IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received July 13, 2018, accepted August 22, 2018, date of publication August 29, 2018, date of current version September 21, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2867731

Artificial Intelligence Scientific Documentation Dataset for Recommender Systems

FERNANDO ORTEGA[1], JESÚS BOBADILLA[2], ABRAHAM GUTIÉRREZ[2], REMIGIO HURTADO[2,3], AND XIN LI[4]

[1]U-tad: Centro Universitario de Tecnología y Arte Digital, Calle Playa de Liencres, 28290 Las Rozas de Madrid, Spain
[2]Universidad Politécnica de Madrid, Computer Science, 28031 Madrid, Spain
[3]Universidad Politécnica Salesiana, Computer Science, Cuenca 010102, Ecuador
[4]Beijing Institute of Technology, Beijing 100081, China

Corresponding author: Jesús Bobadilla (jesus.bobadilla@upm.es)

ABSTRACT The existing scientific documentation-based recommender systems focus on exploiting the citations and references information included in each research paper and also the lists of co-authors. In this way, it can be addressed the recommendation of related papers and even related authors. The approach we propose is original because instead of using each paper citations and co-authors, we relate each of the papers with their main research topics. This approach provides a semantic level superior to that currently used, which allows us to obtain useful results. We can use collaborative filtering recommender systems to recommend research topics related to each paper and also to recommend papers related to each research topic. In order to face this innovative proposal, we have solved a series of challenges that allow us to offer various resources and results in the paper. Our main contributions are: 1) making a data mining of scientific documentation; 2) creating and publishing an open database containing the data mining results; 3) extracting the research topics from the available scientific documentation; 4) creating and publishing a recommender system data set obtained from the database and the research topics; 5) testing the data set through a complete set of collaborative filtering methods and quality measures; and 6) selecting and showing the best methods and results, obtained using the open data set, in the context of scientific documentation recommendations. Results of the paper show the suitability of the provided data set in collaborative filtering processes, as well as the superiority of the model-based methods to face scientific documentation recommendations.

INDEX TERMS Dataset, scientific documentation, recommender systems, machine learning, data mining, artificial intelligence, Scopus, topics.

I. INTRODUCTION

Scientific development in the field of Recommender Systems often requires public datasets in which testing new methods and algorithms. A very significant example of this situation is the strong impulse that the public dataset *MovieLens* had in the Recommender Systems (RS) research field [1]. *MovieLens* [2] is run by *GroupLens*, a research lab at the University of Minnesota. Several datasets have emerged that also contain the preferences of users about movies: *FilmTrust* [3] and *Netflix* [4]. Additionally, there are public datasets focused on other topics, such as books: *BookCrossing* [5], jokes: *Jester* [6] or music: *LastFM* [7].

The existence of diverse and public datasets makes it possible the design and improvement of suitable Artificial Intelligence (AI) methods and algorithms, which can be tested in these datasets. Researchers have, in this way, the possibility of comparing the results of their proposed methods with the results of the most advanced and modern existing baselines. Results are obtained by applying standard techniques of cross-validation and by using standard quality measures, which in the case of the RS usually are: prediction accuracy (*MAE, RMSE*), recommendation accuracy (*precision, recall, F1*) [8], rank recommendation (*ndcg*), performance, and beyond accuracy measures (*novelty, diversity, reliability, serendipity*) [9].

Public datasets not only greatly facilitate the design, development and testing of new generic Artificial Intelligence (AI) methods and algorithms; they also encourage research in the specific field of each dataset. As an example: e-commerce recommendation has evolved in a particular way thanks to specific RS approaches, whose methods and algorithms are not identical to those used in films recommendations, for example: films recommendation usually involves collaborative filtering approaches, whereas e-commerce recommendations usually are based on content-based and hybrid approaches. In this way, the Scientific Documentation Dataset [10], [11] that we provide should serve to promote research in AI applied to the Scientific Documentation (SD), and more specifically to the research in RS Machine Learning (ML) [12] methods. In this way, the main **objective** of this paper is to provide the necessary resources and experimentation background so that the SD field can be enriched by the advances currently experienced by the RS field, and beyond. The dataset that we provide will be called **SD4AI** (Scientific Documentation for Artificial Intelligence).

In this paper we do not restrict ourselves to provide and explain *SD4AI*; we also test it using a powerful RS own framework [13]. We show a large amount of relevant information about the quality of the results that our dataset provides: we offer quality results for prediction and recommendation, by applying a large amount of ML metrics, methods and algorithms. All these results provide the basis for researchers to use their own approaches, methodologies and architectures in search of an improvement of the SD prediction and recommendation results.

There is a variety of RS methods; all these methods are based on filtering the data to obtain the results. The most intuitive method is *content-based filtering* [14], where recommendations are made based on the characteristics of the products or services consumed by the user; e.g: to recommend a historical movie to a user who bought some historical books. You can also perform *demographic-filtering* [15], where a user is recommended based on the preferences of other users who match her in demographic characteristics: sex, age, geographical situation, etc. *Social-filtering* [16] exploits the social information (followers, followed, likes) to recommend a user based on the preferences of her social environment. Currently, *context-based filtering* [17] has emerged and it uses information implicitly obtained: GPS coordinates, RFID signals, IoT data. The most important filtering approach in the RS is the *collaborative-filtering* (CF) [18], where each active user is recommended based on the preferences of her neighbors: users with similar preferences to the active user. In a dataset containing tens of thousands of users, it is very likely to find users with tastes very similar to you and who value products or services that you probably do not know. CF offers the best accuracy results and there are ML methods that efficiently address its processing. Finally, it is relevant to indicate that commercial RS usually focus on *hybrid* [14], [19] designs, taking several types of data filtering and performing an aggregation of results.

Most of the existing public CF RS datasets contain the explicit ratings with which each user has valued a set of items; e.g .: ⟨Bill, Avatar, 4⟩, ⟨Zoe, The clone wars, 3⟩, etc. There are also datasets that contain the same information, but taken implicitly, for example songs listened to by each user [14]. These datasets usually contain millions of ratings; if we structure this information as ratings matrices (''users · items'') we will obtain the mathematical vision of the dataset and we will realize that the matrices are extraordinarily sparse: each user only has ratings about a very limited proportion of the available items. In addition to the ratings, some datasets incorporate demographic information (often incomplete), and a few ones incorporate social-information (often obtained from some social network).

Once mentioned the importance of the CF approach [20] and the usual content of the RS datasets, we are able to explain the originality of the dataset we offer in this paper (*SD4AI*): as far as we know, there is no public dataset containing the explicit preferences of the researchers about the published papers or the current research topics. We do not have this explicit information either. In this way, our approach is to fill in the dataset with implicit information. We will not collect information that can be filtered by using content-based filtering, since we know that this approach is not the one that provides the best accuracy results. Our goal is to create a dataset prepared to be used by applying CF techniques. Our information source comes from the main research publishers; they provide a huge amount of papers that contain, basically the following information: authors, affiliations, title, abstract, keywords, content and references. Our main objective is to create an SD dataset that contains ''ratings'' values and that can be processed using CF RS methods. In addition, *SD4AI* may contain additional information that will be used to feed demographic-filtering and social-filtering methods. Using all the information contained in *SD4AI*, the creation of a hybrid RS can be addressed.

To address the aforementioned objectives, we have run a data mining process using the information provided by *Scopus*. Specifically, we have collected papers from the most relevant JCR Q1 journals in the *Computer Sciences, Artificial Intelligence* area, during the years: 2016, 2017 and the first half of 2018; more than 14,000 papers, in total. All this information has been stored in a database that we provide openly, and that should not be confused with the dataset processed from this database. The fundamental fields of the database contain, for each paper, its title, authors, abstract, keywords, etc. In the following sections of this publication, the structure and contents of both: database and dataset are explained in detail. It is important to note that we have chosen the ''Artificial Intelligence'' area as an example for the SD dataset. It is possible to choose any other areas of knowledge and to provide datasets in those research fields.

While a typical CF RS dataset is structured using tuples: ⟨user, item, rating⟩, in *SD4AI* we change the semantics to convert it to: ⟨paper, topic, cardinality⟩. The most relevant is the choice of ''topics'' as the RS items [21], [22]. The meaning of topic is ''research topic'', and it refers to any of the current research sub-fields in the area covered by the dataset; in our case: Artificial Intelligence. As an example, AI topics are: Neural networks, fuzzy sets, learning systems, decision making, genetic algorithms, clustering, factorization, computational linguistics, set theory, recommender systems, image processing, data mining, etc. The most important work to obtain the dataset from the database is to extract the topics from the information contained in the set of papers. The main sources of information to extract the topics are: keywords and titles, but we also process abstracts. In the RS tuple ⟨paper, topic, cardinality⟩, the usual meaning of cardinality is the rating that the user has explicitly given to the item, or the number of times that, implicitly, the item has been consumed or used. In our case, its direct meaning is: number of times the topic appears in the paper, although the ''number of times'' is modulated according to the importance of the place where it appears; e.g.: if the topic appears in the title of the paper it is much more representative than if it appears only once in the paper abstract.

