Personalized Recommendation and Explanation by using Keyphrases Automatically extracted from Scientific Literature

Dario De Nart, Felice Ferrara and Carlo Tasso
Artificial Intelligence Laboratory,
Department of Mathematics and Computer Science, University of Udine, Udine, Italy

Keywords: Adaptive Personalization, Scientific Paper Recommendation, Concept-based Recommendation, User Modelling.

Abstract: Recommender systems are commonly used for discovering potentially relevant papers in huge collections of scientific documents. In this paper we propose a concept-based recommender system where relevant concepts are automatically extracted from scientific resources in order to both model user interests and generate recommendations. Differently from other work in the literature, our concept-based recommender system does not depend on specific domain ontologies and, on the other hand, is based on an unsupervised, domain independent keyphrase extraction algorithm that identifies relevant concepts included in a scientific paper. This semantic-oriented approach allows the user to easily inspect and modify his user model and to effectively justify the proposed recommendations by showing the main concepts included in the suggested papers.

1 INTRODUCTION

Discovering relevant papers is an ordinary and time-consuming task for researchers since they need to stay tuned with the most relevant scientific advances. In order to support researchers, several systems (such as CiteseerX, Google Scholar, Research Gate, CiteUlike, and Mendeley) provide facilities and tools, such as recommender systems, in order to simplify the task of accessing the knowledge available in huge collections of scientific papers.

Recommender systems can support scientists by filtering information according to the personal interests of the researchers. Collaborative Filtering (CF) recommender systems, which filter resources according to the opinions of people, have been used to reach this goal. For example, in CiteUlike, two collaborative filtering mechanisms are exploited: (i) an item-based CF recommender system where the tags provided by the users are utilized for identifying the resources similar to those the *active user*[1] previously liked and (ii) a user-based recommender system where the resources liked to the users who share papers with the active user are recommended (Bogers and Van den Bosch, 2008). Content-based recommender systems can be used for identifying potentially relevant resources as well. These recommender systems represent each resource by means of a set of features (such as the metadata associated to the resources or other terms extracted from the papers) and the same set of features is also used for modelling the user interests. Since resources and papers are represented by means of the same set of features, the relevance of a paper for a researcher is computed by matching the user profile against the representation of the specific paper. Obviously, the precision of the recommendations strongly depends on the features exploited by the recommender.

In this work we propose to use more semantic features by automatically extracting the most relevant concepts from scientific papers. By using concepts as features, we built a concept-based recommender that suggests the papers related to the concepts of interest for the active user. More specifically, concepts are identified as keyphrases automatically extracted from the scientific paper. A keyphrase (KP) is a short phrase (typically constituted by up to three/four words) that indicates one of the main ideas/concepts included in a document. A keyphrase list is a short list of keyphrases that reflects the content of a single document, capturing the main topics discussed and providing a brief summary of its content. The proposed recommender system builds a user profile mainly by means of relevance feedback, i.e. by ex-

[1] In this paper we refer the user which is going to receive the recommendations as *active user*.

De Nart D., Tasso C. and Ferrara F..
Personalized Recommendation and Explanation by using Keyphrases Automatically extracted from Scientific Literature.
DOI: 10.5220/0004539000960103
In *Proceedings of the International Conference on Knowledge Discovery and Information Retrieval and the International Conference on Knowledge Management and Information Sharing* (KDIR-2013), pages 96-103
ISBN: 978-989-8565-75-4

