Automatic topics identification for reviewer assignment

S. Ferilli, N. Di Mauro, T.M.A. Basile, F. Esposito, and M. Biba

Dipartimento di Informatica, University of Bari, Italy
{ferilli,ndm,basile,esposito,biba}@di.uniba.it

Abstract. Scientific conference management involves many complex and multi-faceted activities, which would make highly desirable for the organizing people to have a Web-based management system that makes some of them a little easier to carry out. One of such activities is the assignment of submitted papers to suitable reviewers, involving the authors, the reviewers and the conference chair. Authors that submit the papers usually must fill a form with paper title, abstract and a set of conference topics that fit their submission subject. Reviewers are required to register and declare their expertise on the conference topics (among other things). Finally, the conference chair has to carry out the review assignment taking into account the information provided by both the authors (about their paper) and the reviewers (about their competencies). Thus, all this subtasks needed for the assignment are currently carried out manually by the actors. While this can be just boring in the case of authors and reviewers, in case of conference chair the task is also very complex and time-consuming.
In this paper we proposes the exploitation of intelligent techniques to automatically extract paper topics from their title and abstract, and the expertise of the reviewers from the titles of their publications available on the Internet. Successively, such a knowledge is exploited by an expert system able to automatically perform the assignments. The proposed methods were evaluated on a real conference dataset obtaining good results when compared to handmade ones, both in terms of quality and user-satisfaction of the assignments, and for reduction in execution time with respect to the case of humans performing the same process.

1 Introduction

Organizing scientific conferences is a complex and multi-faceted activity that often requires the use of a Web-based management system to make some tasks a little easier to carry out, such as the job of reviewing papers. Some of the features typically provided by these packages are: submission of abstracts and papers by Authors; submission of reviews by the Program Committee Members (PCMs); download of papers by the Program Committee (PC); handling of reviewers preferences and bidding; Web-based assignment of papers to PCMs for review; review progress tracking; Web-based PC meeting; notification of acceptance/rejection; sending e-mails for notifications. One of the hardest and most

time-consuming tasks in Scientific Conferences organization is the process of assigning reviewers to submitted papers. Due to the many constraints to be fulfilled, carrying out manually such a task is very tedious and difficult, and does not guarantee to result in the best solution.

In the current practice, before the submission phase starts, the Chair usually sets up a list of research topics of interest for the conference, and all reviewers are asked to specify which of them correspond to their main areas of expertise. On the other hand, during the submission process, authors are asked to explicitly state which conference topics apply to their papers. Such an information provides a first guideline for associating reviewers to papers. One possible source of problems, in the above procedure, lies in the topics selected by the authors being sometimes misleading with respect to the real topic of the paper. For this reason, in order to make the assignment more objective, it would be desirable to automatically infer the paper topics rather than asking the authors to explicitly provide such an information.

While the topics selected by (or inferred for) a reviewer refer to his background competencies, in some cases the reviewers could have specific preferences about papers due to matter of taste or to other vague questions (e.g., the reviewer would like to review a paper just for curiosity; the abstract is imprecise or misleading, etc.). For this reason, the bidding preferences approach is sometimes preferred over the expertise one. We take into account both, but give priority to the one based on the reviewer expertise, assuming that if a paper bid by a reviewer does not match his topics of expertise, this should be considered as a warning. To this concerns, a small pattern language has been defined in the literature that captures successful practice in several conference review processes [8]. In this work two patterns are followed, indicating that papers should be matched, and assigned for evaluation, to reviewers who are competent in the specific paper topics (*ExpertsReviewPapers*), and to reviewers who declared to be willing to review those papers in the bidding phase (*ChampionsReviewPapers*).

This work aims at showing how this complex real-world domain can take advantage of intelligent techniques for indexing and retrieving documents and their associated topics. Specifically, it describes an intelligent component developed to be embedded in scientific Conference Management Systems that is able to automatically:

- identify paper topics, among those of interest for the conference, by exploiting the paper title and abstract;
- identify reviewers expertise, among the conference topics, by exploiting the title of the reviewers publications available in the Internet;
- assign reviewers to papers submitted to a conference.