In a regular CF RS, e.g.: movies, the ratings dataset tells us the value that each user has given to each movie that she has watched and voted. The vast majority of ratings are ''empty'' (not voted), and it is precisely these not voted ratings that are the subject of rating predictions. The CF RS recommends a limited set of items (in our example, movies) that correspond to those that the RS has found as better (higher) predictions. In our *SD4AI* matrix, something very different happens: the ratings matrix has no ''empty'' elements; instead zeros appear, which correspond to the topics that do not belong to each paper. A zero in our SD dataset does not have the meaning of a zero in a regular CF RS dataset based on explicit ratings, where it would be interpreted as an item the user did not like. In a CF RS dataset based on implicit ratings, the zero would be interpreted with its real meaning: the user has not consumed the item, and therefore we do not know, a priori, whether she likes it or not.

Using *SD4AI*, a zero in a topic means that the paper does not contain this topic. As in the implicit ratings-based CF RS, we do not know if the authors or readers of the paper will be interested, or not, in the topic. This is a fundamental concept for our paper; let's see a couple of illustrative examples: 1) Two different papers use convolutional networks: the first one to recognize images, and the second one to recognize speakers. Although the fundamental topics of each paper are different, their common topic ''convolutional networks'' will probably lead to cross-recommendations, and 2) Researchers focused on the NLP (Natural Language Processing) area will be able to receive recommendations about ''Gene ontologies'', if they use *Word2Vec* [23] and the RS identifies it as a topic (*Word2Vec* is a powerful ML method both for NLP and for gene classification). In this way, just as a user will receive the recommendation of movies that she has not watched and that similar users like, an author or reader of a paper will receive the recommendation of topics in which she does not research, but that has a lot to do with papers similar to the one you have written or the one you are reading.

SD4AI, will therefore host, as CF information, the set of tuples ⟨paper, topic, cardinality⟩, where *paper* indicates each of the AI data-mined papers, *topic* refers to one of the AI research areas, and *cardinality* indicates the importance of the topic in the paper (in a limited range). All cases in which cardinality is zero, or they are below a threshold, will be considered as not issued ratings, and therefore their tuples will not appear in the dataset; it also happens with the ratings not issued in the CF datasets. In this way, we match the semantics and also the format of the existing datasets with the provided *SD4AI* dataset. This circumstance presents a very important advantage: The existing RS frameworks can directly process our dataset.

CF RSs face two problems of special importance: their *sparsity* [24] and the *cold-start problem* [25]. The sparsity refers to the huge percentage of absence of ratings: each user only has ratings in a very limited number of the available items. In our case, each paper only contains a very small number of the topics extracted in the whole research area. A high level of sparsity complicates CF processes, especially *memory-based* [26], [27] ones, where similarity measures are employed to find the neighbors of each active user. The *model-based* [28] methods first create a model from the dataset and then they make the recommendations based on the model. The *Matrix Factorization* [29], [30] methods obtain high levels of accuracy in very sparse datasets, such as *SD4AI*. The cold-start problem refers to situations where the active user, the active item or the dataset still do not contain enough ratings to be able to address an accurate recommendation process. This problem does not exist in our dataset, because we start from a situation in which there is a large number of ratings (cardinalities), and in which all papers and topics contain sufficient ratings to make recommendations.

In addition to the implicit ratings that the dataset houses, there is information about authors who publish together and also about related papers. Related papers could be obtained from each paper references. This type of relationship has a similarity with the information obtained in social networks, and *SD4AI* may be processed, too, using a social-filtering approach. Likewise, there is some information in the database (basically titles and abstracts) from which content-based filtering can be performed. Therefore, it is possible to create a hybrid [19] RS composed of the aggregation of three different filtering methods: collaborative, social and content-based. This first version of *SD4AI* only includes CF information.

RS are not only capable of making predictions and recommendations: they also process the data in a way that makes it easy to relate the information. From our *SD4AI* dataset we can obtain: 1) Related papers, 2) Related authors, 3) Authors related to a paper or papers related to an author (directly and indirectly), 4) Topics related to each other, 5) Topics related to a paper, 6) Papers related to a topic, 7) Trees and graphs showing the previous relationships [31], 8) Dynamic navigation systems through the trees [32], and 9) Clusters of papers and clusters of topics [33].

The rest of the paper is structured as follows: Section II abstracts the most relevant related work. Section III explains the necessary data-mining to obtain the database from Scopus, and the process to make the dataset from the database. Section IV explains the RS experiments design, shows the results and discusses them. Finally, section V contains conclusions and future works.

II. RELATED WORK

The existing research papers most related to our work are those that make use of citations and co-authors. There is a research field in which the co-authors, citations and references information contained in each paper is used: with this information a matrix is created that relates each paper with its main citations, and from this matrix CF is used to recommend research papers. This same approach can also be used by substituting papers by authors, in such a way that a matrix with related authors is created; in this case, what is recommended are suitable researchers to make collaborations. It is important to note that the vast majority of these papers focus on how to identify the most representative citations from the total set of citations of each paper. The CF approach of these papers is usually solved using the KNN algorithm and some statistical similarity measure, such as Pearson correlation, cosine or Jaccard. The approach we propose is original for several reasons: 1) We identify research topics, 2) We create matrices of ''papers · topics'', 3) We provide an open database and a public dataset that relates the papers with the topics, and 4) We carry out a wide set of CF RS experiments to find the CF methods that best suit the provided dataset. In this section we also show several representative papers covering the extraction of research topics: the methods they choose, the uses made with the topics and the used RS approaches.

There is a series of current works that address the CF recommendation of research papers. They take as information the citations of each of the papers; a representative example [34] uses a dataset containing data from 50 computer science researchers. Authors retrieve every reference and citation from these 50 researchers, obtaining one hundred thousand referenced papers. Finally, they create a matrix in which each row represents one of those hundred thousand research papers, and each column represents their referenced papers. Authors propose a simple memory-based method based on the Jaccard similarity measure. With a structure similar to that of the previous paper and a total of 186,000 papers, in [35] authors use classic memory-based methods (item-to-item and user-to-user), Bayesian classifier and a graph search, to test the quality of recommendations. In [35] they measure quality using the feedback of 120 users, and also using non-standard quality measures. One more example of recommendation of papers based on their citations is found in [36]; in this case, the authors utilize two datasets: the first one feeds from the 2003 KDD Cup, and the second one from the High Energy Physics dataset. Authors utilize the classic memory-based KNN method, making use of the similarity measure cosine and the recommendation quality measures: precision, recall & F1.

Using the same type of information as in the previous cases (papers and their citations), in [37] authors propose to structure relationships based on an ontological framework composed of: researcher ontology, references ontology and research study ontology. The proposed recommendation model is the application of five simple rules: Identity, non-negative, minimal distance, depth and transitivity. A hybrid approach to the recommendation of research papers is proposed in [38]. In this work, a working framework is provided: ''Scienstein''. As in the previous cases, the weight of the system is based on research papers and their references. The similarity measure of documents is based on the citation distance and the text frequency. Another paper based on co-author analysis is presented in [39], in order to facilitate the search of possible research collaborators. In this case, a co-authorship network is used and they formulate recommendations using graph weighted link predictions. In [40] a RS of research papers is offered, which is internally fed with the citations of each paper, to form a matrix composed of papers as rows and cited papers as columns. They make researcher profiles, distinguishing between junior and senior researchers. Later, they assign weights to the factors that form the profiles. Using an enriched database, [41] proposes to use a RS for finding collaborators with respect to research interests. The recommendation problem is formulated as a link prediction within the co-authorship network.

Following the analysis of the citations of each paper, in [42] a refinement is proposed in which it is tried to alleviate the sparsity problem of the generated matrices. The authors propose a pre-filtering, in which the ''potential citations papers'' are extracted. Additionally, they investigate which sections of papers contain the best references information. The same authors of the paper [42] extend their previous work by proposing an adaptive neighbor selection method [43]. They use the KNN method based on the Pearson correlation similarity measure. Their study claims that the most important fragments to obtain potential citations papers are: full text and conclusions. Reference [44] addresses the same research recommendation goal, but in this case the authors turn to a content-based approach: their framework applies content-based recommendation algorithms to rank the candidates.