ploiting the keyphrase lists extracted from the papers that are considered and explicitly stated as relevant by the active user. Then, in order to compute the relevance of a new article, the user profile is matched against the keyphrase list extracted from that article. The domain-independent keyphrase extraction avoids a manual classification of papers and it still identifies a significant set of concepts as we showed in (Ferrara and Tasso, 2013). The idea of using more semantic features is due to two main goals. First, our concept based recommender system can explain why the system recommended the documents by showing: (i) the keyphrases which are both in the user model and in the paper and (ii) other keyphrases found in the document which are not yet stored in the user model but can support the user in understanding/evaluating the new paper. The explanation of recommendations by means of keyphrases produces several benefits. First of all, the user satisfaction can be increased since explanations save his time: the user is not forced to read the entire document in order to catch the main contents of the paper. Second, the system allows the users to take a look to the main concepts stored in the user model. In this way, a user can explicitly evaluate his interest for the various concepts and can increase or decrease his interest level for specific concepts or even remove them from his profile. By allowing users to provide this new feedback the system can generate a more accurate user profile improving, in this way, the accuracy of the recommendation process and, consequently, the user satisfaction. In this paper we show that these two goals can be reached by providing, at the same time, accurate recommendations.

The paper is organized as follows: Section 2 reviews related work, a brief architectural overview of the system is presented in Section 3, the proposed recommendation method is described in Section 4, the evaluation performed so far is described in Section 5, and Section 6 concludes the paper.

2 RELATED WORK

Several works in the literature deal with the problem of finding relevant scientific literature, mostly from an Information Retrieval perspective, such as in (Bollacker et al., 2000), where CiteSeer is introduced. However there are several authors who have taken into account more personalization-based approaches to the problem, leading to the creation of recommender systems rather than search engines. Several examples analyze the textual contents of scientific papers in order to provide recommendations to researchers. Some of them take into account specific sections of the papers such as the bibliography which can be used to build, navigate, and, moreover, mine the citation graph (i.e. the directed graph in which each vertex represents an academic publication and each edge represents a citation from one publication to another) in order to generate the recommendations. For instance, the citation graph is browsed by the recommender system described in (Huynh et al., 2012), where a set of liked papers is used as seed for navigating the citation graph.

On the other hand, our work aims at extracting from the papers the main ideas and concepts in order to describe the user interests from a more semantic perspective. Similarly, the feedback of the users of social systems, such as CiteUlike and BibSonomy, has been also used for identifying the concepts of interests of researchers. The authors of (Jiang et al., 2012), for example, extract the tags provided by the users of CiteUlike for generating a dictionary which can be used for identifying relevant concepts in the abstracts of scientific publications. In (Ferrara and Tasso, 2011), the tags of the users of BibSonomy are instead exploited for discovering if the user may be interested in several distinct *Topics of Interest (ToI)*. In this case a clustering mechanism is utilized for joining together tags with similar meanings where the similarity depends on the number of times two tags have been applied to the same resource. Such tag clusters allow to organize papers into different collections, each one associated to a specific ToI for the single user. Only opinions of users interested in a specific ToI are then considered for computing recommendations. More specifically, resources labelled by tags which are evaluated as more similar to the tags associated to a ToI are considered more relevant than other resources, and resources bookmarked by users more similar to the active user are more relevant than others as well. The precision of these approaches depends on the active participation of the users whereas the content-based recommender system described in this paper is solely based on the automatic extraction of the main concepts from a scientific resource.

The textual content of scientific papers is also analyzed in a concept-based recommender system proposed in (Chandrasekaran et al., 2008), where authors and papers are modeled by trees of concepts: using the ACM Computing Classification System (CCS), the authors trained a vector space classifier in order to associate concepts of the CCS classifications to documents. The hierarchical organization of the CCS allows the system to represent user interests and documents by trees of concepts. A user profile and a paper representation are then compared by a tree edit-

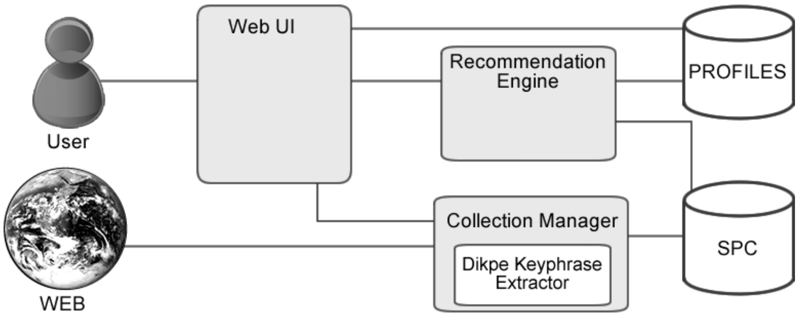


Figure 1: System Architecture Overview.

