Noname manuscript No.
(will be inserted by the editor)

VeTo+: Improved Expert Set Expansion in Academia

Serafeim Chatzopoulos · Thanasis Vergoulis · Theodore Dalamagas · Christos Tryfonopoulos

Received: date / Accepted: date

Abstract Expanding a set of known domain experts with new individuals, sharing similar expertise, is a problem that has various applications, such as adding new members to a conference program committee or finding new referees to review funding proposals. In this work, we focus on applications of the problem in the academic world and we introduce VeTo+, a novel approach to effectively deal with it by exploiting scholarly knowledge graphs. VeTo+ expands a given set of experts by identifying scholars having similar publishing habits with them. Our experiments show that VeTo+ outperforms, in terms of accuracy, previous approaches to recommend expansions to a set of given academic experts.

Keywords Expertise retrieval · Expert finding · Scholarly knowledge graphs · Data mining.

1 Introduction

The problem of expanding a set of known domain experts with new individuals of similar expertise emerges in many real-life applications, many of which coming from the field of academia. As an indicative example, consider a conference organiser who attempts to add new members to its program committee or a funding agency officer seeking new referees to review funding proposals. Such problems provided motivation for the broad research field known as *expert finding* [13].

T. Vergoulis · Theodore Dalamagas
IMSI, "Athena" Research Center, Athens, 15125, Greece
E-mail: {vergoulis, dalamag}@athenarc.gr

S. Chatzopoulos · C. Tryfonopoulos
Univ. of the Peloponnese, Dep. of Informatics & Tel/tions, Tripoli, 22100, Greece
E-mail: {schatzop, trifon}@uop.gr

Early work in this field assumes that the person seeking for experts provides a set of keywords that describe the desired topics of expertise. The respective expert finding approaches (e.g., [4]) attempt to match these topics to experts by utilising the co-occurrences of topic keywords with person names in text corpora (e.g., websites, publications). Nevertheless, it is often difficult to explicitly express the desired topics as concrete sets of keywords. This motivated various methods (e.g., [5, 11]) which adopt the *query by example* approach: the name of a known expert is provided as input and, based on this, other individuals with similar profiles are sought. Usually, the profiles are constructed based on analysing existing text corpora; for instance, by applying language processing or topic modeling techniques. Although most such methods search for individuals that are similar to a single expert, some of them are also able to identify similarities to groups of experts [2].

Nevertheless, the effectiveness of all previously described approaches depends on the availability of concrete text corpora that contain information about the expertise of the individuals. In the case of academic experts, this means that these approaches rely on the availability of a large set of scientific publications. Unfortunately, the full texts of publications are often restricted behind paywalls and, thus, it is practically impossible to construct a concrete set of the relevant texts. Moreover, even if it was possible to construct a corpus containing an adequate number of relevant publications, its size would be vast and, thus, gathering and processing it in a regular basis would be a tedious and time-consuming task. This problem motivated the introduction of alternative methods that, instead, utilise *scholarly knowledge graphs* (e.g., [12]). In late years, due to the systematic effort of various developing teams, a

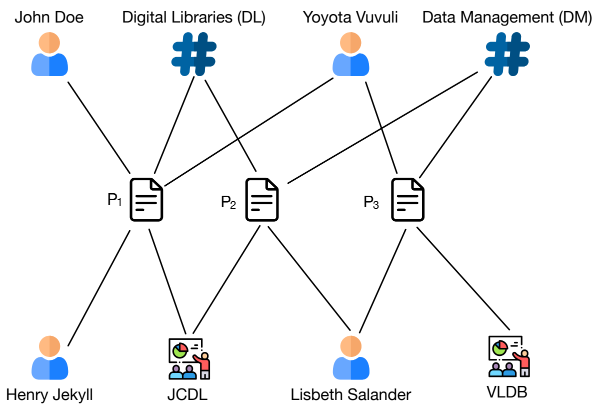


Fig. 1 An example scholarly knowledge graph including academics, papers, venues and topics.

variety of large scholarly knowledge graphs has been made available (e.g., the AMiner's DBPL-based datasets [30], the Open Research Knowledge Graph [14], the OpenAIRE Research Graph [19, 20]). These heterogeneous graphs correspond to a very rich and relatively clean source of information about academics, their publications and relevant metadata. Figure 1 presents an illustrative example of such a graph comprising academics, papers, venues, and topics.

In this context, in a previous work we introduced VeTo [32], a knowledge graph-based approach to deal with the problem of expanding a set of known experts with new individuals of similar expertise. VeTo is based on an advanced graph structure similarity technique [33], tailored for heterogeneous graphs, to identify similarities between researchers based on their publishing habits. In particular, it takes advantage of latent patterns in the way academics select the venues to publish their work and the topics of their respective publications.

Although VeTo was found to outperform competition, we have identified room for various improvements. As a result, in this work we present VeTo+, a new approach that extends VeTo achieving improved effectiveness in the context of the expert set expansion problem. VeTo+ improves upon its predecessor by (i) introducing a flexible weighting scheme, that enables different levels of attention to the respective similarity measures which are used by the approach (Section 3.2.1), (ii) supporting the use of an alternative metapath-based similarity (the 'focused' `APV`-based similarity, Section 3.2.2), which can provide more precise recommendations for some academic domains, and (iii) studying the effect of additional rank aggregation algorithms, some of which may result in improved effectiveness (Section 3.2.3).

The aforementioned extensions, in many cases, result in considerably improved performance and offer more options in fine-tuning the approach. According to our experiments, which follow an extended version of the experimental framework we introduced in our previous work, VeTo+ achieves significant improved accuracy in recommending expansions to a set of given academic experts, in comparison to its predecessor and other rival approaches.

In addition, we worked on extending the open dataset of expert sets, that we developed in our previous work, by adding two extra venues. Since our previous dataset included venues for only one discipline (namely 'Data Management') we have selected to include venues from another discipline ('Digital Libraries') so that to increase its multi-disciplinarity, allowing the generalisation of results, or reveal possible particularities that may exist in different disciplines.

2 Background

The focus of this work is on a specific expert finding problem applied in the academic world: to reveal, among a set of candidate researchers C, the n most suitable of them, to extend a set of known experts E_{kn}. We refer to this as the *expert set expansion* problem; it is also known as the *finding similar experts* problem (e.g., in [2]), however we prefer the former term, since it captures the notion with less ambiguity. This has various real-life applications like reviewer recommendation, collaborator seeking, new hire recommendation, etc.

