February 2022

Recommending Scientific Datasets Using Author Networks in Ensemble Methods

Xu WANG [a,b,1], Frank VAN HARMELEN [a,b] and Zhisheng HUANG [a]
[a] *Vrije Universiteit Amsterdam, De Boelelaan 1105, 1081 HV Amsterdam, The Netherlands*
[b] *Discovery Lab, Elsevier, The Netherlands*

Abstract. Open access to datasets is increasingly driving modern science. Consequently, discovering such datasets is becoming an important functionality for scientists in many different fields. We investigate methods for *dataset recommendation*: the task of recommending relevant datasets given a dataset that is already known to be relevant. Previous work has used meta-data descriptions of datasets and interest profiles of authors to support dataset recommendation. In this work, we are the first to investigate the use of co-author networks to drive the recommendation of relevant datasets. We also investigate the combination of such co-author networks with existing methods, resulting in three different algorithms for dataset recommendation. We obtain experimental results on a realistic corpus which show that only the ensemble combination of all three algorithms achieves sufficiently high precision for the dataset recommendation task.

Keywords. Dataset Recommendation, Dataset Discovery, Data Science, Co-author Network

1. Introduction

The availability of open data is increasingly driving modern science in different fields, ranging from astrophysics [1] to earth sciences [2,3], and from medicine [4,5] to AI [6]. Scientists are encouraged to publish dataset using the FAIR principles [7], and this practice is increasingly being made compulsory by funding agencies. It is widely acknowledged that such open datasets contribute to both the transparency of science, to its quality, it's reproducibility and indeed to the speed of scientific developments. Publishing open datasets has been widely acknowledged as a key factor in the rapid scientific response to the COVID-19 pandemic [8].

Given these developments, the task of *finding* relevant datasets is becoming increasingly important for scientists. At the same time, the increasing volume of scientific datasets that is available online brings with it a need for intelligent tooling to support this task. Commercial providers have started offer dataset search services to scientists, such as Dataset Search from Google [9], Mendeley Data(https://data.mendeley.com/) search from Elsevier, and dedicated repositories such as Figshare . These search engines index millions of datasets, but provide only keyword based search, which is often not suffi-

[1] Corresponding Author: Xu Wang, Vrije Universiteit Amsterdam, De Boelelaan 1105, 1081 HV Amsterdam, the Netherlands, and: Discovery Lab, Elsevier, The Netherlands ; E-mail: xu.wang@vu.nl

ciently powerful to locate relevant datasets with sufficient precision: a keyword-based query such as “diabetes risk” will return 200 results on Google Dataset Search, 2000 results on Mendeley, and over 20.000 on Figshare.

Besides keyword search, a well-known alternative search paradigm is *recommendation search* [10], where a known item of interest (e.g. a product) is used to recommend similar items that are also of interest. This paradigm has also been applied to scientific publications (see [11] for a recent survey). In this paper, we will develop algorithms for recommendation search for scientific *datasets* instead of publications. Recommendation search has been explored for scientific dataset in earlier work (see our discussion in Section 2, but we are the first to propose the use of co-author networks as a major information source for the recommendation algorithm. Our working hypothesis is that we provide a new hypothesis for dataset recommendation: ”If the authors of two datasets have a strong relationship in the co-author network, then these two datasets can be connected with a recommendation link”. Furthermore, we will combine the co-author-network-based algorithm with existing methods for dataset recommendation. We use both a graph embedding method and a ranking method from information retrieval to construct an ensemble method for dataset recommendation. The graph embedding approach transfers the authors from the co-author network into a vector space that allows us to calculate the similarity between authors. The ranking method from information retrieval improves the ranking of datasets that are similar to the given dataset. We perform experiments with these methods on a realistic corpus of datasets and co-author relations. Among these different methods, only the ensemble method that combines all three of them results in a reasonable precision (0.75), although at the cost of low recall.

The main contributions of this paper are: 1) We construct a co-author network based on the Microsoft Academic Knowledge Graph (MAKG) to represent the academic publication-relationship between authors. 2) We provide three dataset recommendation algorithms: the first algorithm uses only a graph walk in the co-author network to recommend datasets, the other two algorithms combine this with graph embeddings and a ranking approach. 3) We perform experiments which use these algorithms on real-world data. Our results show that only the performance of this ensemble method yields sufficient precision for a realistic recommendation algorithm for scientific dataset search..

2. Related Work and Motivation

Co-author networks play a very import role in the study of academic collaborations, and in attempts to provide maps of academic fields of study. In [12], a co-author network was used to searching promising researchers via network centrality metrics. Even more ambitiously, [13] used a co-author network to predict possible future strong researchers. Sun et al. [14] provided an approach to predict future co-author relationships with the help of heterogeneous bibliographic networks.

Because of the increasing importance of open datasets for modern science, a number of *dataset search engines* can be found online nowadays, including Google Dataset Search [2], Mendeley Data[3], Microsoft Rearch Open Data[4] and others. These dataset

[2] https://datasetsearch.research.google.com/

[3] https://data.mendeley.com/

[4] https://msropendata.com/

search engines help researchers to find datasets based on an input query consisting of keywords.

An alternative search process is to *recommend* datasets based on the datasets which were found by other search engines. In our previous papers [15,16], we also adopted such a recommendation paradigm ”if you like this dataset/query, you’ll also like these datasets......”. There are several other interesting works on dataset recommendation. Michael et al. [17] propose a system that recommends suitable datasets based on a given research problem description. Chen et al. [18] study the problem of recommending the appropriate datasets for authors, by using a multi-layer network learning model on the information from a three-layered network composed by authors, papers, and datasets. Ellefi et al. [19] provide a dataset recommendation approach by considering the overlap between the schema of two datasets. Altaf et al. [20] provide a dataset recommendation method based on a set of research papers given by user. Giseli et al. [21] present two approaches for datasets recommendation, based on Bayesian classifiers and on on Social Network connections. Both of their approaches use vocabularies, classes and properties of datasets to rank the dataset for recommendation. In contract, Gogal et al. [22] represent the researchers in vector space based on the publications of researchers, and then use cosine similarity between the vectors of the publications of researchers and the vectors of the datasets to do recommendation.

Many of these approaches use both the content and the meta-data of the datasets. For example, Kato et al. test datasets from 74 dataset search engines with a dataset retrieval task by using the content of these datasets [23]. However, the contents of scientific datasets is in general extremely heterogeneous, ranging from numerical time sequences, to genetic codes, to astrophysical observations, to geodata, to spreadsheets with economic indicators and many others. Furthermore, the specific type of a dataset is often not even explicitly indicated. In our previous work we have therefore limited ourselves to the use of *only the meta-data descriptions* of datasets as the signal to do dataset recommendation. In [16], we used ontology-based concept similarity, a machine learning approach for text similarity and an information retrieval approach, all applied only to the title and other meta-data fields of the dataset. Our experimental results showed that the information retrieval approach could outperform others, but the performance of this approach was still relatively low.

This provides us with the motivation to search for other signals that can be used to improve the results of dataset recommendation, besides title and meta-data, while abstaining from the contents of the dataset. A very little explored signal for dataset recommendation is the academic co-author network. Even though from existing work, we know that co-author networks can be used to make meaningful predictions and analyses, it is not a priori clear whether such links between co-authors can also be exploited to find relevant links between datasets, in order to drive dataset recommendation.

The motivating question for this paper is therefore whether a co-author network can contribute to dataset recommendation. And more elaborately, whether we can use such co-author analysis in an ensemble combination with other approaches to obtain maximally good result. Our research questions are therefore as follows:

1. How to do dataset recommendation by using a co-author network? And how to combine existing dataset recommendation methods with such a co-author network based approach?

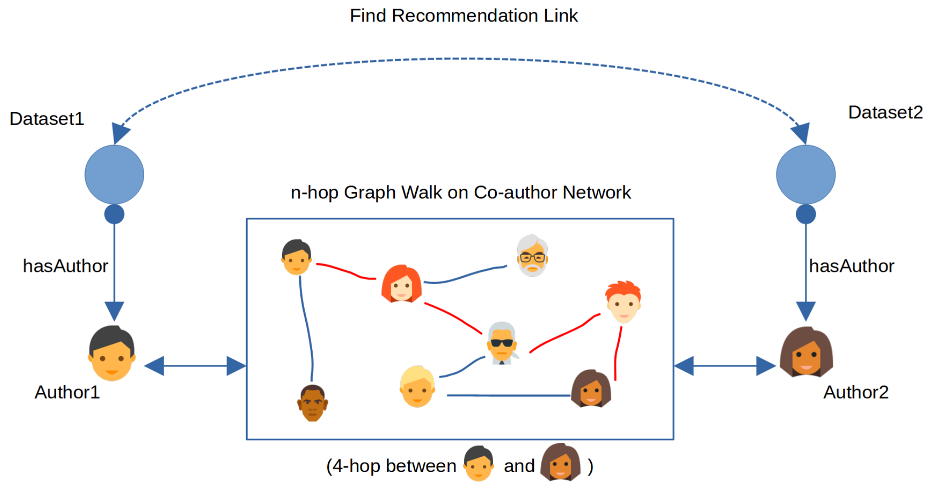

Figure 1. Recommendation Pathway between Dataset1 and Dataset2 based on co-author network.

