IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

SPECIAL SECTION ON TRUSTED COMPUTING

Received February 7, 2018, accepted April 1, 2018, date of publication April 6, 2018, date of current version April 25, 2018.

*Digital Object Identifier 10.1109/ACCESS.2018.2823720*

# Aggregating Author Profiles from Multiple Publisher Networks to Build a List of Potential Collaborators

**KARIM ALINANI[1], ANNADIL ALINANI[1], DUA HUSSAIN NAREJO[1], AND GUOJUN WANG[2], (Member, IEEE)**

[1]School of Information Science and Engineering, Central South University, Changsha 410083, China
[2]School of Computer Science and Educational Software, Guangzhou University, Guangzhou 510006, China

Corresponding author: Guojun Wang (csgjwang@gzhu.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61632009 and Grant 61472451, in part by the Guangdong Provincial Natural Science Foundation under Grant 2017A030308006 and by the High-Level Talents Program of Higher Education in Guangdong Province under Grant 2016ZJ01.

**ABSTRACT** Recommender systems have roots in numerous fields, and their use is widespread in the modern world. The scientific community is striving to enhance the quality of life by breaking innovative barriers and developing solutions that had never previously been considered. In an ideal world, an individual researcher would participate in various fields of research and make cumulative impactful contributions to benefit society. However, in reality, this goal is difficult to attain without a team of collaborators. Collaboration refers to the information of partnerships that bring uniquely talented researchers together around a common idea. However, efforts to seek such co-authors not only are challenging but also occasionally yield no significant results. In this paper, we propose a recommender system to aggregate author information from multiple publisher networks. It evaluates the trustworthiness of the author recommendations based on the impact of the authors' contributions and the recency and popularity of their work as well as the correlations among these factors. On this basis, the system generates a list of prospective collaborators who might be of interest to a given researcher.

**INDEX TERMS** Recommender system, correlation, co-author relationship, collaboration, co-author, trust.

## I. INTRODUCTION

Scientific growth drives the development of our society. It is fostered by intellectuals who selflessly devote their time and effort to enhancing the quality of life. Researchers work as individuals as well as in teams, where their various skill sets collectively contribute to remarkable achievements that have an impact on various aspects of life. Collaboration is one of the key components of substantial research contributions; however, successful collaborations often requires a great deal of effort to discover the best fit among a set of researchers with diverse expertise who can work together to produce a premium research contribution.

In this paper, we propose a recommender system to assist researchers in discovering a list of authors who not only have produced work pertinent to a certain field of interest but also have quality contributions under their belt.

An efficient system for a researcher who has just started, as the system would suggest him the top contributors of the field to read and understand their contributions as well as grasp the various techniques along with getting the idea of the line of research as well as the open problems to work. Furthermore, an equally efficient system for the existing researchers who have uplifted their research to the next level, as the system would point out the latest advancement in their field of interest. Therefore, an existing researcher could take benefit of the emerging advancement by utilizing the up to date knowledge to come up with a better solution to open problems.

The scientific contributions of this paper can be summarized as:

- In our previous paper, we proposed a recommender system [20] that could fetch essential details from the offline ACM dataset based on one or more specified keywords. These details were supplemented by calling the Scopus API. However, we observed that there was still a need to incorporate abundant additional

information about the candidate authors to improve the recommendation quality. Hence, we now propose the use of a far superior combination of information from both the offline and online ACM Digital Library (ACM DL) datasets to sketch a better picture of each author prior to further supplementing the author data by aggregating Scopus results.
- The results from the different publisher networks are aggregated to capture the merits of the authors and generate better recommendations. The enhancements consist of citations (to indicate the quality of each publication), the impact factor of the journal, the research areas of the author (for relevance), the ability use keywords to target a specific body of literature, and the ability to estimate the expertise of an author based on his contributions.

This paper is organized into several sections. Related work is discussed in section II. Our solution is proposed in section III. The details of the experimental design are elaborated upon in section IV, and section V presents the results and a discussion of our work. The conclusions of this study and plans for future work are described in section VI.

## II. PROBLEM STATEMENT & LITERATURE REVIEW

Gaining a comprehensive understanding of someone is a complex task, particularly when the relevant interaction involves a machine on one end and a human on the other. Machine-human interactions require different input and output devices, and the absence of true intelligence in machines remains a substantial challenge in the growth of the current technological era. Consequently, research on machine learning algorithms and techniques for enhancing the overall human experience with various systems is being conducted worldwide. Recommender systems are among the solutions built to address such issues. Recommender systems can function most effectively when provided with a user's usage history. Based on past usage, a profile is constructed, and the system utilizes these details to evaluate the best possible choices for the user. This evaluation considers various factors, such as age, location, gender, customs, the influence of friends and family, and many others. These recommender systems operate in an environment consisting of a vast array of other systems that are used in our everyday lives, in contrast to an educational or research environment.

In the context of scholastic recommender systems, many profile generation techniques have been proposed for capturing background information about scholars [4], including usage of pre-build domain ontologies for concept extraction and have further enhanced it using terminology builder. In this scenario, complete domain knowledge is not required, and such a system tends to identify items in which a user might be interested [5], [6]. A context-aware semantic recommender system architecture was proposed in [1] which solves magic barrier problem by efficiently dealing with incoherent items. Whereas Raamkumar *et al.* [2] proposed an academic recommender system that generates a preliminary reading list of scientific articles to help a person to become acquainted with a new field or to enhance his knowledge by getting relevant articles from offline ACM dataset. Tejeda-Lorente *et al.* [6], the authors proposed a recommender system that constructs a user profile based on publications submitted by the user and generated a list of recommendations based on the profile; the list is emailed to the user with feedback to evaluate the performance. A system that recommends a set of video lectures on various subjects based on the knowledge level of the user was proposed in [7], this system leverages fuzzy linguistic tool specifically for medical students. Various recommendation techniques for profile building have been proposed in [9], including the use of statistical information extraction, along with mediated profiles as well as approaches for knowledge unification. A user-centric visualization approach was used in [10], to generate suitable publications based on his interest while taking into account various factors including age, gender, track record to evaluate usability, effectiveness as well as trust and accuracy of recommendations. To cope up with the gigantic data growth performance of various collaborative filtering algorithms have been evaluated in [11] and reduction of time has been achieved by the author by implementing their proposed method. Amini *et al.* [8] analyzed web taxonomies to build background knowledge on scholars, for cross-validation they identified connections between various taxonomies matching their semantic concept. Bauer and Nanopoulos [12] have leveraged quantitative implicit feedback from the customer to propose a recommendation algorithm that handles distributional assumptions using matrix factorization. Various contextual information methods have been reviewed in [13] that are being used in digital libraries, they have categorized the recommendations into user, document and environment context. They have also proposed some suggestion to design content-aware recommender system for digital libraries effectively. MapReduce brute force algorithm has been introduced in [14] to enhance the collaborative filtering for libraries with a large number of users to avoid similarity computation problem.