For reasons elaborated in Section 1, we focus on approaches that exploit scholarly knowledge graphs to deal with the problem. *Knowledge graphs*, also known as *heterogeneous information networks* [28], are graphs that contain nodes and edges of multiple types. They represent entities (nodes) from a domain of interest and the different types of relationships between them (edges). For instance, consider the scholarly knowledge graph depicted in Figure 1. This graph contains information about three papers (P_1, P_2, P_3), their venues ($JCDL$, $VLDB$), their topics (DL, DM), and the academics that have authored them. In fact this is a simplistic, toy-example, since real-life scholarly knowledge graphs (like the Open Research Knowledge Graph or the OpenAIRE Research Graph) contain a larger variety of entity types (e.g., academic institutions, funding organisations, research projects).

Knowledge graphs capture rich information about their respective domains encoding not only direct relationships of the involved entities, but also more complex ones that correspond to larger paths in the graph. In particular, all paths that involve the same sequence of entity and edge types capture relationships of exactly the same semantics between their first and last nodes. These generalised path patterns are widely known as *metapaths* and we refer to the paths that follow these patterns as their *instances*. For example, in the graph of Figure 1, the paths `‘John Doe -` P_1 `- DL’` and `‘Yoyota Vuvuli -` P_3 `- DM’` both are instances of the metapath `‘Academic - Paper - Topic’` (or `APT`, for brevity) and both have the same interpretation: they relate an academic with a topic through a paper authored by them.

A recent line of work in knowledge graphs supports calculating the similarity of two entities (nodes) of the same type according to the particular semantics of a given metapath [26, 29, 33]. Such *metapath-based similarity measures* have been proven to be valuable in various applications, since they can be used to reveal latent similarities of entities taking into consideration complex and rich semantics which are captured in the structure of the respective graphs. This is why approaches that efficiently calculate them (e.g., based on dynamic programming [27]) provide a powerful framework to effectively exploit the information captured in knowledge graphs.

The main intuition behind metapath-based similarity measures is that they consider the way the nodes of the graph are connected, according to paths that are instances of the given metapath. For example, academics `John Doe` and `Henry Jekyll` in the graph of Figure 1 seem similar based on the topics of their published papers (i.e., based on the semantics of the `APT` metapath) since they both have only one paper connecting them to the `DL` topic (i.e., one `APT` instance) and no paper connecting them to the `DM` topic (i.e., 0 `APT` instances).

A well-known metapath-based similarity measure that follows the previous intuition is JoinSim [33]. Consider a knowledge graph $\mathcal{G}$ and one of its metapaths $m = F - \cdots - L$, where $F, \ldots, L$ are valid entity/node types in $\mathcal{G}$. In addition, let $\{f_1, \ldots, f_n\}$ and $\{l_1, \ldots, l_k\}$ be the sets of $\mathcal{G}$ nodes of type F and L, respectively. For each f_i (with $i \in [1, n]$), let $\mathbf{v^m_{f_i}}$ be a vector of size k, where $\mathbf{v^m_{f_i}}[j]$ corresponds to the number of m instances connecting f_i to l_j. Then, the JoinSim similarity of f_α and f_β according to m is given by the cosine similarity of the vectors $\mathbf{v^m_{f_\alpha}}$ and $\mathbf{v^m_{f_\beta}}$, i.e.,:

$$sim(f_\alpha, f_\beta, m) = \frac{\mathbf{v^m_{f_\alpha}} \cdot \mathbf{v^m_{f_\beta}}}{||\mathbf{v^m_{f_\alpha}}|| \; ||\mathbf{v^m_{f_\beta}}||} \quad (1)$$

The intuition of this formula is that the JoinSim similarity of f_α and f_β is large if they are connected with a comparable number of paths to a similar set of nodes of type L. Going back to the example of Figure 1, given the metapath `APT`, JoinSim first constructs for each academic a vector with the topics related to the papers they have authored, and then calculates similarity scores between the academics based on these vectors.

3 The VeTo+ Approach

In this section, we describe how VeTo+ deals with the expert set expansion problem; we first provide a detailed description of VeTo (Section 3.1), our previous approach, which provides the basis of VeTo+. Then, we elaborate on the improvements introduced on top of it by VeTo+, our current approach (Section 3.2). Finally, in Section 3.3, we discuss a few interesting remarks about both approaches.

3.1 Basic VeTo

VeTo's main intuition is that it considers the metapath-based similarity of academics to known experts to form a list of recommended candidates for a given expert expansion problem. In particular, two metapaths are utilised: `APT` and `APV`, each one capturing a distinct "publishing habit". The former takes into consideration the venues in which academics select to publish their articles, while the latter the topics of these articles. VeTo+ combines the similarities of academics according to these two metapaths to create the list of suggested experts.

Given a set of known experts E, a set of candidates C, and n the number of expansions to be made, VeTo performs the following steps:

1. For each expert $e \in E$, the top-k most similar academics based on the `APV` metapath are identified as candidates (k is an approach parameter). These metapath-based similarities are calculated according to the JoinSim measure[1] (Eq. 1). The ranked list $R^e_{APV} = \{c_1, c_2, ..., c_k | c_i \in C\}$ is produced, containing the candidates having the higher similarity scores with expert e, according to `APV`.
2. Given the set $\{R^e_{APV} | e \in E\}$ containing the ranked lists based on metapath `APV` for all experts $e \in E$, a rank aggregation algorithm is applied on it to produce R_{APV} the aggregated ranked list that ranks all candidates considering their similarities to all experts according to `APV` (in descending order).
3. A similar procedure to the one performed in Steps 1 & 2 is performed to produce the ranked list R_{APT}

[1] Following the definitions in Section 2, for the case of the `APV` metapath, the corresponding vectors have length equal to the number of distinct venues in the dataset. Of course, since these vectors are very sparse, in practice sparse vector representations can be used to reduce the memory footprint.

that ranks all candidates according to their "aggregated" similarity score based on the `APT` metapath in descending order (again, the JoinSim similarity, defined in Eq. 1, is used).
4. A rank aggregation algorithm is applied on ranked lists R_{APV} and R_{APT} to produce the final aggregated list R_{fin} that takes into account the similarities between experts and other academics based on both metapaths.
5. The top-n academics of R_{fin}, when sorted in descending order based on the aggregated score, constitute the final result to the given expert set expansion problem.