2. How to evaluate the recommendation approach between datasets and to evaluate the quality of recommendation links built by our recommendation approach?
3. How to obtain and use real data for our experiments on the recommendation and evaluation approach?

3. Dataset Recommendation Approaches

In this section, we will introduce three dataset recommendation algorithms that we will test in our experiments: the first is based on computing paths in a co-author network, the second is based on vector embeddings of author computed from the academic network, and the third is a ranking method often used in information retrieval. As we mentioned before, the goal of dataset recommendation is to map one or more given datasets to a collection of recommended datasets. This makes dataset recommendation different from dataset search which amounts to mapping a query to a collection of datasets. Before introducing specific dataset recommendation algorithms, we give the general definition of dataset recommendation:

Definition 1 (Dataset Recommendation)**.** *Let* $D = \{d_1, d_2, \ldots\}$ *be a set of datasets. Dataset recommendation is a function* $Rec : D \to 2^D$ *such that* $Rec(d_i) = \{d_j | d_j$ *is recommended to* $d_i, d_j \in D\}$.

Based on this definitions, the goal of this paper is to compute recommendation relationships between datasets. We will propose different dataset recommendation algorithms that implement the function *Rec*

3.1. Co-author network based approach

In this section we will briefly introduce the idea of a dataset recommendation algorithm based on a co-author network. The intuition is to construct a pathway from one dataset to another with the help of a co-author network, as shown in Figure 1.

To find such relationships, we take into account the authors of the datasets. The authors of the datasets are then matched to the authors of in the publication co-author network. At this point we can use the co-author network to find (potential) links between the authors of the datasets. Eventually we build recommendation links between datasets through links between authors. Then, as shown in Figure 1, the recommendation pathway from Dataset1 to Dataset2 is "Dataset1 → Author1 → (Co-)Author network → Author2 → Dataset2".

We will now formalise the co-author network based approach. First off, we define a co-author network:

Definition 2 (Co-Author Network)**.** *Let $A = \{a_1, a_2, ...\}$ be a set of authors. Let $P = \{p_1, p_2, ...\}$ be a set of papers. The authors of a paper are denoted by a function $Author : P \rightarrow 2^A$, which means that for each $p \in P$, $Author(p) = \{a | a \in A, Author(p, a)\}$, where $Author(p, a)$ means a is the author of p. Co-author is a predicate $CoAuthor : A \rightarrow A$, which means that for $a_1, a_2 \in A$, $CoAuthor(a_1, a_2) \leftrightarrow a_1 \in Author(p_x), a_2 \in Author(p_x), p_x \in P$. A co-author network is a set of co-author relationships, $\{CoAuthor(a_i, a_j) | a_i \in A, a_j \in A\}$.*

In the co-author network definition, we have the definition of co-author relationship between two author, denoted by $CoAuthor(a_1, a_2)$. We then define the co-author distance between a_1 and a_2 to be 1, or we can also say that a_2 is 1-hop walk from a_1 in co-author network. If we then also have $CoAuthor(a_2, a_3)$, this makes the co-author distance between a_1 and a_3 is 2, and a_3 is 2-hop walk from a_1 in co-author network.

Then we will introduce the connection between authors in co-author network. We define this connection as a co-author path between authors.

Definition 3 (Co-author Path)**.** *Let $A = \{a_1, a_2, ...\}$ be a set of authors. Let CoNet be a co-author network. A co-author path between authors is a function $AuthorPath_{CoNet}(a_i, a_j) \leftrightarrow CoAuthor(a_i, ...), ..., CoAuthor(..., a_j)$, where $CoAuthor(a_i, ...), ..., CoAuthor(..., a_j) \in CoNet$. This also means a_i and a_j have co-author path if and only if there is co-author relationship pathway between a_i and a_j in co-author network $CoNet$.*

In this definition, the co-author path between two authors can also be considered as an n-hop walk from one author to another. So we also call this approach as graph walk based approach. For instance, when we have 3-hop walk from author a_m to a_n in co-author network $CoNet$, we also have that $AuthorPath_{CoNet}(a_m, a_n) \leftrightarrow CoAuthor(a_m, a_1), CoAuthor(a_1, a_2), CoAuthor(a_2, a_n)$, where a_m, a_n, a_1, a_2 are authors; $CoAuthor(a_m, a_1)$, $CoAuthor(a_1, a_2)$ and $CoAuthor(a_2, a_n)$ are in $CoNet$. We can also say that $AuthorPath_{CoNet}(a_m, a_n) = 3$ here.

Then, we have the definition of dataset recommendation based on a co-author network.

Definition 4 (Recommendation based on Co-author Network)**.** *Let $A = \{a_1, a_2, ...\}$ be a set of authors. Let $D = \{d_1, d_2, ...\}$ be a set of datasets. Let CoNet be a co-author network. Dataset recommendation based on a co-author network is a function $Rec_{CoNet} : D \rightarrow 2^D$, such that for each $d_i \in D$, $Rec_{CoNet}(d_i) = \{d_j | a_i \in Author(d_i) \cap A, a_j \in Author(d_j) \cap A, AuthorPath(a_i, a_j), d_j \in D\}$, where $AuthorPath_{CoNet}(a_i, a_j)$ means that there exists path between a_i and a_j in $CoNet$.*

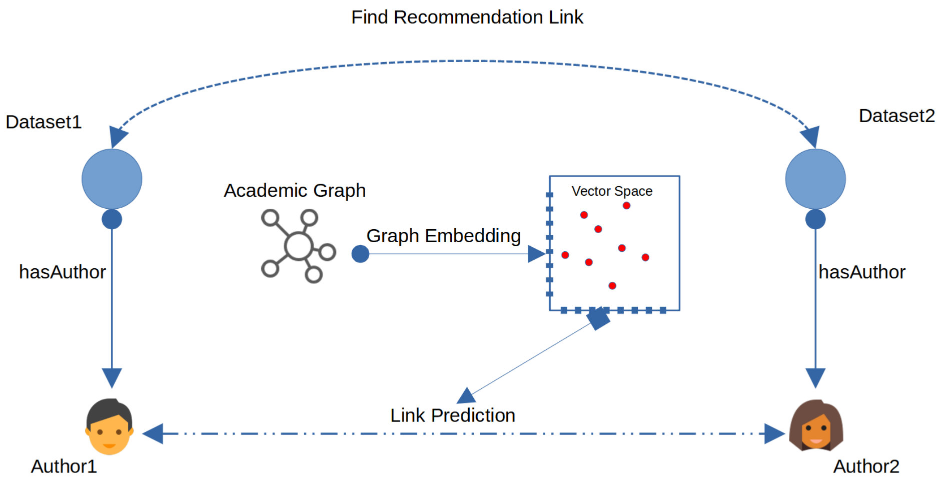

Figure 2. Recommendation Pathway between Dataset1 and Dataset2 based on graph embedding.

We can specialise this into a definition of dataset recommendation based on an n-hop graph walk in a co-author network.

Definition 5 (Dataset recommendation based on n-hop walk on co-author network)**.** *Let $A = \{a_1, a_2, \ldots\}$ be a set of authors. Let $D = \{d_1, d_2, \ldots\}$ be a set of datasets. Let CoNet be a co-author network. Dataset recommendation based on an n-hop walk in a co-author network is a function $Rec^n_{co-author} : D \to 2^D$, such that for each $d_i \in D$, $Rec^n_{co-author}(d_i) = \{d_j | a_i \in Author(d_i) \cap A, a_j \in Author(d_j) \cap A, AuthorPath_{CoNet}(a_i, a_j) \leq n, d_j \in D\}$, where $AuthorPath_{CoNet}(a_i, a_j) \leq n$ means that the shortest path between a_i and a_j in CoNet is not more than n.*

3.2. Knowledge graph embedding based approach

A knowledge graph embedding is the transformation of the entities and relationship of a knowledge graph into a vector space [24]. There are many existing and popular knowledge graph embedding models, such as ComplEx[25], TransE[26], TransR[27], RESCAL[28] and many others. See [29,30] for a survey. An embedding of a co-author graph in a vector spaces allows us to generate new (predicted) links between the authors. We can then use such predicted links between authors as a way to recommend datasets, just as we used the existing co-author links between authors above. In this paper, we will use the pre-trained author entity embedding, which is trained by ComplEx on the Microsoft Academic knowledge graph.

Figure 2 shows the overview of using a graph embedding for dataset recommendation. We would use the graph embedding of co-author network to construct the vector space which contains all the vectors of authors. Then we use the cosine similarity metric between author vectors to do link prediction: to predict a link between authors with a high similarity. We then use the predicted links to build the recommendation links between datasets, based on the similarity between the authors of datasets.

Based on this intuition, we have the following definition of link prediction based on authors with graph embedding.

