo.org).

2 Problem statement

Existing cataloging interfaces are designed to reduce the bottleneck of creating, editing, and refining bibliographic records (Gómez, 2015; Reese, 2015). These interfaces provide a convenient framework, but ease of data entry provides only a partial solution to the problems of cataloging—the cataloger still has the harder task of deciding what information to include. The intellectual effort which is expended at this stage is time-consuming, costly, and leads to bottlenecks in resource processing.

Informal discussions with catalogers indicate that when cataloging, they often pause for significant amounts of time wondering what information to include. They usually look at existing catalogs (e.g., the Library of Congress Catalog, https://catalog.loc.gov/) and other information on the Web for metadata to associate with incoming resources. Through bibliographic database services such as those provided by OCLC, Inc. (2015a, 2015b), catalogers have easy access to up-to-date records for many kinds of resources. However, these tools require the user to explicitly request a bibliographic record, a request that can only be fulfilled if the resource is already cataloged in the databases of bibliographic records that are accessible to the cataloger.

Some of the resources that arrive at a library, such as doctoral and master's theses or other rare material, are not cataloged in these databases. In spite of that, bibliographic records of material that is topically similar to the to-be-cataloged resource can be helpful at the moment of generating metadata such as subject areas and descriptive keywords. Thus, intelligent tools that identify similar material and generate suggestions could provide substantial benefits for cataloging. This approach motivated previous work done to expand cataloging tools with intelligent aides (Delgado, Maguitman, Ferracutti, & Herrera, 2011; Dini, Varela, Antúnez, Maguitman, & Herrera, 2010).

Such tools are necessary to optimize cataloger productivity and can save libraries the burden of investing in a task that can be replaced to a large extent by automatic processing. The functionality of these tools not only increases the efficiency of the cataloging process, and thereby saves money and reduces bottlenecks, but it can also improve the quality of the catalog itself and enable more complete cataloging.

The work described here is part of an effort carried out as a collaboration between the main library of the Universidad Nacional del Sur (Bahia Blanca, Argentina) and members of the Knowledge Management and Information Retrieval Research Group. The SciELO Suggester tool is one in a series of prototypes developed with the purpose of leveraging existing bibliographic records to assist the cataloger.

3 Literature review

Unlike manual catalog creation, semi-automatic generation of catalog entries relies on support tools that assist the cataloger to effectively identify the most appropriate metadata for the resource under analysis. For several years, in domains other than library science, intelligent support tools have served the purpose of expanding the user's natural capabilities, for example by acting as intelligence or memory augmentation mechanisms (Engelbart, 1962; Licklider, 1960). Many of these systems are highly autonomous and are based on the intelligent agent metaphor (Bradshaw, 1997; Laurel, 1997; Maes, 1994; Negroponte, 1997) while others adopt a user-driven approach and need to be initiated by commands or direct manipulation interfaces (Shneiderman, 1992; Sutherland, 1963; Ziegler & Fahnrich, 1988). An intermediate group of support tools reconciles both approaches, giving rise to mixed-initiative user interfaces (Horvitz, 1999). In general, these tools complement the users' abilities and enhance their performance by offering proactive or on-demand context-sensitive support. A recent review of intelligence augmentation systems is presented in Xia and Maes (2013).

Support tools that anticipate the user's next steps and offer automation of predicted actions have been popular for several years, mostly in word processing and programming. The Eager system (Cypher, 1991) is an early example of such predictive support tools. Eager is an aid for the HyperCard environment that monitors the user's activity and draws from ideas of programming by example (Smith, 1977) to generalize the user's repetitive patterns in order to anticipate what the user will do next. The system highlights menus and objects on the screen to indicate its predictions. If a correct anticipation has been generated the user can tell *Eager* to complete the task automatically. Another tool, Writer's Aid (Babaian, Grosz, & Shieber, 2002), is a collaborative interface that uses a planning system to support an author's writing efforts by facilitating the insertion of bibliographic records. The more recently developed Zotero (Puckett, 2011; Zotero, 2015) is a support tool that functions as an aid in writing papers, managing references, and organizing research materials.

Knowledge acquisition and modeling is another domain for which there have been several proposals for mixed-initiative user interfaces. For instance, the EXTENDER system (Leake, Maguitman, & Reichherzer, 2014) takes a knowledge model under construction and applies an incremental technique to build up context descriptions. Its task is to generate brief descriptions of new topics relevant to an existing knowledge model. Other suggester tools attempt to reduce the bottleneck of knowledge acquisition in the construction of domain ontologies. In Hsieh, Lin, Chi, Chou, and Lin (2011) text mining techniques are applied to support the extraction of concepts, instances, and relations from a handbook of a specific domain in order to quickly construct a basic domain ontology.

In the domain of library science, a number of methods have been used in an attempt to automatically extract semantic metadata for digital resources; these are reviewed in Albassuny (2008); Greenberg (2009); Greenberg et al. (2005); and Park and Lu (2009). Some of these tools, such as the ones presented in Paynter (2005) automatically create metadata, including not only title and authors, but also some derived information, for example keyphrases and the subject of a given resource. Automatic keyphrase extraction has typically been addressed as a supervised learning problem. In most existing methods, documents are treated as a set of phrases that the learning algorithm must learn to classify as positive or negative examples of keyphrases. Different classifiers have been applied to learn this classification task, for instance, Kea (Frank, Paynter, Witten, Gutwin, & Nevill-Manning, 1999; Witten, Paynter, Frank, Gutwin, & Nevill-Manning, 1999) applies a keyphrase extraction domain-specific method based on the naïve Bayes classifier. In Turney's work (2000) the learning task is achieved by using the C4.5 decision tree induction algorithm. Methods based on support vector machines have also been successfully applied to the problem of keyphrase extraction (Zhang, Xu, Tang, & Li, 2006). A number of proposals rely on external sources to improve keyphrase extraction. For instance, Maui (Medelyan, Frank, & Witten, 2009) is a successor to *Kea* which uses semantic information extracted from Wikipedia to automatically extract keyphrases from documents. HUMB (Lopez & Romary, 2010b) is a key term extraction system that makes use of knowledge from Wikipedia and GRISP, a large scale terminological database for technical and scientific domains (Lopez & Romary, 2010a), to produce a set of lexical and semantic features. A list of ranked key term candidates is then generated using a machine learning algorithm. The authors report that bagged decision trees appeared to be the most efficient algorithm to complete this task. HIVE (Greenberg et al., 2011) is a system that relies on Kea for keyphrase extraction and is complemented by the use of a vocabulary server supporting automatic metadata generation by simultaneously drawing descriptors from multiple controlled vocabularies.

Recent overviews of various state-of-the art methods for keyphrase extraction (Hasan & Ng, 2014; Kim, Medelyan, Kan, & Baldwin, (2013) show that the methods for identifying potential keyphrases rely mostly on text and natural language processing techniques, for example n-grams or part-of-speech (POS) sequences. These methods differ from this proposal, which relies on CBR to identify potential cataloged entries from which metadata can be directly obtained. In addition, keyphrase extraction approaches differ from this proposal in that they attempt to automatically extract keywords from a resource to be cataloged. Instead, the goal of the SciELO Suggester system is to automatically select keywords (and subject areas) that are associated with other digital resources and to suggest them as potentially useful metadata which can be associated with the to-be-cataloged resource.

More closely related to this proposal are those frameworks that automatically generate metadata based on pre-existing metadata of related resources. For instance, the methods proposed by Rodriguez, Bollen, and Sompel (2009) rely on existing repository metadata to enrich metadata-poor resources. This is accomplished by constructing an associative network of repository resources based on occurrence and co-occurrence metadata. A spreading activation algorithm is then used to propagate metadata. These methods resemble this proposal as they both use pre-existing metadata to populate other resources. However, they use only the metadata to compute the similarity between the resources, while the method used by the SciELO Suggester system also measures the similarity between abstracts. In addition, the methods proposed by Rodriguez et al. (2009) require re-computing the associative network of metadata-rich and metadata-poor resources for each new resource to be cataloged. The SciELO Suggester system, on the other hand, is based on the use of an index of cases created from existing cataloged resources.

Finally, as pointed out by Knijnenburg, Reijmer, and Willemsen (2011), Ozok, Fan, and Norcio (2010), and Pu, Chen, and Hu. (2011), researchers have recently started to examine issues related to users' subjective opinions and to develop additional user-oriented criteria to evaluate recommender systems. Indeed, recommendation technology is becoming widely accepted as an important component as it provides user benefits while enhancing productivity. In this setting, the ResQue evaluation framework (Pu et al., 2011) addresses the most important issues associated with the design of recommender systems from a user-centered perspective. However, more specific user-centered models have yet to be developed or adapted for the design of suggester systems in the context of the library cataloging task.