distance which computes a similarity measure among trees. Our approach, on the other hand, does not need a training phase and it also does not depend on specific ontologies for identifying relevant concepts (i.e. keyphrases constituted by n-grams) in the papers.

In (Govindaraju and Ramanathan, 2012), the authors propose a content-based filtering system based on a simple, unsupervised, keyphrase extraction technique to identify relevant concepts and entities. Such keyphrases are then organized, for each document, in a graph model, clustered, and matched against other KPs graphs in order to measure the degree of similarity between documents. However, their KP extraction technique does not take into account linguistic features (terms are extracted accordingly to their frequency in the document), keyphrases are considered as atomic entities and recommendation is based on the Jacquard similarity measure and metadata-driven criterias rather than on an actual comparison of the graph models.

3 SYSTEM OVERVIEW

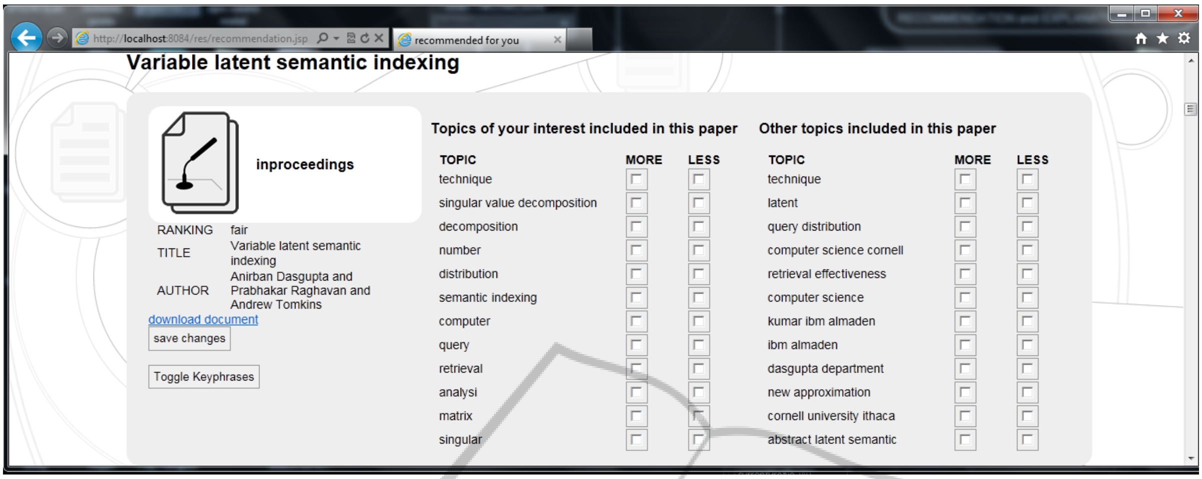
In order to support our claims and to test our approach we have developed a specific recommender system for scientific publications, named *Recommender and Explanation System* (*RES*), described in the following. The main goal of RES is providing personalized access to documents retrieved from CiteSeerX. The architecture of the system, showed in Figure 1, includes a database called *Scientific Paper Collection (SPC)*, a repository for user profiles and registry, and the following three main modules:

1) A *Web User Interface* devoted to (i) let the user create and manage profiles, (ii) specify one or more documents of interest, to be used as positive relevance feedback, either by browsing a list of articles within the SPC or uploading new ones, (iii) query CiteSeerX, and (iv) request recommendations. These are presented as a ranked list of documents where the top items are those that better match the user profile. For each document two lists of Keyphrases are shown: the first includes KPs representing concepts that actually match the user profile, the latter is constituted of relevant KPs extracted from the document but not matching the user profile. This information, shown in Figure 2, serves two goals: it briefly explains why a document was recommended by highlighting its main concepts and, secondly, offers the user a way to provide relevance feedback. Users can adjust the weight of each KP in their profiles by checking the “more” or the “less” checkbox.