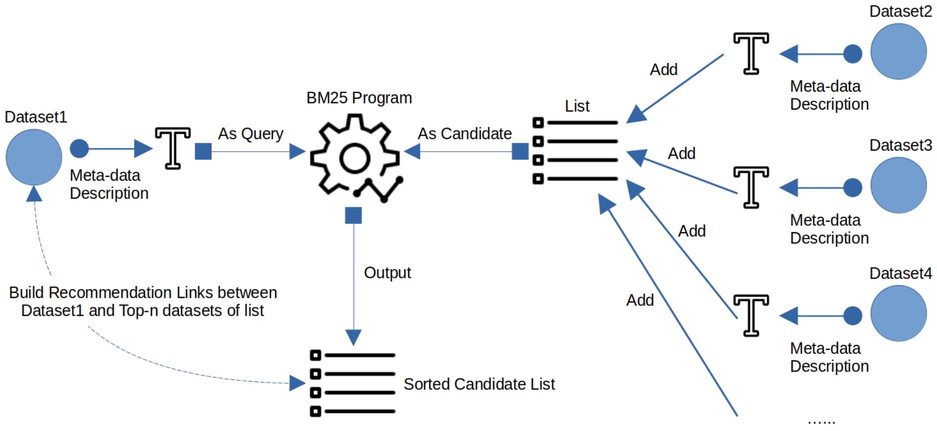

Figure 3. Dataset Recommendation based on BM25

Definition 6 (Link Prediction for Author-based Graph Embedding)**.** *Let $A = \{a_1, a_2, ...\}$ be a set of authors. Let Graph be an academic graph. Let VS_{Graph} be the vector space of the graph embedding of Graph. Let T be a threshold for link prediction, which means that we will predict a link between two authors when the cosine-similarity between the vectors of two authors is bigger than this threshold. We have a function $Sim : A \times A \to [0,1]$ such that for $a_i, a_j \in A$, in vector space VS_{Graph}, $Sim(a_i,a_j) = CosSim(Vec(a_i),Vec(a_j))$. Then we have $Link_{predicted}(a_i,a_j) \leftrightarrow Sim(a_i,a_j) \geq T$, for $a_i, a_j \in A$.*

where $Vec(a_i)$ and $Vec(a_j)$ is the vector of author a_i and a_j in VS_{Graph}, respectively; $ConSim(Vec(a_i),Vec(a_j))$ is the cosine similarity of vector $Vec(a_i)$ and vector $Vec(a_j)$; $Link_{predicted}(a_i,a_j)$ means that there exists a predicted link between authors a_i and a_j.

Finally, we have the definition of dataset recommendation with graph embedding on an academic graph.

Definition 7 (Dataset Recommendation with graph embedding on an academic graph)**.** *Let $A = \{a_1, a_2, ...\}$ be a set of authors. Let $D = \{d_1, d_2, ...\}$ be a set of datasets. Let Graph be an academic graph. Let VS_{Graph} be the vector space of graph embedding on Graph. Dataset recommendation with graph embedding on Graph is a function $Rec^{Graph}_{embedding} : D \to 2^D$, such that for $d_i \in D$, $Rec^{Graph}_{embedding}(d_i) = \{d_j | a_i \in Author(d_i) \cap A, a_j \in Author(d_j) \cap A, Link_{predicted}(a_i,a_j), d_j \in D\}$.*

3.3. BM25 based approach

BM25 (also known as Okapi BM25) is a ranking function used by search engines to estimate the relevance of documents to a given search query in information retrieval [31]. BM25 uses IDF(inverse document frequency) to add weight to each keyword in the query. The documents will then be sorted by the keywords contained in each document to be ranked.

As already motivated in the introduction, in this paper we will only consider and treat the meta-data description (title and description) of one dataset as one document, without looking into dataset itself. This is because there is too much variety in the format of datasets. Our work is in contrast to for example [32], which said that the challenge of data reuse could also be applied to dataset search, including data in formats that are difficult or expensive to use. In practice, too much variety in the format of datasets would bring difficulty to recommendation in our work and would make it expensive to use the dataset itself. However, compared to the dataset itself, the metadata of the dataset is much easier to use in our work. We treat the meta-data description of given dataset as given document (query), and the meta-data descriptions of candidate datasets as candidate documents to rank. With the given document, BM25 can rank the candidate documents based on the meta-data description of the given dataset. The definition of the meta-data description is as follows.

Definition 8 (Meta-data Description). *Let $D = \{d_1, d_2, ...\}$ be a set of datasets. The meta-data description of dataset is a function $meta-data : D \rightarrow String$, which means that for $d \in D$, $meta-data(d) = Title(d) \cup Description(d)$ where $Title(d)$ is the title of dataset d and $Description(d)$ is the description (or abstract) of dataset d.*

We also have the definition of BM25 based on the meta-data description of datasets.

Definition 9 (BM25 on Meta-data Description of Datasets). *Let $D = \{d_1, d_2, ...\}$ be a set of datasets. Let $meta-data : D \rightarrow String$ be a function to get the meta-data description of a dataset. Then we define the function $BM25_{Dataset} : D \rightarrow 2^D$, such that for $d_i \in D$ $BM25_{Dataset}(d_i) = \{d'_1, d'_2, ...\} \subseteq D$, where $Score_{BM25}(d_i, d'_1) \geq Score_{BM25}(d_i, d'_2) \geq ...$; where $Score_{BM25}(d_i, d'_1)$ is the BM25 score of d'_1 for query dataset d_i.*

We can now give the definition of dataset recommendation with BM25 on meta-data descriptions of datasets.

Definition 10 (Dataset Recommendation with BM25 on Meta-data Description). *Let $D = \{d_1, d_2, ...\}$ be a set of datasets. Let $BM25_{Dataset} : D \rightarrow 2^D$ be a function of using BM25 for dataset ranking. Let T_{BM25} be the threshold for BM25 ranking on datasets, which means that we only consider the datasets that appear in the top-T_{BM25} of a BM25-sorted dataset set. Then we have a function $Rec_{BM25} : D \rightarrow 2^D$, such that for $d_i \in D$, $Res_{BM25}(d_i) = \{d_j | d_j \in BM25'_{Dataset}(d_i)\}$, where $BM25'_{Dataset}(d_i) \subseteq BM25_{Dataset}(d_i)$ and $BM25'_{Dataset}(d_i) = \{d'_1, d'_2, ..., d'_{T_{BM25}}\}$.*

Note that this threshold is less than or equal to the size of the dataset set returned by BM25 based on the query dataset, which means that $T_{BM25} \leq BM25_{Dataset}(d_i)$ in Definition 10.

4. Dataset Recommendation Algorithms

In this section, we will provide the algorithms based on the recommendation approaches introduced in previous section.

4.1. Recommendation Algorithm with Co-author Network

The first recommendation algorithm uses dataset recommendation approach based on co-author network.

Algorithm 1: Graph walk in co-author network: $GW(A_S, G, n)$

```
Input  : Seed author $A_S$ to start the walk from.
         Co-author graph $G$.
         Hop number $n$ is the max shortest-path between $A_S$ and any author on
         a walk in graph $G$.
Output: A set of authors from $G$, denoted by $L_A$
1 $L_A \leftarrow \emptyset$;
2 foreach $Author\ A_C \in G$ do
3     Shortest path between $A_S$ and $A_C$ in $G$: $SPath_G(A_S, A_C)$;
4     if $SPath_G(A_S, A_C) \leq n$ then
5         Add $A_C$ to $L_A$: $L_A \leftarrow A_C$;
6     end
7 end
8 return $L_A - A_S$;
```

Algorithm 2: Dataset recommendation with graph walk: $DR_{GW}(D_G, L_D, G, n)$

```
Input  : Given dataset $D_G$ as the source author to be walked from.
         A list of candidate datasets for recommendation $L_D$.
         A co-author graph $G$.
         The hop number $n$ as the max shortest-path between the seed author
         and any authors in graph $G$.
Output: A set of recommended datasets, denoted by $L_{RD}$
$L_{RD} \leftarrow \emptyset$;
foreach $Author\ A\ in\ D_G$ do
    Get all reachable authors $L_{WA} = GW(A, G, n)$;
    foreach $WA \in L_{WA}$ do
        $L_{RD} \leftarrow \{D | WA \in author(D), D \in L_D\}$;
    end
end
return $L_{RD}$
```

We first explain how to perform a graph walk in a co-author network with Algorithm 1. For this algorithm, we need inputs that contain the max. hop number in addition to the starting authors (seed authors) and the network itself. The max-hop number nn is used to limit the maximum distance of our graph walk from the seed author. The maximum distance mentioned here refers to the shortest distance between the seed author and the target author in the co-author graph. Note that, all authors within distance n are treated equally as candidate recommendations, and are not ranked based on distance. In later experiments, we iterate over different versions of n to measure the effect of distance in the co-author network. Lines 3-6 of the Algorithm 1 are about how to use the hop number to limit the result of the graph walk.