Significant contributions to research rely on the dedication of experts in various fields combining the knowledge for the benefit of society. The significance of co-author relationships has been explored in many previous works such as [15], [16], and the integrity of such contributions has also been elaborated upon in [17]. Another important factor to consider is the trustworthiness of collaborators which comprises of various aspects such as popularity, reliability, and activeness as discussed in [26]. Hence, for a researcher, the task of finding a suitable collaborator for scientific work can be rather tedious.

In our recent publication [20], we proposed a system that fetches essential details from the offline ACM DL dataset based on specified keywords. It finds relevant publications, extracts author information and supplements this information using Scopus API to generate better results. Based on the retrieved information, it recommends to the user a list of

authors with whom he would be most likely to team up. The recommendations are generated on the basis of the authors' relevance to the field of interest and many other related factors. However, in the previously proposed system, many details were missing that could have contributed to better recommendations, as the ACM offline dataset has a limited corpus from 1982 to 2015 and was not updated on a regular basis, hence a ton of updated valuable information was unavailable. Therefore, to improve the quality of the results, we have enhanced the system by incorporating data from ACM DL online dataset, which is updated with various details that are not included offline dataset, before querying Scopus.

## III. IMPROVED SYSTEM

In the previous section, we discussed the existing recommender systems related to author correlations and relevant aspects of the present literature. Below, we list a few scenarios that remain to be addressed to potentially enhance the overall recommendation performance:

- *Scenario 1:* the credit that an author receives for his contribution to a multi-author publication may not be consistent with the extent to which he collaborated on the relevant work.
- *Scenario 2:* a researcher may have a very diverse background, and older background information might be irrelevant to the present situation, adding noise that can affect the recommendation quality.
- *Scenario 3:* different fields may use very similar keywords, which might result in unreliable recommendations.
- *Scenario 4:* talented researchers who are new to the field and do not yet have many contributions, or do not have a recognized profile, may be overlooked.

Based on the above scenarios, there is a gap that requires attention. Addressing these scenarios could assist scholars in obtaining improved recommendations for collaborators for their scientific contributions. To address scenario 1, we could credit the authors of a publication based on their naming order, assigning higher contribution levels to authors listed earlier. To address the 2nd scenario, the system should be sufficiently adaptable to recognize a change in a scholar's focus and discard older, unassociated information. Addressing the 3rd scenario would require the results based on not only the keywords entered but also the author's field of interest to prevent ambiguous results. Addressing scenario 4 would require considering a larger set of authors and comparing a greater number of features to avoid disregarding skilled scholars who are potentially new to scientific research.

This section focuses on our proposal for addressing the scenarios discussed above. To generate a list of prospective collaborators, the system proceeds through various stages, as visualized in Fig. 1, based on the field of interest and the specified keywords. In the following subsections, we will discuss each of these stages in detail.

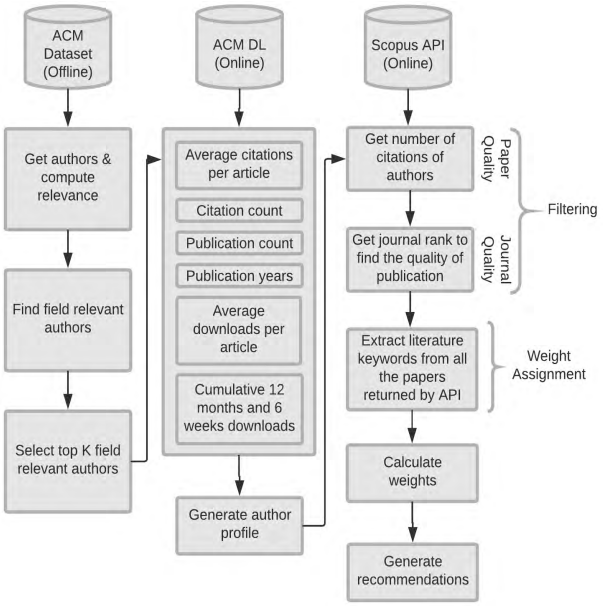


**FIGURE 1.** System flow diagram.

### A. STAGE 1 - FILTERING DATA

The proposed system allows a scholar to enter one or more topic keyword and performs a search that proceeds through various steps to produce a list of potential collaborators. The steps executed in the data filtering stage are as follows:

#### 1) USAGE HISTORY PROFILE

To ensure that the system can function effectively, it is essential to gain a comprehensive understanding of the scholar's various requirements and compose a profile accordingly. This is accomplished by keeping a log of the scholar's search history and his interaction with the system. His queries are aggregated, and frequently used keywords are weighted more highly, whereas the weights of keywords with lower frequencies are reduced. This helps the system to recognize changes in the scholarś background and future requirements.

$$S_i = Aggregation(S_s, K_s) \begin{Bmatrix} S_i = K_c \\ S_i = K_s \end{Bmatrix} \quad (1)$$

Equation 1 is derived from the keyword aggregation mechanism described in [6]. The scholar's interest profile, $S_i$, is obtained by aggregating the keywords searched for by the scholar, $S_s$, along with the keywords entered in the scholar's own profile, $K_s$, and the keywords extracted from the publications of potential collaborators accessed by the scholar, $K_c$; this last set of keywords is included because it is assumed that the scholar will only access the profiles of collaborators in whom he is most likely to be interested. Additional search keywords are added to the system over time to allow it to adapt to changes in the scholarś requirements.