2) A *Collection Manager Module*, devoted to: (i) execute queries on CiteSeer and crawl results, (ii) pre-process articles by extracting KPs from full text, and (iii) store their representations, as a list of KPs, into the SPC. This module has been developed using the Dikpe KP extraction algorithm described in (Ferrara et al., 2011), which has proven to perform significantly better than other known systems. The Dikpe KP extractor provides, as output, a list of KPs extracted from the document where each KP has a weight called *Keyphraseness* that summarizes the several linguistic and statistical indicators exploited in the extraction process. The higher the Keyphraseness, the more relevant is the KP in the document.

3) A *Recommendation Engine Module* devoted to: build and maintain individual user profiles; retrieve from the SPC the set of documents returned by a query, and then recommend the most promising papers.

The SPC is a crucial part of the system since Keyphrase Extraction, being an advanced Information Extraction task, takes time and processing a set of hundreds of query results cannot be done in an interactive way. In order to address this issue, we decided to let RES process retrieved documents only once, in an asynchronous way, and save their representation for later use. On the other hand, when the document KPs are known, the recommendation algorithm proposed is very efficient and it is able to rank large sets in a short time.



Variable latent semantic indexing

inproceedings

RANKING fair
TITLE Variable latent semantic indexing
AUTHOR Anirban Dasgupta and Prabhakar Raghavan and Andrew Tomkins

download document

save changes

Toggle Keyphrases

Topics of your interest included in this paper

TOPIC	MORE	LESS
technique	□	□
singular value decomposition	□	□
decomposition	□	□
number	□	□
distribution	□	□
semantic indexing	□	□
computer	□	□
query	□	□
retrieval	□	□
analysi	□	□
matrix	□	□
singular	□	□

Other topics included in this paper

TOPIC	MORE	LESS
technique	□	□
latent	□	□
query distribution	□	□
computer science cornell	□	□
retrieval effectiveness	□	□
computer science	□	□
kumar ibm almaden	□	□
ibm almaden	□	□
dasgupta department	□	□
new approximation	□	□
cornell university ithaca	□	□
abstract latent semantic	□	□

Figure 2: Recommendation screenshot.

4 PROPOSED METHOD

In the RES system, both user profile and document content are represented by a network structure called *Context Graph* (CG). For each document stored in the SPC, a CG is built by processing its weighted KP list. User profiles are represented by CGs built from a pool of KPs belonging to one or more SPC documents marked by the user as interesting and, possibly, enriched with other KPs gathered via relevance feedback, for example by providing a fragment of text or a specific paper not previously included in the SPC, or a specific list of KPs or keywords.

CGs are built by taking into account each single term belonging to each KP; each term is stemmed and then represented as a node of the graph; if two terms belong to the same KP, their corresponding nodes are connected by an arc. Both nodes and arcs are assigned a weight which is computed according to the Keyphraseness values associated to each KP containing the corresponding terms. In Figure 3 is shown the small CG formed by the KP list [information filtering, adaptive web personalization, adaptive filtering, content based filtering, social web, web usage, collaborative filtering].

As new KPs are added into the CG, either by direct article insertion or relevance feedback, both provided by the user, related concepts tend to link together, creating, in such a way, extensive networks of terms. Consider for example the profile CG shown in Figure 4, which has been built from four articles dealing with 'Content-based Recommender Systems' and 'Information Extraction' . On the other hand, unrelated concepts, form different, non-connected groups, as we can see in Figure 5 where two unrelated articles (the first dealing with Machine Learning, the second with Mechanical Engineering) are fed into a profile. If a user expresses multiple domains of interest in his profile, they will form different groups in the corresponding CG. This fact makes CGs expressive tools, able to model both short term and long term interests.