The identification of paper topics and reviewers expertise is performed starting from the output of an automatic system for document analysis and then exploiting NLP methods for automatically extracting significant topics. Then, the assignment process is performed by an expert system that takes as input this information. Thus, the methods that we propose aim at exploiting intelli-

gent techniques in the *ExpertsReviewPapers* pattern, that so far was applicable only if some steps were manually performed by the users.

2 Reviewers Assignment: the General Framework

In order to perform the assignment, the Chair needs to know both the conference topics selected for each submitted paper and the topics that better describe the reviewers expertise. In the following we show how it is possible to automatically acquire such a knowledge by means of the *Latent Semantic Indexing* (LSI) technique. As regards the papers, a system for the automatic processing of the submitted documents will be presented. It will be exploited in order to automatically extract the significant components, i.e. title and abstract, from the paper without the author do it manually. As concern the reviewers, the information needed from the application of the LSI was extracted from the online repository of their publications (at the moment this task is carried out manually). Successively, an expert system that automatically performs the assignments based on the extracted knowledge about the papers/reviwers topics will be presented.

2.1 Latent Semantic Indexing

A problem of most existing word-based retrieval systems consists of their ineffectiveness in finding interesting documents when the users do not use the same words by which the information they seek has been indexed. This is due to a number of tricky features that are typical of natural language. One of the most common concerns the fact that there are many ways (words) to express a given concept (*synonymy*), and hence the terms in a user's query might not match those of a document even if it could be very interesting for him. Another one is that many words have multiple meanings (*polysemy*), so that terms in a user's query will literally match terms in documents that are not semantically interesting to the user.

The LSI technique [3] tries to overcome the weaknesses of term-matching based retrieval by treating the unreliability of observed term-document association data as a statistical problem. Indeed, LSI assumes that there exists some underlying latent semantic structure in the data that is partially obscured by the randomness of word choice with respect to the retrieval phase and that can be estimated by means of statistical techniques. LSI relies on a mathematical technique called *Singular-Value Decomposition* (SVD). Starting from a (large and usually sparse) matrix of term-document association data, the SVD allows to build and arrange a semantic space, where terms and documents that are closely associated are placed near to each other, in such a way to reflect the major associative patterns in the data, and ignore the smaller, less important influences. As a result, terms that do not actually appear in a document may still end up close to it, if this is consistent with the major association patterns in the data. Position in the space thus serves as a new kind of semantic indexing, and retrieval proceeds by using the terms in a query to identify a point in the

space, and returning to the user documents in its neighbourhood. It is possible to specify a reduction parameter that intuitively represents the number of different concepts to be taken into account, among which distributing the available terms and documents.

The large amount of items that a document management system has to deal with, and the continuous flow of new documents that could be added to the initial database, require an incremental methodology to update the initial LSI matrix. Indeed, applying from scratch at each update the LSI method, taking into account both the old (already analysed) and the new documents, would become computationally inefficient. Two techniques have been developed in the literature to update (i.e., add new terms and/or documents to) an existing LSI generated database: Folding-In [1] and SVD-Updating [9]. The former is a much simpler alternative that uses the existing SVD to represent new information but yields poor-quality updated matrices, since the information contained in the new documents/terms is not exploited by the updated semantic space. The latter represents a trade-off between the former and the recomputation from scratch.

2.2 The Document Management System

This section presents the current version of DOMINUS (DOcument Management INtelligent Universal System) [5], a system for automated electronic documents processing characterized by the intensive exploitation of intelligent techniques in each step of the document management process: acquisition, layout analysis, document image understanding, indexing, for categorization and information retrieval purposes. It can deal with documents in standard formats, such as PostScript (PS) or its evolution Portable Document Format (PDF).

The layout analysis process on documents in electronic format, sketched in Figure 1, is now reported along with the steps performed by the system going from the original PDF/PS document to the text extraction and indexing.