**Algorithm 1** Fetching Basic Details

```
Require: ACM_D = ACM offline dataset;
Data: Keywords k entered by the scholar;
Result: ACM_L = list of relevant author details from the ACM offline dataset;
l = GetLiterature(item);
a = GetAuthors(l);
p = GetAuthorPublications(a);
y = GetPublicationYears(a);
c = GetCitationCount(a);
for Each item r do
    if y < 3 years then
        if c > 0 then
            add item to a temporary list L_t
        end if
    end if
end for
ACM_L = SortByKeywordCount(L_t);
Return(ACM_L);
```

#### 2) FETCHING BASIC DETAILS (OFFLINE DB)

Based on the current keywords specified by the scholar, the system performs a basic search to fetch some basic details from the ACM offline dataset [18]. The returned results include a set of publications containing the specified keywords; from these, the system extracts the author names and assigns weights based on their naming order. As depicted in algorithm 1, to gain basic insights of the entered keyword, we get literature details from ACM offline dataset, from the results we extract authors, their publications, year of publication, and citations. Once these details are extracted we filter out all the publications that are more than three years old. Finally, we sort the results based on keyword count.

### B. STAGE 2 - EXTENDING THE KNOWLEDGE GRAPH - ACM DL (ONLINE)

#### 1) EXTENDING THE KNOWLEDGE GRAPH - ACM DL (ONLINE)

To build an author knowledge graph consisting of each author's background, expertise, and contributions, the system requires additional details that are not readily available in the offline dataset due to the unavailability of up-to-date information. Since the creation of the offline dataset, various additions may have been made to the potential collaborators' publications and references; thus, it is better to fetch the updated details from the online ACM DL author profile pages. In Algorithm 2, it is evident that the system leverages existing details from ACM offline dataset and further polishes it by gaining updated insights from ACM DL (online). The details regarding author collaborations, citations and publication count, publication years, affiliations, and download records are accumulated. Based on these details, author is assigned a weight, and results are sorted on competence level. An author profile page on ACM DL site [21] includes the following details as shown in Fig. 2 and also discussed in our previous contribution [20]:

**Bibliometrics**: publication history

| | |
|---|---|
| Average citations per article | 3.80 |
| Citation Count | 19 |
| Publication count | 5 |
| Publication years | 2015-2016 |
| Available for download | 4 |
| Average downloads per article | 428.25 |
| Downloads (cumulative) | 1,713 |
| Downloads (12 Months) | 478 |
| Downloads (6 Weeks) | 51 |

**FIGURE 2.** Author publication history - ACM DL.

*a: AFFILIATION*

The affiliation history includes a list of institutions with which the author has been associated. It is vital to match the institution details with the author's name because this helps the system to identify the correct author from a list of authors with similar names.

*b: AVERAGE CITATIONS PER ARTICLE*

The average citation count is the ratio of the total citation count to the publication count.

*c: CITATION COUNT*

The citation count is the total number of times that the author's work has been cited by others in publications contained in ACM's bibliographic database. It includes citations in journals and proceedings but does not include citations in books, dissertations, and technical reports.

*d: PUBLICATION COUNT*

The publication count is the total number of works by the author in any field published by ACM.

*e: PUBLICATION YEARS*

The publication years correspond to the active years of the author from his very first publication in ACM to his most recent.

*f: AVAILABLE FOR DOWNLOAD*

This field shows the number of full-text articles that are available for download from ACM.

*g: AVERAGE DOWNLOADS PER ARTICLE*

The average download count is the ratio of the total downloads to the number of articles that are available for download.

*h: DOWNLOADS (CUMULATIVE)*

The all-time download count is the number of times that the author's works have been downloaded from ACM and is updated monthly.

*i: DOWNLOADS (ANNUAL)*

The number of downloads over the last twelve months is updated biweekly.

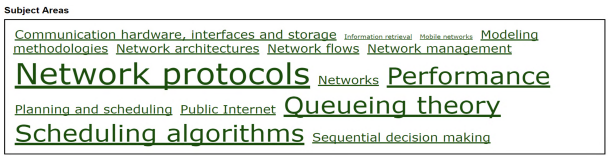


**FIGURE 3.** Areas of interest - ACM DL.

*j: DOWNLOADS (SIX WEEKS)*

The number of downloads over the last six weeks is also updated biweekly. We need the details listed above to filter the information and add weights based on various factors to improve the overall recommendation quality.

#### 2) AREAS OF EXPERTISE

A scholar may work in various areas throughout his research career; thus, to generate better recommendations, it is essential to understand the extent of his expertise. We assume that an author will have numerous publications in each of his fields of interest. Based on this assumption, the system creates an interest graph based on his publications in various fields, as shown in Fig. 3.

#### 3) COLLABORATIVE EXPERTISE

The previous collaboration history of an author is an essential factor in identifying the quality of his contributions. We assume that quality work is the result of talented collaborators directing their efforts toward achieving impactful contributions, and hence, we need to consider the profiles of the co-authors with whom an author has previously collaborated. This information is used to further weight the author's fields of interest and his level of competence.

### C. STAGE 3 - SUPPLEMENTING THE RESULTS

Up to this point, we have retrieved many details related to pertinent authors; however, to gain a more comprehensive understanding, we broaden our information set by gathering further data as listed below.