An interesting and innovative research paper predicts scientific success based on co-authorship [45]; they study centrality in the co-authorship networks, differentiating between high cited and non-high cited authors. This paper predicts, with high accuracy, whether an article will be highly cited five years after publication. To measure research impact, [46] defines six indicators: degree, closeness, betweenness centrality, team exploration, publishing tenure, and prolific co-author count. It investigates how these indicators interact and affect citations for publications. With the aim of finding relevant papers, [47] incorporates various citation relations for a proper set of papers. They use both a metric and a model: the metric, called ''Local Relation Strength'', is defined to measure the dependency between cited and citing papers. The model, called ''Global Relation Strength'', is proposed to capture the relevance between two papers in the whole citation graph. Co-authorship networks are used for strategic research planning [48]; they generate valuable information relevant to the strategic planning, implementation and monitoring. A novel CF recommendation approach is proposed in [49]; they create a matrix ''users · factors'', and MF is used. This paper provides an interpretable latent structure for

TABLE 1. **Data mined artificial intelligence JCR journals. Years 2016, 2017 and the first half of 2018. Number of added papers to the database and impact factor of each journal.**

Journal	#papers	Impact factor
ACM Transactions on Intelligent Systems and Technology	157	3.196
Applied Soft Computing Journal	1608	3.541
Artificial Intelligence	202	4.797
Cognitive Computation	200	3.441
Data Mining and Knowledge Discovery	144	3.16
Decision Support Systems	286	3.222
Engineering Applications of Artificial Intelligence	523	2.894
Evolutionary Computation	53	3.826
Expert Systems with Applications	1623	3.928
IEEE Computational Intelligence Magazine	10	6.343
IEEE Transactions on Affective Computing	159	3.149
IEEE Transactions on Cybernetics	936	7.384
IEEE Transactions on Evolutionary Computation	131	10.629
IEEE Transactions on Fuzzy Systems	506	7.671
IEEE Transactions on Image Processing	1097	4.828
IEEE Transactions on Knowledge and Data Engineering	499	3.438
IEEE Transactions on Neural Networks and Learning Systems	727	6.108
IEEE Transactions on Pattern Analysis and Machine Intelligence	507	8.329
Information Fusion	161	5.667
Integrated Computer-Aided Engineering	54	5.264
International Journal of Computer Vision	174	8.222
International Journal of Intelligent Systems	34	2.929
International Journal of Neural Systems	95	6.333
Journal of Intelligent Manufacturing	400	3.035
Knowledge-Based Systems	726	4.529
Medical Image Analysis	209	4.188
Neural Networks	200	5.287
Neurocomputing	1860	3.317
Pattern Recognition	678	4.582
Semantic Web	44	2.889
Swarm and Evolutionary Computation	125	3.893
Swarm Intelligence	18	3.115
Total	14143	

users and items, and it can form recommendations about both existing and newly published articles. A graph where nodes are scientists is provided in [50]; scientists are connected if they have co-authored a paper. Authors are not used to recommend papers, but they offer valuable information about collaboration patterns, such as: the numbers of papers each author writes, what the typical distance between scientists is through the network, and how patterns of collaboration vary between subjects and over time.

A news topic RS is provided using the Bing news dataset [51]. They make experiments processing both content and collaborative-filtering. In [52] authors use keywords as topics, from a movies database, making content-based filtering. They compare LDA versus LSA to handle the topics model: LSA has been revealed to be better than LDA. A research paper recommendation with topic analysis is proposed in [53]. They use LDA to extract topics in a tiny set of 122 research papers. They do not provide quality results. A topic-centre RS [54] exploits the latent author-topic and author-author relationships; this paper uses the DBLP dataset and it processes the KNN algorithm, based on the Jaccard similarity measure. They only test the recall quality measure. A topic RS [55] has been structured in three steps: 1) Creating graphs from tags, 2) Extracting topics from graphs, and 3) Recommending and visualizing topics. This paper is focused in topic extraction, and it does not use CF to make recommendations. Collaborative topic regression is used to utilize items content (attributes) as auxiliary information [56]; this is not a topic extraction from the research papers, instead it can be considered as a tag information. A probabilistic topic model is used to analyze papers' latent topics [57]; authors create a paper cooperation network and they utilize LDA to relate topics. In [58] authors obtain topics of projects/experts based on LDA model, and then they use the topics to feed a CF RS algorithm. A probabilistic approach to extract topics is provided in [59]; authors apply the methodology to a corpus of 160,000 abstracts and 85,000 authors from the CiteSeer digital library. Each author is represented by a probability distribution over topics, and each topic is represented as a probability distribution over words for that topic. No recommendation processing is made in this paper. To discover topics and topical phrases, [60] provides a topical n-grams model.

III. SCIENTIFIC DOCUMENTATION DATA MINING AND DATASET CONFIGURATION

This section is divided into three parts: III-A) Performed scientific documentation data mining and description of the obtained database, III-B) Extraction of topics algorithm, and III-C) Dataset structure, features and their main frequency distributions.

A. DATABASE

As mentioned in the introduction section, we have taken Scopus as data source. From *Journal Citation Reports* (JCR)

TABLE 2. Data mined information from each paper.

Data mined information	Database field type
TITLE	varchar(240)
AUTHOR	varchar(400)
NUM_CITATIONS	smallint(5)
JOURNAL_ID	int(11)
YEAR	varchar(6)
PAGES	varchar(20)
DOI	varchar(35)
URL	varchar(200)
ABSTRACT	varchar(4000)
INDEX_KEYWORDS	varchar(1000)
AUTHOR_KEYWORDS	varchar(1000)
REFERENCES_FIELD	text
COUNTRIES	varchar(500)
ADDRESS	varchar(400)
PUBLISHER	varchar(100)
LANGUAGE_ARTICLE	varchar(50)
DOCUMENT_TYPE	varchar(25)
SOURCE	varchar(25)

TABLE 3. Journals information.

Data mined information	Database field type
JOURNAL_ID	int(11)
JOURNAL	varchar(250)
IMPACT_LEVEL	decimal(10,5)
POSITION_RANK	int(11)

we have chosen the area: *Computer Sciences. Artificial Intelligence*. From this area, the first third of the first quartile (Q1) journals have been selected during the years: 2016, 2017 and the first half of 2018. Table 1 shows the data mined journals, the number of papers added to the database and the impact factor of each journal.

From each paper, the data mined information and its database field type is shown in Table 2. Paper contents have not been extracted for legal restrictions. Finally, a database table is used to hold each journal main information; Table 3 shows this information.

In order to promote research both in the recommender systems field and the scientific documentation field, and to make it possible the reproducibility of our experiments, we have made this database open available through the URL: rs.etsisi.upm.es/sd4ai/.

B. TOPICS EXTRACTION

From the data mined information we have created a database. Now, from the database we create a RS dataset: *SD4AI* (Scientific Documentation for Artificial Intelligence). As mentioned in the Introduction section, our dataset is formed by a large set of tuples: $\langle$*paper, topic, cardinality*$\rangle$. Determination of these tuples is not immediate because, first, we must obtain the representative set of topics in the area; in our case, the artificial intelligence research area. Subsequently, each *cardinality* value will represent, in its tuple, the importance of the *topic* in *the paper*. To extract topics from papers we establish the next algorithm and we explain it by using a real example. Note that to obtain tokens from a sentence we first take the words of the sentence, and later we filter the stop words and the words contained in our black list.