4 Using case-based reasoning to support cataloging

The cataloger's task is to create records of existing library resources. When previously-built records exist, those records may be seen as a set of cases, reflecting past catalogers' decisions about what to include in the record. If a new resource is similar to an already cataloged resource, the choices made by others may provide useful advice. As a consequence, CBR may be useful in providing suggestions.

Starting from the abstract of a resource that needs to be cataloged, the SciELO Suggester tool identifies and suggests keywords and subject areas as potential cataloging metadata for inclusion in the resource record. The implemented tool is a human-in-the-loop system: it automates part of the cataloging process, by searching for useful material, but relies on the user to make the final decision as to what information to include. Figure 1 outlines the SciELO Suggester processing cycle. The system starts from a to-be-cataloged incoming resource, which is used to automatically generate queries. The queries are submitted to a library of cases consisting of previously cataloged resources. The retrieved cases that are sufficiently similar to the to-be-cataloged resource under analysis are used to produce suggestions of keywords and subject areas. The cataloger is in charge of deciding which suggestions to include in the new bibliographic record. Once generated, this new record becomes part of the library of cases, expanding the set of suggestions that can be provided by the tool in future requests.

[Figure 1 about here.]

To address the challenges posed by the implementation of the SciELO Suggester tool, methods inspired by information retrieval and CBR have been developed to index earlier library records, perform context-based retrieval, and suggest appropriate elements of prior records to aid catalogers in their task. The suggester tool searches prior records for relevant cases and presents them for the cataloger to adapt. The newly generated records are added to the library to enrich the set of records that can be mined in the future.

CBR (Leake, 1996) is a paradigm used to build intelligent systems where the main sources of knowledge are not rules but cases or episodes. These systems generate solutions by recovering relevant stored cases and adapting them to new situations. The CBR paradigm is based on two premises about the nature of the world. The first premise is that the world is regular, and because of this regularity, solutions that were useful for previous problems can serve as a starting point to solve new problems. The second premise establishes that the kind of problems that an agent finds tend to be recurrent and therefore new problems may be similar to problems found in the past. CBR systems are built on these premises to store, adapt, and reuse solutions to previous problems.

When a new case is recovered and used at the right time, it becomes an important source of information, saving the time and effort necessary to develop solutions from scratch. CBR systems have been successfully applied to areas such as design, planning, diagnosis, knowledge management, and legal reasoning. Some CBR systems operate autonomously, while others are integral parts of collaborative systems, where the user and the system complement and help each other with the purpose of solving problems.

Applying CBR to assist with the cataloging process presents specific challenges. First, CBR generally considers cases as being segmented into problem-solution pairs. In the scenario under analysis, the problem is represented by a resource that needs to be cataloged, while the solution consists of the subject areas and the descriptive keywords that have been chosen by the cataloger to describe the resource. The relevance of a stored problem-solution pair to an incoming resource that needs to be cataloged must be decided based on the similarity between the incoming resource and the stored case.

5 The SciELO Suggester system

The SciELO Suggester system is fully functional and documented and it is available for download under the GNU Affero General Public License[1].

5.1 The SciELO Suggester crawler

A web crawler is a program that visits pages of the World Wide Web, typically with the purpose of creating a copy of the visited pages. These pages are usually processed later and indexed to implement a software application, such as a classifier or a search engine. A crawler has been developed which collects resources from the SciELO library in order to populate the CBR library with problem-solution pairs. This library contains more than a thousand academic journals in diverse subject areas. The number of articles indexed by SciELO is close to half a million and they are available in XML format. Since the initial analysis was carried out in a Spanish-speaking university, the SciELO crawler only collected articles written in Spanish, amounting to 51,637 articles from 361 journals. However, this could easily be extended and applied to other languages.

5.2 The SciELO Suggester index

The articles collected by the SciELO crawler have a rich structure but they required an additional processing stage to become well-structured cases. It should be noted that each journal may be associated with one or more subject areas. To generate a useful CBR library, it was necessary to identify the most suitable subject area for each article. This was accomplished by applying an automatic classifier that was trained using journals associated with a single subject area and used to classify articles from journals associated with multiple subject areas. To identify a single subject area for a given article, the three most similar subject areas were obtained from the training set. Then cosine similarity (Salton, 1971) was used to compare the vector representation of the article at hand to the vector representation of the three retrieved subject areas. If one of the three retrieved subject areas matched one of the subject areas of the article's journal, this subject area was associated with the article, otherwise the article was discarded. After completing this process only 293 articles were discarded. Those articles for which it was possible to identify a single subject area were incrementally added to the training set. Finally, a CBR library of problem-solution pairs was built by indexing all the articles for which a single subject area was identified. Each case was a cataloged article, where its title, keywords and abstract represented "the problem" while the subject area and the descriptive keywords represented its corresponding "solution". Apache Lucene (Apache Software Foundation [ASF], 2015a) was used to create this index of cases.

5.3 The SciELO Suggester web service

A Simple Object Access Protocol (SOAP) web service was implemented running on a Tomcat Apache server (ASF, 2015c). The main purpose of the SciELO Suggester web service was to run the method needed to generate suggestions for the client's request. This

[1] `https://github.com/fariel/sugerencias-palabras-clave`

method takes the client request, consisting of a selected text, and returns a string containing the top three suggestions obtained by using the Lucene's MoreLikeThis library (ASF, 2015b) in combination with the SciELO Suggester index. The MoreLikeThis library works by comparing the text for which suggestions are requested against the indexed resources. This allows for the retrieval of content related to the text at hand. The keywords, title and abstract of the cases (articles) indexed by the SciELO Suggester index are examined so as to identify similar content. Finally, the top three cases are selected and three suggestions are constructed based on the subject areas and keywords associated with these cases.

5.4 The SciELO Suggester add-on

To facilitate the use of the suggester tool, the browser context menu was expanded with a new option for requesting suggestions for subject areas and keywords. This was implemented through an add-on for the Firefox browser. The decision to use a browser add-on was based on the fact that most cataloging tools have a web-based interface. Therefore, a cataloger examining a resource to be cataloged can highlight a portion of the text describing this resource, such as the resource's abstract, and after clicking on the right button of the mouse, a context-menu will appear (Fig. 2). The resource to be cataloged can be in plain text, HTML or PDF format, as the Firefox browser allows viewing any of these file types.

[Figure 2 about here.]

If the user requests suggestions for subject areas and keywords, the system will initiate a search process for relevant suggestions and will then show a panel with suggestions (Fig. 3). In the first place, the tool presents the most relevant terms from the text highlighted by the user when the request was initiated. Then, it presents the top-three suggestions found by the system. Each suggestion is composed of a subject area and a set of keywords, which are extracted from those cases that were identified as the most similar ones to the text that was highlighted by the user.

[Figure 3 about here.]

6 Evaluation

The system was evaluated from three points of view. A user study was completed to determine whether the keywords suggested by the system were useful to the cataloger. In addition, traditional information retrieval metrics were used to assess the accuracy of the subject areas suggested by the system. Finally, a usability study was completed to assess the ease of use of the SciELO Suggester interface.

6.1 Assessing the usefulness of the suggested keywords

Seven catalogers working at different libraries of the Universidad Nacional del Sur were invited to participate in this experiment. The participants were asked to complete an online form which contained an introductory text with the experiment instructions followed by the test itself. In the test phase, each participant answered five questions about five different cataloging tasks. Each question presented the title and the abstract of a potential resource to be cataloged and a list of potentially relevant keywords (Fig. 4). Participants were asked to examine the title, the abstract and the list of keywords, and to determine which keywords were useful to describe the potential resource. Each list contained a balanced set of keywords suggested by the SciELO Suggester systems and unrelated keywords randomly selected from off-topic resources. The unrelated keywords were introduced into the experiment for control purposes. The suggested and the unrelated keywords were mixed and appeared in a random order every time a question was accessed.

[Figure 4 about here.]

Many of the suggested keywords were selected by most of the participants, pointing out the usefulness of the suggestions made by the system (Table 1). Also, none of the unrelated keywords were selected by any of the participants.

[Table 1 about here.]