#### 1) EXTENDED AUTHOR EVALUATION - SCOPUS (ONLINE)

To enhance the scope of our author understanding, we leverage the Scopus API [18], which yields additional results regarding the publications of the authors of interest from a comprehensive library of Elsevier publications that includes numerous journals on a range of subjects. There are two core objectives of using Scopus; to acquire additional information about the authors identified from the ACM DL and to further extend the set of identified authors. The flow of Algorithm 3 specifies that the system leverages the details aggregated from offline as well as online ACM DL. It assesses the authors with existing Scopus profiles and fetches basic details such as publications, citations, H-index, co-authors, and area of interest. These results are evaluated as per relevance to the requirement and are sorted accordingly. We discuss the details in the following subsections:

**Algorithm 2** Fetching Supplementary Details

Require: $ACM_L$ = results from the ACM offline dataset;
Data: Results $r$ returned from the ACM DL;
Result: $ACM_A$ = list of relevant author details from ACM DL;
$a$ = GetAuthorDetails(item);
$cc_a$ = GetAverageCitationCount(a);
$cc_t$ = GetTotalCitationCount(a);
$py$ = GetPublicationYears(a);
$ah$ = GetAffiliationHistory(a);
$pc$ = GetPublicationCount(a);
$ad$ = GetAvailableDownloads(a);
$d_c$ = GetCumulativeDownloads(a);
$d_y$ = GetYearDownloads(a);
$d_6$ = GetSixWeeksDownloads(a);
**for** Each item $r$ **do**
  **if** $py < 3$ years **then**
    **if** $cc_a > 0$ **then**
      add item to a temporary list $L_t$
    **end if**
  **end if**
**end for**
$w$ = GetWeight($L_t$);
$ACM_A$ = SortByWeight($W$);
Return($ACM_A$);

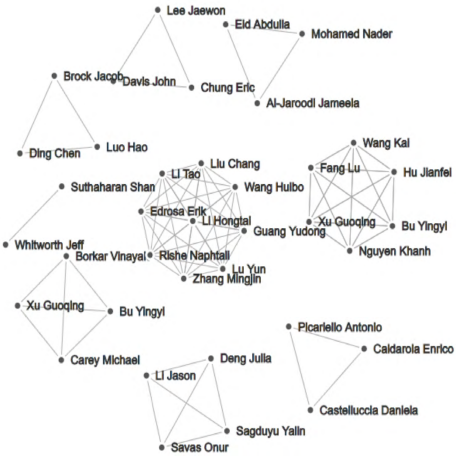


**FIGURE 4.** Co-author graph.

*a: SUPPLEMENTING THE AUTHOR DETAILS*

The Scopus API is used to search for the authors identified from the ACM network as described above to obtain further details. We are interested in collecting details regarding an author's publications in Scopus and the co-authors who have existing partnerships with the author, as shown in Fig. 4, to gain a broader view of the author's contributions. The details that we retrieve include the title and abstract of each publication, from which we can discover the publication's relevance to the search term based on term frequency; the author names, from which we can identify co-author

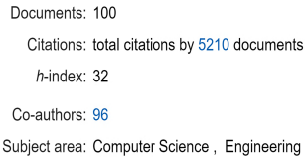


**FIGURE 5.** Author publication history - Scopus.

**Algorithm 3** Extending Author Details

```
Require: ACM_A = ACM author list;
Data: Results r returned by the Scopus API;
Result: SCP_A = list of relevant authors from Scopus
a = GetAuthorDetails(item);
p = GetPublications(a);
c = GetCitations(a);
i = GetHIndex(a);
co = GetCo-Authors(a);
sub = GetSubject(a);
SCP_A = SortByRelevance(r);
Return(SCP_A);
```

relationships; the publication date, from which we can determine the recency of work; and the journal details, which we can use to assess the quality of the publication. We also collect details regarding the author's total publications, citations, h-index, co-authors and areas of expertise, as shown in Fig. 5.

#### b: OBTAINING A WIDER VARIETY OF RESULTS

The second purpose of utilizing the Scopus API is to gather details on more authors related to the specified keywords. We search for available literature related to the keywords and fetch the relevant author details to accumulate a broad spectrum of authors who are associated with the keywords searched by the scholar.

### D. STAGE 4 - GENERATE THE RELEVANT AUTHOR LIST

In the final stage, as depicted in algorithm 4, the aggregated results returned from ACM offline dataset, enhanced by ACM online DL, and further extended by the Scopus are leveraged to compute trust level. The system filters all the publications that are older than three years, relevance is computed based on keywords, whereas popularity and quality is calculated on the bases of citations. Furthermore, weight is assigned on the basis of keyword frequency and trust. Finally, it sorts the recommendations based on weight as expressed in the following sub-sections:

#### 1) COMPUTING TRUST

In human-computer interaction (HCI) trust is one of the key factors affecting system performance. Computing the trustworthiness of recommendations requires a great deal of information. As described in the following subsection, we filter the collected information on the basis of certain assumptions.

#### a: RECENCY

The scope of scientific research is constantly broadening as researchers introduce new solutions to various problems. For this reason, we assume that more recent contributions of an author should have more impact on issues of recent interest compared with many-decades-old research, which might have little current relevance. Due to this assumption, we consider each author's scientific contributions from the last three years to gain an understanding of his current area of research and his level of activity.

#### b: RELEVANCE

The relevance of an author's area of interest should be aligned with the requirements of the scholar issuing the query. It is vital to remove from the final output any undesirable results that might not be of interest to the scholar.

- *Publication Keywords:* Because authors are permitted to selected only a limited number of keywords to elaborate the scope of their publications, it is assumed that the author-designated keywords have been chosen wisely and can be leveraged as one of the key aspects for weighting the relevance of a publication.
- *Author Ownership:* Author contribution is another factor to consider because it reflects the amount of effort put forward by each collaborator. We consider only publications with author lists limited to ten collaborators, and we assume that the contribution percentage varies in proportion to the naming order unless otherwise specified. To calculate the percentage of ownership, we use the following formula:

$$\sum_{i=1}^{10} AC_i$$
$$AC_i = ((1 - 0.1n) * 100) \quad (2)$$

  Here, $i$ is the total number of authors from 1 to 10, $AC_i$ is the author contribution, and $n$ is the naming position of the author. By applying the above mechanism, we can consider different contribution levels for the authors of a publication.
- *Keyword Frequency:* To capture the variation in an author's fields of interest over the past few years, the system uses the keyword frequency in the author's publications, as we assume that majority of the author's contributions will be associated with his fields of interest.
- *Author Similarity:* To find the degree of relevance between similar authors, we calculate the cosine distance between their publications. The result can take any value between 0.0 and 1.0, where 1.0 indicates an exact match. Thus, a value closer to 1.0 indicates a higher degree of relevance between the authors based on the

general cosine distance formula:

$$\frac{\sum_{n}^{i=1} x_i y_i}{\sqrt{\sum_{n}^{i=1} x_i{}^2}\sqrt{\sum_{n}^{i=1} y_i{}^2}} \tag{3}$$

Here, $i$ is an index that runs from 1 to $n$, $x_i$ is the $i$-th component of the publication vector of the first author, and $y_i$ is the $i$-th component of the publication vector of the second author.