Algorithm 2 shows our first algorithm for dataset recommendation. In this algorithm, we first perform a graph walk in the co-author network separately for all authors contained in a given dataset. The graph walks for authors here all respect the given max-

Algorithm 3: Graph walk + Author Embedding in Co-author Network: $GW\&AE(A_S, G, n, VS, T)$

```
   Input : Seed author $A_S$ is the source author to be walked from.
           Co-author graph $G$.
           Hop number $n$ is the max shortest-path between $A_S$ and any authors
           under walks in graph $G$.
           Vector space $VS$ contains vectors of every author in $G$.
           Threshold $T$ for cosine similarity between two vectors of authors.
   Output: A list of authors from $G$, denoted by $L_A$
1  $L_A \leftarrow \emptyset$;
2  foreach $Author\ A_C \in G$ do
3      Compute $SPath_G(A_S, A_C)$;
4      Compute $VS(A_S)$;
5      Compute $VS(A_C)$;
6      if $SPath_G(A_S, A_C) \leq n$ then
7          if $Cosine_{Sim}(VS(A_S), VS(A_C)) \geq T$ then
8              Add $A_C$ to $L_A$: $L_A \leftarrow A_C$;
9          end
10     end
11 end
12 return $L_A - A_S$;
```

hop number. We then find the datasets corresponding to these authors found by the graph walk, and these are the datasets to be recommended.

4.2. Recommendation Algorithm by Combining Co-author Network with Author Embedding

We will now introduce the dataset recommendation algorithm based on the co-author network approach combined with the author embedding approach. The author embedding is based on the knowledge graph embedding approach which was introduced in Definition 7. In contrast to the previous algorithm, this algorithm not only uses the graph walk but also the vector similarity in the vector space. As vector space we use the pre-trained entity embedding provided by MAKG[5], where the embedded entities are authors, publications, journals and conferencse. We use this pre-trained entity embedding to do link prediction (i.e. compute the similarity) between entities represented in the vector space.

Algorithm 3 shows how to combine the co-author network approach with the approached based author similarity in the entity embedding. For all authors obtained from the graph walk, we additionally compute the similarity between these authors and the seed author in the pretrained vector space. Then we select authors whose vector similarity is higher than the threshold we give. This is as mentioned in lines 5-9 of the Algorithm 3.

After introducing the graph walk and vector similarity combination algorithms, we will introduce our second dataset recommendation method, as shown in Algorithm 4.

[5] `https://makg.org/entity-embeddings/`

Algorithm 4: Dataset recommendation with graph walk and author embedding: $DR_{GW\&AE}(D_G, L_D, G, n, VS, T)$

```
   Input  : Dataset $D_G$ is the source author to be walked from.
            A list of candidate datasets $L_D$.
            Co-author graph $G$.
            Hop number $n$ is the max shortest-path between seed author and any
            authors under walks in graph $G$.
            Vector space $VS$ which contains vectors of every author in $G$.
            Threshold $T$ for cosine similarity between two vectors of authors.
   Output: A list of recommended datasets, denoted by $L_{RD}$
1  $L_{RD} \leftarrow \emptyset$;
2  foreach Author A in $D_G$ do
3      Get all walked authors $L_{WA} = GW\&AE(A, G, n, VS, T)$;
4      foreach $WA \in L_{WA}$ do
5          $L_{RD} \leftarrow \{D | WA \in D, D \in L_D\}$;
6      end
7  end
8  Duplicate $L_{RD}$;
9  return $L_{RD}$
```

Different from Algorithm 2, Algorithm 4 requires the additional input of the MAKG pre-trained vector space VS and a threshold T for vector similarity. Another difference is that for the SEED author, the return result of our co-author network will refer to Algorithm 3, which means that not only the graph walk is considered, but also the vector similarity. Because it applies more restrictions, this algorithm will result in less output than Algorithm 2.

4.3. Recommendation Algorithm by Combining Co-author Network, Author Embedding and BM25

We will now discuss the dataset recommendation algorithm by using the combination of graph walk, author embedding and BM25 approach. This algorithm is based on Definition 10, which uses the descriptions of datasets for dataset ranking.

In Algorithm 5, $TitleDes(D_G)$ denotes the title and description of the given dataset D_G; and $L_{RD}[0 : T_{BM25}]$ means the top-T_{BM25} list of L_{RD}, where T_{BM25} is the threshold for BM25. Also, as shown in Algorithm 5, we give BM25 a threshold T_{BM25} to determine how many datasets we will consider in the top of the ranked list.

After introducing the algorithm that uses BM25 for dataset ranking, we will introduce our third dataset recommendation algorithm, which is a combination of co-author network, author embedding and dataset ranking, as shown in Algorithm 6.

In Algorithm 6, line2-7 is same as the steps in Algorithm 4, for using a graph walk and author embedding to find a list of datasets with the help of co-author network. After that, we use BM25 to rank and filter the list of obtained datasets, as shown in line 8 of Algorithm 6. Finally our algorithm returns a filtered list of datasets as the recommended datasets for the given datasetD_G.

Algorithm 5: Dataset ranking with BM25: $DR_{BM25}(D_G, L_D, T_{BM25})$

```
Input  : Given dataset $D_G$ is the seed author to be walked from.
         A list of candidate datasets $L_D$.
         Threshold $T_{BM25}$ for BM25: the max size of the list returned by BM25.
         [Optional] Argument $k1$ for BM25.
         [Optional] Argument $b$ for BM25.
Output: A list of ranked datasets, denoted by $L_{RD}$
$L_{RD} \leftarrow \emptyset$;
Default $BM25.k1 = 1.2$;
Default $BM25.b = 0.75$;
if $k1$ is given then
    $BM25.k1 = k1$
end
if $b$ is given then
    $BM25.b = b$
end
$BM25.query = TitleDes(D_G)$;
$BM25.document = \{TitleDes(D) | D \in L_D\}$;
Run $BM25$ ranking;
$L_{RD} = BM25.result$;
Duplicate $L_{RD}$;
Sort $L_{RD}$;
if $Size(L_{RD}) > T_{BM25}$ then
    $L_{RD} = L_{RD}[0 : T_{BM25}]$
end
return $L_{RD}$
```

5. Dataset

We will now introduce the experimental data we used in the recommendation experiments we performed to evaluate the algorithms above.

5.1. Mendeley Data

Elsevier provided a very large and popular dataset search engine, Mendeley Data [6], containing more than 20 million datasets[7] from different kinds of data repositories (such as Zenodo[8]).

Each Mendeley dataset contains several types of metadata: descriptive metadata (e.g. title, description, authors and ID), administrative metadata (e.g. creation date) and legal metadata (e.g. dataset creators and public licence), shown in Table 1. What will be covered in this paper is the descriptive metadata, where the title and description are used for the BM25 method, and title is used to match with datasets from other sources.

[6] https://data.mendeley.com/

[7] https://data.mendeley.com/research-data/?repositoryType=NON_ARTICLE_BASED_REPOSITORY

[8] https://zenodo.org/

Algorithm 6: Dataset recommendation with graph walk, author embedding and dataset ranking: $DR_{GW\&AE+DRank}(D_G, L_D, G, n, VS, T, T_{BM25})$

```
Input  : Given dataset $D_G$ is the source author to be walked from.
         A list of candidate datasets $L_D$.
         Co-author graph $G$.
         Hop number $n$ is the max shortest-path between seed author and any
         authors under walked in graph $G$.
         Vector space $VS$ which contains vectors of every author in $G$.
         Threshold $T$ for cosine similarity between two vectors of authors.
         Threshold $T_{BM25}$ for BM25: Max size of the list returned by BM25.
Output: A set of recommended datasets, denoted by $L_{RD}$
$L_{RD} \leftarrow \emptyset$;
foreach Author A in $D_G$ do
    Get all walked authors $L_{WA} = GW\&AE(A, G, n, VS, T)$;
    foreach $WA \in L_{WA}$ do
        $L_{RD} \leftarrow \{D | WA \in Author(D), D \in L_D\}$;
    end
end
$L_{RD} = DR_{BM25}(D_G, L_{RD}, T_{BM25})$;
return $L_{RD}$
```

Table 1. The metadata types contained in Mendeley dataset, ScholeXplorer dataset and MAKG dataest.

Contained Metadata		Mendeley Dataset	ScholeXplorer Dataset	MAKG Dataset
Descriptive Metadata	Title	Yes	Yes	Yes
	DOI/URL	Some	Some	Some
	Description	Yes	Yes	Yes
	Author	Some	Some	Yes
Administrative Metadata	Creation date	Yes	Yes	Yes
	Publication date	Yes	Yes	Yes
Legal Metadata	Dataset creator	Some	Yes	Some
	licence	Yes	Yes	Yes

5.2. ScholeXplorer

ScholeXplorer[9] is a huge database containing datasets and literature objects as well as links between them. All the links between literature and dataset objects or between dataset objects are provided by data sources, and these can be considered as trusted links of high quality.

Different from Mendeley Data, ScholeXplorer stores data in the format of dataset-pairs. For each pair, it contains one link and two datasets. For each dataset, it also has descriptive metadata, administrative metadata and legal metadata, shown in Table 1.