1. WINE: Rewrites basic PostScript operators to turn their drawing instructions into objects. It takes as input a PDF/PS document and produces (by an intermediate vector format) the initial document's XML basic representation, that describes it as a set of pages made up of basic blocks.
2. Rewriting rules: Identifies rewriting rules that could suggest how to set some parameters in order to group together rectangles (words) to obtain lines. Specifically, such a learning task was cast to a Multiple Instance Problem (MIP) and solved by exploiting the kernel-based method proposed in [4].
3. DOC: Collects semantically related basic blocks into groups by identifying frames that surround them based on whitespace and background structure analysis. This is a variant of Breuel's algorithm [2], that finds iteratively the maximal white rectangles in a page. The modification consisted in a bottom-up grouping of basic blocks into words and lines and in the empirical identification of a stop criterion to end the process before finding insignificant white spaces such as inter-word or inter-line ones.

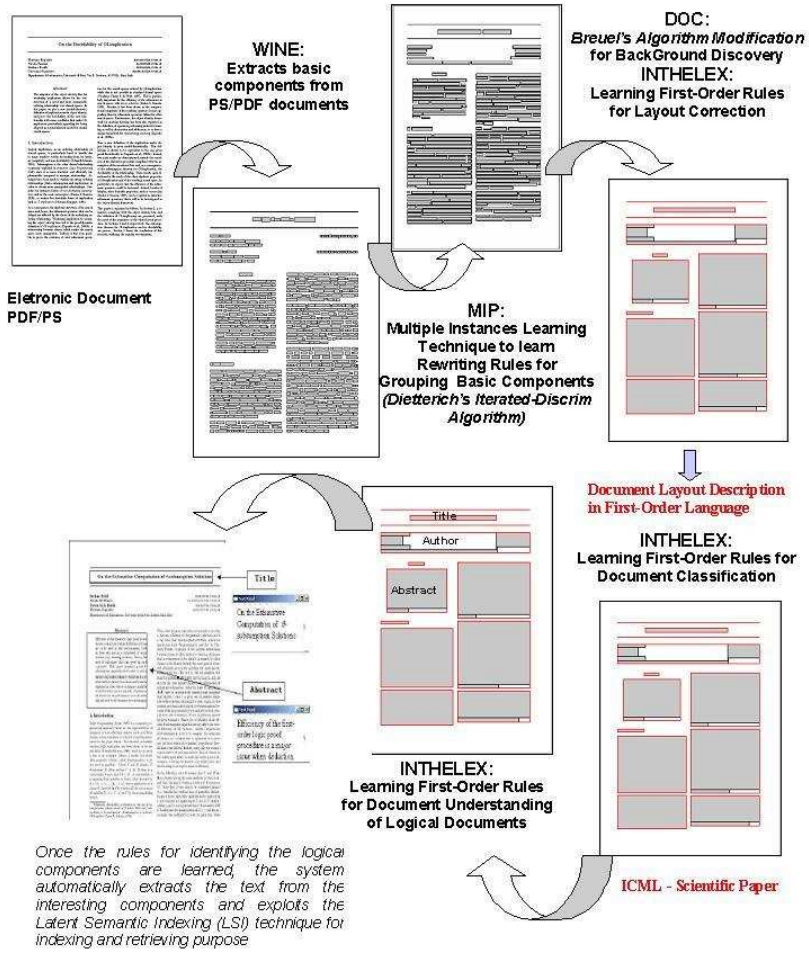

Fig. 1. Document Management System

4. Layout Correction: At the end of the previous step it could be possible that some blocks are not correctly recognized, i.e. background areas are considered content ones and vice versa. In such a case a phase of layout correction is needed, that is automatically performed in DOC by applying embedded rules automatically learned for this task. To this purpose, we firstly collect the manual corrections performed on some documents and describe them by means of a first-order language representing both the situations *before* and *after* the manual correction, then we exploit INTHELEX [6] (a first-order logic learner) on this training set in order to identify correction rules.
5. Classification: Associates the document to a class that expresses its type (e.g., scientific/newspaper article, etc.). Since the logical structure is obviously different according to the kind of document, classification of the document is a

preliminary step before recognizing the relevant components for that document (e.g., a sender is significant for a mail but non for a newspaper article). INTHELEX is also exploited to learn rules for the automatic identification of the class.