#### c: IMPACT

The impact of an author's contributions reflects their quality. The fundamental data used to compute an author's impact are listed below.

- *Citation Count:* Citations reflect the significance of a contribution because they indicate that a publication has been used by other authors to support their own work. Part of the beauty of scientific research that researchers leverage the work done by other authors while crediting them for their efforts. We assume that this acknowledgement is of the utmost importance; thus, the higher the citation count is, the greater is the impact of a contribution.
- *Downloads:* Another vital factor to consider is the number of times a publication has been downloaded. The download count reflects the number of people who were interested in learning more about the contribution of the article and therefore download it.
- *Collaboration Colleagues:* The third aspect to consider is the quality of the collaborators with whom an author has worked. This consideration is vitally important, as even a slight contribution on the part of collaborators might yield an impactful result.

#### d: USER EXPERTISE

As discussed in section III.B.2, we retrieve the basic details about the scholar's area of interest from his ACM profile; however, we also utilize the term-frequency/inverse-document-frequency (TF/IDF) method to discover the keywords used in those of the scholar's publications that are accessible. A term identified in this way is added to the existing keyword set if it has not yet been specified or if it would increase the weight of an existing keyword. In this way, we can gain a more complex understanding of the scholar's scope of interest and can suggest better options to him.

#### e: QUALITY

To ensure that the generated recommendations are closely relevant to the scholar, we must consider the contribution quality of candidate author. The aspects considered when assessing quality are listed below.

- *Literature:* We assume that the recency of research is vital. Hence, the quality of a publication can be determined from its number of citations and its level of interest as indicated by its recent download count, as described in previous sections. Equation 4 gives the formula used to compute the quality $Q_L$ of a body of literature.
- *Journal:* We assume that the literature published in high-ranked journals has undergone a stringent review process before being published, and hence, an article that has met the criteria for publication in a certain journal should be of a quality level consistent with that of the journal. For this reason, we consider the journal impact factor in the quality evaluation.
- *Downloads:* Another important aspect to acknowledge as a quality factor is the number of times a publication has been downloaded. It is vital to evaluate the number of people who found the article interesting and worth to go though and therefore downloaded it.

$$Q_L = \frac{CC_L}{TP_A} * \sum_{i=1}^{n} JQ_i * \sum_{j=1}^{3} D_j \tag{4}$$

Here, $CC_L$ is the citation count of a body of literature $L$, $TP_A$ is the total number of publications by author $A$, sum of journal quality is $JQ_i$ and the sum over $D_j$ represents the summation of several download factors $D_{1,2,3}$, where $D_1 = \frac{D_T}{n^2}$, $D_2 = \frac{D_Y}{n}$, and $D_3 = D_{6W}$. $\mathrm{D_Y}$, $\mathrm{D_{6W}}$, and $\mathrm{D_T}$ are the download count over the last twelve months, the download count over the last six weeks, and the total download count, respectively, for all publications, and n is a constant with a value of 10 that is applied to reduce the relative weights of the download counts calculated over longer time intervals.

As seen above, we consider a sum of terms related to the total downloads as well as the downloads over the last six weeks and the last twelve months. This approach is motivated by the assumption that the recent download counts can serve as indicators of popularity, whereas the total download count emphasizes the impact of the author's contributions.

### 2) SORTING THE RECOMMENDATIONS

Once the various criteria have been assessed, the final list of potential collaborators for the scholar is generated and sorted based on their relevance and impact weights determined as described in previous sections. Hence, the author with the highest relevance is listed in the top position, and the others are listed after that based on their weights.

## IV. EXPERIMENTATION

The system operates in three phases to generate the list of recommended collaborators based on information from various publishers, as specified in the following subsections.

### A. PRIMARY BASE

The primary base for our recommendations, we use the ACM DL [19] offline dataset. Some technical details on the dataset and the analysis of the data are presented below.

**Algorithm 4** Aggregating Results & Computing Trust

Obtain author details $ACM_A$ from the ACM DL and $SCP_A$ from Scopus;
Require: $ACM_L$ = ACM DL author list, $SCP_A$ = Scopus author list;
Data: Results $r$ obtained from the ACM DL and Scopus;
Result: *Authors* = list of relevant authors after aggregation;
$Data_A$ = AggregateData(item);
*Trust* = ComputeTrust($Data_A$);
**for** Each item $Data_A$ **do**
    **if** $y < 3$ years **then**
        **if** $k > 0$ keywords **then**
            **if** $c > 0$ citations **then**
                **if** $k_f > 0$ keyword frequency **then**
                    add item to a temporary list *Trust*
                **end if**
            **end if**
        **end if**
    **end if**
**end for**
*Weight* = AssignWeight(*Trust*);
*Authors* = SortByWeight(*Weightage*);
Return(Authors);

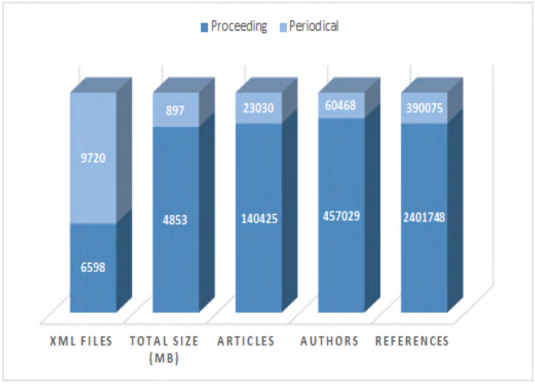


**FIGURE 6.** ACM dataset.

### *a: TECHNICAL DETAILS*

We obtained the ACM dataset in the form of XML files along with DTD files that specify the XML structure of the data. We converted the XML data into the MySQL format to achieve a more comfortable level of management and faster retrieval.

The XML data contain publication details for a period of 33 years, from 1982 to 2015, separated into two subsets: periodicals and proceedings. In Fig. 6, we show the number of XML files, the total data size, the number of authors, and the number of references for each of these subsets.