6.2 Testing the suggester's accuracy

The second experiment was designed to establish how well the suggester can predict the subject area of a given article. To complete this analysis a training corpus consisting of 45,742 cases was used. The cases belonged to 12 different subject areas (Table 2). For each of these subject areas, a set consisting of 10 articles not belonging to the training set was used to test the suggester tool. Because the 120 articles used for testing were part of the SciELO Scientific Electronic Library, these articles were already cataloged. Each of these articles was associated with a specific journal and therefore it was possible to identify the article's subject area by applying the approach discussed in section 5.2. The existence of a classification based on subject areas made it possible to test the precision of the tool automatically.

[Table 2 about here.]

To complete this test, the suggester was run to obtain the top-three suggestions for each of the test articles. The article's title, abstract and keywords were used as the selected text to initiate the search for suggestions. After that, it was determined for each article whether the article's actual subject area matched the first suggestion, one or two of the top-two suggestions, or if it matched one, two or three of the top-three suggestions. Based on these results, a statistical analysis was completed to establish the number of correctly predicted subject areas (Table 3).

[Table 3 about here.]

The predicted subject areas agreed with the actual ones in more than half of the cases. Moreover, the mean number of correct predictions is significantly superior by considering the top 2 and top 3 predictions. These results indicate that the prediction success was high for a twelve-class classification problem.

To further analyze the prediction accuracy of the suggester, the confusion matrices associated with the first, top-two and top-three predictions were computed. By means of a confusion matrix M it is possible to show the classifier's accuracy. Entry $M[i, j]$ represents the number of cases belonging to class i that were assigned to class j by the classifier. The values in the main diagonal correspond to the number of correctly classified instances. Therefore, for a perfect classifier only diagonal elements $M[i, i]$ would be nonzero. Tables 4, 5 and 6 show the numerical and grayscale confusion matrices for the first, top-two and top-three predictions respectively. The grayscale levels represent the number of cases in each entry. The analysis of these confusion matrices makes it easy to recognize which subject areas that tend to be correctly predicted and which are the poorly predicted ones. In addition, the grayscale matrices allow visual identification when a particular subject area is typically confused with another one.

[Table 4 about here.]

[Table 5 about here.]

[Table 6 about here.]

All the cases belonging to health science (T3) are correctly classified (i.e., $M[3, 3] = 10$). On the other hand, only one out of the ten cases belonging to mathematics (T11) is correctly classified, which is indicated by the value $M[11, 11] = 1$. These results are a natural consequence of the fact that the training set contains a large number of cases associated with health science (23,754) while there are very few cases for mathematics (9). This highlights the importance of having a rich case base, which is crucial for generating correct predictions. By further analyzing this confusion matrix, it becomes evident that social sciences (T6) and linguistics, literature and arts (T10) are many times confused with humanities (T8). This follows immediately from visualizing high grayscale values (dark boxes) away from the matrix diagonal associated with the entries $M[6, 8] = 7$ and $M[10, 8] = 6$.

Finally, the precision and recall measures for each of the subject areas were computed (Tables 7 through 18). Given a subject area, precision is the fraction of positive predictions for that subject area that are correct, while recall is the fraction of cases associated with that subject area that are correctly predicted. For each subject area T_i, the following values were computed:

- $[T_i, T_i]$: total number of articles that belong to T_i that were correctly classified (i.e., true positives).

- $[T_i', T_i']$: total number of articles that do not belong to T_i and were not predicted as being of T_i (i.e., true negatives).
- $[T_i', T_i]$: total number of articles that do not belong to T_i but were predicted as being of T_i (i.e., false positives).
- $[T_i, T_i']$: total number of articles that belong to T_i but were not predicted as being of T_i (i.e., false negatives).

Then, precision and recall for T_i were computed as follows:

$$\text{Precision}(T_i) = [T_i, T_i] / ([T_i, T_i] + [T_i', T_i]).$$

$$\text{Recall}(T_i) = [T_i, T_i] / ([T_i, T_i] + [T_i, T_i']).$$

[Table 7 about here.]

[Table 8 about here.]

[Table 9 about here.]

[Table 10 about here.]

[Table 11 about here.]

[Table 12 about here.]

[Table 13 about here.]

[Table 14 about here.]

[Table 15 about here.]

[Table 16 about here.]

[Table 17 about here.]

[Table 18 about here.]

Let **T** be the set containing the twelve subject areas T_i. Based on the precision and recall value for each T_i the micro-averaged precision (M_μ^P), micro-averaged recall (M_μ^R), macro-averaged precision (M_m^P) and macro-averaged recall (M_m^R) were computed as follows:

$$M_{\mu}^{P} = \frac{\sum_{T_i \in \mathbf{T}} [T_i, T_i]}{\sum_{T_i \in \mathbf{T}} ([T_i, T_i] + [T_i', T_i])} = 0.53.$$

$$M_{\mu}^{R} = \frac{\sum_{T_i \in \mathbf{T}} [T_i, T_i]}{\sum_{T_i \in \mathbf{T}} ([T_i, T_i] + [T_i, T_i'])} = 0.58.$$

$$M_{m}^{P} = \frac{\sum_{T_i \in \mathbf{T}} \text{Precision}(T_i)}{|\mathbf{T}|} = 0.61.$$

$$M_{m}^{R} = \frac{\sum_{T_i \in \mathbf{T}} \text{Recall}(T_i)}{|\mathbf{T}|} = 0.53.$$

Finally, the F_1 score for the harmonic mean of the macro-averaged precision and the macro-averaged recall was computed as follows:

$$F_1 = \frac{2 * M_m^P * M_m^R}{M_m^P + M_m^R} = \frac{2 * 0.61 * 0.53}{0.61 + 0.53} = 0.56$$

The reported performance measures show that the system achieves good average precision values, indicating that it provides more relevant suggestions than irrelevant ones. In the meantime, the average recall values obtained by the system point out that it has good prediction coverage for most of the twelve analyzed subject areas.

6.3 Usability study

Usability is a software attribute usually associated with the ease of use and learning of a given interactive system, and largely recognized in the literature as “the extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use” (International Organization for Standardization, 1998) where the context of use is a description of the actual conditions under which the interactive system is being assessed, or will be used in a normal working situation. Nowadays usability evaluation is an important part of software development, providing results based on quantitative and qualitative estimations. The importance of taking into account usability lies in the fact that for non-experts in recommender systems, the SciELO Suggester system is just what its interface provides. The user’s acceptance is therefore directly proportional to its quality, since a satisfactory experience should lead to higher productivity and applicability of the tool.

The observed field for this study was limited to the Universidad Nacional del Sur, a public university where a large library system is spread over several departments. While the main library coordinates and centralizes general resources, each departmental library maintains special collections covering various fields associated with the department. Systems development and support staff working at the main library provide support to the entire system. Catalogers and librarians work both at the main and at the departmental libraries, combining common cataloging norms, formats and guidelines with specific conditions associated with each department. The specific existing conditions at each departmental library are the result of using different collections of digital and print resources and different cataloging strategies. In spite of this heterogeneity, the cataloging process has been normalized. Taking this scenario as case study is an opportunity to observe and interview catalogers working in different buildings and under different conditions, but all involved in a normalized cataloging process.

The first goal of the usability study was to observe catalogers while joining them in their activity, working in their workplace to see how they use the existing tools for cataloging. The goal was to achieve an understanding of the cataloging process. Catalogers from the main library and the departmental libraries were observed, excluding the Mathematics Department library, which was used to triangulate and validate the collected data. Instead of formal interviews, the first stage was focused on dialog and observation, which helped to gain trust with the observed staff. During this stage qualitative descriptions were preferred over quantitative data. In order to minimize personal and cultural biases in usability experts and catalogers, members of the system development and support staff of the main library were present during the observations, taking note on the process. As a result, common problems encountered during the cataloging process were identified. In addition, this stage allowed researchers to identify a list of critical tasks as well as to obtain an accurate description of the context of use and the profiles of the SciELO Suggester system's users. All the data collected during this stage was used as input for completing the usability study. On the basis of the data collected during the observation stage, two strategies were chosen to cope with the usability study: first, a heuristic evaluation handled by usability experts, and second, a participatory cognitive walkthrough performed by catalogers working at the main library and at the departmental libraries. In the case of the cognitive walkthrough, members of the system development and support staff of the main library participated just as external observers without interfering with the walkthrough itself.