Regarding the rank aggregation algorithm required by various steps of the VeTo workflow, in theory, any such algorithm can be used (see Section 3.2.3). In our initial work [32] we have implemented and tested the following:

– *Borda Count (BC)*: Given $I = \{i_1, \ldots, i_k\}$, a set of items to be ranked, and $\{R_1, \ldots, R_m\}$, a set of alternative rankings of these items, where each ranking R_j is a set of k pairs (one for each $i \in I$) of the form $\langle i, s^i_j \rangle$, with $i \in I$ and s^i_j being i's ranking score according to R_j, Borda Count produces a new ranked list:

$$R_{BC} = \{ \ \langle i, s^i_{BC} \rangle : i \in I, \ \ s^i_{BC} = \sum_{j \in [1,m]} k - rank(i, R_j) + 1 \ \}$$

where $rank(i, R_j)$ is a function that returns i's rank (i.e., its order) based on the ranking R_j. Intuitively, the Borda Count score of an item i is determined based on the number of of items ranked lower than it according to all rankings.
– *Sum*: This is a simple algorithm that ranks items based on their ranking scores according to the individual rankings to be aggregated. Using the same notation as for Borda Count, this algorithm produces a ranked list:

$$R_S = \{ \ \langle i, s^i_S \rangle : i \in I, s^i_S = \sum_{j \in [1,m]} s^i_j \ \}$$

3.2 VeTo+ extensions

Although our experiments [32] showed that VeTo outperformed its rivals in the context of expert set expansion, we have identified that there is room for improvements. In the next sections, we elaborate on a series of additions, that transformed VeTo into a new, improved approach, called VeTo+.

3.2.1 Weighting different similarity types

The basic VeTo approach implicitly assumes that `APT`-based and `APV`-based similarities are equally important. In particular, during the fourth step of VeTo, the two similarity scores for each candidate are simply combined in the way dictated by the used rank aggregation algorithm. However, it is uncertain that the similarity of two academics based on the topics of their articles (i.e., their `APT`-based similarity) is equally important to their similarity based on the venues they publish (i.e., their `APV`-based similarity). It is possible that, for some disciplines, the one or the other similarity plays a more important role.

To alleviate this issue, we introduced a weighting scheme for VeTo+: during the rank aggregation, the similarity scores of the candidates in the R_{APT} and R_{APV} lists are further weighted based on two new configuration parameters, namely α and β, respectively. The sum of these parameters should always be equal to 1 (i.e., $\alpha + \beta = 1$). Consequently, larger α values promote candidates that are more similar with the known experts in E based on the `APT` metapath, while as β increases, more emphasis is given to candidates with higher similarity according to the `APV` metapath.

3.2.2 Using 'focused' `APV`-based similarities

A major application area of the expert set expansion problem is the expansion of the reviewer base (program committee or editorial board) of a given venue (conference/workshop or journal, respectively). During our experiments it became evident that, for some datasets (e.g., TPDL), the usage of 'unconditional' `APV`-based academics similarities, was not performing adequately well. Contrary, applying a venue-based constraint on the same metapath, so that only a 'focused' set of venues will be considered, worked well in these cases.

The exact reason why these 'focused' `APV`-based similarities might work better for a particular dataset can be various and are application-dependent. In the case of our experiments, this could be due to the fact that people involved in the program committees of the corresponding venues (e.g., TPDL) originate from significantly different backgrounds having, apart from digital libraries, also other, very diverse, research interests.

Based on this, we adapted VeTo+ so that it can support this functionality, as well. The user should first determine the set of venues that are of interest to them, and then configure the approach to consider the similarities according to the focused `APV`-based similarities. For our work, before proceeding with our experiments, we conducted a preliminary evaluation of the four datasets we are using, and the results revealed that the focused `APV`-based similarities provide improvement only for the case of TPDL and JCDL[2]. Consequently, in our evaluation, we use the focused similarities only for them and the default similarities for the rest (i.e., VLDB and SIGMOD).

3.2.3 Examining other rank aggregation algorithms

In addition to the Borda Count and the Sum aggregation algorithms, for the needs of VeTo+, we examine two extra rank aggregation algorithms. Following the same notation as in Section 3.1, we have implemented and tested the following:

- *Reciprocal Rank Fusion (RRF)* [6]: This algorithm is inspired by the fact that highly ranked elements are important; however, at the same time, RRF recognises that the importance of lower ranked ones should be preserved. RRF produces a ranked list:

$$R_R = \{\langle i, s_R^i \rangle : \ i \in I, \ s_R^i = \sum_{j \in [1,m]} \frac{1}{\lambda + rank(i, R_j)}\}$$

  where constant λ is a parameter that can be used to adjust the importance of elements based on their ranking in the input lists. Small values of λ essentially promote highly ranked elements, while large values mitigate their importance compared to those in lower ranks.
- *CombMNZ* [10]: It assigns higher weights to elements that are present in multiple ranking lists. Due to its to its simplicity and effectiveness, it serves as a baseline method when comparing different rank aggregators and data fusion approaches [3, 16, 17]. CombMNZ calculates ranking scores as:

$$R_C = \{ \ \langle i, s_C^i \rangle : i \in I, s_C^i = \sum_{j \in [1,m]} s_j^i * |s^i > 0| \ \}$$

  where $|s^i > 0|$ is the number of non-zero scores assigned to element i from all alternative rankings R_j. Note that CombMNZ requires normalisation of scores before aggregation.

It is worth mentioning that, in theory, it is possible to use other rank aggregators for the two rank aggregation steps of the algorithm. However, not all types of rank aggregation methods are suitable to be incorporated in VeTo. For example, there is a family of aggregators that require training to correlate items coming from different sources (e.g., [16, 17]). This type of methods require that a fixed set of sources, each having different quality, exist (since they rely on learning the particularities of each source). In the aggregation tasks of Steps 2 & 3 of VeTo, though, we have a different set of ranked lists for each query (each expert set); even the number of the list may differ from query to query. Since the sources are not fixed, learning their statistical properties is not possible. Therefore the aggregation methods of this family cannot be used by VeTo.

3.3 Final Remarks

VeTo+ is a configurable approach having the following parameters:

- α & β: these are the weights for the different similarity types (see Section 3.2.1).
- k: this parameter is used by Steps 1 & 3 of the basic VeTo approach; it dictates how many of the top similar academics will be considered as candidates. Attention should be given to avoid confusing k with the problem parameter n (usually $k >> n$).
- $rank$: this parameter determines the rank aggregation algorithm to be used and can have one of the following values: 'BC', 'Sum', 'RRF' or 'CombMNZ'.
- λ: this is a parameter of the the RRF aggregation algorithm (see Section 3.2.3), hence it is used in case that $rank = RRF$.
- $focus$: this is a boolean parameter, that determines if the optimization for the 'focused' `APV`-based similarities (see Section 3.2.2) will be used.

It should be noted that, despite the fact that both VeTo and VeTo+ exploit similarities based on two particular metapaths (namely APT and APV), this is not a limitation of their approach. In case a richer knowledge graph is used and if there are extra metapaths that seem to be useful, VeTo+ can handle their inclusion. In fact, VeTo+ is able to use an arbitrary number of metapaths: Steps 1 & 2 should be executed for each metapath to produce a ranked list of candidates based on its similarities, then that list should be taken into consideration in the aggregation of Step 4.