### *b: DATA ANALYSIS*

Because we transformed the XML data into the MySQL format, it was fairly easy for us to analyze various aspects of the data and identify the patterns relevant to our work. Python was our language of choice because it offers a wide variety of libraries that can be used to handle fairly large and robust datasets.

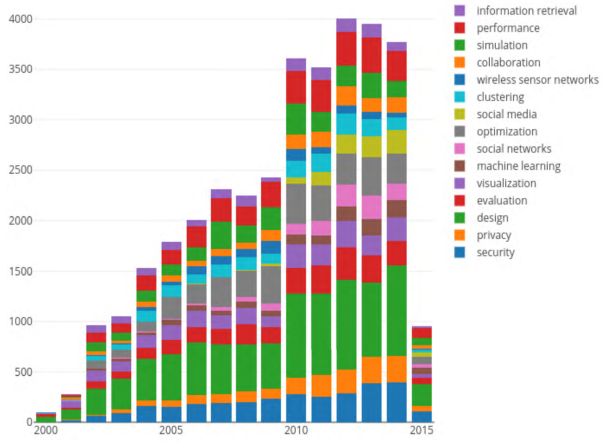


**FIGURE 7.** Distribution of the top 15 keywords amount publications sin the proceeding dataset for the 15-year period from 2000 to 2015 (for the precise interpretations of the colors and for an interactive graph, the reader is referred to the web version of this graph [24]).

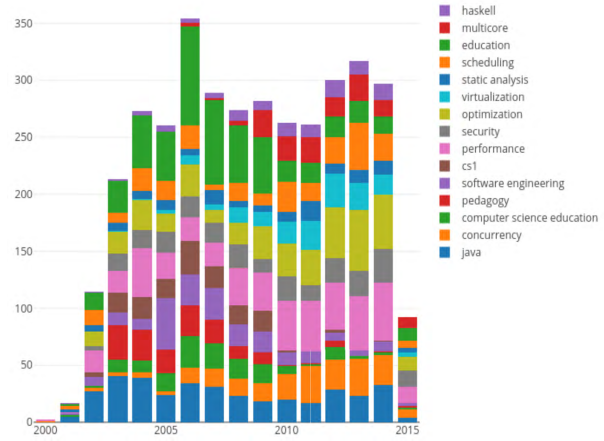


**FIGURE 8.** Distribution of top 15 keywords among publications in periodicals dataset for 15-year period from 2000 to 2015 (for the precise interpretations of the colors and for an interactive graph, the reader is referred to the web version of this graph [25]).

Fig. 7 illustrates the distribution of the top 15 keywords among publications in the proceedings dataset for the 15-year period from 2000 to 2015. The reason we considered only the last fifteen years of the dataset is that the most current patterns in the literature were not clearly visible before this time period. For this reason, we truncated the duration of the data sample to allow the graphs to be plotted on a suitable scale. Similarly, Fig. 8 shows the fifteen-year trends of the top 15 keywords in periodicals dataset.

## *B. ADDITIONAL DETAILS COLLECTED FROM THE ACM DL*

To supplement the data, we collected the latest results from the ACM DL online dataset to compute the author

| Person Id | Name | Quality Scopus | Pubs in base | Pubs w/keyword | ACM info | ACM score | Scopus Info | Scopus score |
|---|---|---|---|---|---|---|---|---|
| P4152680 | *Hidden* | 100 | 1 | 1 | Pubs in ACM DL: 24<br>Num of pubs: 57<br>Citations: 1016<br>Downloads - 6 weeks: 222<br>Downloads - 12 months: 2080<br>Downloads - Total: 18333<br>Earliest pub: 1994<br>Latest pub: 2015<br>ACM DL Id: 81100072677 | 10932 | Num of pubs: 100<br>Id: 7102955788<br>Num of coauthors: 97<br>Num of cited docs: 5177 | 5374 |
| P4148699 | *Hidden* | 100 | 1 | 1 | Pubs in ACM DL: 19<br>Num of pubs: 27<br>Citations: 347<br>Downloads - 6 weeks: 79<br>Downloads - 12 months: 688<br>Downloads - Total: 12083<br>Earliest pub: 2001<br>Latest pub: 2015<br>ACM DL Id: 81100597598 | 3452 | Num of pubs: 29<br>Id: 12800602900<br>Num of coauthors: 85<br>Num of cited docs: 563 | 677 |
| P4959824 | *Hidden* | 100 | 2 | 2 | Pubs in ACM DL: 9<br>Num of pubs: 23<br>Citations: 378<br>Downloads - 6 weeks: 49<br>Downloads - 12 months: 366<br>Downloads - Total: 12090<br>Earliest pub: 1995<br>Latest pub: 2015<br>ACM DL Id: 81100305920 | 3394 | Num of pubs: 29<br>Id: 6602185366<br>Num of coauthors: 74<br>Num of cited docs: 972 | 1075 |

**FIGURE 9.** Recommended list of authors for the keyword 'Big Data'

quality weights based on up-to-date citation and download counts.

### C. ADDITIONAL DETAILS COLLECTED FROM SCOPUS

Furthermore, the Scopus API was called to get additional details on each author from his publication in various journals and proceedings that Scopus supports. These details include publication counts, co-author counts, and citations received. In parallel, the system also acquired the journal details to be used in the calculation of the quality weights.

## V. RESULTS & DISCUSSIONS

When a scholar performs a search for potential collaborators, the system gathers information from the offline dataset based on the specified keywords. As soon as a filtered list of publications is obtained, a new array of authors associated with these publications is created. For each of the authors, the ACM DL is queried to fetch the publication count, years of activity, fields of interest, co-authors, citations, and downloads associated with the author's contributions. The author names are also searched via the Scopus API to discover additional information regarding each author's contributions, including his publication count, co-authors, citations, h-index, subject areas and years of publication. In parallel to this, the details of the journals in which the author has published his scientific work are also queried to fetch the cite score, the SJR (Scientific Journal Rankings) score and the SNIP (Source-Normalized Impact per Paper) score, all of which are used to weight the quality of the author's work. More details on these research metrics can be found in [3] and [23]. These details are aggregated to calculate the relevance and impact of each author's contributions. Based on the author-assigned keywords and considering the assigned weights, a list of recommended authors is generated. This list is closely related to the scholar's query and indicates which authors are most likely to be the best fits as potential collaborators for his scientific contributions, as shown in Fig. 9.