The datasets from ScholeXplorer are used as candidate recommendation datasets in this paper. The links of each pair contained in ScholeXplorer are used as a gold stan-

[9] `https://scholexplorer.openaire.eu`

dard evaluation criterion for our recommendation algorithms. We also use the title and description for BM25-based dataset ranking approach.

5.3. Microsoft Academic Knowledge Graph

Microsoft Academic Graph (MAG) is a huge graph containing information on research outcomes (publications and datasets) and researchers, and the relationships between these [33]. Microsoft academic knowledge graph (MAKG) is a large RDF knowledge graph based on Microsoft Academic Graph [34]. MAKG contains information on publications, authors, indexes, journals, institutions, etc., as well as their scholarly relationships with each other. MAKG also provides an NTriple RDF dump[10], a SPARQL Endpoint and a pre-trained entity embedding to allow other researchers to use MAKG more easily. In Table 1, we show the metadata contained in MAKG dataset.

In this paper we will use the triples with the 'creator' predicate from MAKG as co-author network, since these triples show the creator relationship between authors and research outcomes. We also use title of MAKG datasets for matching with datasets from different sources. For author embeddings, we will use the pre-trained entity embedding[11] from MAKG.

5.4. Co-Author Network based on MAKG

We build our author network with the help ofthe MAKG academic graph, using the MAKG data-set [34]. The schema[12] of MAKG gives us the possibility of using the create-link relationship between author and paper to build a co-author network. Then this co-author network will represent whether two authors are co-author in same paper.

For instance, if we take two RDF triples from MAKG:

```
<:100000002> <http://purl.org/dc/terms/creator> <:1885406747> .
<:100000002> <http://purl.org/dc/terms/creator> <:2756955588> .
```

where each triple means the create-relationship between paper (subject) and author (object). Then we use the "Construct" function of SPARQL[13] to build a co-author triple. The SPARQL query is:

```
PREFIX dct: <http://purl.org/dc/terms/>
Construct{
  ?author1 <:hasCoAuthor> ?author2.
  ?author2 <:hasCoAuthor> ?author1.
}
WHERE {
  ?paper dct:creator ?author1.
  ?paper dct:creator ?author2.
}
```

This gives us the co-author triples:

[10] https://makg.org/rdf-dumps/
[11] https://makg.org/entity-embeddings/
[12] https://makg.org/schema-linked-dataset-descriptions/
[13] https://www.w3.org/TR/rdf-sparql-query/

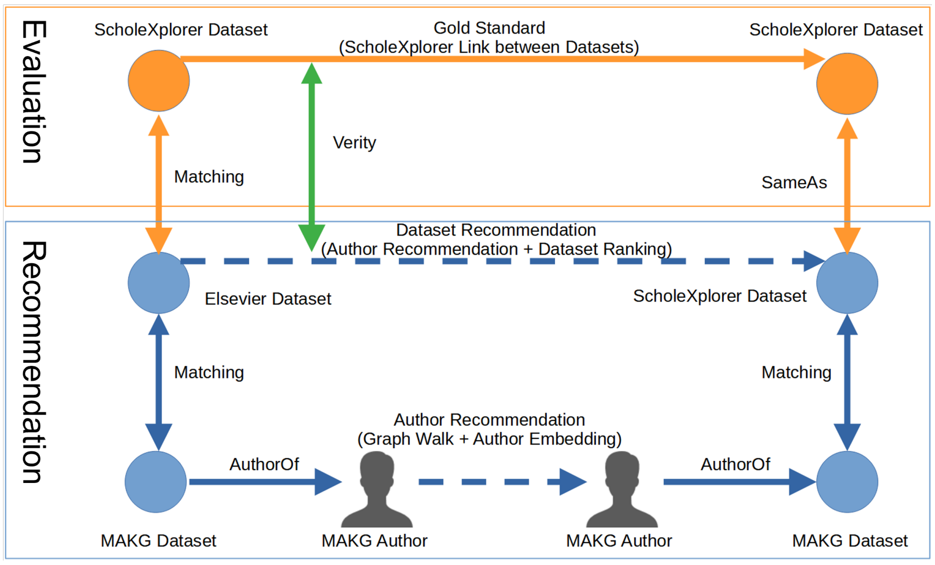

Figure 4. The overviwe of dataset recommendation and evaluation in our experiments.

```
<:1885406747> <:hasCoAuthor> <:2756955588> .
<:2756955588> <:hasCoAuthor> <:1885406747> .
```

With the help of this SAPRQL query, we create the co-author network from the SPARQL endpoint of all the MAKG author-paper triples. We have made this co-authors graph available online[14].

6. Experimental Design and Validation

In this section we will introduce our experimental design and the validation of our gold standard.

First off, we will give a brief introduction of how dataset recommendation based on ensemble methods with co-author network works in our experiments, which is shown in Figure 4. Our dataset recommendation experiment is to recommend datasets from ScholeXplorer source for given Mendeley datasets. We use a matching approach to match Mendeley datasets and ScholeXplorers datasets with MAKG datasets. Then we recommend target MAKG datasets for seed MAKG datasets, based on dataset authors by using the co-author-based recommendation approach. We then combine the author recommendation approach with the dataset ranking approach to do the final dataset recommendation.

6.1. Experiment Validation

Here we will introduce the gold standard for evaluation and how to evaluate our different recommendation algorithms in the experiments.

[14] https://api.krr.triply.cc/datasets/XuWang/MAKG/services/MAKG-CoAuthor/sparql

6.1.1. Gold Standard

The gold standard we use is from ScholeXplorer. ScholeXplorer provides links between datasets, as well as links between dataset and paper. These links are from providers of datasets, data centers or organizations that provide data storage and management, such as CrossRef, DataCite, and OpenAIRE. So these links between datasets are convincing to be used as the gold standard for our evaluation.

6.1.2. Evaluation for Dataset Recommendation

For a given dataset, our evaluation takes into account two lists: the list of datasets returned by the recommendation algorithm, and the list of datasets that are linked to the given dataset in the gold standard. Then we use the F1-measure to evaluate it. The intersection of the two lists we mentioned is the true positive in the F1-measure. The list of datasets returned by the recommendation algorithm is the predicted condition positive in the F1-measure, and the list of datasets that are linked to the given dataset in the gold standard is the Actual condition positive in the F1-measure. We then can calculate the recall, precision and F1-score with these, and obtain the evaluation results of dataset recommendation approach. using the standard mathematical formulae for F1-score:

$$TP = output \cap gold \tag{1}$$

$$TN = complement(output) \cap complement(gold) \tag{2}$$

$$FP = output \cap complement(gold) \tag{3}$$

$$FN = complement(output) \cap gold \tag{4}$$

$$recall = \frac{|output \cap gold|}{|gold|} \tag{5}$$

$$precision = \frac{|output \cap gold|}{|output|} \tag{6}$$

$$F1 = 2 \times \frac{precision \times recall}{precision + recall} \tag{7}$$

where $output$ is the output of recommendation approach, $gold$ the gold standard, $complement(output)$ the complement of output.

6.1.3. Matching between Datasets from Different Sources

In this part, we will introduce the approach to match datasets from different sources or in different format.

As we discussed before, we will use the co-author network of MAKG to construct a recommendation pathway from a Mendeley dataset to a ScholeXplorer dataset (shown in Figure 4). Also for our gold standard, we will match Mendeley datasets with ScholeXplorer datasets to obtain the gold standard recommendation links. These two task all require the matching between datasets from different sources. We must therefore use an approach for matching such datasets from different sources, and we will use a simple approach for this, based on matching titles of the datasets:

Table 2. Results of evaluating heterogeneous datasets matching approach.

Same content	Count	Percentage
DOI/URL + Title	16,286	60.48%
Author + Title	5,822	21.62%
Publisher + Title	272	1.01%
Count Number of Author + Title	1,391	5.17%
SUM	23,771	88.28%
Only Title (100% baseline)	26,928	100%

Definition 11 (Matching between Datasets from Different Sources)**.** *Given two datasets D_{S1} and D_{S2} from different sources $S1$ and $S2$, where $S1 \neq S2$. D_{S1} and D_{S2} can be matched, denoted as $match(D_{S1}, D_{S2})$ (same as $match(D_{S2}, D_{S1})$), if and only if $title(D_{S1}) = title(D_{S2})$, where $title(D_{S1})$ is the title of dataset D_{S1} and $title(D_{S2})$ is the title of dataset D_{S2}.*

To check if this matching approach performs well, we set up an experiment to evaluate the quality and performance of this matching approach, which will give us an indication of the quality of our gold standard recommendation links. We select 26,928 pairs of datasets, where

- for each pair of datasets, one from ScholeXplorer source and the other from MAKG source, both of them have same title;
- for each dataset in these pairs, it must contain (at least one) DOI or URL in its metadata.

We also set four baselines for evaluating the matching approach:

- (Strong baseline) Two datasets are the same if they have the same title and (at least one) same DOI or URL.
- (moderately strong baseline) Two datasets are the same if they have the same title and their authors are the same.
- (Likely weak baseline) Two datasets are the same if they have the same title and their publishers are the same.
- (Weak baseline) Two datasets are the same if they have the same title and the count numbers of their authors are the same.