The Sirius heuristic-based framework for measuring web site usability proposed in del Carmen Suárez Torrente, Prieto, Gutiérrez, and Sagastegui (2013) was applied to complete the heuristic evaluation. Although Sirius is currently embedded in the Prometheus tool, one of the original matrices of Sirius methodology was applied (see Tables 19, 20, 21 and 22), since Prometheus was developed after the usability study reported here began. The Sirius framework itself supports a heuristic approach focusing on critical tasks to evaluate usability adapted to a web resource. Besides, Sirius offers numerical usability metrics that quantify the usability level achieved, adding quantitative aspects without disregarding the qualitative results. In this study, the category "web resource" was chosen among Sirius alternatives for web type. The list of critical tasks characterized during the observation stage was used as input. The quality of the information architecture provided by the SciELO Suggester system was measured through 13 Sirius items related to authorship and content. In the same way, the quality of the SciELO Suggester system form was assessed by considering 11 Sirius items associated with accessibility and usability. The heuristic evaluation was carried out in two steps. First, usability experts marked all the items individually and then a general consensus was achieved. Finally, the results for the two dimensions (quality of information and quality of the form) were weighed according to del Carmen Suárez Torrente et al. (2013). Based on the qualitative data collected during the observation stage, the value 0.5 was used to equally weigh the information architecture and form dimensions. A final numerical value for the SciELO Suggester system was calculated. The possible range for this value was between 0 (worst value) and 5 (best possible value). After reaching a consensus, the usability experts assigned an overall value equal to 4.

[Table 19 about here.]

[Table 20 about here.]

[Table 21 about here.]

[Table 22 about here.]

The heuristic evaluation shows an agreement among usability experts regarding the information and format quality of the system confirming the usability of the tool according to the adopted evaluation framework.

Finally, a cognitive walkthrough was completed with the participation of seven catalogers from the main library and departmental libraries of the Universidad Nacional del Sur. Once again, members of the system development and support staff of the main library participated as observers in this stage of the study. After performing some cataloging tasks supported by the current prototype of the SciELO Suggester tool, catalogers completed the questionnaire in which they were asked to mark different items focused on the assessment of the usability of the tool's interface (Table 23). In this way, the evaluation results reported in sections 6.1 and 6.2 were complemented with data reflecting the system usage experience. Some suggestions provided by Pu et al. (2011) were used to define the content of the questionnaire. In addition, the users were asked to support the assigned marks with a statement expressed in natural language. The last question of the questionnaire was directed towards gathering additional qualitative data. Therefore, catalogers were requested to give general impressions, general suggestions and global evaluation of the experience of using the SciELO Suggester system.

[Table 23 about here.]

The scores assigned by catalogers were high for most of the questions (typically the value was 4 or 5). Therefore, the quantitative results validated the score obtained in the cognitive walkthrough. However, some interesting suggestions pointed to the need for including an extended help option in the interface, as well as better explanations associated with the error messages. Furthermore, the inclusion of a brief explanation of the criteria used by the tool to generate each suggestion seems to be necessary to support each suggestion. The differentiation between "keywords" and "relevant words" provided by the tool seems to confuse some catalogers. Another relevant consideration is that the current version of the SciELO Suggester system does not include a clear metaphor to indicate the current stage of the process and the end of the program itself. Some functional limitations of the tool such as the support for only one language (Spanish) and the need to use a particular web browser (Firefox) were discussed with the participants of the study. Although the usability study focused on the quality of non-functional requirements of the current version of the SciELO Suggester system, these functional limitations were also reported.

7 Discussion

A key difference between the proposed system and most of the existing cataloging support tools is that the CBR approach does not require the suggested keywords to be part of the article to be cataloged. This is due to the fact that the suggested keywords are obtained from the metadata of similar resources and not from the to-be-cataloged resource itself. Expensive natural language processing is avoided by using CBR to suggest subject areas and keywords, but this more cost-effective approach may result in a loss of accuracy. A more thorough analysis of the costs and benefits of adopting this approach needs to be completed. A possible modification to avoid a loss of precision would be to limit the proposed suggestions to keywords that match with keywords that actually occur in the article to be cataloged, after stemming is applied. Another factor that affects performance is that the suggestions are based on the subject areas defined by the SciELO library, which are too general when compared with other classifications. The use of other sources for refining this classification, such as the monthly subject heading updates from the Library of Congress or subject taxonomies such at the one provided by PLOS (http://www.plosone.org/taxonomy) could greatly enhance the system's functionality.

Further improvements in performance would be possible if other libraries beyond the SciELO scientific electronic library were used to populate the case library. In particular, indexing more articles for specific subject areas that are poorly represented in the case library, such as mathematics, will be a crucial step to achieve higher precision and recall. Because the system has been implemented using a scalable and high-performance indexing platform, the system is expected not to decrease in efficiency after the number of cases in the index increases. A current limitation of the system is that it relies on a digital version of the resource abstract or another representative piece of text to take as a starting point for suggestion generation. To catalog hard copies of books and journals, additional steps and resources need to be considered. In particular, the use of a scanner and optical character recognition (OCR) technology would provide an easy-to-implement solution.

Future work is intended to include a full usability evaluation carried out in different cataloging scenarios. More data is required to validate and improve results. In particular, the usability evaluation methodology $QUTC_{KDD}$ (González, Lorés, & Granollers, 2008) will be applied. $QUTC_{KDD}$ provides a datamining-based technique for detecting common usability problems of particular contexts of use. The application of this approach to characterizing the most relevant usability problems at the cataloging domain is currently under consideration. The current version of the SciELO Suggester system is being modified to improve its weaknesses on the basis of the usability results obtained.

8 Conclusion

Evaluations and user studies demonstrate the effectiveness and usefulness of the SciELO Suggester system. CBR is a cost-effective yet powerful solution to the problem of identifying appropriate subject areas and keywords that can be associated with incoming material. The heuristic evaluation and cognitive walkthrough revealed strengths and limitations of the system's interface. As a secondary contribution, since there do not appear to be user-centered models for assessing the usability of suggester systems in the context of library cataloging, this study proposes criteria for carrying out such assessment.

Cataloging is a fundamental process in library services and remains a bottleneck in library management. The use of intelligent tools to assist with this task can improve cataloger productivity and efficiency, enhance the quality of the catalog, and enable more complete cataloging. The SciELO Suggester system offers a cost-effective and powerful step in this direction.

Acknowledgments

We gratefully acknowledge the contribution of Luis A. Herrera, María Alejandra Dini, Mariana Anahí Varela, Verónica Antúnez, Pablo Delgado and Jerónimo Spadaccioli for their contribution in the development of earlier versions of the SciELO suggester system. We also thank the Libraries Staff at Universidad Nacional del Sur, in particular Nélida Benavente, Guillermina Castellano, Paola Cruciani, Patricia Di Carlo, Marcela Esnaola, Marcela Garbiero, Fernando Gómez, Marta Ibarlucea, Nelly José, Edith Lopez and Marcela Sanchez, whose useful insights helped improve the quality of this work. This research was partially supported by PICTO-ANPCyT 2010-0149, PICT-ANPCyT 2014-0624, PIP-CONICET 112-2012010-0487, PIP-CONICET 112-2009010-0863, PIP- CONICET, 112-2011010-1000, PIP-CONICET 112-200801-02798, PGI-UNS 24/N029 and PGI-UNS 24/N030.

References

`Albassuny, B. M. (2008). Automatic metadata generation applications: A survey study. *International Journal of Metadata, Semantics and Ontologies*, *3*(4), 260–282.

`Apache Software Foundation (2015a). Apache Lucene library [Software]. Retrieved from http://lucene.apache.org

`Apache Software Foundation (2015b). Apache Lucene MoreLikeThis library [Software]. Retrieved from https://lucene.apache.org/core/3_0_3/

`Apache Software Foundation (2015c). Apache Tomcat [Software]. Retrieved from http://tomcat.apache.org

`Babaian, T., Grosz, B. J., & Shieber, S. M. (2002). A writer's collaborative assistant. In *Proceedings of the International Conference on Intelligent User Interfaces*, *7*, 7–14.

`Bradshaw, J. M. (1997). An introduction to software agents. In J. M. Bradshaw (Ed.), *Software agents* (pp. 3–46). Menlo Park, CA: AAAI Press.

`Buckland, M. (1992*). Redesigning library services: A manifesto.* Chicago, IL: American Library Association.

`Cypher, A. (1991). Eager: Programming repetitive tasks by example. In S. P. Robertson, G. M. Olson, & J. S, Olson (Eds.), *Proceedings of the 1991 SIGCHI Conference on Human Factors in Computing Systems* (pp. 33–39). New York, NY: ACM Press.

`del Carmen Sua´rez Torrente, M., Prieto, A. B. M., Gutiérrez, D. Á., & de Sagastegui, M.E. A. (2013). Sirius: A heuristic-based framework for measuring web usability adapted to the type of website. *Journal of Systems and Software*, *86*(3), 649–663.