### A. CHALLENGES

Acquiring information from various publishers helps us to enhance the quality of our recommendations; however, our system still encounters challenges. Some of the challenges that will require further consideration are discussed below.

#### 1) NO RESULTS

An author who has secured publication by one publisher will not necessarily have publications or a profile with another publisher. Therefore, it is entirely possible that even if an author has publications in the ACM database, it will not be possible to further verify his work via Scopus.

#### 2) MULTIPLE RESULTS

In the case of multiple authors with similar names, details may be fetched that might be irrelevant to the queried author. Although we have added additional filters based on data such as author affiliations to find better matches, there is still no way to associate an author from one publisher network with an author from another with perfect confidence.

#### 3) CHANGES IN RESEARCH BACKGROUND

It is somewhat difficult to precisely assess changes in an author's background or areas of interest. Because many details are not directly available from the ACM DL and Scopus, fetching such details would require additional queries and calculations. Hence, to compute whether an author has an interest in various fields, we would need to determine the publication counts for each related keyword and research area, which are not available out of the box.

#### 4) COMPUTING TRUST

The challenges discussed above also affect the trust computation because this computation depends on various factors, such as recency, popularity and impact, which can be determined with complete accuracy only if we can guarantee that the information we acquire is also accurate and complete.

#### 5) INCOMPLETE & EVOLVING USER PROFILE

When a user profile is incomplete, either because he has recently joined the system, usage is not sufficient, or he has not contributed his interests, or his profiles keep evolving as he might frequently be using un-related queries, the efficiency of the system is at stake. Though there are some ways to handle this situation to some extent lets say using demographic information, and leverage his usage history, but this is still a challenge to come up the mark of user expectations while the system is blindfolded with no input at all. Hence, it would take more time for the system to adapt user profile and suggest relevant recommendations.

## VI. CONCLUSION & FUTURE WORK

Recommender systems have assisted us in enhancing the level of machine-human understanding and, hence, in improving the quality of life in our society. In the field of scientific research, smart systems enhance the quality of research by reducing the time required to perform laborious tasks. In our previous paper [20], we proposed a system to assist researchers in discovering the best candidates with whom to collaborate based on specified keywords and their profile histories. These results were fetched primarily from the ACM offline database and then were further supplemented with missing details via the Scopus API from Elsevier. Because the scope of the system was limited, we encountered quite a few problems, such as discrepancies in the profile information of an author between different publisher networks. There is no readily available way to collect the overall publication and citation details for a researcher from each and every publisher; therefore, to address this issue, we propose an advanced profile structuring process that aggregates information from multiple publisher networks. In this system, our basic details about an author are supplemented with the latest up-to-date information regarding the author from the ACM DL before queries are sent to Scopus. By gaining more information about each author, we can acquire more correct author profiles when utilizing the Scopus API, as the author profiles in the two publisher networks are not directly related to each other.

We have mentioned some of the challenges we face in section V; in our future work, we will focus on resolving some of these issues while aggregating data from more publishers to make the system more robust and to achieve a more comprehensive understanding of the user's requirements to recommend more suitable results including relevant details from various fields to understand the dimensions and scope in depth. We will also work on enabling the system to adapt to changes in the research background of an author, and we will attempt to incorporate a mechanism for considering rising talent based on recent research activity. We will evaluate the algorithms presented and analyze their performance along with comparing these with existing solutions to find what works best for our system. We will also consider many other dimensions of the problem to further improve the recommendation quality.

## ACKNOWLEDGMENTS

The ACM Digital Library dataset is provided for non-commercial research purposes courtesy of the Association for Computing Machinery, Inc., 2017.

## REFERENCES

[1] L. Boratto, S. Carta, G. Fenu, and R. Saia, "Semantics-aware content-based recommender systems: Design and architecture guidelines," *Neurocomputing*, vol. 254, pp. 79–85, Sep. 2017, doi: http://dx.doi.org/10.1016/j.neucom.2016.10.079

[2] A. S. Raamkumar, S. Foo, and N. Pang, "Using author-specified keywords in building an initial reading list of research papers in scientific paper retrieval and recommender systems," *Inf. Process. Manage.*, vol. 53, no. 3, pp. 577–594, 2017, doi: http://doi.org/10.1016/j.ipm.2016.12.006

[3] (Aug. 15, 2017). *Journal Metrics in Scopus: Source Normalized Impact Per Paper (SNIP)*. [Online]. Available: https://blog.scopus.com/posts/journal-metrics-in-scopus-source-normalized-impact-per-paper-snip

[4] B. Amini, R. Ibrahim, M. S. Othman, and H. Rastegari, "Incorporating scholar's background knowledge into recommender system for digital libraries," in *Proc. 5th Malaysian Conf. Softw. Eng. (MySEC)*, 2011, pp. 516–523, doi: http://doi.org/10.1109/MySEC.2011.6140721

[5] J. Lu, D. Wu, M. Mao, W. Wang, and G. Zhang, "Recommender system application developments: A survey," *Decision Support Syst.*, vol. 74, pp. 12–32, Jun. 2015.

[6] Á. Tejeda-Lorente, C. Porcel, J. Bernabé-Moreno, and E. Herrera-Viedma, "REFORE: A recommender system for researchers based on bibliometrics," *Appl. Soft Comput.*, vol. 30, pp. 778–791, May 2015.