The strong baseline, requiring the same DOI or URL, will give a very strong matching link between two datasets. This is because DOIs and URLs are used as identifiers of datasets. Note however that DOIs/URLs are not a perfect key to match two datasets or papers. This is because,although two different datasets or papers will rarely if ever have the same URI or DOI, a single dataset or paper will often have multiple DOIs. For example, the DOIs `http://dx.doi.org/10.2139/ssrn.997829` and `https://doi.org/10.13016/epy6-dyne` are two entities with the same title, abstract, publication date, and authors but different DOIs. Weakening the criteria somewhat to the moderately strong baseline, it is plausible that two datasets are equal if they have same title and authors. Similarly, it is plausible to consider two datasets equal if they have the same title and publisher. Only the weak baseline (with equal title and number of authors) is potentially unconvincing.

We have therefore tested the different matching criteria on a set of 26,928 pairs from ScholeXplorer and MAKG that have at least one DOI or URL each, and that share the same title, with the results shown in Table 2. We find that 60% of pairs meet the strong baseline which means that the datasets share at least one DOI or URL when they have same title. And if we consider all the plausible baselines (i.e. all baselines except the weak baseline), twe have about 83% pairs which is counted in correct matching. This result means that our approach for matching datasets from different sources is convincing in most cases.

6.2. Experiments Design

Here we will introduce the real-world data we used in our experiments.

Co-author Network The co-author network we used is built with 130,638,555 papers from MAKG, where each of these papers contains at least two authors. (There are a further 107,993,471 single-authored MAKG papers, which cannot help us provide the co-author relationship between authors). As said above, the links between authors here are based on the co-authorship relationship between the authors, which means that if two authors have a co-authorship, then there will be a link between those two authors.

List of Given Datasets We use 2370 datasets from Mendeley Data. These were obtained by first randomly selecting 1 million datasets from Mendeley, and then selecting all those that can be matched against MAKG datasets with our matching approach discussed above.

Gold Standard We match each of the 2370 given dataset with ScholeXplorer datasets, and count all the datasets linked to matched ScholeXplorer datasets as Gold Standard datasets. we found a total of 38,655 datasets in ScholeXplorer that have a gold standard link to any of the 2370 given datasets.

List of Candidate Datasets We use 28,981 of these ScholeXplorer datasets as candidate datasets. All these ScholeXplorer datasets are the ones which are not only matched to MAKG datasets but are also contiained in the Gold Standard. We have used here only these datasets from the gold standard as alternative datasets, as they already meet our requirements for doing experiments:

- They guarantee the possibility that we recommend the correct linked dataset: for each given dataset the linked dataset from the gold standard is in the candidate dataset.
- We also have ”noisy” datasets: for each given dataset, the datasets linked in the gold standard for the other given datasets can be considered as ”noisy” datasets,
- Each candidate dataset can be found by the co-author network recommendation method: all candidate datasets can be matched to the MAKG, which ensures that the MAKG-based co-author network can potentially find the datasets by walking the author graph through them to find the authors of the datasets.

6.3. Experimental Set-up

Here we will introduce our experiments. Our experiments are based on the previously proposed recommendation algorithms for the dataset. Our experiments are step-by-step incremental, meaning that we start with just using the co-author network, we then add other methods step by step, finally the full ensemble method. For experiments using only graph walk, we use 1-hop to 3-hop walking settings for the graph walk in the co-author network, respectively. For experiments using graph walk and author embedding for recommendations, we set minimal thresholds for the cosine-similarity between the author embeddings from 0.3 to 0.7, increasing these in steps of 0.1 each time, while keeping the same graph walk settings. We did not investigate similarity thresholds of 0.1 and 0.2: requiring only such a low similarity demand means that there is almost no benefit from having the embedding. The reason for dropping minimal similarity thresholds of 0.8 and up is that these lead to very small answer sets. Finally, for the recommendation experiments considering the dataset ranking with BM25, we gave BM25 thresholds of two and three times the number of gold standard results, respectively. This means that if for a given dataset we can find n gold standard linked datasets, then the threshold of BM25 is $2n$ or $3n$ respectively.

This gives us three different types of experiments: GraphWalk based experiments, GraphWalk+Embedding based experiments, and GraphWalk+Embedding+BM25 based experiments.

1. *GW*: **Graph walk based experiments:** We use 1-hop, 2-hop and 3-hop walks to walk through the co-author network to find relevant authors. Then we find the dataset in ScholeXplorer, which have the authors matched to the found authors. The purpose of this experiment is to test whether using only graph walks for dataset recommendation can make the recall acceptable (although it will make the precision low).
2. *GW&AE*: **GraphWalk+Embedding based experiments:** We not only use 1-hop, 2-hop and 3-hop graph walks, but also use author embeddings in vector space to calculate the cosine similarity between author vectors, and we select only authors that meet the similarity threshold (from 0.3 to 0.7). The purpose of this experiment is to test whether combining graph walks and author embedding methods for dataset recommendation can make precision acceptable (compared to the method that uses only graph walk).
3. *GW&AE&DRank*: **GraphWalk+Embedding+BM25 based experiments:** Here we follow the previous 1 to 3 hops of graph walking plus author similarity in vector space (and its threshold), while we add the ranking method BM25 to help us filter the recommended datasets. As discussed, the thresholds we use for BM25 are $2n$ and $3n$. The purpose of this experiment is to test whether the addition of the BM25 dataset ranking method can further improve the accuracy as measured by the F1 score.

7. Results and Analysis

After the above introduction of our experimental design and validation, we will analyse and discuss the results of our experiments in this section.

7.1. Results of Experiments

Tables 3, 4, 5 and 6 show all the results for the three types of experiments. Through these three experiments we find that the recommendation algorithm that combines graph walk, author embedding and dataset ranking methods performs best and reached the maximum

Table 3. Results of *GW* experiments

Hop	Recall	Precision	F1
1	0.1986	0.11687	**0.14715**
2	0.27468	0.01254	0.02399
3	0.38137	0.00257	0.00511

Table 4. Results of *GW&AE* experiments

Hop	Threshold T	Recall	Precision	F1
1	0.3	0.1883	0.12278	0.14864
1	0.4	0.16233	0.15235	0.15718
1	0.5	0.11517	0.17389	0.13856
1	0.6	0.0136	0.17716	0.02527
1	0.7	0.00095	0.10946	0.00189
2	0.3	0.26498	0.01375	0.02615
2	0.4	0.23965	0.01854	0.03442
2	0.5	0.19736	0.04231	0.06968
2	0.6	0.17446	0.18796	**0.18096**
2	0.7	0.15293	0.29247	**0.20084**
3	0.3	0.36768	0.00386	0.00765
3	0.4	0.32404	0.00359	0.00711
3	0.5	0.23469	0.00726	0.01408
3	0.6	0.17865	0.04972	0.07779
3	0.7	0.17112	0.28017	**0.21247**

Table 5. Results of *GW&AE&DRank* ($T_{BM25} = 2n$) experiments.

Hop	Threshold T	Recall	Precision	F1
3	0.3	0.08068	0.07671	0.07865
3	0.4	0.08839	0.08770	0.08805
3	0.5	0.10932	0.12686	0.11744
3	0.6	0.14981	0.30454	0.20083
3	0.7	0.16592	**0.74244**	**0.27124**

Table 6. Results of *GW&AE&DRank* ($T_{BM25} = 3n$) experiments.

Hop	Threshold T	Recall	Precision	F1
3	0.3	0.08684	0.05741	0.06912
3	0.4	0.09683	0.06701	0.07920
3	0.5	0.11809	0.09891	0.10765
3	0.6	0.15573	0.25864	0.19441
3	0.7	0.16745	**0.71327**	**0.27123**

F1 score at hop-2 graph walk with an author embedding threshold of 0.7. We will now analyze the results of each experiment in detail.

First we discuss the results of the *GW* experiments, using only the graph walk. We can conclude from Table 3 that when we just use the graph walk for the recommendation task, the recall is good but the precision becomes unacceptably low as the number of hops grows. Consequently, also the F1 score will become very low as the number of hops grows. Since the precision in this experiment is particularly low (the low F1 score is also due to this reason), it is necessary to add more restrictions to reduce the list of recommended datasets for the output of the recommendation algorithm. Therefore, we need to look at the results of the later experiments with the addition of author embedding or BM25 to determine whether they are valid.

We then discuss the results of the *GW&AE* experiments, combining the graph walk with the author embedding, shown in Table 4. As mentioned in the experimental setup, the main purpose of this experiment is to improve the precision in the results of the previous experiment. First we see that the precision generally becomes higher when we do the hop-1 graph walk. Only when the author vector similarity threshold is 0.7, the precision becomes lower than the previous result. This is because at a threshold of 0.7, the set of recommended data to be found becomes very small. When the graph walk is hop-2 and hop-3, we can conclude that as the similarity threshold of the author vector increases, the recall becomes lower and the precision becomes higher, which makes the F1 score higher as well. This is because the rate of precision increase is greater than the rate of recall decrease. The maximum value of F1 in this experiment is 0.21247, which appears in the hop3 graph walk with the author's similarity threshold set to 0.7.