`Delgado, P. H., Maguitman, A. G., Ferracutti, V. M., & Herrera, L. A. (2011). Using thematic contexts and previous solutions for maintaining and accessing institutional repositories. *Journal of Computer Science and Technology*. Retrieved from http://journal.info.unlp.edu.ar/journal/journal31/papers/JCST-Oct11-1.pdf

`Dini, M. A., Varela, A., M., Antúnez, V., Maguitman, A., & Herrera, L. (2010, November). *Soporte inteligente para el mantenimiento y acceso contextualizado a repositorios institucionales* [Intelligent support for the maintenance of and contextualized access to institutional repositories]. Paper presented at the 8*th* Jornada sobre la Biblioteca Digital Universitaria, Buenos Aires, Argentina. Retrieved from http://www.amicus.udesa.edu.ar/documentos/8jornada/8jornada.html

`Engelbart, D. (1962). *Augmenting human intellect: A conceptual framework. Summary report, Stanford Research Institute, on Contract AF 49(638)–1024*. Retrieved from http://www.dougengelbart.org/pubs/augment-3906.html

`Frank, E., Paynter, G. W., Witten, I. H., Gutwin, C., & Nevill-Manning, C. G. (1999). Domain-specific keyphrase extraction. In *Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence* (pp. 668-671). Retrieved from http://ijcai.org/Past%20Proceedings/IJCAI-99%20VOL-2/PDF/002.pdf

`Gómez, F. J. (2015). Catalis [Software]. Retrieved from http://inmabb.criba.edu.ar/catalis/

`González, M. P., Lorés, J., & Granollers, T. (2008). Enhancing usability testing through datamining techniques: A novel approach to detecting usability problem patterns for a context of use. *Information & Software Technology*, *50*, 547–568.

`Greenberg, J. (2009). Theoretical considerations of lifecycle modeling: An analysis of the dryad repository demonstrating automatic metadata propagation, inheritance, and value system adoption. *Cataloging & Classification Quarterly*, *47*, 380–402.

`Greenberg, J., Losee, R., Agera, J. R. P., Scherle, R., White, H., & Willis, C. (2011). HIVE: Helping interdisciplinary vocabulary engineering. *Bulletin of the American Society for Information Science and Technology*, *37*(4), 23–26.

`Greenberg, J., Spurgin, K., & Crystal, A. (2005). *Final report for the AMeGA (Automatic Metadata Generation Applications) project.* Retrieved from www.loc.gov/catdir/bibcontrol/lc amega final report.pdf

`Hasan, K. S., & Ng, V. (2014). Automatic keyphrase extraction: A survey of the state of the art. *Proceedings of the Association for Computational Linguistics, 52*, 1262-1273. Retrieved from http://acl2014.org/acl2014/P14-1/pdf/P14-1119.pdf

`Horvitz, E. (1999). Principles of mixed-initiative user interfaces. In M. G. Williams & M. W. Alton (Eds.), *Proceedings of the 1999 SIGCHI Conference on Human Factors in Computing Systems* (pp. 159–166). New York, NY: ACM Press.

`Hsieh, S.-H., Lin, H.-T., Chi, N.-W., Chou, K.-W., and Lin, K.-Y. (2011). Enabling the development of base domain ontology through extraction of knowledge from engineering domain handbooks. *Advanced Engineering Informatics*, *25*, 288–296.

`International Organization for Standardization. (1998). *ISO 9241: Ergonomic requirements for office work with visual display terminals (VDTs) – Part 11. Guidance on usability*. Retrieved from https://www.iso.org/obp/ui/#!iso:std:16883:en

`Kim, S. N., Medelyan, O., Kan, M.-Y., & Baldwin, T. (2013). Automatic keyphrase extraction from scientific articles. *Language Resources and Evaluation*, *47*(3), 723–742.

`Knijnenburg, B. P., Reijmer, N. J. M., & Willemsen, M. C. (2011). Each to his own: how different users call for different interaction methods in recommender systems. *Proceedings of the ACM Conference on Recommender Systems*, *5*, 141–148.

`Laurel, B. (1997). Interface agents: Metaphors with character. In Bradshaw, J. (Ed.), *Software agents* (pp. 67-77). Menlo Park, CA: AAAI Press.

`Leake, D. (1996). CBR in context: The present and future. In Leake, D. (Ed.), *Case-based reasoning: Experiences, lessons, and future directions* (pp. 3-30). Menlo Park, CA: AAAI Press.

`Leake, D., Maguitman, A., & Reichherzer, T. (2014). *Experience-based support for human-centered knowledge modeling. Knowledge-Based Systems*, Amsterdam, Netherlands: Elsevier, 68(0), 77 – 87.

`Levy, D. M., & Marshall, C. C. (1995). Going digital: A look at assumptions underlying digital libraries. *Communications of the ACM*, *38*(4), 77–84.

`Licklider, J. C. R. (1960). Man-computer symbiosis. *IRE Transactions on Human Factors in Electronics*, *HFE-1*(1), 4–11.

`Lopez, P. & Romary, L. (2010a). GRISP: A massive multilingual terminological database for scientific and technical domains. *Proceedings of the International Conference on Language Resources and Evaluation*, *7*. Retrieved from https://hal.inria.fr/inria-00490312/document

`Lopez, P. & Romary, L. (2010b). Humb: Automatic key term extraction from scientific articles in grobid. *Proceedings of the International Workshop on Semantic Evaluation*, *5*, 248–251.

`Maes, P. (1994). Agents that reduce work and information overload. *Communications of the ACM*, *37*(7), 30–40.

`Medelyan, O., Frank, E., & Witten, I. H. (2009). Human-competitive tagging using automatic keyphrase extraction. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing* (pp. 1318–1327). Stroudsburg, PA: Association for Computational Linguistics.

`Negroponte, N. (1997). Agents: From direct manipulation to delegation. In Bradshaw, J. (Ed.), *Software agents* (pp. 57–66). Menlo Park, CA: AAAI Press/MIT Press.

`OCLC, Inc. (2015a). Catexpress [Software]. Retrieved from https://www.oclc.org/catexpress.en.html

`OCLC, Inc. (2015b). Connexion [Software]. Retrieved from https://www.oclc.org/connexion.en.html

`Ozok, A. A., Fan, Q., & Norcio, A. F. (2010). Design guidelines for effective recommender system interfaces based on a usability criteria conceptual model: Results from a college student population. *Behaviour & Information Technology*, *29*(1), 57–83.

`Park, J., & Lu, C. (2009). Application of semi-automatic metadata generation in libraries: Types, tools, and techniques. *Library & Information Science Research*, *31*, 225–231.

`Paynter, G. W. (2005). Developing practical automatic metadata assignment and evaluation tools for internet resources. *Proceedings of the ACM/IEEE-CS Joint Conference on Digital Libraries*, *5*, 291–300.

`Pu, P., Chen, L., & Hu, R. (2011). A user-centric evaluation framework for recommender systems. *Proceedings of the ACM Conference on Recommender Systems*, *5*, 157–164.

`Puckett, J. (2011). *Zotero: A guide for librarians, researchers, and educators*. Chicago IL: ACRL.

`Reese, T. (2015). Marcedit [Software]. Retrieved from http://people.oregonstate.edu/~reeset/marcedit/html/index.php

`Rodriguez, M. A., Bollen, J., & Sompel, H. V. D. (2009). Automatic metadata generation using associative networks. *ACM Transactions on Information Systems*, *27*(2), 7.

`Salton, G. (1971). *The SMART retrieval system: Experiments in automatic document processing*. Upper Saddle River, NJ: Prentice-Hall.

`Shneiderman, B. (1992). *Designing the user interface: Strategies for effective human-computer interaction*. Essex, UK: Addison-Wesley.

`Smith, D. C. (1977). *Pygmalion: A computer program to model and stimulate creative thought*. Basel, Switzerland: Birkhäuser.

`Sølvberg, I. (2001). Digital libraries and information retrieval. In M. Agosti, F. Crestani, & G. Pasi (Eds.), *Lectures on information retrieval* (pp. 139–156). London, UK: Springer-Verlag.

`Sutherland, I. (1963). Sketchpad: A man–machine graphical communication system. In E. C. Johnson (Chair), *Proceedings of the May 21-23 Spring Joint Computer Conference* (pp. 329–346). New York, NY: ACM Press.

`Turney, P. D. (2000). Learning algorithms for keyphrase extraction. *Information Retrieval*, *2,* 303–336.