Real example of the algorithm to extract topics from papers:

Paper title: *"On the use of convolutional neural networks for robust classification of multiple fingerprint captures"*

Paper abstract: *"Fingerprint classification is one of the most common approaches to accelerate the identification in large databases of fingerprints. Fingerprints are grouped into disjoint classes, so that an input fingerprint is compared only with those belonging to the predicted class, reducing the penetration rate of the search. The classification procedure usually starts by the extraction of features from the fingerprint image, frequently based on visual characteristics. In this work, we propose an approach to fingerprint classification using convolutional neural networks, which avoid the necessity of an explicit feature extraction process by incorporating the image processing within the training of the classifier. Furthermore, such an approach is able to predict a class even for low-quality fingerprints that are rejected by commonly used algorithms, such as FingerCode. The study gives special importance to the robustness of the classification for different impressions of the same fingerprint, aiming to minimize the penetration in the database. In our experiments, convolutional neural networks yielded better accuracy and penetration rate than state-of-the-art classifiers based on explicit feature extraction. The tested networks also improved on the runtime, as a result of the joint optimization of both feature extraction and classification."*

Paper keywords: *convolutional neural networks* | *deep learning* | *deep neural networks* | *fingerprint classification*

Tokenization:
Title tokens: *use* | *convolutional* | *neural* | *fingerprint* | *captures*
Abstract tokens: *fingerprint* | *common* | *approaches* | *accelerate* | *large* | *databases* | *fingerprints* | *fingerprints* | *grouped* | *disjoint* | *classes* | *input* | *fingerprint* | *compared* | *belonging* | *predicted* | *reducing* | *penetration* | *rate* | *procedure* | *usually* | *starts* | *extraction* | *fingerprint* | *frequently* | *visual* | *characteristics* | *work* | *propose* | *approach* | *fingerprint* | *using* | *convolutional* | *neural* | *avoid* | *necessity* | *explicit* | *extraction* | *incorporating* | *processing* | *within* | *training* | *furthermore* | *approach* | *able* | *predict* | *even* | *fingerprints* | *rejected* | *commonly* | *used* | *fingercode* | *study* | *gives* | *special* | *importance* | *robustness* | *different* | *impressions* | *fingerprint* | *aiming* | *minimize* | *penetration* | *database* | *experiments* | *convolutional* | *neural* | *yielded* | *better* | *accuracy* | *penetration* | *rate* | *art* | *classifiers* | *explicit* | *extraction* | *tested* | *also* | *improved* | *runtime* | *result* | *joint* | *extraction* | *wiley* | *periodicals* | *inc*

Algorithm 1 To Extract Topics From Papers

```
create topic set $T$ with the keywords of each paper
for each paper $p$:
    get $title\_tokens$ from $title$
    get $abstract\_tokens$ from $abstract$
    for each topic $t$ in $T$:
        get $topic\_tokens$ from $t$
        if ($topic\_tokens\ size$ is 1):
            set $nt$ as the number of occurrences of the single topic in the $title\_tokens$
            set $na$ as the number of occurrences of the single topic in the $abstract\_tokens$
            if ($nt \geq 1$ or $na \geq 2$):
                increment cardinality of paper $p$ to topic $t$ with $(2.5 * nt + na)/2$
        else:
            for each $tt1$ in $topic\_tokens$:
                for each $tt2$ in $topic\_tokens$:
                    if ($tt1 = tt2$): continue
                        set $nt1$ as the number of occurrences of the $tt1$ in the $title\_tokens$
                        set $na1$ as the number of occurrences of the $tt1$ in the $abstract\_tokens$
                        set $nt2$ as the number of occurrences of the $tt2$ in the $title\_tokens$
                        set $na2$ as the number of occurrences of the $tt2$ in the $abstract\_tokens$
                        if (($nt1 \geq 1$ or $nt2 \geq 1$) and ($nt1 \geq 2$ or $nt2 \geq 2$))
                            increment cardinality of paper $p$ to topic $t$ with $2.5 * min(nt1, nt2) + min(na1, na2)$
    for each $keyword$ of paper $p$:
        increment cardinality of paper $p$ to topic $keyword$ with 5
remove each topic used less than 5 times
remove each topic whose maximum cardinality is 1.5
```

Cardinality:

Example: Given the topic “convolutional neural networks”, we tokenize it into two tokens: “convolutional | neural”. “networks” has been filtered because it is included in a blacklist of generic words. The number of occurrences of the token “convolutional” in the title (nt1) is 1. The number of occurrences of the token “convolutional” in the abstract (na1) is 2. The number of occurrences of the token “neural” in the title (nt2) is 1. The number of occurrences of the token “neural” in the abstract (na2) is 2. So, the cardinality of this topic for this paper is $(2.5 * min(1, 1) + min(2, 2) = 4.5$. Furthermore, “convolutional neural networks” is a keyword of the paper, so the cardinality is increased by 5. Final cardinality will be $4.5 + 5.0 = 9.5$.

C. DATASET

The created dataset *SD4AI* has the same format than the *Movielens* ones, except for the missing timestamp field; in this way, it can be read using the existing open RS frameworks. Its file structure is formed by tuples: *paperId, topicId, cardinality*. The lines within the dataset are ordered first by *paperId*, then, within paper, by *topicId*. *SD4AI* has been made open available through the URL: rs.etsisi.upm.es/sd4ai/.

TABLE 4. ***SD4AI* size and composition.**

# papers	14,143
# topics	18,502
# ratings	1,389,094
sparsity	99.47%
range	[1..160] step 0.25

Table 4 shows the size and composition of *SD4AI*. It is remarkable the high sparsity of this dataset: with a similar size to *Movielens-1M* it presents a sparsity level equal to the *Movielens-20M* dataset. As an example: *Movielens-1M* has the sparsity level 95.75%, *Movielens-10M*: 98.69% and *Movielens-20M*: 99.47%.

Below we present frequency distributions representative of the *SD4AI* dataset. Figure 1 shows the frequency distribution of the dataset ratings (cardinalities). As it can be seen, most of the cardinalities have a very low value (in the range [1..2]); they correspond with topics that have a low relation with the paper, but a existing relation, anyway. These topics can lead to novel recommendations. There is a second group of cardinalities (in the range [2..7]) that corresponds to topics that have an appreciable relationship with the paper; It is expected that these topics will offer diverse recommendations. Finally, cardinalities greater than 8 correspond to topics closely related to the paper; Usually, these topics will offer not novel recommendations, but with a high degree of accuracy.

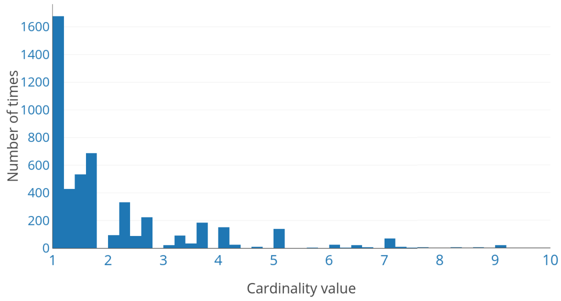

FIGURE 1. Frequency distribution of the dataset cardinalities. *X*-axis: most representative cardinality values from the *cardinality* range [1..160]. *Y*-axis: frequency of each cardinality (number of times this cardinality appears in the dataset).

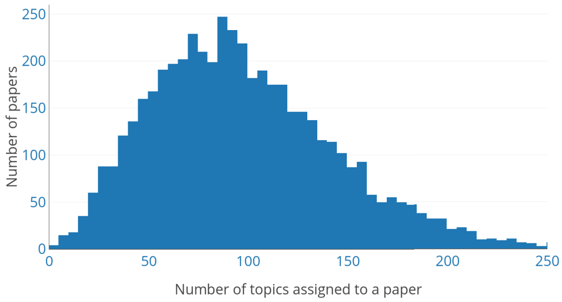

FIGURE 2. Frequency distribution of the papers with a fixed number of assigned topics. X-axis: number of assigned topics. Y-axis: frequency of papers. e.g.: there are approximately 250 papers containing from 85 to 90 topics each one of them.

The frequency distribution in Figure 2 answers the question: How many topics are associated with each paper? As we can see, this is a Gaussian distribution with a rough mean of 90 topics. As an example: there are approximately 150 papers with 45 to 50 assigned topics, and very few papers with 250 assigned topics. Results are compatible with a dataset that meets the following features: a) There is a sufficient number of topics assigned to each paper to relate papers with each other and to implement a CF RS that provides accurate recommendations, and b) The number of topics in each paper is low enough so that most of them are representative of the paper's content. In Figure 2, the main central area of the Gaussian distribution fulfils both features. The extreme area on the left of Figure 2 would not adequately fulfil feature a). The extreme area on the right of Figure 2 would not adequately fulfil feature b).

Figure 3 shows the frequency distribution of the topics: there is a large amount of topics that are specific to their papers; e.g.: there are more than 1000 topics that have been assigned to less than 10 papers from the thousands of papers in the dataset. On the other hand, there is a little quantity of "universal" topics that have been asigned to more than 200 papers (rigth side of Figure 3).

IV. COLLABORATIVE FILTERING RESULTS

The IV-A subsection explains the experiments design details: open framework used to test the dataset, cross-validation values, used recommendation methods and chosen quality measures. The IV-B subsection shows and explains the prediction and recommendation quality results obtained using each tested CF method. Finally, IV-C subsection compares quality results on diverse RS datasets, including the proposed one.