However, we still find one problem in Table 4: when performing the hop-3 graph walk, the original precision (experimental results in Table 4) was too low, resulting in the author's vector similarity threshold of 0.3 to 0.6 failing to raise the precision to an acceptable range, thus making the F1 score still very low. So we need to reduce the list of

Table 7. The size of returned datasets by recommendation approaches (the baseline=38,655 is the size of gold standard for all the seed datasets).

Hop1	T=0.3	T=0.4	T=0.5	T=0.6	T=0.7
T_BM25=0	59,283	41,187	25,602	2,969	338
T_BM25=2n	16,405	13,028	7,641	1,284	97
T_BM25=3n	18,610	14,396	8,246	1,424	121
Hop2	T=0.3	T=0.4	T=0.5	T=0.6	T=0.7
T_BM25=0	744,508	499,551	180,308	35,879	24,084
T_BM25=2n	34,606	31,130	22,245	11,461	8,218
T_BM25=3n	47,819	42,168	27,738	12,597	8,552
Hop3	T=0.3	T=0.4	T=0.5	T=0.6	T=0.7
T_BM25=0	3,675,576	3,480,196	1,249,505	138,890	23,610
T_BM25=2n	40,656	38,958	33,310	19,015	8,639
T_BM25=3n	58,474	55,859	46,150	23,275	9,075

recommended datasets found by the recommendation algorithm even further to improve the precision.

We therefore turn to the results of the *GW&AE&Drank* experiment shown in Table 5 and Table 6. The purpose of this experiment is to continue to improve the precision and thus the F1 score. For this purpose, we added the ranking method BM25 to this experiment. We used two different thresholds for BM25 to test whether different thresholds for the number of results returned by BM25 would have a significant effect on the results of our experiments. Table 5 shows the results for a BM25 threshold of $2n$, while Table 6 shows the results for a BM25 threshold of $3n$. The results shown in these two tables lead to the conclusion that adding BM25 will slightly reduce the recall, but both the precision and F1 scores will improve, with a maximum F1 score of 0.27124 and 0.27123 in experiments with BM25 thresholds is $2n$ and $3n$, respectively.

We also show the size of returned datasets per recommendation approach in Table 7 and Figure 5. As can be seen from the size of the returned datasets, our best precision approach (hop-3, T=0.7 and T_Bm25=2n,3n) only returns about 9,000 datasets (the baseline is 30,000, the size of gold standard). This means that our ensemble method enforces many restriction on the recommendation set (in order to remove as many ”noisy” datasets as possible) and thus cannot guarantee to return many results, thereby limiting the recall.

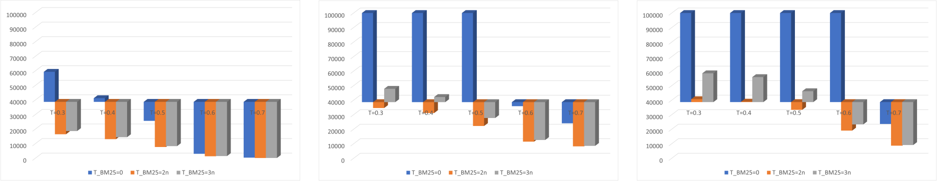

Figure 5. Comparison between baseline=38,655 (the size of gold standard) and the size of returned datasets by recommendation approaches: 1hop (left), 2hop (middle) and 3hop (right).

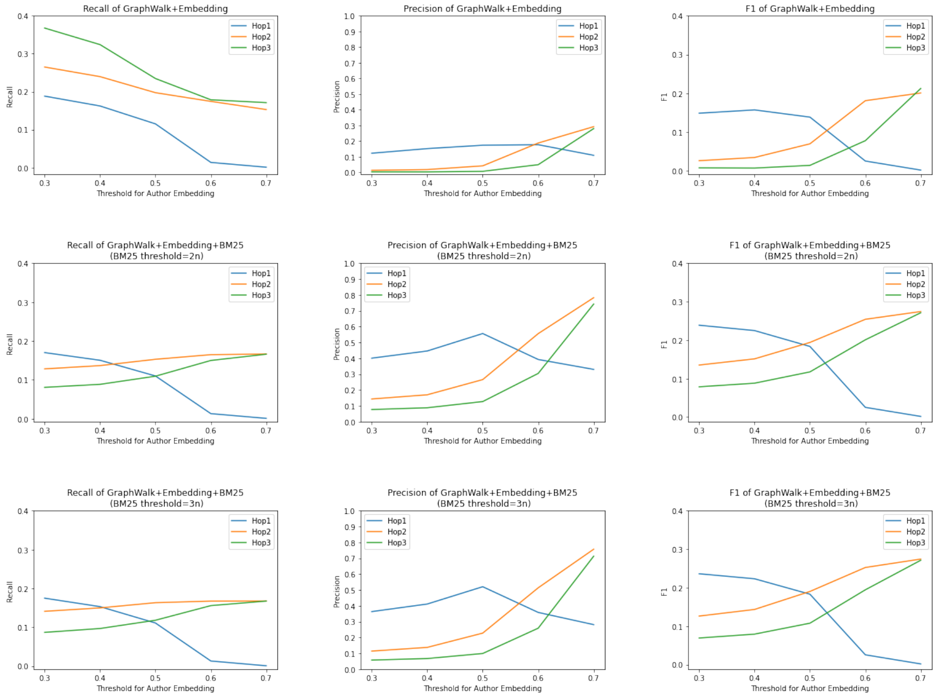

Figure 6. Recall (column 1), Precision (column 2) and F1 score (column 3) for $GW\&AE$ experiment (row 1), $GW\&AE + DRank$ experiemnt with $T_{BM25} = 2n$ (row 2) and $GW\&AE + DRank$ experiemnt with $T_{BM25} = 3n$ (row 3).

To be able to compare more intuitively the changes in recall, precision and F1 score of the results in Table 4, Table 5 and Table 6, we introduce Figure 6. The first row of Figure 6 shows the results of Table 4, the second row shows the results of Table 5, and the third row shows the results of Table 6.

By looking at the first column (recall), we can clearly see that the recall decreases after adding the BM25 method, especially when the author similarity threshold is 0.3 to 0.5. We can also see that the recall of hop-1 is higher than hop-2 and hop-3 for lower embedding threshold T after adding BM25. This is because the BM25 approach reduces relatively less the size of the returned datasets on hop-1 than on hop-2 and hop-3, which is shown in Table 7. Removing relatively more datasets means more returned gold standard datasets may be removed. And because recall is based on the size of the gold standard datasets in the returned datasets, this leads to the reason why recall is lower in hop-2 and hop-3.

Then, by looking at the second column, we can see that the precision has improved significantly after adding the BM25 method, and that the maximum value of precision can be increased from about 0.3 to about 0.8 for hop-2 and hop-3. For hop-1, we can see that the precision decreases when the embedding threshold increases. This is because the size of the returned dataset is very small at this point, with about one thousand returned at T=0.6 and only about one hundred returned at T=0.7. When the size of the returned dataset is so small, we cannot guarantee some conclusion will hold (e.g., the higher the embedding threshold, the higher the precision).

Thanks to the significant improvement in the precision, we can conclude from the third column that the F1 score will improve with the addition of the BM25 method.

We then proceed to analyze Figure 6 to see if the BM25 threshold affects the experimental results. By comparing the second and third rows, we can see that there is no significant change in recall, precision and F1 score for different BM25 thresholds. This shows that the BM25 threshold (i.e., the maximum number of returned data sets) does not affect our experimental results.

7.2. Analysis

We will now summrise the effect of the different algorithms, hops and thresholds by analysing all the aforementioned results.

The first effect concerns different algorithms:

- When using only the graph walk algorithm, the precision is too low, which means that the recommended datasets are too often not the ones in the gold standard.
- By adding author embedding similarity, the precision increases but is still not high, with some cost of recall.
- By adding BM25 ranking, the precision is good enough but recall is still low.

High precision means that most of datasets recommended by our algorithm are correct (as judged against the gold standard). This also means that our algorithm often returns useful datasets for users, but does succeed in return all useful datasets. This trade-off of a high precision against a low trade-off is similar to the behaviour of typical search engine.

Then we will discuss the effect of the similarity threshold for author embeddings. AHhgher threshold causes recall to go down, precision to go up, and F1 score to go up as well. This means that a high similarity threshold for author embeddings benefits our algorithm, again causing a trade-off of high precision against lower recall.

Finally, concerning the effect of multi-hops, Table 4, shows that a high hop count will cause an increase in recall, which means that our algorithm would cover more correct datasets from the gold standard.