`Witten, I. H., Paynter, G. W., Frank, E., Gutwin, C., & Nevill-Manning, C. G. (1999). Kea: Practical automatic keyphrase extraction. *Proceedings of the ACM Conference on Digital Libraries, 4,* 254–255.

`Xia, C., & Maes, P. (2013). The design of artifacts for augmenting intellect. *Proceedings of the Augmented Human International Conference, 4,* 154–161.

`Zhang, K., Xu, H., Tang, J., & Li, J. (2006). Keyword extraction using support vector machine. *Proceedings of the International Conference on Advances in Web-Age Information Management, 7,* 85–96.

`Ziegler, J. E. & Fahnrich, K. P. (1988). Direct manipulation. In M. Helander (Ed.), *Handbook of human-computer interaction* (pp. 123-132). Amsterdam, Netherlands: Elsevier.

`Zotero [Software] (2015). Retrieved from http://www.zotero.org

Figure 1: The SciELO Suggester's cycle.

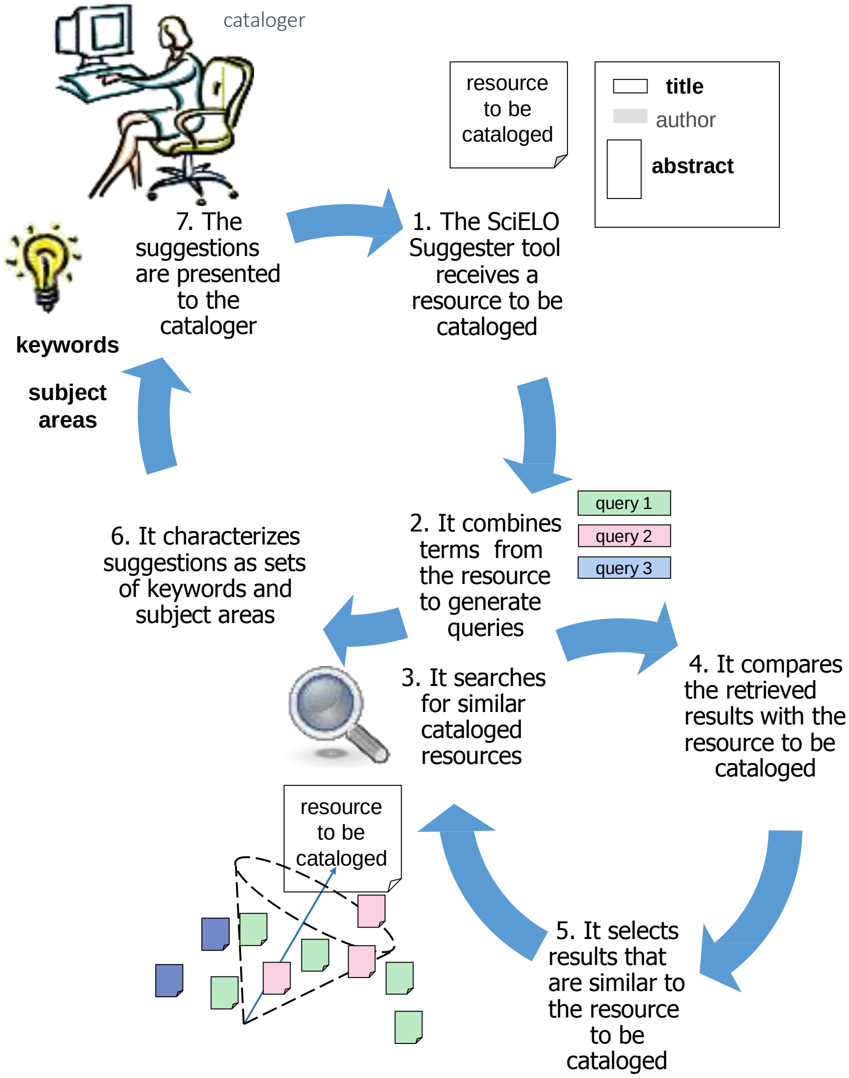

Figure 2: The SciELO Suggester’s context menu (the last option triggers the suggestions)

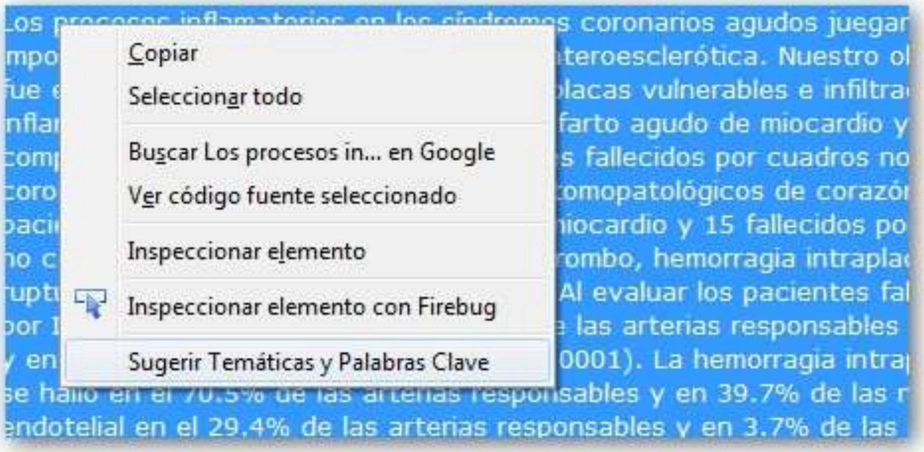

Figure 3: The SciELO Suggester's interface showing the relevant terms and generated suggestions, including the subject areas and their associated keywords

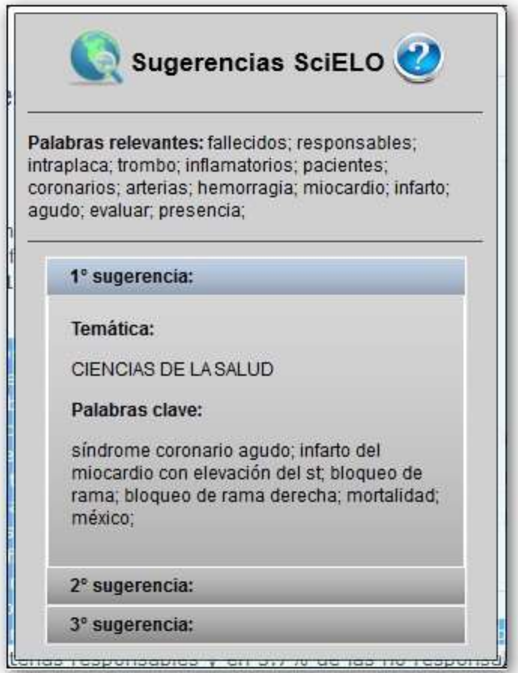

Figure 4: Example of a question used during the user study for assessing the usefulness of the suggested keywords.