Finally, like its predecessor, although VeTo+ is tailored to the problem of expert finding in academia, it is possible to be adapted and applied to other domains given the existence of appropriate knowledge graphs and metapaths. For instance, given a knowledge graph containing information about movies (e.g. the FILM dataset from [18]), one may want to form the cast for a new movie. Given that they have already selected a set of known actors, one may want to expand that set. VeTo+ can exploit similarities based on the *Actor* -

[2] For TPDL and JCDL, we used the following venues to calculate the focused `APV`-based similarities: TPDL (and its predecessor ECDL), JCDL, and IJDL. For SIGMOD and VLDB: SIGMOD, VLDB, EDBT, ICDE and TODS.

Movie - *Genre* and *Actor* - *Movie* - *Director* metapaths to suggest possible expansions. The former essentially considers similarity of actors based on the genres of movies they have played while the latter captures similarity based on the directors they have collaborated with in the past.

4 Evaluation Framework

A common issue in various expert finding problems is that it is not easy to evaluate the effectiveness of a given approach, since it is impossible to construct an objective ground truth. However, this is not the case for the expert set expansion problem. In particular, in our previous work [32], we introduced a novel evaluation framework that can be used to assess the effectiveness of an approach based on a fairly objective ground truth. In this work, we expand this framework, adding Mean Average Precision (MAP) as an extra evaluation measure, and exploit it to measure the effectiveness of VeTo+ against VeTo and other state-of-the-art approaches. We next elaborate on the relevant technical details.

The intuition behind this framework is to gather available expert lists from real-life applications (e.g., the PC members of a conference, editorial boards of journals) and, then, use each of them as dataset for a k-fold cross validation process. This means that, for each expert list E, a given expert set expansion approach is assessed as follows:

1. E is shuffled and, then, split in k disjoint sets $E_1, \ldots, E_k$, all of equal size[3] $n = \lfloor |E|/k \rfloor$.
2. For each E_i (with $i \in [1, k]$), a pair of training and testing set $\{E_i^{train}, E_i^{test}\}$ is constructed, where $E_i^{train} = \bigcup_{j \neq i} E_j$ and $E_i^{test} = E_i$.
3. For each $\{E_i^{train}, E_i^{test}\}$ pair:
   - we use E_i^{train} as the set of known experts (i.e., $E = E_i^{train}$)
   - we apply the expert set expansion approach on E_i^{train} and get O_i, its output
   - we examine false & true positives and negatives in O_i^x, the top-x items of O_i, based on E_i^{test} and we calculate proper information retrieval measures based on them, for $x \in [1, n]$ (where, $n = |O_i| = |E_i^{test}|$).

Regarding the information retrieval measures that are suitable to be used in Step 3 of the aforementioned process, we proposed the use of top-x precision, recall, and F_1 score that can be defined as follows:

[3] The last one may be larger than the others, however it is easy to take this into consideration.

$$Precision_x = \frac{|O_i^x \cap E_i^{Test}|}{x}, \quad Recall_x = \frac{|O_i^x \cap E_i^{Test}|}{n},$$

$$F_{1x} = 2 \cdot \frac{Precision_x \cdot Recall_x}{Precision_x + Recall_x}$$

The larger the values of these measures are, the better the effectiveness of the method based on the given list E at the corresponding measuring point x. The values of all measuring points could be used to construct a line plot.

Moreover, after completing the previous process for E, we proposed to also calculate, for the same expert set, the Mean Reciprocal Rank (MRR) based on all outputs O_i (for all $i \in [1, k]$) which can be calculated as follows:

$$MRR = \frac{1}{k} \sum_{i=1}^{k} \frac{1}{rank_i}$$

where, $rank_i$ refers to the rank position of the first true positive element in the output O_i.

In this work, we additionally calculate the Mean Average Precision (MAP) that also considers the order of the returned elements. MAP aggregates the Average Precision (AvgP) value over all outputs O_i, $i \in [1, k]$ as follows:

$$MAP = \sum_{i=1}^{k} \frac{AvgP(O_i)}{k}$$

where $AvgP$ refers to the average precision after each true positive is retrieved.

The described evaluation framework was initially used for the experiments presented in [32] and we adopt it for the experiments of the current work (see Section 5). In particular, we use the list of program committee members of four well-known computer science conferences. Similarly to our previous experiments, we use the PC members of SIGMOD & VLDB, two well-known conferences from the field of Data Management; additionally we gathered the members of two top-tier conferences from the field of Digital Libraries (TPDL & JCDL). After the data collection, we applied the process of the framework on all four PCs using the aforementioned information retrieval measures.

A semi-automatic process was used to gather all program committee members using Web scrapping tools. In fact, our collected data could be used by third parties as benchmarks to evaluate the effectiveness of their own expert set expansion approaches. This is why we provide them as open datasets (more details in Section 5.1).

5 Experiments

In this section, we describe the experiments we have conducted to evaluate the effectiveness of our approach. Section 5.1 discusses the experimental setup, i.e. the approaches considered and the datasets used. Next, in Section 5.2 we study the effectiveness of VeTo+ while investigating different configurations. Finally, in Section 5.3 we demonstrate comparative experiments of VeTo+ against its competitors.

5.1 Setup

Datasets. For our experiments, we used the following sets of data:

- *DBLP Scholarly Knowledge Graph (DSKG) dataset.* It contains data for approximately 1.5M academics, their papers in the period 2000-2017, the corresponding venues and the involved topics. DSKG is based on the AMiner's DBLP citation network [30], enriched with topics assigned to papers by the CSO Classifier [22,23] (based on their abstracts). Finally, DSKG contains approximately 3.9M and 34.1M `APV` and `APT` metapath instances, respectively.
- *Program Committees (PC) dataset.* It contains program committee data from two established conferences from the field of data management (the ACM SIGMOD conference and the VLDB conference) and two well-known conferences from the field of digital libraries (TPDL and JCDL). The data were gathered by scrapping the official Web pages of these conferences for the years $2007-2017$ and, then, applying a semi-automatic cleaning process to properly map the PC members to academics in the DSKG dataset.

The DSKG dataset was used as a knowledge base that the various approaches could take advantage of. The PC dataset, on the other hand, was used to create the required training and testing sets for the evaluation based on the framework described in Section 4 (we use 5-fold validation for our experiments). This latter dataset was also made openly available at Zenodo[4] so other researchers could use it as benchmark to assess the effectiveness of their own approaches.