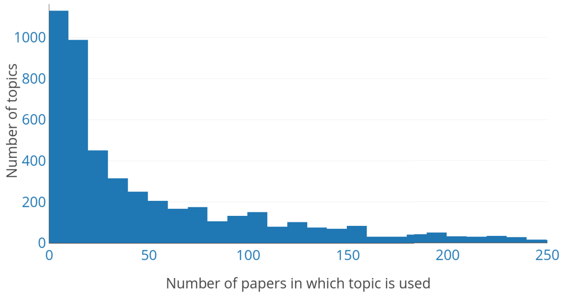

FIGURE 3. Frequency distribution of the topics contained in the papers. *X*-axis: number of papers. *Y*-axis: frequency of topics. e.g.: there are approximately 200 topics that have been assigned to more than 50 and less than 60 papers each one of them, but only a very small number of topics have been popular enough to be assigned to 250 papers.

TABLE 5. Recommender Systems libraries comparative.

Feature	CF4J	LibRec	Mahout
Extensivility	****	****	****
Performance	****	**	**
Efficiency	*	****	****
Abstraction	****	*	**
Flexibility	****	*	**
Scalability	*	*	****

A. EXPERIMENTS DESIGN

To run the experiments in this paper, we use our open source *Collaborative Filtering for Java (CF4J)* project [13]. This is a Java software stored in the *GitHub* repository, under *Apache License* version 2.0. *CF4J* provides an object-oriented framework to be used in CF RS research projects. The main contribution of this software is its flexibility: it makes easy to researchers accessing to all the stored data, to the intermediate results, such as the MF hidden factors, and to the prediction and recommendations final results. *CF4J* has been designed to simplify experiments implementation. It includes all the necessary objects to deal with the research trial and test processes. Researchers can run their experiments using the implemented cross-validation techniques and benefiting from the parallel *CF4J* execution threads. A comparative of the *CF4J* with other well known RS frameworks (*LibRec* and *Mahout*) is presented in Table 5.

The main goal of *CF4J* is to facilitate the reproducibility of experiments to the research community. Using this framework, researchers can: a) Load the provided RS datasets, b) Choose the CF methods and algorithms, c) Run the process, d) Obtain the prediction and recommendation results, e) Test the quality of the results, f) Extend the *CF4J* objects to incorporate their own methods and algorithms, and g) Compare their proposed algorithms with the provided baselines.

TABLE 6. Cross-validation values used in the experiments.

General parameters	
Testing-Items%	20%
Test-Users%	20%
Training-Items%	80%
Training-Users%	80%
#Neighbors	{50,..., 1000}, step 50
#Recommendations	{5,...,20}, step 1
Precision & Recall thresholds	Percentile 85: cardinality 3.75
Precision & Recall #Neighbors	350
PMF factorization parameters	
#Factors	16
#Iterations	150
Lambda	0.01

CF4J incorporates different CF datasets, it implements several singularity measures and matrix factorization methods, and a variety of quality measures are included. The details of the framework, including its architecture and representative examples are explained in [13].

In order to check the quality of the prediction and recommendation results of our dataset, we apply the *CF4J* framework to the *SD4AI* dataset. We select three types of baseline methods: a) Traditional statistic metrics, b) Current memory-based similarity measures, and c) The model-based *Probabilistic Matrix Factorization* (PMF) method. As traditional statistic methods we have chosen *Pearson correlation* (COR) and *cosine* (COS). As current memory-based similarity measures we have chosen PIP [61] and JMSD [26]. We test the predictions quality, making use of the quality measure *Mean Absolute Error* (MAE). We test the quality of the recommendations, making use of the quality measures *Precision*, *Recall* and *Normalized Discounted Cumulative Gain*. Table 6 shows the main parameters and values of the cross-validation experiments.

B. EXPERIMENTS RESULTS

This subsection shows the CF results, both for prediction and for recommendation. These results will tell us the quality of the proposed dataset for RS tasks. They will also show the best methods to make recommendations. Moreover, reaching good prediction results will allow us to tackle some other interesting goals such as papers and topics clustering, recommendation to groups of papers or to groups of topics, obtaining reliability, novelty, diversity and serendipity measures, recommendation explanation, etc.

Figure 4 shows the prediction accuracy obtained using the proposed dataset *SD4AI*. The *Mean Absolute Error* (MAE) has been used to return the overall error of all the possible predictions in the cross-validation process. The best (error) results are the smallest ones: the lowest in Figure 4. As explained before, the tested methods are: *Probabilistic Matrix Factorization* (PMF), JMSD, PIP, *Pearson correlation* (COR) and *cosine* (COS). As expected, the model-based PMF obtains a good result, since the dataset is very sparse and model-based methods perform comparatively better in these situations. The competitive Pearson (COR) result is surprising, because traditional statistical metrics work better in datasets with higher ratings density. As we will see later, this metric (COR) will not maintain this good behaviour when be used to obtain recommendations. PMF does not use the neighborhood approach, so its shape is an straight line.

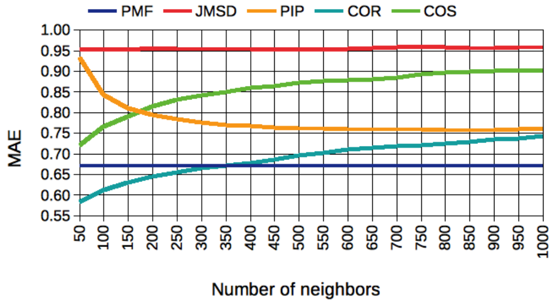

FIGURE 4. ***Mean Absolute Error*** **values obtained using the proposed** ***SD4AI*** **dataset.** ***X*****-Axis: number of neighbors of each KNN run.** ***Y*****-Axis: Prediction quality achieved; since we are facing an error measure, the best results are the lowest in the graph.**

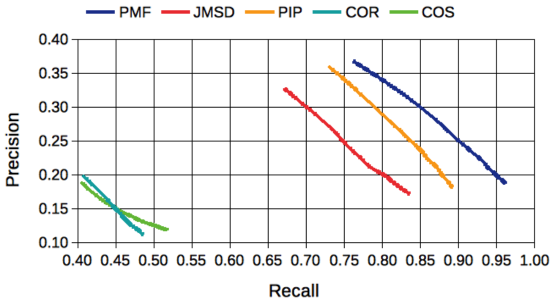

FIGURE 5. ***Precision*** **and** ***Recall*** **values obtained using the proposed** ***SD4AI*** **dataset.** ***X*****-Axis:** ***Recall*** **results.** ***Y*****-Axis:** ***Precision*** **results. Recommendation threshold: percentile 85 (rating 3.75). Best results are the highest in the graph: top-right corner.**

Recommendation accuracy (*precision & recall*) is shown in Figure 5. *Precision* is measured in the y-axis, while *recall* is measured in the x-axis. Here, the best results are the highest in the graph: top-right corner. As it can be seen in Table 6, we have settled the cardinalities percentile 85 to consider a recommendation as relevant. The range of the *precision & recall* results (numeric values in the y-axis and the x-axis) depend on this threshold: the higher the threshold, the lower the accuracy results. This means that the essential information of the graph is found in the relative positions of the results, and not in the absolute values obtained.

It is very interesting to check how the recommendation accuracy follows the expected behaviour: a) Traditional statistics metrics (COR and COS) obtain a low accuracy in this sparse dataset, b) Current CF methods (JMSD and PIP) obtain good accuracy results, because they handle more information than just the numerical values of the cardinalities, and c) The model-based PMF method gets the best recommendation accuracy results: the matrix factorization methods work with hidden semantics extracted in factors. These factors agglutinate all the dataset information and they

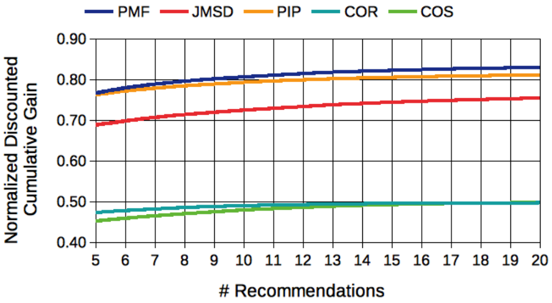

FIGURE 6. ***Normalized Discounted Cumulative Gain (NDCG)*** **results obtained using the proposed *SD4AI* dataset. *X*-Axis: Number of recommendations in the ranked list. *Y*-Axis: *NDCG* results. Recommendation threshold: percentile 85 (rating 3.75). Best results are the highest in the graph.**

behave especially well in heavy sparse environments, like ours.