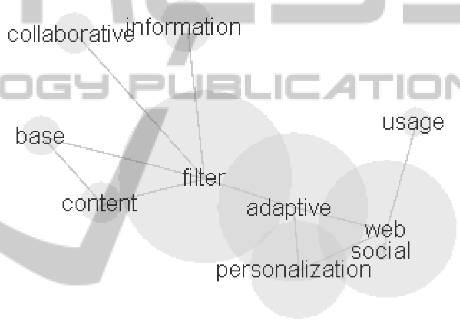


Figure 3: A simple Context Graph.

CGs allow to create, for each term, a meaningful context of interest by simply checking its adjacency list. If, in two different documents, the same term is used in similar contexts (i.e. in the two respective CGs the same nodes are connected in the same or similar way), it reasonably refers to the same concept, proving a certain degree of similarity between the two items. This mechanism also represents our solution to the problem of disambiguating polysemic terms.

When, as result of a user-specified query, a set of documents is retrieved from CiteSeer, RES extracts a list of KPs from each one of the retrieved articles, builds a CG for each KP list and generates a recommendation.

Recommendations are generated in three steps: Matching/Scoring, Ranking, and Presentation. In the

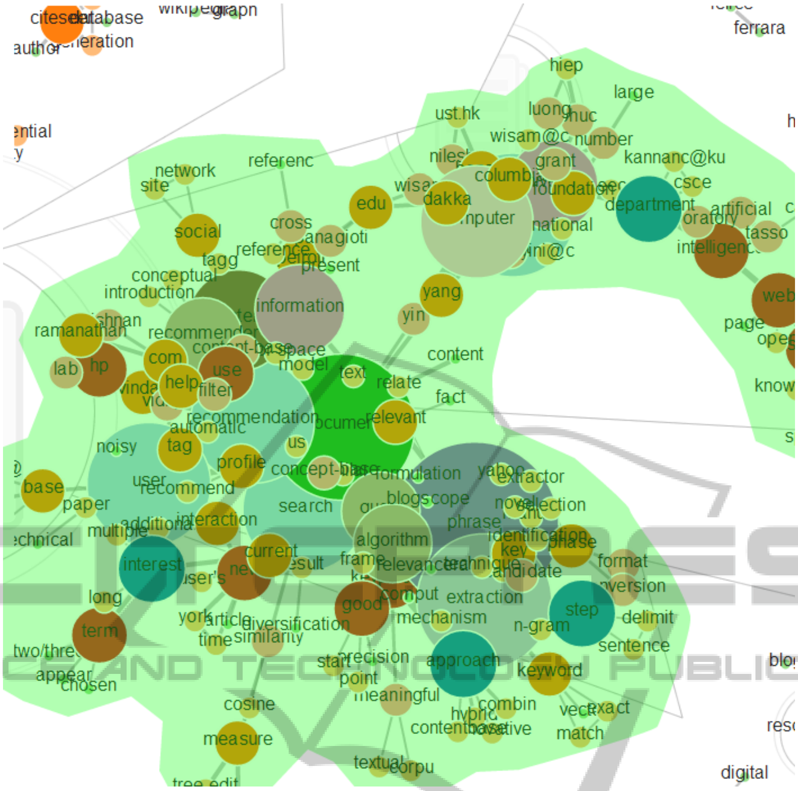


Figure 4: A CG built from 4 articles dealing with related topics.

first step every document (D) in the SPC is matched against the user profile (U) by calculating the following parameters: *Coverage (C)*, *Relevance (R)* and *Similarity (S)*.

C represents the percentage of concepts in D which are also of interest for the user, since they are already included in the profile U.

$$C(D,U) := \frac{sharedTerms(D,U)}{totalTerms(D)} \qquad (1)$$

By default, if less than 10% of the document nodes do not match those in the user profile, the document is not ranked, since there are not enough shared nodes for a meaningful evaluation of the other two parameters.