[7] Á. Tejeda-Lorente, J. Bernabé-Moreno, C. Porcel, P. Galindo-Moreno, and E. Herrera-Viedma, "A dynamic recommender system as reinforcement for personalized education by a fuzzly linguistic Web system," *Procedia Comput. Sci.*, vol. 55, pp. 1143–1150, Jan. 2015, doi: http://doi.org/10.1016/j.procs.2015.07.084

[8] B. Amini, R. Ibrahim, M. S. Othman, and M. A. Nematbakhsh, "A reference ontology for profiling scholar's background knowledge in recommender systems," *Expert Syst. Appl.*, vol. 42, no. 2, pp. 913–928, 2015, doi: http://doi.org/10.1016/j.eswa.2014.08.031

[9] B. Amini, R. Ibrahim, M. S. Othman, and A. Selamat, "Capturing scholar's knowledge from heterogeneous resources for profiling in recommender systems," *Expert Syst. Appl.*, vol. 41, no. 17, pp. 7945–7957, 2014, doi: http://doi.org/10.1016/j.eswa.2014.06.039

[10] S. Bruns, A. C. Valdez, C. Greven, M. Ziefle, and U. Schroeder, "What should I read next? A personalized visual publication recommender system," in *Human Interface and the Management of Information. Information and Knowledge in Context* (Lecture Notes in Computer Science), vol. 9173. Cham, Switzerland: Springer, 2015, pp. 89–100, doi: http://doi.org/10.1007/978-3-319-20618-9_9

[11] F. Zhang, T. Gong, V. E. Lee, G. Zhao, C. Rong, and G. Qu, "Fast algorithms to evaluate collaborative filtering recommender systems," *Knowl.-Based Syst.*, vol. 96, pp. 96–103, Mar. 2015, doi: http://doi.org/10.1016/j.knosys.2015.12.025
[12] J. Bauer and A. Nanopoulos, "Recommender systems based on quantitative implicit customer feedback," *Decision Support Syst.*, vol. 68, pp. 77–88, Dec. 2014, doi: http://doi.org/10.1016/j.dss.2014.09.005
[13] Z. D. Champiri, S. R. Shahamiri, and S. S. B. Salim, "A systematic review of scholar context-aware recommender systems," *Expert Syst. Appl.*, vol. 42, no. 3, pp. 1743–1758, 2015, doi: http://doi.org/10.1016/j.eswa.2014.09.017
[14] L. C. Chen, P. J. Kuo, and I. E. Liao, "Ontology-based library recommender system using MapReduce," *Cluster Comput.*, vol. 18, no. 1, pp. 113–121, 2015, doi: http://doi.org/10.1007/s10586-013-0342-z
[15] Q. Yu, C. Long, Y. Lv, H. Shao, P. He, and Z. Duan, "Predicting Co-author relationship in medical Co-authorship networks," *PLoS ONE*, vol. 9, no. 7, pp. 1–7, 2014, doi: http://doi.org/10.1371/journal.pone.0101214
[16] B. de Paula Fonseca e Fonseca, R. B. Sampaio, M. V. de Araújo Fonseca, and F. Zicker, "Co-authorship network analysis in health research: method and potential use," *Health Res. Policy Syst.*, vol. 14, no. 1, p. 34, 2016, doi: http://doi.org/10.1186/s12961-016-0104-5
[17] D. Rennie, "Integrity in scientific publishing," *Health Services Res.*, vol. 45, no. 3, pp. 885–896. 2010, doi: http://doi.org/10.1111/j.1475-6773.2010.01088
[18] (Aug. 15, 2017). *Accelerate Academic Research Using Scopus APIs* [Online]. Available: https://blog.scopus.com/posts/accelerate-academic-research-using-scopus-apis
[19] (Nov. 8, 2017). *ACM Digital Library*. [Online]. Available: https://dl.acm.org/
[20] K. Alinani, G. Wang, A. Alinani, and D. H. Narejo, "Who should be my co-author? Recommender system to suggest a list of collaborators," in *Proc. 16th IEEE Int. Conf. Ubiquitous Comput. Commun. (IUCC)*, Dec. 2017, pp. 1427–1433.
[21] (Nov. 8, 2017). *Author Page Sample—ACM Digital Library*. [Online]. Available: https://dl.acm.org/author_page.cfm?id=94658610794
[22] (Nov. 8, 2017). *Biblomatrics—ACM Digital Library*. [Online]. Available: https://dl.acm.org/author_page.cfm?dsp=bibliohelp
[23] (Nov. 13, 2017). *Research Metrics—Elsevier*. [Online]. Available: https://www.elsevier.com/solutions/scopus/features/metrics
[24] (Nov. 13, 2017). *Interractive Graph—Proceeding Dataset*. [Online]. Available: https://plot.ly/~research_publications/3.embed
[25] (Nov. 13, 2017). *Interractive Graph—Periodical Dataset*. [Online]. Available: https://plot.ly/~research_publications/1.embed
[26] A. Jøsang, R. Ismail, and C. Boyd, "A survey of trust and reputation systems for online service provision," *Decision Support Syst.*, vol. 43, no. 2, pp. 618–644, 2007, doi: https://doi.org/10.1016/j.dss.2005.05.019

**KARIM ALINANI** received the B.E. degree in software engineering from the Mehran University of Engineering and Technology, Pakistan, and the M.Sc. degree in computer application and technology from Central South University, China, where he is currently pursuing the Ph.D. degree in computer application and technology. His research interests include primarily on recommender systems, machine learning, and deep learning.

**ANNADIL ALINANI** received the B.E. degree in software engineering from the Mehran University of Engineering and Technology, Pakistan, and the M.Sc. degree in computer application and technology from Central South University, China, where she is currently pursuing the Ph.D. degree in computer application and technology. Her research interests include primarily on NDN and vehicular networks.

**DUA HUSSAIN NAREJO** received the B.S. degree in software engineering from Isra University, Pakistan. She is currently pursuing the M.Sc. degree in computer application and technology from Central South University, China.

**GUOJUN WANG** (M'08) received the B.Sc. degree in geophysics, the M.Sc. degree in computer science, and the Ph.D. degree in computer science from Central South University, China. He has been a Professor at Central South University, a Visiting Scholar at Temple University and Florida Atlantic University, USA, a Visiting Researcher at the University of Aizu, Japan, and a Research Fellow at the Hong Kong Polytechnic University. He is currently a Pearl River Scholarship Distinguished Professor and the Vice Dean of the School of Computer Science and Educational Software, Guangzhou University, China, where he is also the Director of the Institute of Computer Networks. His research interests include cloud computing, big data, trusted computing, and information security. He is a Distinguished Member of CCF and a member of the ACM and the IEICE.