Precision & Recall test the recommendations accuracy. That is, the proportion of hits in the set of recommendations made. Now, we are going to show a new recommendation quality study, where recommendations are no longer presented as a set, but as an ordered list. Ranked quality measures reward more the successes of the first recommendations of the list, and they also penalize more their errors. In this way, the classic *Normalized Discounted Cumulative Gain (NDCG)* measures de usefulness (gain) of recommendations based on their position in the list. Figure 6 shows the *NDCG* results when the tested methods are applied to the proposed *SD4AI* dataset. Results confirm the previous behaviour: model-based PMF gets the best results, followed by the memory-based current baselines: PIP and JMSD; finally, traditional metrics: *Correlation* and *Cosine* are not appropriate choices for this dataset.

C. RECOMMENDER SYSTEMS DATASETS COMPARATIVE

In this section we compare the quality results obtained when different RS datasets are tested. In this way, the proposed *SD4AI* dataset behaviour can be validated as CF dataset if its quality results values and trends are similar to those showed by the current public datasets that we have taken as baselines. This section is divided in three sub-sections: the first one compares prediction accuracy values, the second one tests recommendation accuracy values, and the third sub-section compares recommendation accuracy ranks.

1) PREDICTION ACCURACY COMPARATIVE

Traditional accuracy testing is adequate to compare different methods and metrics that are applied to the same dataset. In this way, looking at Figures 4 & 5 we can claim that PMF gets the best accuracy results when applied to *SD4AI* dataset. What about comparing accuracy on several datasets? Using *Mean Absolute Error* (MAE) we can not compare directly absolute results values when datasets have different ratings ranges, such as in this case, where *SD4AI* ranges from 1 to 160, while the other datasets range from 1 to 5. To normalize results we will introduce the variance of the ratings values: the smaller the variance of the ratings, the easier it will be to predict, and vice-versa. The chosen prediction accuracy quality measure is independent of the range and average of the ratings. We will call MAE_{RN} to the ratings range normalized prediction accuracy; the higher its value, the better the accuracy.

TABLE 7. **Ratings range normalized prediction accuracy.**

Dataset	Optimum #factors	$\mathrm{MAE}_{\mathrm{RN}}$
Filmtrust	4	0.248
Movielens-1M	6	0.438
Netflix	8	0.390
SD4AI	16	0.894

$$MAE_{RN} = 1 - \frac{MAE}{\sigma^2_{ratings}}$$

where $\sigma^2_{ratings} = 0 \Leftrightarrow prediction = ratings\ values$.

Table 7 shows the normalized prediction accuracy results on several RS datasets. We have selected the PMF model-based method to make this experiment. The optimum number of factors for each dataset have been calculated and, using these values, each MAE_{RN} has been processed. Table 7 results show that the comparative prediction quality of the proposed dataset (*SD4AI*) is higher than the traditional RS datasets ones.

2) COMPARATIVE OF RECOMMENDATION ACCURACY VALUES

In this sub-section we test *Precision & Recall* recommendation quality results when the model-based PMF method is applied to several datasets. We test the proposed *SD4AI* dataset and compare its results with the baseline datasets: *FilmTrust*, *Movielens 1M* and *Netflix*. Figure 7 shows that *SD4AI Precision* results are similar to the *FilmTrust* ones, and something worsts than the *Netflix* and *Movielens 1M* ones. *Recall* results show a better behaviour of the proposed dataset, compared to *Netflix* and *Movielens*, and a worst behaviour than the *FilmTrust* one. Overall, the proposed *SD4AI* dataset shows a similar behaviour to the CF RS chosen baselines, what validates its choice as CF dataset.

3) COMPARATIVE OF RECOMMENDATION ACCURACY RANKINGS

In the previous sub-section, the quality of the recommendations was shown according to their numerical values. In this sub-section, the quality of the recommendations is shown according to the correct or the erroneous order in which the recommendations are presented. The basic idea is that users perceive as more erroneous to fail in the first recommendation than to fail in the last one: we are more permissive with the failures of the last recommendations of a list than with the failures of the first recommendations of that list. To test the quality of the list (the ranking), the classic *Normalized Discounted Cumulative Gain (NDCG)* measures de usefulness

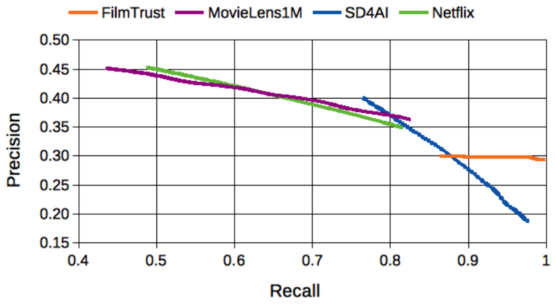

FIGURE 7. ***Precision* and *Recall* values obtained using the proposed *SD4AI* dataset and the baselines *FilmTrust*, *Movielens 1M* and *Netflix* datasets. Collaborative filtering method: PMF. *X*-Axis: *Recall* results. *Y*-Axis: *Precision* results. Best results are the highest in the graph: top-right corner. Recommendation thresholds: *FilmTrust*: 4.0, *MovieLens*: 5.0, *Netflix*: 5.0, *SD4AI*: 3.75.**

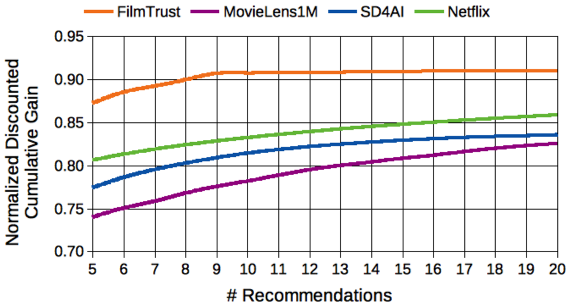

FIGURE 8. ***Normalized Discounted Cumulative Gain (NDCG)* values obtained using the proposed *SD4AI* dataset and the baselines *FilmTrust*, *Movielens 1M* and *Netflix* datasets. Collaborative filtering method: PMF. *X*-Axis: number of recommendations. *Y*-Axis: *NDCG* results. Best results are the highest in the graph.**

(gain) of recommendations based on their position in the list. Figure 8 shows the standard behaviour of the proposed *SD4AI* dataset: their ranking results are better than the *Movielens 1M* dataset ones, similar to the *Netflix* ones and something worse than the *FilmTrust* ones. Beyond the comparative details, the fundamental concept is the suitability of *SD4AI* as CF RS dataset.

V. CONCLUSIONS

The main contribution of this paper is to provide a collaborative filtering dataset containing scientific documentation. The size, structure, and frequency distributions of the dataset are compatible with the recommender systems processing. Moreover, it has been tested using representative prediction methods and suitable quality measures. Results show a usual collaborative filtering behaviour: they provide stable trends, appropriate results values and the expected behaviour of the recommendation methods put to the test. Making use of this dataset, we can make recommendations of topics from a paper, recommendations of papers from a topic, related papers, related topics, etc.

A special feature of the proposed dataset is its especially high degree of sparsity. Due to this, the matrix factorization methods are presented as the most appropriate to implement recommendation processes. In particular, we have verified that the Probabilistic Matrix Factorization method (PMF) obtains very accurate results. Current model-based similarity measures obtain results of reasonable quality, while traditional metrics are not suitable for use in this dataset.

In the usual collaborative filtering datasets, ratings are obtained by explicit votes cast by users, or by implicit ratings obtained through user interactions. In our dataset, instead of users we have research papers, instead of items we use research topics and instead of ratings we assign cardinalities obtained from the papers contents. The semantics of the dataset varies, but the recommendation process remains useful.

In addition to the provided dataset, we make public its scientific documentation source database and an open framework to process recommendations. In this way, researchers can create their own datasets, vary the selected topics and test different collaborative filtering methods by using several quality measures. Topics extraction is a sensitive process that can be customized to provide new topics; NLP methods can be designed to accomplish specific requirements in different fields of research.

The provided scientific documentation dataset and database open a wide range of future works: a) Testing new collaborative filtering methods, tailored to the special features of the provided dataset, b) Recommendation to authors: papers, topics and related researchers, c) Recommendation to groups: papers, authors and topics, d) Extraction of topics, e) Creation of new datasets from the open database, f) Scientific documentation analytics, g) Clustering of topics, papers and authors, h) Recommendations explanation, and i) Recommendation of novel, diverse or reliable papers or topics.

REFERENCES

[1] J. Bobadilla, F. Ortega, A. Hernando, and A. Gutiérrez, “Recommender systems survey,” *Knowl.-Based Syst.*, vol. 46, pp. 109–132, 2013.

[2] F. M. Harper and J. A. Konstan, “The MovieLens datasets: History and context,” in *Proc. ACM Trans. Interact. Intell. Syst.*, vol. 5, Dec. 2015, pp. 19:1–19:19.