R estimates the importance of the concepts shared by the user profile (U) and the document (D). It is computed as the average tf-idf measure of the terms corresponding to the shared nodes between the user and the document CG with reference to the retrieved document set.

$$R(D,U) := \frac{\sum_{i \in terms(D) \bigcap terms(U))} tf\text{-}idf(i,D)}{sharedTerms(D,U)} \qquad (2)$$

Finally, S is intended to assess the local overlap between the two CGs and to measure how relevant are the shared arcs, i.e. determine how similar are the contexts in which shared terms are used, the stronger the shared association, the higher the score. S is computed by considering the sub-graph () constituted by shared nodes of the user CG; the parameter is evaluated as the sum of the weights (w) of the arcs in ΠU (E(ΠU)) which are also included in D (indicated as E(D)) divided by the overall weight of the arcs in ΠU.

$$S(D,U) := \begin{cases} 0 & \text{if } E(\Pi U) = \varnothing \\ \dfrac{\sum_{i \in E(\Pi U) \bigcap E(D)} w(i)}{\sum_{j \in E(\Pi U)} w(j)} & \text{otherwise} \end{cases} \qquad (3)$$

S varies between 0 and 1 In this way, each document is considered a point in a 3-dimensional space where each dimension corresponds to one of the three above parameters. In the Ranking phase, the 3-dimensional space is subdivided into several subspaces according to the value ranges of the three parameters, identifying in such a way different regions in terms of potential interest for the user. High values for all three parameters identify an excellent potential interest, while values lower than specific thresholds decrease the potential interest. Five subspaces are identified from *excellent* to *not recommended*, as shown in Figure 6, and each document is ranked according to where its three-dimensional representation is located. In the current experimental prototype, the interest threshold for each

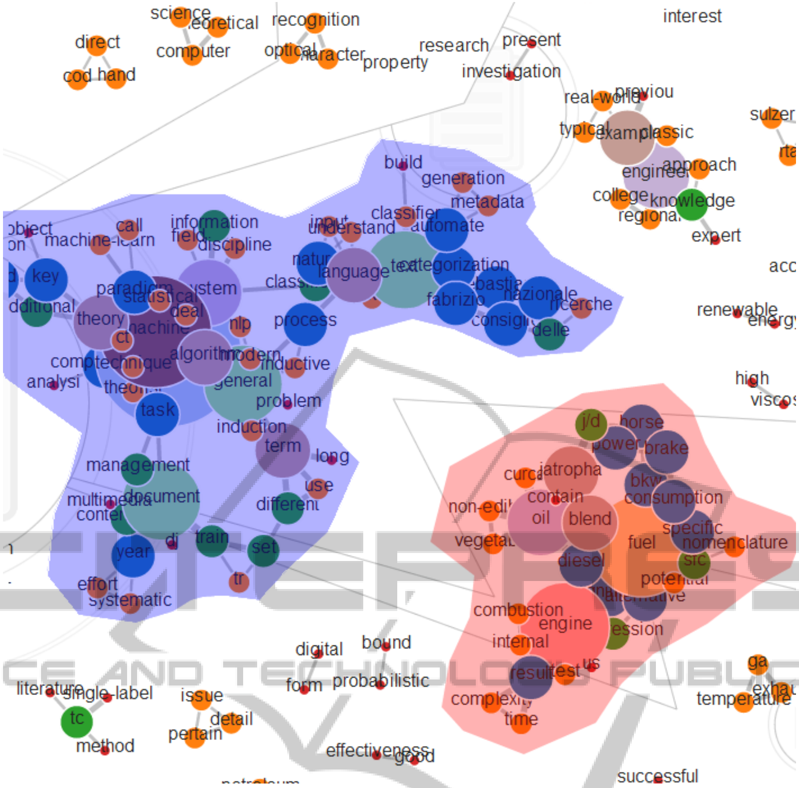


Figure 5: A CG built from two articles dealing with non-related topics.