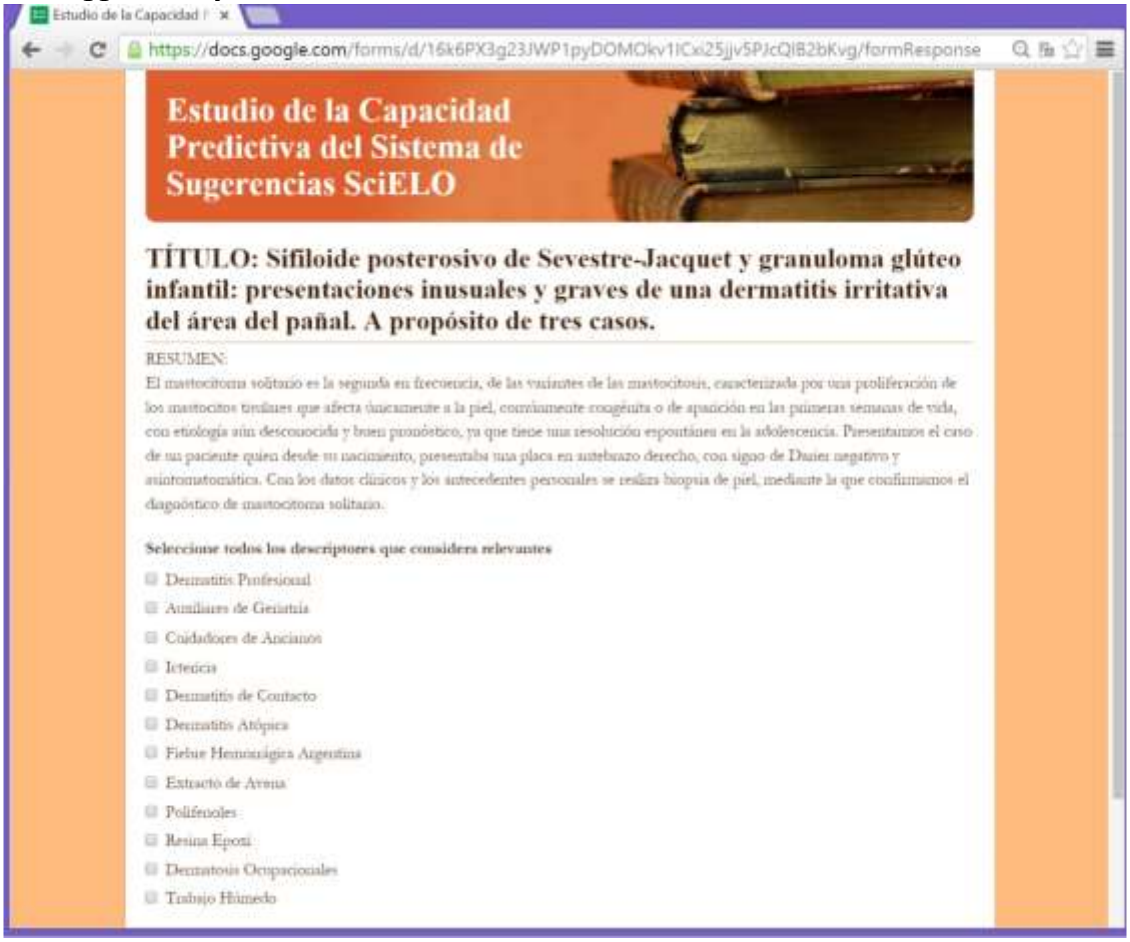

Table 1: Usefulness of the suggested keywords.

Questions:	Q_1	Q_2	Q_3	Q_4	Q_5	M
Number of displayed keywords (suggested and random keywords)	14	12	12	12	10	12
Number of suggested keywords selected by at least one participant	6	4	2	6	5	4.6
Number of suggested keywords selected by most participants	3	2	1	5	2	2.6
Number of suggested keywords selected by all participants	2	1	0	0	1	0.8
Number of unrelated keywords selected by at least one participant	0	0	0	0	0	0

Table 2: Number of articles associated with each subject area

Subject area	***Number of articles***
T1: Agricultural Science	2,403
T2: Biological Science	4,250
T3: Health Science	23,754
T4: Earth Science	2,775
T5: Geosciences	197
T6: Social Sciences	1,705
T7: Applied Social Sciences	4,421
T8: Humanities	4,652
T9: Engineering	1,289
T10: Linguistics, Literature and Arts	164
T11: Mathematics	9
T12: Chemistry	123
Total number of articles	45,742

Table 3: Prediction success for the suggested subject areas (N = 120).

	M	*SD*	*SE*	*95% CI*	
First prediction	0.53	0.15	0.03	0.51	0.56
Top 2	0.95	0.26	0.05	0.90	1.00
Top 3	1.38	0.34	0.06	1.32	1.44

Table 4: Numerical and grayscale confusion matrices for the first prediction

	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12
T1	9	1	0	0	0	0	0	0	0	0	0	0
T2	1	7	1	1	0	0	0	0	0	0	0	0
T3	0	0	10	0	0	0	0	0	0	0	0	0
T4	0	0	1	5	0	2	1	1	0	0	0	0
T5	0	2	0	1	7	0	0	0	0	0	0	0
T6	0	0	0	0	0	2	0	7	0	1	0	0
T7	1	0	1	1	0	1	4	1	1	0	0	0
T8	0	0	1	0	0	0	3	6	0	0	0	0
T9	1	1	0	2	0	0	2	0	4	0	0	0
T10	0	0	0	0	0	1	0	6	0	3	0	0
T11	1	0	2	3	0	0	0	0	3	0	1	0
T12	0	1	0	1	0	0	0	0	2	0	0	6

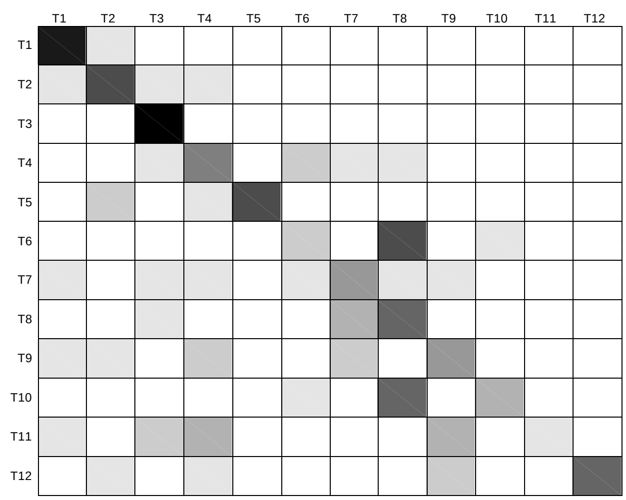

Table 5: Numerical and grayscale confusion matrices for the top-two predictions

	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12
T1	18	2	0	0	0	0	0	0	0	0	0	0
T2	1	15	1	3	0	0	0	0	0	0	0	0
T3	0	0	20	0	0	0	0	0	0	0	0	0
T4	0	2	2	6	0	3	4	3	0	0	0	0
T5	0	3	0	6	11	0	0	0	0	0	0	0
T6	0	0	0	0	0	3	2	14	0	1	0	0
T7	2	0	3	1	0	2	8	3	1	0	0	0
T8	0	0	1	0	0	3	7	9	0	0	0	0
T9	3	2	0	6	0	0	2	0	7	0	0	0
T10	0	0	0	0	0	3	1	11	1	4	0	0
T11	1	0	2	10	0	0	0	1	4	0	2	0
T12	2	1	0	2	0	0	1	0	5	0	0	9

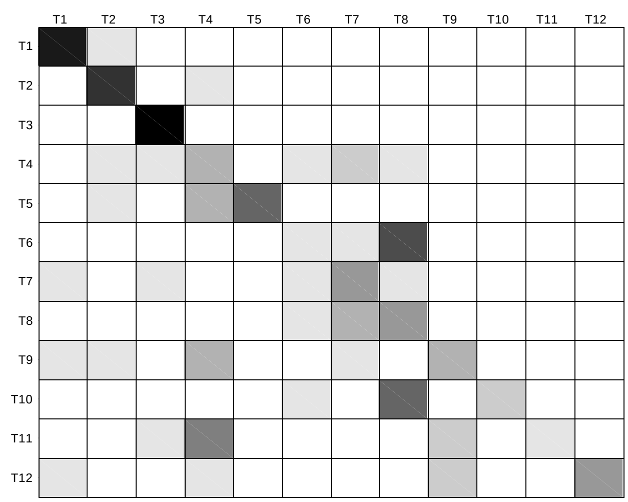

Table 6: Numerical and grayscale confusion matrices for the top-three predictions

	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12
T1	27	3	0	0	0	0	0	0	0	0	0	0
T2	1	21	2	6	0	0	0	0	0	0	0	0
T3	0	1	29	0	0	0	0	0	0	0	0	0
T4	0	3	4	6	0	4	8	5	0	0	0	0
T5	0	4	0	10	15	0	0	1	0	0	0	0
T6	0	0	1	0	0	4	5	19	0	1	0	0
T7	2	0	4	1	0	5	13	3	2	0	0	0
T8	0	0	1	0	0	3	9	17	0	0	0	0
T9	4	3	0	8	0	0	2	1	12	0	0	0
T10	0	0	0	0	0	7	1	16	1	5	0	0
T11	1	0	3	17	0	0	0	2	5	0	2	0
T12	6	2	1	3	0	0	1	0	7	0	0	10

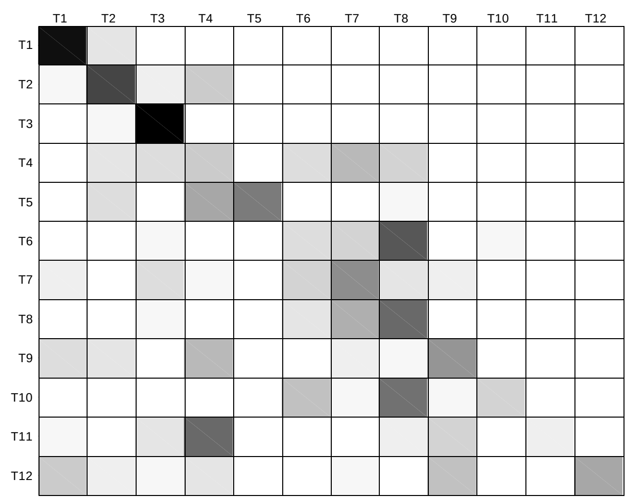

Table 7: Precision and Recall for Agricultural Science.