[3] J. Golbeck and J. Hendler, “FilmTrust: Movie recommendations using trust in Web-based social networks,” in *Proc. 3rd IEEE Consum. Commun. Netw. Conf. (CCNC)*, vol. 1, Jan. 2006, pp. 282–286.

[4] J. Bennett and S. Lanning, “The Netflix prize,” in *Proc. KDD Cup Workshop*, New York, NY, USA, 2007, p. 35.

[5] Z. Zaier, R. Godin, and L. Faucher, “Evaluating recommender systems,” in *Proc. Int. Conf. Autom. Solutions Cross Media Content Multi-Channel Distrib.*, Nov. 2008, pp. 211–217.

[6] K. Goldberg, T. Roeder, D. Gupta, and C. Perkins, “Eigentaste: A constant time collaborative filtering algorithm,” *Inf. Retr.*, vol. 4, no. 2, pp. 133–151, 2001.

[7] Ò. Celma, *Music Recommendation and Discovery*. Berlin, Germany: Springer, 2010.

[8] G. Shani and A. Gunawardana, “Evaluating recommendation systems,” in *Recommender Systems Handbook*. Boston, MA, USA: Springer, 2011, pp. 257–297.

[9] H. Jungkyu and H. Yamana, “A survey on recommendation methods beyond accuracy,” *IEICE Trans. Inf. Syst.*, vol. E100-D, pp. 2931–2944, Dec. 2017.

[10] A. Karlsson *et al.*, “Modeling uncertainty in bibliometrics and information retrieval: An information fusion approach,” *Scientometrics*, vol. 102, pp. 2255–2274, Mar. 2015.

[11] J.-P. Lis-Gutiérrez, M. Gaitán-Angulo, P. V. Robayo, D. Aguilera-Hernandez, and A. Viloria, “Academic production patterns in public administration: An analysis based on scopus,” *J. Eng. Appl. Sci.*, vol. 12, no. 11, pp. 2904–2909, 2017.

[12] S. Angra and S. Ahuja, "Machine learning and its applications: A review," in *Proc. Int. Conf. Big Data Anal. Comput. Intell. (ICBDAC)*, Mar. 2017, pp. 57–60.

[13] F. Ortega, B. Zhu, J. Bobadilla, and A. Hernando, "CF4J: Collaborative filtering for Java," *Knowl.-Based Syst.*, vol. 152, pp. 94–99, Jul. 2018.

[14] M.-L. Wu, C.-H. Chang, and R.-Z. Liu, "Integrating content-based filtering with collaborative filtering using co-clustering with augmented matrices," *Expert Syst. Appl.*, vol. 41, no. 6, pp. 2754–2761, 2014.

[15] M. Y. H. Al-Shamri, "User profiling approaches for demographic recommender systems," *Know.-Based Syst.*, vol. 100, pp. 175–187, May 2016.

[16] J. Yu, M. Gao, W. Rong, Y. Song, and Q. Xiong, "A social recommender based on factorization and distance metric learning," *IEEE Access*, vol. 5, pp. 21557–21566, 2017.

[17] N. M. Villegas, C. Sánchez, J. Díaz-Cely, and G. Tamura, "Characterizing context-aware recommender systems: A systematic literature review," *Knowl.-Based Syst.*, vol. 140, pp. 173–200, Jan. 2018.

[18] Z. Yang, B. Wu, K. Zheng, X. Wang, and L. Lei, "A survey of collaborative filtering-based recommender systems for mobile Internet applications," *IEEE Access*, vol. 4, pp. 3273–3287, 2016.

[19] T. K. Paradarami, N. D. Bastian, and J. L. Wightman, "A hybrid recommender system using artificial neural networks," *Expert Syst. Appl.*, vol. 83, pp. 300–313, Oct. 2017.

[20] J. B. Schafer, D. Frankowski, J. Herlocker, and S. Sen, *Collaborative Filtering Recommender Systems*. Berlin, Germany: Springer, 2007, pp. 291–324.

[21] K. Lu, X. Cai, I. Ajiferuke, and D. Wolfram, "Vocabulary size and its effect on topic representation," *Inf. Process. Manage.*, vol. 53, pp. 653–665, May 2017.

[22] B. S. Khan and M. A. Niazi. (Aug. 2017). "Emerging topics in Internet technology: A complex networks approach." [Online]. Available: https://arxiv.org/abs/1708.00578

[23] E. Altszyler, M. Sigman, S. Ribeiro, and D. F. Slezak. (Oct. 2016). "Comparative study of LSA vs Word2vec embeddings in small corpora: A case study in dreams database." [Online]. Available: https://arxiv.org/abs/1610.01520

[24] J. Bobadilla and F. Serradilla, "The effect of sparsity on collaborative filtering metrics," in *Proc. 20th Australas. Conf. Australas. Database (ADC)*, vol. 92. Darlinghurst, NSW, Australia: Australian Computer Society, 2009, pp. 9–18.

[25] J. Bobadilla, F. Ortega, A. Hernando, and J. Bernal, "A collaborative filtering approach to mitigate the new user cold start problem," *Knowl.-Based Syst.*, vol. 26, pp. 225–238, Feb. 2012.

[26] J. Bobadilla, F. Serradilla, and J. Bernal, "A new collaborative filtering metric that improves the behavior of recommender systems," *Knowl.-Based Syst.*, vol. 23, no. 6, pp. 520–528, 2010.

[27] J. L. Sanchez, F. Serradilla, E. Martinez, and J. Bobadilla, "Choice of metrics used in collaborative filtering and their impact on recommender systems," in *Proc. 2nd IEEE Int. Conf. Digit. Ecosyst. Technol.*, Feb. 2008, pp. 432–436.

[28] D. Bokde, S. Girase, and D. Mukhopadhyay, "Matrix factorization model in collaborative filtering algorithms: A survey," *Procedia Comput. Sci.*, vol. 49, pp. 136–146, 2015, doi: 10.1016/j.procs.2015.04.237.

[29] X. Guan, C.-T. Li, and Y. Guan, "Matrix factorization with rating completion: An enhanced SVD model for collaborative filtering recommender systems," *IEEE Access*, vol. 5, pp. 27668–27678, 2017.

[30] A. Hernando, J. Bobadilla, and F. Ortega, "A non negative matrix factorization for collaborative filtering recommender systems based on a Bayesian probabilistic model," *Knowl.-Based Syst.*, vol. 97, pp. 188–202, Apr. 2016.

[31] A. Hernando, R. Moya, F. Ortega, and J. Bobadilla, "Hierarchical graph maps for visualization of collaborative recommender systems," *J. Inf. Sci.*, vol. 40, no. 1, pp. 97–106, 2014.

[32] A. Hernando, J. Bobadilla, F. Ortega, and A. Gutiérrez, "Method to interactively visualize and navigate related information," *Expert Syst. Appl.*, vol. 111, pp. 61–75, Nov. 2018.

[33] J. Bobadilla, R. Bojorque, A. H. Esteban, and R. Hurtado, "Recommender systems clustering using Bayesian non negative matrix factorization," *IEEE Access*, vol. 6, pp. 3549–3564, 2018.

[34] K. Haruna, M. A. Ismail, D. Damiasih, J. Sutopo, and T. Herawan, "A collaborative approach for research paper recommender system," *PLoS ONE*, vol. 12, p. e0184516, Oct. 2017.

[35] S. M. McNee *et al.*, "On the recommending of citations for research papers," in *Proc. ACM Conf. Comput. Supported Cooperat. Work (CSCW)*. New York, NY, USA: ACM, 2002, pp. 116–125.

[36] H. Liu, X. Kong, X. Bai, W. Wang, T. M. Bekele, and F. Xia, "Context-based collaborative filtering for citation recommendation," *IEEE Access*, vol. 3, pp. 1695–1703, 2015.

[37] K. Haruna and M. A. Ismail, "An ontological framework for research paper recommendation," *Int. J. Soft Comput.*, vol. 11, pp. 96–99, Apr. 2016.

[38] B. Gipp, J. Beel, and C. Hentschel, "Scienstein: A research paper recommender system," in *Proc. Int. Conf. Emerg. Trends Comput.*, 2009, pp. 309–315.

[39] I. Makarov, O. Bulanov, O. Gerasimova, N. Meshcheryakova, I. Karpov, and L. E. Zhukov, "Scientific matchmaker: Collaborator recommender system," in *Analysis of Images, Social Networks and Texts*, W. M. van der Aalst *et al.*, Eds. Cham, Switzerland: Springer, 2018, pp. 404–410.