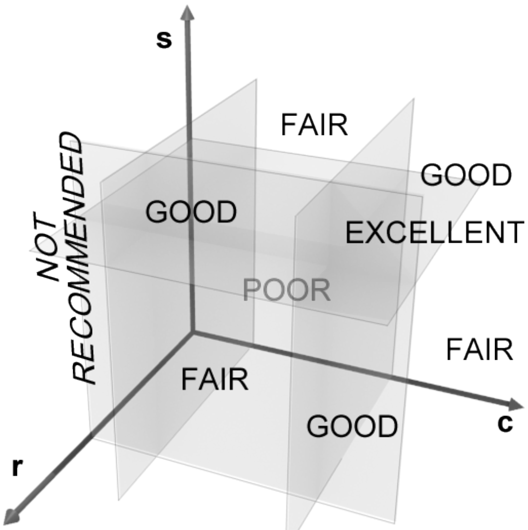


Figure 6: The five sub-spaces according to whom items are ranked.

parameter can be adjusted at runtime, for fine tuning the matching algorithm. Finally, in the Presentation step, documents are sorted by descending ranking order and the top ones are suggested to the user; documents not ranked are put at the very bottom of the list. As shown in Figure 2, both matching and not matching KPs are shown and the user can provide relevance feedback for fine adjustments of his profile and inclusion of serendipitous concepts indicated by not matching KPs.

5 EVALUATION

In the first development stage of the system, we have performed a limited number of offline formative tests, mainly aimed at experimenting different system tunings. A set of over 300 scientific papers dealing with Recommender Systems and Adaptive Personalization was collected and classified by users, identifying 16 sub-topics. Later, 200 uncategorized documents dealing with several random ICT topics were added in order to create noise in the data set and the whole set was processed and stored in a test SPC. 250 user profiles were automatically generated for each one of the 16 topics using groups of 2, 4, 6, 10, and 15 relevant seed documents respectively; then, for each user profile, RES and a baseline reference system (ad-hoc developed), based upon the well-known and established tf-idf metric, recommended ten items from the whole SPC. The baseline system produced its recommendations according to keyword vector models of

Table 1: Average accuracy results of comparative testing.

Seed documents	TF-IDF system	RES
2	0,42	0,57
4	0,53	0,66
6	0,55	0,70
10	0,60	0,72
15	0,60	0,72

both user interests and document contents, where keywords were extracted from texts according to their tf-idf and recommendation was evaluated upon the number of shared terms between the user and the document vector and their average weight (again, tf-idf). For each recommendation test run, every recommended item dealing with the same topic as the seed document was considered a good recommendation. We have defined the *accuracy* as the average part of good recommendations over the total number of recommended items. Results gathered so far are very promising since RES significantly outperforms the baseline mechanism for any given number of seed documents, as shown in Table 1.

In particular, the first evaluation of RES highlights how the proposed method is able to discriminate among similar domains with very fine granularity. For example, in the test SPC we included a small set of documents dealing with ’segment injection attacks’ [2] together with several others dealing with various kinds of ’attack’ to commercial recommender systems, such as ’random, average and bandwagon attacks’. When the two systems exploited in the evaluation phase were asked to recommend items similar to a limited number of articles extracted from that subset, the average RES accuracy was 0.59 while the average baseline accuracy was 0.15; Figure 7 shows the average accuracy results for this test. Such good results in this scenario may be a direct consequence of the high polisemy of terms such as ’attack’ and ’segment’, which RES handles by taking into account a significant and non-trivial context for each one of them.