	T1’	T1
T1’	106	4
T1	1	9

Precision(T1) = 9 / (9 + 4) = 0.69
Recall(T1) = 9 / (1 + 9) = 0.90

Table 8: Precision and Recall for Biological Science.

	T2’	T2
T2’	105	5
T2	3	7

Precision(T2) = 7 / (7 + 5) = 0.53
Recall(T2) = 7 / (3 + 7) = 0.70

Table 9: Precision and Recall for Health Science.

	T3’	T3
T3’	104	6
T3	0	10

Precision(T3) = 10 / (10 + 6) = 0.62
Recall(T3) = 10 / (0 + 10) = 1

Table 10: Precision and Recall for Earth Science.

	T4’	T4
T4’	101	9
T4	5	5

Precision(T4) = 5 / (5 + 9) = 0.35

Recall(T4) = 5 / (5 + 5) = 0.50

Table 11: Precision and Recall for Geosciences.

	T5'	T5
T5'	110	0
T5	3	7

Precision(T5) = 7 / (7 + 0) = 1
Recall(T5) = 7 / (3 + 7) = 0.70

Table 12: Precision and Recall for Social Sciences.

	T6'	T6
T6'	106	4
T6	8	2

Precision(T6) = 2 / (2 + 4) = 0.33
Recall(T6) = 2 / (8 + 2) = 0.20

Table 13: Precision and Recall for Applied Social Sciences.

	T7’	T7
T7’	104	6
T7	6	4

Precision(T7) = 4 / (4 + 6) = 0.40
Recall(T7) = 4 / (6 + 4) = 0.40

Table 14: Precision and Recall for Humanities.

	T8’	T8
T8’	95	15
T8	4	6

Precision(T8) = 6 / (6 + 15) = 0.28
Recall(T8) = 6 / (4 + 6) = 0.60

Table 15: Precision and Recall for Engineering.

	T9’	T9
T9’	104	6
T9	6	4

Precision(T9) = 4 / (4 + 6) = 0.40
Recall(T9) = 4 / (6 + 4) = 0.40

Table 16: Precision and Recall for Linguistics, Literature and Arts.

	T10’	T10
T10’	109	1
T10	7	3

Precision(T10) = 3 / (3 + 1) = 0.75
Recall(T10) = 3 / (7 + 3) = 0.30

Table 17: Precision and Recall for Mathematics.

	T11’	T11
T11’	110	0
T11	9	1

Precision(T11) = 1 / (1 + 0) = 1
Recall(T11) = 1 / (9 + 1) = 0.10

Table 18: Precision and Recall for Chemistry.

	T12'	T12
T12'	110	0
T12	4	6

Precision(T12) = 6 / (6 + 0) = 1
Recall(T12) = 6 / (4 + 6) = 0.60

Table 19: Sirius matrix for testing SciELO Suggester system usability as a web resource (part 1). Rating column.

Information Quality			
Category	*Item*	*Rating*	*Comment*
Authorship	Are the authors identified within the resource?	3	It assesses whether the web resource is perfectly identified.
	Are the authors well qualified?	5	It determines whether authors' titles, background, experience and resume are related to the web topic. In the case of institutions, institution type is taken into account (.edu, .com, .org, etc.).
	Is the authors' contact information available on the web?	3	It determines whether the web page offers information to contact authors or web administrators.
Content	Is the information provided precise and concise?	5	It determines information accuracy in relation to information background and formatting.
	Are web resources properly updated?	5	It determines traceable updating frequency (creation date, versions update dates, information about re-editions or changes, etc.) both in the main web page and sections liable to turning obsolete.
	Do the resource and information provided cover the web topic properly?	3	It determines completeness, data omission or resource limitations. It determines the existence of links to other web sites that complement the provided resources and information.
	Are web information and resources presented objectively?	4	It determines whether there are proper arguments supporting the given information, and if the vocabulary is appropriate to the purpose and audience (informing, providing a resource, convincing, etc.).

Table 20: Sirius matrix for testing SciELO Suggester system usability as a web resource (part 2). Rating column.

Information			
Category	*Item*	*Rating*	*Comment*
Content	Is the information provided original enough?	5	It determines whether the resource contains original information or a fresh viewpoint regarding the topic or some contribution to improve the state of the art.
	Is the information provided useful in reference to the web resource?	5	It determines the relevance of the provided information in context.
	Are there links or references supporting the provided information?	2	It determines whether the information is supported by statistical data or numbers, or by some opinions and conclusions regarding some particular data. It determines whether there are references or links explicitly stated.
	Are the grammar and syntax of the information provided correct?	5	It determines whether the formal elements presented in the text were completely covered and if there is evidence that the information included has been properly
	Does the resource provide a summary or an outline so as to quickly overview its main structure?	1	A good summary shows how the information has been structured without having to consult the entire resource.

Table 21: Sirius matrix for testing SciELO Suggester system usability as a web resource (part 3). Rating column.

Format Quality			
Category	*Item*	*Rating*	*Comment*
Accessibility	Is it easy to access the resource?	3	It is related to the number of steps or actions needed to open the resource.
	Is the download time appropriate? Does the resource have a progress bar or some way to indicate the remaining time to complete the download?	5	It allows comparing the download time with the download time of similar resources.
	Is the resource free?	5	Even if the use of the resource requires some kind of registration, access and free use of the available information has to be granted to all users.
Usability	Are the color combination, text and graphics pleasant?	4	The quantity and quality of the objects included in the resource have to be appropriate, as well as the organization and interrelationships between them.
	Is the design minimalist?	5	It determines whether the content is easy to read, if figures and images are easily recognized from the text and the background and if affordance is achieved in a reasonable degree.
	Are graphics and images correctly used?	5	Do graphics and images add value to the text
	Is it easy to browse the resource?	5	It determines the navigation quality. It assesses the possibility of reaching the main resource page from anywhere.

Table 22: Sirius matrix for testing SciELO Suggester system usability as a web resource (part 4). Rating column.

Format Quality			
Category	*Item*	*Rating*	*Comment*
Usability	Is the information easy to identify and to access?	4	It assesses whether the resource information can be consulted by following the resource links.
	Does the resource have a general index?	1	A good summary shows how the information has been structured without having to consult the entire resource.
	Does the resource have a link to related news and novelties?	1	This type of links allows to quickly access novel information or news related to the resource.
	Does the resource have online help?	1	A help web page is valuable support for users, providing information on good practices.

Table 23: Questionnaire used for the cognitive walkthrough.

Observation: Answer each question using values 0 (worst score) to 5 (best score).	
1.	Is the system operation clear or are there circumstances or actions that you cannot identify? 0 - 1 - 2 - 3 - 4 - 5 If there are any actions or situations that cannot be identified, specify which they are.
2.	If you want to undo a completed action a. Is it easy to undo the action? 0 - 1 - 2 - 3 - 4 - 5 b. Is the number of steps needed to undo the action appropriate? 0 - 1 - 2 - 3 - 4 - 5 c. Does the system acknowledge that the action has been undone? 0 - 1 - 2 - 3 - 4 - 5
3.	Are all the options needed to complete the cataloging process available through the system or are there any options that are required but cannot be found? 0 - 1 - 2 - 3 - 4 - 5 If your answer is positive, specify what the options that you cannot find are.
4.	Does the system screen present irrelevant elements that distract you? 0 - 1 - 2 - 3 - 4 - 5 If your answer is positive, specify which they are.
5.	Does the system screen indicate in some way what action is being carried out? 0 - 1 - 2 - 3 - 4 - 5
6.	Can you clearly distinguish the different actions that are being carried out? 0 - 1 - 2 - 3 - 4 - 5
7.	Is the vocabulary familiar? Is it appropriate for the cataloging processes that you are performing or is it too technical (informatics oriented instead of catalogers oriented)? 0 - 1 - 2 - 3 - 4 - 5
8.	If you make a mistake or there is a system error a. Are the error messages clear? 0 - 1 - 2 - 3 - 4 - 5 b. Are there instructions on how to proceed when an error is detected? 0 - 1 - 2 - 3 - 4 - 5
9.	What general score would you choose to evaluate the system? (0=worst score to 10=best score) Briefly justify the selected score. What aspects do you consider should be improved or changed?