[40] K. Sugiyama and M.-Y. Kan, "Scholarly paper recommendation via user's recent research interests," in *Proc. 10th Annu. Joint Conf. Digit. Libraries (JCDL)*. New York, NY, USA: ACM, 2010, pp. 29–38.

[41] I. Makarov, O. Bulanov, and L. E. Zhukov, "Co-author recommender system," in *Models, Algorithms, and Technologies for Network Analysis*. Springer, Jun. 2017, pp. 251–257. [Online]. Available: https://link.springer.com/book/10.1007%2F978-3-319-56829-4

[42] K. Sugiyama and M.-Y. Kan, "Exploiting potential citation papers in scholarly paper recommendation," in *Proc. 13th ACM/IEEE-CS Joint Conf. Digit. Libraries (JCDL)*. New York, NY, USA: ACM, Jul. 2013, pp. 153–162.

[43] K. Sugiyama and M.-Y. Kan, "A comprehensive evaluation of scholarly paper recommendation using potential citation papers," *Int. J. Digit. Libraries*, vol. 16, pp. 91–109, Jun. 2015.

[44] C. Nascimento, A. H. F. Laender, A. S. da Silva, and M. A. Gonçalves, "A source independent framework for research paper recommendation," in *Proc. 11th Annu. Int. ACM/IEEE Joint Conf. Digit. Libraries (JCDL)*. New York, NY, USA: ACM, Jun. 2011, pp. 297–306.

[45] E. Sarigöl, R. Pfitzner, I. Scholtes, A. Garas, and F. Schweitzer, "Predicting scientific success based on coauthorship networks," *EPJ Data Sci.*, vol. 3, p. 9, Sep. 2014.

[46] E. Y. Li, C. H. Liao, and H. R. Yen, "Co-authorship networks and research impact: A social capital perspective," *Res. Policy*, vol. 42, no. 9, pp. 1515–1530, 2013.

[47] Y. Liang, Q. Li, and T. Qian, "Finding relevant papers based on citation relations," in *Web-Age Information Management*, H. Wang, S. Li, S. Oyama, X. Hu, and T. Qian, Eds. Berlin, Germany: Springer, 2011, pp. 403–414.

[48] C. M. Morel, S. J. Serruya, G. O. Penna, and R. Guimarães, "Co-authorship network analysis: A powerful tool for strategic planning of research, development and capacity building programs on neglected diseases," *PLoS Neglected Tropical Diseases*, vol. 3, p. e501, Aug. 2009.

[49] C. Wang and D. M. Blei, "Collaborative topic modeling for recommending scientific articles," in *Proc. 17th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*. New York, NY, USA: ACM, 2011, pp. 448–456.

[50] M. E. J. Newman, "Coauthorship networks and patterns of scientific collaboration," *Proc. Nat. Acad. Sci. USA*, vol. 101, pp. 5200–5205, Apr. 2004.

[51] Z. Lu, Z. Dou, J. Lian, X. Xie, and Q. Yang, "Content-based collaborative filtering for news topic recommendation," in *Proc. 29th AAAI Conf. Artif. Intell. (AAAI)*. Menlo Park, CA, USA: AAAI Press, 2015, pp. 217–223.

[52] S. Bergamaschi and L. Po, "Comparing LDA and LSA topic models for content-based movie recommendation systems," in *Web Information Systems and Technologies*, V. Monfort and K. H. Krempels, Eds. Cham, Switzerland: Springer, 2015, pp. 247–263.

[53] C. Pan and W. Li, "Research paper recommendation with topic analysis," in *Proc. Int. Conf. Comput. Design Appl.*, vol. 4, Jun. 2010, pp. V4-264–V4-268.

[54] A. P. Singh, K. Shubhankar, and V. Pudi, "Topic-centric recommender systems for bibliographic datasets," in *Advanced Data Mining and Applications*, S. Zhou, S. Zhang, and G. Karypis, Eds. Berlin, Germany: Springer, 2012, pp. 689–700.

[55] Á. Bogárdi-Mészöly, A. Rövid, H. Ishikawa, S. Yokoyama, and Z. Vámossy, "Tag and topic recommendation systems," *Acta Polytechn. Hungarica*, vol. 10, no. 6, pp. 171–191, 2013.

[56] H. Wang and W. J. Li, "Relational collaborative topic regression for recommender systems," *IEEE Trans. Knowl. Data Eng.*, vol. 27, no. 5, pp. 1343–1355, May 2015.

[57] S. Gao, X. Li, Z. Yu, Y. Qin, and Y. Zhang, "Combining paper cooperative network and topic model for expert topic analysis and extraction," *Neurocomputing*, vol. 257, pp. 136–143, Sep. 2017.

[58] S. Gao, Z. Yu, L. Shi, X. Yan, and H. Song, "Review expert collaborative recommendation algorithm based on topic relationship," *IEEE/CAA J. Autom. Sinica*, vol. 2, no. 4, pp. 403–411, Oct. 2015.

[59] M. Steyvers, P. Smyth, M. Rosen-Zvi, and T. Griffiths, "Probabilistic author-topic models for information discovery," in *Proc. 10th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*. New York, NY, USA: ACM, 2004, pp. 306–315.

[60] X. Wang, A. McCallum, and X. Wei, "Topical N-grams: Phrase and topic discovery, with an application to information retrieval," in *Proc. 7th IEEE Int. Conf. Data Mining (ICDM)*, Oct. 2007, pp. 697–702.

[61] H. J. Ahn, "A new similarity measure for collaborative filtering to alleviate the new user cold-starting problem," *Inf. Sci.*, vol. 178, no. 1, pp. 37–51, Jan. 2008.

FERNANDO ORTEGA was born in 1988. He received the B.S. degree in software engineering and the Ph.D. degree in computer sciences from the Universidad Politécnica de Madrid, Madrid, in 2010 and 2015, respectively. He is currently an Assistant Professor with the Department of Engineering, U-tad: Centro Universitario de Tecnología y Arte Digital. His main research fields are machine learning, data analysis, and artificial intelligence. He has published several papers in the most relevant international journals. He also actively collaborates in various projects with technology companies.

JESÚS BOBADILLA received the B.S. degree in computer science from the Universidad Politécnica de Madrid and the Ph.D. degree in computer science from Universidad Carlos III. He is currently a Lecturer with the Department of Applied Intelligent Systems, Universidad Politécnica de Madrid. He is a habitual author of programming languages books working with McGraw-Hill, Ra-Ma, and Alfa Omega publishers. His research interests include information retrieval, recommender systems, and speech processing. He is in charge of the FilmAffinity.com research team working on the collaborative filtering kernel of the web site. He has been a Researcher with the International Computer Science Institute, Berkeley University, and the Head of the Research Group, The Sheffield University.

ABRAHAM GUTIÉRREZ was born in Madrid, Spain, in 1969. He received the B.S. and the Ph.D. degrees in computer science from the Universidad Politécnica de Madrid. He is currently a Lecturer with the Department of Applied Intelligent Systems, Universidad Politécnica de Madrid. He is a habitual author of programming languages books working with McGraw-Hill, Ra-Ma, and Alfa Omega publishers. He is in charge of this group innovation issues, including the commercial projects. His research interests include P-systems, social networks, and recommender systems.

REMIGIO HURTADO was born in 1989. He received the B.S. degree in systems engineering from the Universidad Politécnica Salesiana, Ecuador, in 2012, the master's degree in information and software technology from the Instituto Tecnológico y de Estudios Superiores de Monterrey, México, in 2014, and the master's degree in computer science and technology from the Universidad Politécnica de Madrid, Spain, in 2017. He is currently a Lecturer with the Universidad Politécnica Salesiana. His research interests include the recommender systems and natural language processing. He is currently a member of the Research Team, Artificial Intelligence, and Assistive Technology, his team is collaborating with Universidad Politécnica de Madrid.

XIN LI received the B.Sc. and M.Sc. degrees from Jilin University, China, and the Ph.D. degree from Hong Kong Baptist University. She was with Hong Kong Baptist University as a Post-Doctoral Teaching Fellow. She is currently an Associate Professor with the School of Computer Science, Beijing Institute of Technology. Her research interests include representation learning, deep reinforcement learning, with its applications to healthcare and robotics. She has published over 30 papers in refereed international journals and conference proceedings, including AAAI, IJCAI, ICML, KAIS, TKDE, and TOC. Her team received the Best Paper Award at IPCCC 2013. She also served/is serving as a PC for AI conferences, including CIKM, WSDM, and IJCAI. She has supervising multiple projects as the principle supervisor funded by the National Natural Science Foundation of China, National Program on Key Basic Research Foundation, Doctoral Programs Foundation, Ministry of Education, China, since 2011.