Approaches. In our experimental evaluation we consider the following five approaches that can be applied to the expert set expansion problem.

- *VeTo*, our initial approach presented in [32].[5]
- *VeTo+*, our new approach that improves upon *VeTo* by implementing the extensions described in Section 3.2.
- *Baseline*, an approach that counts the number of papers an academic has published in the corresponding conference, ranks academics based on this number, and then provides the top academics as the most suitable expansions.
- *ADT*, the best performing graph-based approach proposed in [12], that attempts to capture the association strength between two academics by considering the weighted paths that relate them to topics (based on their papers).[6]
- *DOC*, an approach proposed in [2], which considers academic similarities according to the number of their common publications.[7], WG performed worse in all cases and its results were omitted from the experimental section for presentation reasons.

The different approaches were implemented in Python, although the data preprocessing was implemented in C++ for improved efficiency.[8] In addition, metapath-based similarities required by VeTo and VeTo+ are calculated using the JoinSim [33] algorithm (see also Section 2). For this purpose, the open entity similarity Java library HeySim[9] was utilised.

5.2 Effectiveness Analysis & Configuration of VeTo+

In this section, we examine different configurations of our approach and we investigate the effect they have in its effectiveness.

[4] `https://doi.org/10.5281/zenodo.3739315`

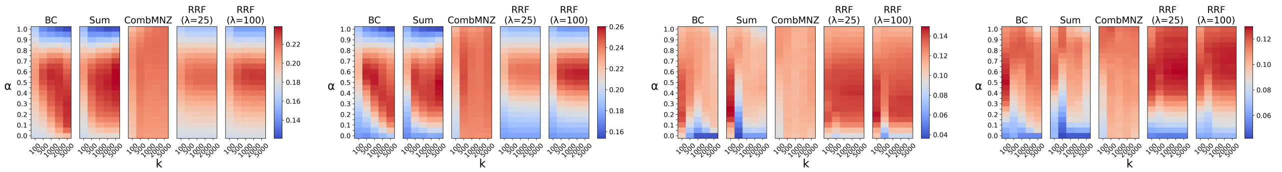
[5] In this work, the configuration of VeTo and VeTo+ was done by selecting the same parameter value for all experiments performed on the same dataset; the selection was made according to the value that yield the best F_1 results). This experimental design is different to the one used in our previous work [32], where the best configuration of VeTo was selected for each of the respective setups (e.g., $k = 100$ and $k = 200$ was used for the F_1 and the MRR experiment for the SIGMOD dataset, respectively). More details for the configuration of VeTo+ can be found in Section 5.2.1).

[6] Note that, since the DSKG dataset does not contain weighted edges between papers and topics, we assigned weights in correspondence to the number of topics connected to each paper, i.e., assuming that a paper is connected with n topics, the weight assigned to each edge is equal to $1/n$.

[7] We have also conducted experiments using *WG*, the alternative graph-based approach proposed in the same paper. However, similarly to the results in [2]

[8] `https://github.com/smartdatalake/HMiner`

[9] `https://github.com/schatzopoulos/HeySim`



(a) SIGMOD (b) VLDB (c) TPDL (d) JCDL

Fig. 2 Parameter configuration based on average $F_1 score$ of different rank aggregation methods.

5.2.1 Parameter configuration

In this experiment, we examine VeTo+'s performance in terms of the average F_1 score using different parameter configurations. In particular, we vary parameters α, β, and k. For α and β we examine all the values in the range $[0, 1]$ with a step of 0.05 (recall that a and b values are dependent since $a + b = 1$); for k we examine the values in the set $\{100, 500, 1000, 2000, 5000\}$. Additionally, we examine different values of parameter λ (in the range $[0, 100]$ with step 25) for the cases when the RRF rank aggregation algorithm is used. In Figure 2, we use heatmaps to visualise the average F_1 score achieved by the various VeTo+ configurations for all datasets based on the aforementioned setup. It should be noted that, for presentation reasons, we only visualise the results for $\lambda = 25$ and $\lambda = 100$, as these values achieve the best results. Finally, for parameter $focus$ we select the best option for each dataset based on a set of preliminary measurements we conducted (see Section 3.2.2). For SIGMOD and VLDB the best option was to set $focus = false$, while for TPDL and JCDL $focus = true$. The detailed parameter configurations that found to perform best for each dataset are presented in the Appendix A.

Based on our experiments, the Sum rank aggregation algorithm achieves the best average F_1 score for SIGMOD that is equal to 0.238 when $\{\alpha = 0.55, \beta = 0.45, k = 5000\}$ with BC and RRF being on par with 0.234 and 0.231 respectively. CombMNZ underperforms in SIGMOD achieving 0.221 when $\{\alpha = 0.65, \beta = 0.55, k = 500\}$. For VLDB, all methods result in comparable scores with BC being slightly better with 0.258 at $\{\alpha = 0.55, \beta = 0.45, k = 1000\}$. For TPDL, RRF at $\{\alpha = 0.2, \beta = 0.8, k = 100, \lambda = 100\}$ achieves slightly better F_1 score with 0.149 compared to Sum that achieves 0.147, with BC and CombMNZ performing worse with merely 0.14 and 0.122 respectively. RRF at $\{\alpha = 0.6, \beta = 0.4, k = 5000, \lambda = 25\}$ is also the best performing rank aggregation algorithm for JCDL with 0.129 with BC being slightly worse with 0.126. Meanwhile, CombMNZ and Sum score 0.12 and 0.118 respectively. Table 5 summarises the best parameter configurations found. Taken into consideration our findings, we use RRF as the default rank aggregation algorithm of VeTo+ that achieves the best results in TPDL and JCDL and has comparable performance for VLDB and SIGMOD.

Table 1 MRR of different variants based on the folds of each dataset. The highest MRR score for each conference is in bold.