6. Understanding: Identifies the significant layout components for the class previously recognized and associates to each of them a tag that expresses its role (e.g., title, author, abstract, etc.). Again, INTHELEX is exploited to learn rules for the automatic identification of the logical components.
7. Extraction: Extracts the text from the significant components.
8. Indexing: Exploits the Latent Semantic Indexing technique to index the documents.

The following scenario can give an idea of how DOMINUS can be exploited in the submission phase, and of what advantages it can bring to the involved people. An Author connects to the Internet and (after registering, or after logging in if already registered) opens the submission page, where he can browse his hard disk and submit a paper by choosing the corresponding file in one of the accepted formats. The paper is received and undergoes the various processing steps. The layout analysis algorithm is applied, in order to single out its layout components. Then, it is translated into a first-order logic description and classified by a proper module according to the theory learned so far for the acceptable submission layout standards (e.g., full paper, poster, demo). Depending on the identified class, a further step exploits the same description to locate and label the layout components of interest for that class (e.g., title, author, abstract and references in a full paper). The text that makes up each of such components is read, stored and used to automatically file the submission record (e.g., by filling its title, authors and abstract fields).

If the system is unable to carry out any of these steps, such an event is notified to the Conference administrators, that can manually fix the problem and let the system complete its task. Such manual corrections are logged and used by the incremental learning component to refine the available classification/labeling theories in order to improve their performance on future submissions. Nevertheless, this is done off-line, and the updated theory replaces the old one only after the learning step has been successfully completed: this allows further submissions to take place in the meantime, and makes the refinement step transparent to the Authors. Alternatively, the fixes can be logged and exploited all at once to refine the theory when its performance falls below a given threshold. Successively a categorization of the paper content according to the text read is performed, with the purpose of allowing to match the paper topics against the reviewers' expertise, in order to find the best associations for the final assignment. Specifically, the text contained in the title and abstract is exploited, since we assume they compactly summarize the subject and research field the paper is concerned with, respectively.

2.3 The Papers-Reviewers Assignment Phase

GRAPE (Global Review Assignment Processing Engine) [7], is an expert system, written in CLIPS, for solving the reviewers assignment problem, that takes advantage of both the papers content (topics) and the reviewers expertise and preferences (biddings). It could be used by exploiting, in addition to the papers topics, the reviewers expertise only, or both the reviewers expertise and biddings. In the following a brief description of the system is given.

Let $P = \{p_1, \ldots, p_n\}$ denote the set of n papers submitted to the conference C, regarding t topics (*conference topics*, TC), and $R = \{r_1, \ldots, r_m\}$ the set of m reviewers. The goal is to assign the papers to reviewers, such that the following basic constraints are fulfilled:

1. each paper is assigned to exactly k reviewers (usually, k is set to 3 or 4);
2. each reviewer should have roughly the same number of papers to review (the mean number of reviews per reviewer is equal to nk/m);
3. papers should be reviewed by domain experts;
4. reviewers should revise articles based on their expertise and preferences.

As regards constraint 2, GRAPE can take as input additional constraints indicating that some specific reviewer r must review at most h paper. These constraints override the general principle and must be taken into account for calculating the mean number of reviews for the other reviewers.

Two measures were defined to guide the system during the search of the best solutions: the *reviewer's gratification* and the *article's coverage*. The former represents the gratification degree of a reviewer, calculated on the basis of the papers assigned to him. It is based on the *confidence degree* between the reviewer and the assigned articles (the confidence degree between a paper p_i concerning topics Tp_i and the reviewer r_j expert in topics Tr_j is defined as the number of topics in common) and on the number of assigned papers that were actually bid by the reviewer. The article's coverage represents the coverage degree of an article after the assignments. It is based on the *confidence degree* between the article and the reviewers it was assigned to (the same as before), and the *expertise degree* of the assigned reviewers (represented by the number of topics in which they are expert, and computed for a reviewer r_j as Tr_j/TC). GRAPE tries to maximize both measures during the assignment process, in order to fulfil the basic constraints 3 and 4. To reach this goal a fundamental requirement is that each reviewer must provide at least one topic of preference, otherwise the article coverage degree would be always null.