8. Conclusion and Discussion

In this paper we have investigated the use of a co-author network in ensemble methods for scientific dataset recommendation. Our recommendation algorithm involves three methods: a graph walk in a co-author network, author similarity in a graph embedding, and the ranking of datasets based on textual descriptions. We used real-world open source data to experiment with and evaluate the recommendation algorithm. The final results confirm that when we combine all three methods and use the farthest possible graph walking distance and the most stringent threshold for the graph embedding similarity threshold, we can obtain high precision recommendation results, albeit at a low recall. This means that our ensemble method is able to recommend relatively good datasets but will not recommend all good datasets. This behaviour is similar to that of most widely used search engines.

After obtaining a reasonably high precision performance with our ensemble method, the next challenge will be to improve the recall performance. In this work, the upper bound of recall is based on the number of returned datasets following the co-author relationship between authors. In future work, we will use other information sources such the affiliation of the author, or the research domain of the author, or a citation-graph among authors to expand the set of potentially related authors without sacrificing precision.

Acknowledgements

This work was funded by Elsevier's Discovery Lab. This work was also funded by the Netherlands Science Foundation NWO grant nr. 652.001.002 which is also partially funded by Elsevier. The first author is funded by by the China Scholarship Council (CSC) under grant nr. 201807730060.

References

[1] Zuiderwijk A, Spiers H. Sharing and re-using open data: A case study of motivations in astrophysics. International Journal of Information Management. 2019;49:228-41.

[2] Chao TC. Disciplinary reach: Investigating the impact of dataset reuse in the earth sciences. Proceedings of the American Society for Information Science and Technology. 2011;48(1):1-8.

[3] Baumann P, Mazzetti P, Ungar J, Barbera R, Barboni D, Beccati A, et al. Big data analytics for earth sciences: the EarthServer approach. International Journal of Digital Earth. 2016;9(1):3-29.

[4] Eysenbach G, et al. Medicine 2.0: social networking, collaboration, participation, apomediation, and openness. Journal of medical Internet research. 2008;10(3):e1030.

[5] Mayer-Schönberger V, Ingelsson E. Big Data and medicine: a big deal?. Wiley Online Library; 2018.

[6] Gundersen OE, Kjensmo S. State of the art: Reproducibility in artificial intelligence. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 32; 2018. .

[7] Wilkinson MD, Dumontier M, Aalbersberg IJ, Appleton G, Axton M, Baak A, et al. The FAIR Guiding Principles for scientific data management and stewardship. Scientific data. 2016;3(1):1-9.

[8] Škorić L, Glasnović A, Petrak J. A publishing pandemic during the COVID-19 pandemic: how challenging can it become? Croatian medical journal. 2020;61(3):79-81.

[9] Brickley D, Burgess M, Noy N. Google Dataset Search: Building a search engine for datasets in an open Web ecosystem. In: The World Wide Web Conference; 2019. p. 1365-75.

[10] Isinkaye FO, Folajimi YO, Ojokoh BA. Recommendation systems: Principles, methods and evaluation. Egyptian informatics journal. 2015;16(3):261-73.

[11] Bai X, Wang M, Lee I, Yang Z, Kong X, Xia F. Scientific Paper Recommendation: A Survey. IEEE Access. 2019;7:9324-39.

[12] Fujita M, Inoue H, Terano T. Searching Promising Researchers through Network Centrality Measures of Co-author Networks of Technical Papers. In: 2017 IEEE 41st Annual Computer Software and Applications Conference (COMPSAC). vol. 2; 2017. p. 615-8.

[13] Daud A, Ahmad M, Malik M, Che D. Using machine learning techniques for rising star prediction in co-author network. Scientometrics. 2015;102(2):1687-711.

[14] Sun Y, Barber R, Gupta M, Aggarwal CC, Han J. Co-author relationship prediction in heterogeneous bibliographic networks. In: 2011 International Conference on Advances in Social Networks Analysis and Mining. IEEE; 2011. p. 121-8.

[15] Wang X, Huang Z, van Harmelen F. Evaluating Similarity Measures for Dataset Search. In: Huang Z, Beek W, Wang H, Zhou R, Zhang Y, editors. Web Information Systems Engineering – WISE 2020. Cham: Springer International Publishing; 2020. p. 38-51.

[16] Wang X, van Harmelen F, Huang Z. Biomedical dataset recommendation. In: Quix C, Hammoudi S, van der Aalst W, editors. Proceedings of the 10th International Conference on Data Science, Technology and Applications, DATA 2021. SciTePress; 2021. p. 192-9. 10th International Conference on Data Science, Technology and Applications, DATA 2021 ; Conference date: 06-07-2021 Through 08-07-2021.

[17] Färber M, Leisinger AK. In: Recommending Datasets for Scientific Problem Descriptions. New York, NY, USA: Association for Computing Machinery; 2021. p. 3014–3018. Available from: https://doi-org.vu-nl.idm.oclc.org/10.1145/3459637.3482166.

[18] Chen Y, Wang Y, Zhang Y, Pu J, Zhang X. AMENDER: An Attentive and Aggregate Multi-layered Network for Dataset Recommendation. In: 2019 IEEE International Conference on Data Mining (ICDM); 2019. p. 988-93.

[19] Ellefi MB, Bellahsene Z, Dietze S, Todorov K. Dataset recommendation for data linking: An intensional approach. In: European Semantic Web Conference. Springer; 2016. p. 36-51.

[20] Altaf B, Akujuobi U, Yu L, Zhang X. Dataset recommendation via variational graph autoencoder. In: 2019 IEEE International Conference on Data Mining (ICDM). IEEE; 2019. p. 11-20.

[21] Rabello Lopes G, Paes Leme LAP, Pereira Nunes B, Casanova MA, Dietze S. Two Approaches to the Dataset Interlinking Recommendation Problem. In: Benatallah B, Bestavros A, Manolopoulos Y, Vakali A, Zhang Y, editors. Web Information Systems Engineering – WISE 2014. Cham: Springer International Publishing; 2014. p. 324-39.

[22] Patra BG, Roberts K, Wu H. A content-based dataset recommendation system for researchers—a case study on Gene Expression Omnibus (GEO) repository. Database. 2020 11;2020. Baaa064. Available from: https://doi.org/10.1093/database/baaa064.

[23] Kato MP, Ohshima H, Liu YH, Chen HL. In: A Test Collection for Ad-Hoc Dataset Retrieval. New York, NY, USA: Association for Computing Machinery; 2021. p. 2450–2456. Available from: https://doi-org.vu-nl.idm.oclc.org/10.1145/3404835.3463261.

[24] Wang Q, Mao Z, Wang B, Guo L. Knowledge Graph Embedding: A Survey of Approaches and Applications. IEEE Transactions on Knowledge and Data Engineering. 2017;29(12):2724-43.

[25] Trouillon T, Welbl J, Riedel S, Gaussier E, Bouchard G. Complex Embeddings for Simple Link Prediction. In: Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48. ICML'16. JMLR.org; 2016. p. 2071–2080.

[26] Bordes A, Usunier N, Garcia-Durán A, Weston J, Yakhnenko O. Translating Embeddings for Modeling Multi-Relational Data. In: Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2. NIPS'13. Red Hook, NY, USA: Curran Associates Inc.; 2013. p. 2787–2795.

[27] Nickel M, Tresp V, Kriegel HP. A Three-Way Model for Collective Learning on Multi-Relational Data. In: Proceedings of the 28th International Conference on International Conference on Machine Learning. ICML'11. Madison, WI, USA: Omnipress; 2011. p. 809–816.

[28] Lin Y, Liu Z, Sun M, Liu Y, Zhu X. Learning Entity and Relation Embeddings for Knowledge Graph Completion. In: Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence. AAAI'15. AAAI Press; 2015. p. 2181–2187.

[29] Nickel M, Murphy K, Tresp V, Gabrilovich E. A Review of Relational Machine Learning for Knowledge Graphs. Proceedings of the IEEE. 2016 Jan;104(1):11–33. Available from: http://dx.doi.org/10.1109/JPROC.2015.2483592.

[30] Wang Q, Mao Z, Wang B, Guo L. Knowledge Graph Embedding: A Survey of Approaches and Applications. IEEE Transactions on Knowledge and Data Engineering. 2017;29(12):2724-43.

[31] Robertson S, Walker S, Jones S, Hancock-Beaulieu MM, Gatford M. Okapi at TREC-3. In: Overview of the Third Text REtrieval Conference (TREC-3); 1995. p. 109-26.

[32] Chapman A, Simperl E, Koesten L, Konstantinidis G, Ibáñez LD, Kacprzak E, et al. Dataset search: a survey. The VLDB Journal. 2020;29(1):251-72.

[33] Sinha A, Shen Z, Song Y, Ma H, Eide D, Hsu BJP, et al. An Overview of Microsoft Academic Service (MAS) and Applications. In: Proceedings of the 24th International Conference on World Wide Web. WWW '15 Companion. New York, NY, USA: Association for Computing Machinery; 2015. p. 243–246.

[34] Färber M. The Microsoft Academic Knowledge Graph: A Linked Data Source with 8 Billion Triples of Scholarly Data. In: Proceedings of the 18th International Semantic Web Conference. ISWC'19; 2019. p. 113-29. Available from: https://doi.org/10.1007/978-3-030-30796-7_8.