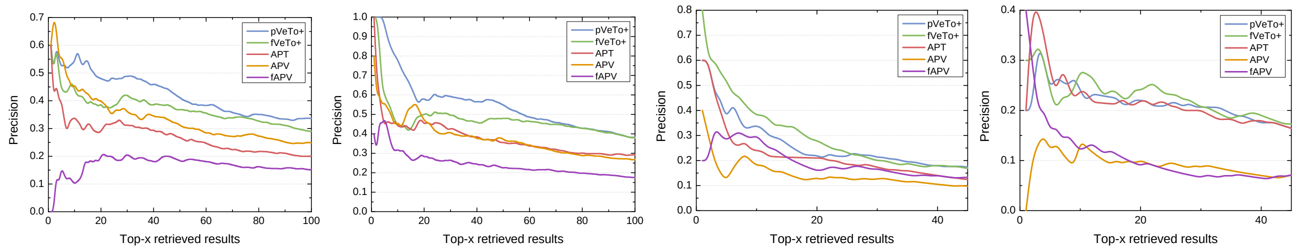
	APT	**APV**	**fAPV**	**pVeTo+**	**fVeTo+**
SIGMOD	0.74	0.766	0.213	**0.8**	**0.8**
VLDB	**1**	**1**	0.6	**1**	**1**
TPDL	0.84	0.558	0.5	0.75	**0.866**
JCDL	0.488	0.24	**0.563**	0.378	0.458
Average	0.767	0.648	0.469	0.732	**0.781**

5.2.2 Studying the effect of the used metapaths

VeTo+'s approach considers similarities of academics based on two criteria: their similarity based on the venues they prefer to publish (captured by the `APV` metapath) and on the topics of their published papers (captured by the `APT` metapath). In this experiment we investigate the effect of each of these metapaths by examining the following VeTo+'s variants:

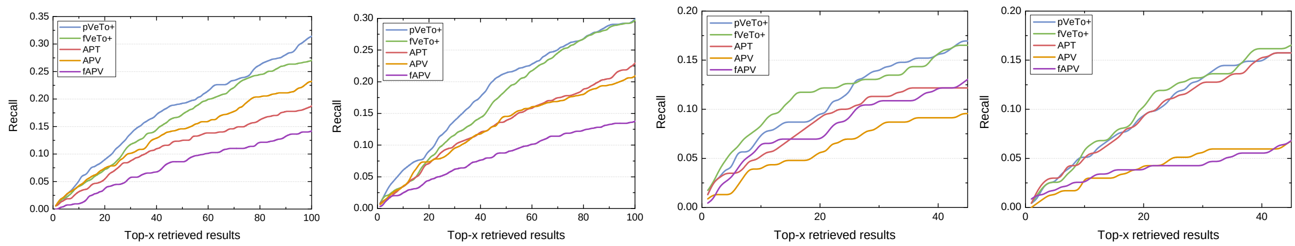
– *APT* considers similarities of academics based only on the topics of the papers they publish (only `APT` metapath).
– *APV* considers similarities based only on the venues they choose to publish (only `APV` metapath).
– *fAPV*, a variant of the previous method that calculates similarities considering only venues of the field of the venue of interest (i.e., it is a 'focused' variant, details in Section 3.2.2).
– *pVeTo+*, a 'plain' version of our approach that does not use the focused `APV` metapath (i.e., $focus = false$)
– *fVeTo+*, a variant of VeTo+ that uses *fAPV* instead of `APV` to produce the final result (i.e., $focus = true$).

In the context of this experiment, the aforementioned approaches are configured at their best settings (see Section 5.2.1). Figures 3, 4 and 5 illustrate the measured top-x precision, recall and F_1 score, respectively,



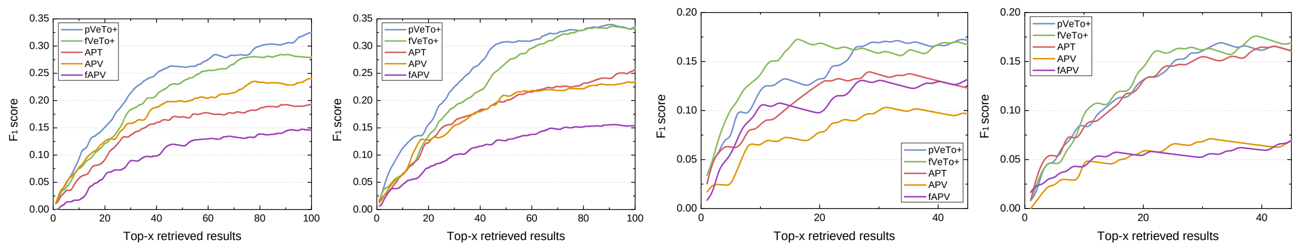
(a) SIGMOD (b) VLDB (c) TPDL (d) JCDL

Fig. 3 Precision of different variants of our method.



(a) SIGMOD (b) VLDB (c) TPDL (d) JCDL

Fig. 4 Recall of different variants of our method.



(a) SIGMOD (b) VLDB (c) TPDL (d) JCDL

Fig. 5 F_1 score of different variants of our method.

Table 2 MAP of different variants based on the folds of each dataset. The highest MAP score for each conference is in bold.

	APT	APV	fAPV	pVeTo+	fVeTo+
SIGMOD	0.324	0.403	0.174	**0.464**	0.427
VLDB	0.395	0.416	0.273	**0.605**	0.492
TPDL	0.404	0.175	0.245	0.326	**0.448**
JCDL	**0.306**	0.109	0.162	0.261	0.288
Average	0.357	0.275	0.213	**0.414**	0.413

of all variants for SIGMOD, VLDB, TPDL and JCDL, while Tables 1 and 2 summarise the MRR and MAP scores respectively.

It is evident that pVeTo+ and fVeTo+ outperform the other variants in (almost) all cases. This indicates that both `APV`-based and `APT`-based similarities can be helpful and that it is beneficial to combine them. Furthermore, it should be noted that APV achieves slightly higher precision and recall than APT in most cases in the SIGMOD dataset. Both these approaches achieve comparable results for the VLDB conference; however APT is significantly better in JCDL and achieves higher precision in the first elements of TPDL. This result is inline with the intuition that for different datasets different metapath-based similarities may be more important (see also Section 3.2.1).

Furthermore, fAPV improves upon APV in TPDL (mainly) and JCDL, failing to achieve similar improvements in SIGMOD and VLDB. As expected this is reflected in the performance of fVeTo+ and pVeTo+, as well. A possible explanation for this behavior could be that the PC members of conferences in digital libraries seem to publish a smaller proportion of their works in digital library venues, compared to the proportion of papers by data management PC members published in data management venues. Indicatively, we have found that TPDL's and JCDL's PC members published only 10.7% and 10.6% of their papers in digital libraries venues, while the same statistics for SIGMOD's and VLDB's PC members is 25.7% and 23.1%, respectively.[10] As a result, fAPV is expected to provide more evident improvements in the case of recommending expansions for the PCs of digital libraries conferences since, in this case, the plain APV metapath will also consider academic similarities based on published work in venues on irrelevant topics.

5.2.3 Studying the effect of different rank aggregations

A rank aggregation algorithm is required in various steps of our approach. In this section, we examine the effect of different rank aggregation algorithms in the performance of VeTo+. In particular, we consider the four rank aggregation algorithms: *Borda Count (BC)*, *Sum*, *CombMNZ* and *Reciprocal Rank Fusion (RRF)* (see also Sections 3.1 & 3.2). In Figures 6, 7 and 8 we present the top-x precision, recall, and F_1 score for the best parameter configuration of VeTo+ using each rank aggregation algorithm.