The assignment process is carried out in two phases. In the former, the system progressively assigns reviewers to papers with the lowest number of candidate reviewers. At the same time, the system *prefers* assigning papers to reviewers with few assignments. In this way, it avoids to have reviewers with zero or few assigned papers. Hence, this phase can be viewed as a search for review assignments by keeping low the average number of reviews *per* reviewer and maximizing the coverage degree of the papers. In the latter phase, the remaining assignments are chosen by considering first the confidence levels and then the expertise level

of the reviewers. In particular, given a paper p_i which has not been assigned k reviewers yet, the system tries to assign it to a reviewer r_j with a high confidence level between r_j and p_i. In case it is not possible, it assigns a reviewer with a high level of expertise.

The assignments resulting from the base process are presented to each reviewer, that receives the list A of the h assigned papers, followed by the list A′ of the remaining ones, in order to actually issue his bidding. When all the reviewers have bid their papers, GRAPE searches for a new solution that takes into account these biddings as well, in addition to the information about expertise. In particular, it tries to change previous assignments in order to maximize both article's coverage and reviewer's gratification. By taking the article's coverage high, the system tries to assign the same number of papers bid with the same class to each reviewer. Then, the solution is presented to the reviewers as the final one.

The main advantage of GRAPE relies in the fact that it is a rule-based system. Hence, it is very easy to add new rules in order to change/improve its behavior, and it is possible to describe background knowledge, such as further constraints or conflicts, in a natural way. For example, one could insert a rule expressing the preference to assign a reviewer to the articles in which he is cited, assuming that he should be an expert in those fields.

3 Evaluation

The system was evaluated on a real-world dataset built by using data from the 18th Conference on Industrial & Engineering Applications of Artificial Intelligence & Expert Systems (IEA/AIE 2005), whose Call for Papers identified 34 topics of interest. The papers submitted were 264 and the reviewers 60. Since the objective was to assess the performance of the automatic topic recognition methodology, in this case only the reviewers' expertise, and not their bidding were exploited by the paper assignment system.

The following steps were carried out. Firstly, the layout of each paper was automatically analyzed in order to recognize the significant components. In particular, the abstract and title were considered the most representative of the document subject, and hence the corresponding text was read. The words contained therein were stemmed according to the technique proposed by Porter [10], resulting in a total of 2832 word stems, on which the LSI technique was applied in order to index the whole set of documents. Then, the same procedure was applied to index the reviewers, resulting in 2204 stems. In this case, the titles of their papers appearing in the DBLP Computer Science Bibliography repository (http://www.informatik.uni-trier.de/∼ley/db/) were exploited. With respect to exploiting their homepages' information on research interests, this ensured a more uniform description. Compared to manually selecting the title of their publications, this ensured more completeness, even if at the cost of not having the abstracts available as well.

In both cases, the LSI parameters were set in such a way that all the conference topics were covered as different concepts. The experiment consisted in performing 34 queries, each corresponding to one conference topic, on both the papers and the reviewers in the database previously indexed, and then in associating to each paper/reviewer the topics for which it/he appears among the first l results. Specifically, the results on document topic recognition showed that 88 documents per query had to be considered, in order to include the whole set of documents. However, returning just 30 documents per query, 257 out of 264 documents (97.3%) were already assigned to at least one topic, which is an acceptable trade-off (the remaining 7 documents can be easily assigned by hand). Thus, 30 documents were considered a good parameter, and exploited to count the distribution of the topics between the documents. Interestingly, more than half of the documents (54.7%) concern between 2 and 4 topics, which could be expected both for the current interest of the researchers in mixing together different research areas and for the nature of the topics, that are not completely disjoint (some are specializations of others). Evaluated by the conference organizers, the result showed a 79% accuracy on average. As to the reviewers, even if taking $l = 6$ already ensured at least one topic for each of them, we adopted a more cautious approach and took $l = 10$, in order to balance the possible inaccuracy due to considering only the titles of their publications. The resulting accuracy was 65%.