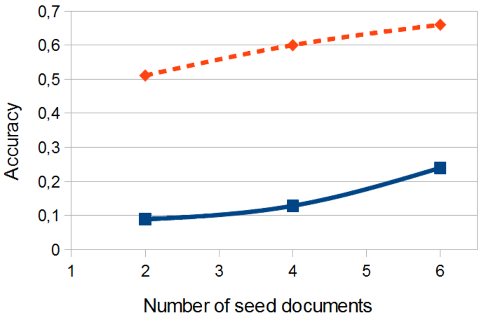


Figure 7: Average accuracy of RES (dotted) and the baseline tf-idf system (solid) in the domain of ’segment injection attacks’.

Evaluation is ongoing and in the future it will address the quality and the impact of the produced explanations on user satisfaction.

6 CONCLUSIONS

Recommender systems can greatly facilitate the task of searching for scientific literature, however, by just filtering collection of papers, state-of-the-art recommender systems still leave a heavy work to researchers who have to spend efforts and time for accessing the knowledge contained in scientific publications. In this paper we present a more semantic approach to the problem, aimed at the creation of a user model that is both based on actual concepts of interest and understandable. The presented RES system is still a testbed and evaluation is ongoing, but results gathered so far are encouraging, proving that our concept-based, human understandable approach is able to generate accurate recommendations. Future work will be aimed at expanding our concept-based strategy by means of ontologies and, eventually, folksonomies, exploiting different sources of knowledge in order to identify synonymous terms and phrases, suggest to the users new concepts related to the ones he considers interesting, and overcome the limitations of a pure content-based approach. Finally, we will also address the possible advantages of utilizing our ideas in other scenarios such as news, patents or legal documents recommendation.

REFERENCES

Bogers, T. and Van den Bosch, A. (2008). Recommending scientific articles using citeulike. In *Proceedings of the 2008 ACM conference on Recommender systems*, pages 287–290, New York, NY, USA. ACM.

Bollacker, K. D., Lawrence, S., and Giles, C. L. (2000). Discovering relevant scientific literature on the web. *Intelligent Systems and their Applications, IEEE*, 15(2):42–47.

Chandrasekaran, K., Gauch, S., Lakkaraju, P., and Luong, H. P. (2008). Concept-based document recommendations for citeseer authors. In *Proceedings of the 5th international conference on Adaptive Hypermedia and Adaptive Web-Based Systems*, AH ’08, pages 83–92, Berlin, Heidelberg. Springer-Verlag.

Ferrara, F., Pudota, N., and Tasso, C. (2011). A keyphrase-based paper recommender system. In Agosti, M., Es-

[2] A particular kind of profile injection attack to collaborative recommenders, exploiting statistical market analysis to alter recommendations.

posito, F., Meghini, C., and Orio, N., editors, *Digital Libraries and Archives*, volume 249 of *Communications in Computer and Information Science*, pages 14–25. Springer Berlin Heidelberg.

Ferrara, F. and Tasso, C. (2011). Extracting and exploiting topics of interests from social tagging systems. *Adaptive and Intelligent Systems*, pages 285–296.

Ferrara, F. and Tasso, C. (2013). Extracting keyphrases from web pages. In Agosti, M., Esposito, F., Ferilli, S., and Ferro, N., editors, *Digital Libraries and Archives*, volume 354 of *Communications in Computer and Information Science*, pages 93–104. Springer Berlin Heidelberg.

Govindaraju, V. and Ramanathan, K. (2012). Similar document search and recommendation. *Journal of Emerging Technologies in Web Intelligence*, 4(1):84–93.

Huynh, T., Hoang, K., Do, L., Tran, H., Luong, H. P., and Gauch, S. (2012). Scientific publication recommendations based on collaborative citation networks. In Smari, W. W. and Fox, G. C., editors, *CTS*, pages 316–321. IEEE.

Jiang, Y., Jia, A., Feng, Y., and Zhao, D. (2012). Recommending academic papers via users' reading purposes. In *Proceedings of the sixth ACM conference on Recommender systems*, RecSys '12, pages 241–244, New York, NY, USA. ACM.