All algorithms achieve comparable results in (almost) all datasets. CombMNZ and Sum perform significantly worse than other options in TPDL and JCDL conferences, respectively. A possible explanation for this may be that, in these conferences, VeTo+ uses the 'focused' venue similarities (i.e., `fAPV` instead of `APV` similarities); in this case, since the number of venues, based on which the similarities are calculated, is very small, it is likely that there are many PC members having no articles published in the respective venues. Consequently, no list of similar academics can be produced for them, thus, the list of recommended experts according to `fAPV` is likely to be produced by aggregating a significantly smaller number of ranked lists. As a result, methods like CombMNZ and Sum, that incorporate the actual similarity scores, may be heavily biased against `fAPV`, since `fAPV` scores are expected to be significantly smaller than `APT` scores. On the other hand, rank aggregators like BC and RRF, which rely on the rank positions and not the exact scores, are not affected by this and are expected to achieve better results. The differences between BC and RRF are marginal; BC achieves slightly higher precision in the top few results in SIGMOD with RRF subsequently managing to overcome, while RRF performs better in the first retrieved elements in TPDL. From these two options, we choose to use RRF as our default rank aggregation algorithm with VeTo+ for the rest of the experimental evaluation.

[10] Our analysis was based on the venue catalogues determined in Section 3.2.2.

5.3 Evaluation against competitors

In this experiment, we compare the effectiveness of our approach against its rivals based on the framework discussed in Section 4 using all expert sets in the PC dataset (SIGMOD, VLDB, TPDL and JCDL), according to the best configurations identified in Section 5.2.1.

5.3.1 Precision, recall & F_1 score

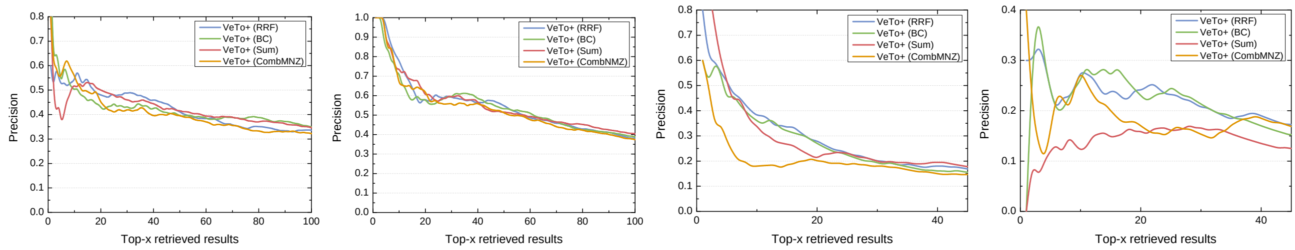
Figures 9, 10 and 11 present the precision, recall and F_1 score of all compared approaches, respectively. Larger values for all measures indicate superior effectiveness. It is evident that VeTo and VeTo+ clearly outperform their competitors in (almost) all scenarios. More importantly, in all datasets, they achieve notably higher precision in comparison to the rest approaches for the top retrieved results, which are usually the most useful ones: for most applications, the required expansion of the expert set involves adding a relatively small number of extra experts.

Furthermore, VeTo+ achieves better results than VeTo in all examined expert sets. It significantly outperforms VeTo for SIGMOD and VLDB after the first 20 results, where they both achieve comparable performance. On the other hand, in the case of JCDL and TPDL the differences between the two approaches are prominent even for small x values. The larger differences between VeTo and VeTo+ can be probably explained by the fact that, for these datasets, VeTo+ is configured to have $focus = true$, resulting in using significantly different APV-based academics similarities than those used by VeTo.

Finally, it should be noted that the baseline approach seems to work pretty well (but, at the same time, significantly worse than VeTo+) in most cases. It performs notably well for TPDL and JCDL, a result indicating that there is a correlation between the academics that publish articles in these conferences and the corresponding PC members. On the other hand, both ADT and DOC do not perform well, with DOC performing significantly better than ADT only in TPDL.

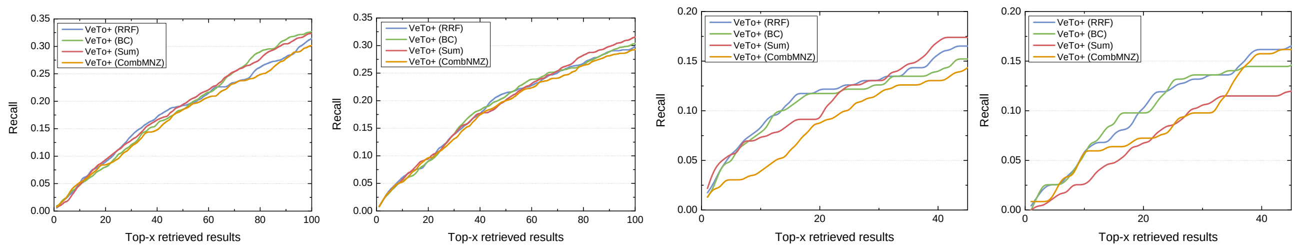
5.3.2 MRR & MAP per conference

Tables 3 and 4 present the assessment of all approaches based on the Mean Reciprocal Rank (MRR) and Mean Average Precision (MAP) considering all expert sets (SIGMOD, VLDB, TPDL and JCDL) over all their folds. They also include the average score for each approach considering all expert sets. Values in bold face highlight the highest scores per conference, while asterisks indicate statistical significance compared to VeTo+



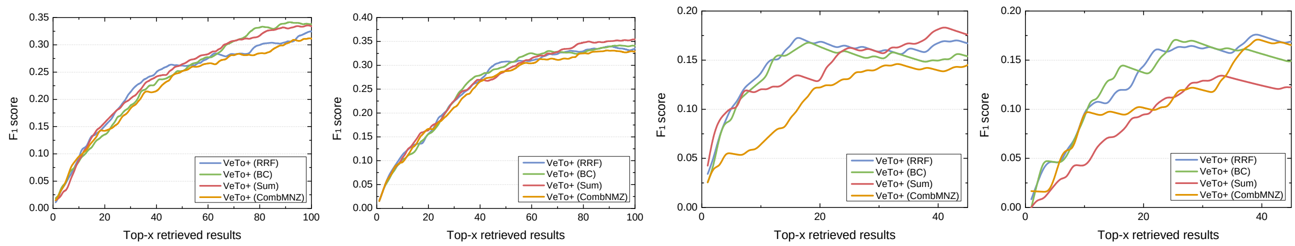
(a) SIGMOD (b) VLDB (c) TPDL (d) JCDL

Fig. 6 Precision of different rank aggregation methods.