Lastly, the topics automatically associated to papers and reviewers were fed to GRAPE in order to perform the associations, with the requirement to assign each paper to 2 reviewers. In order to have an insight on the quality of the results, in the following we present some interesting figures concerning GRAPE's outcome. In solving the problem, the system was able to complete its task in 120 seconds. GRAPE was always able to assign papers to reviewers by considering the topics only, except in two cases. In particular, except for those reviewers that explicitly asked to review less than 10 papers (*MaxReviewsPerReviewer* constraint), it assigned 10 papers to 40 reviewers, 9 to 2 reviewers, 8 to 3 reviewers, 7 and 6 to one reviewer. The experts considered the final associations made by GRAPE very helpful, since they would have changed just 7% of them.

4 Conclusions and Future Works

This paper proposed the application of intelligent techniques as a support to the various phases required for making automatic the task of paper-reviewer assignment in a scientific conference management. Experiments on a real domain prove the viability of the proposed approach.

Different future work directions are planned for the proposed system. First, the conference management system will be extended to cover other knowledge-intensive tasks currently in charge of the organizers, such as final presentations partition and scheduling according to the paper subject. Second, the automatic processing of the bibliographic references of the papers and of the publications of the reviewers will be faced. Furthermore, we plan to process the reviewers home

page to discover all the information needed for their registrations in order to automatically fill in all the fields in the registration form (i.e., affiliation, research interests, etc.). Then, in a more general perspective, the proposed techniques will be applied to the problem of matching the documents in a digital library to the interests of the library users. The use of ontologies for improving matching effectiveness will be investigated as well.

References

1. Michael W. Berry, Susan T. Dumais, and Gavin W. O'Brien. Using linear algebra for intelligent information retrieval. *SIAM Rev.*, 37(4):573–595, 1995.
2. Thomas M. Breuel. Two geometric algorithms for layout analysis. In *Workshop on Document Analysis Systems*, 2002.
3. Scott C. Deerwester, Susan T. Dumais, Thomas K. Landauer, George W. Furnas, and Richard A. Harshman. Indexing by latent semantic analysis. *Journal of the American Society of Information Science*, 41(6):391–407, 1990.
4. Thomas G. Dietterich, Richard H. Lathrop, and Tomas Lozano-Perez. Solving the multiple instance problem with axis-parallel rectangles. *Artificial Intelligence*, 89(1-2):31–71, 1997.
5. Floriana Esposito, Stefano Ferilli, Teresa Maria Altomare Basile, and Nicola Di Mauro. Semantic-based access to digital document databases. In *Foundations of Intelligent Systems, 15th International Symposium (ISMIS 2005)*, volume 3488 of *Lecture Notes in Computer Science*, pages 373–381. Springer Verlag, 2005.
6. Floriana Esposito, Stefano Ferilli, Nicola Fanizzi, Teresa M.A. Basile, and Nicola Di Mauro. Incremental multistrategy learning for document processing. *Applied Artificial Intelligence: An Internationa Journal*, 17(8/9):859–883, September-October 2003.
7. Nicola Di Mauro, Teresa Maria Altomare Basile, and Stefano Ferilli. Grape: An expert review assignment component for scientific conference management systems. In *Innovations in Applied Artificial Intelligence: 18th International Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems (IEA/AIE 2005)*, volume 3533 of *Lecture Notes in Computer Science*, pages 789–798. Springer Verlag, 2005.
8. Oscar Nierstrasz. Identify the champion. In N. Harrison, B. Foote, and H. Rohnert, editors, *Pattern Languages of Program Design*, volume 4, pages 539–556. Addison Wesley, 2000.
9. Gavin W. O'Brien. Information management tools for updating an SVD-encoded indexing scheme. Technical Report UT-CS-94-258, University of Tennessee, 1994.
10. Martin F. Porter. An algorithm for suffix stripping. In J. S. Karen and P. Willet, editors, *Readings in information retrieval*, pages 313–316. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1997.