(a) SIGMOD (b) VLDB (c) TPDL (d) JCDL

Fig. 7 Recall of different rank aggregation methods.



(a) SIGMOD (b) VLDB (c) TPDL (d) JCDL

Fig. 8 F_1 score of different rank aggregation methods.

Table 3 MRR based on the folds of each dataset. The highest MRR score for each conference is in bold. Asterisks indicate statistical significance ($p < 0.05$) when compared to Veto+ using the t-test.

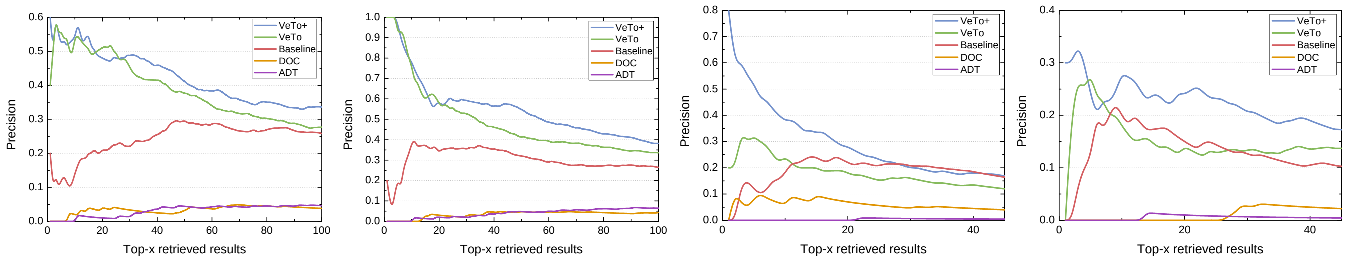
	Baseline	ADT	DOC	VeTo	VeTo+
SIGMOD	0.323∗	0.032∗	0.056∗	0.7	**0.8**
VLDB	0.357∗	0.035∗	0.049∗	**1**	**1**
TPDL	0.201∗	0.012∗	0.209∗	0.483∗	**0.866**
JCDL	0.191∗	0.027∗	0.028∗	0.373	**0.458**
Average	0.268	0.026	0.114	0.639	**0.781**

Table 4 MAP based on the folds of each dataset. The highest MAP score for each conference is in bold. Asterisks indicate statistical significance ($p < 0.05$) when compared to Veto+ using the t-test.

	Baseline	ADT	DOC	VeTo	VeTo+
SIGMOD	0.270∗	0.038∗	0.049∗	0.448	**0.464**
VLDB	0.323∗	0.061∗	0.057∗	0.547∗	**0.605**
TPDL	0.222∗	0.009∗	0.081∗	0.284∗	**0.448**
JCDL	0.199∗	0.014∗	0.020∗	0.221∗	**0.288**
Average	0.253	0.030	0.051	0.375	**0.451**

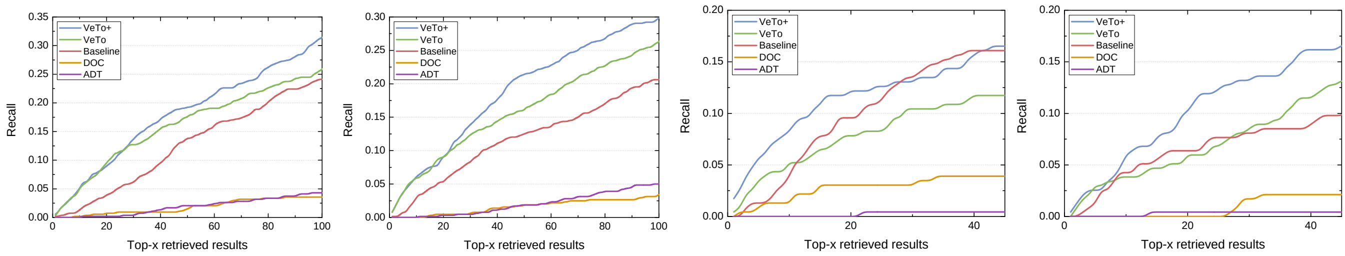
using the t-test with $p < 0.05$. Overall, larger scores indicate better approach effectiveness.

MRR results are in compliance with the previous experiment: since VeTo and VeTo+ achieve notably larger precision for small values of x, they perform notably better than their competitors in terms of MRR (see also MRR definition in Section 4). VeTo+ matches the result of VeTo in VLDB, that is the maximum MRR score (i.e. equal to 1), while achieving higher scores in the remaining datasets. The most prominent difference, that is also statistically significant, is observed for TPDL where VeTo+ manages a 79% improvement over VeTo. This large difference may be due to the fact that VeTo+ is using $foucus = true$ for this dataset; JCDL, which also takes advantage of the same optimisation, also achieves a notable (but smaller) improve-



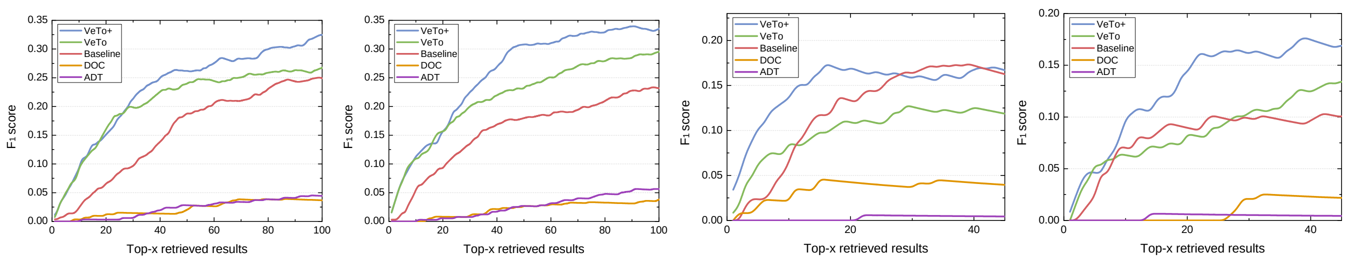
(a) SIGMOD (b) VLDB (c) TPDL (d) JCDL

Fig. 9 Precision against competitors.



(a) SIGMOD (b) VLDB (c) TPDL (d) JCDL

Fig. 10 Recall against competitors.



(a) SIGMOD (b) VLDB (c) TPDL (d) JCDL

Fig. 11 F_1 score against competitors.

ment (22.79%). Last but not least, DOC performs better than ADT overall, mainly due to its noteworthy score in TPDL, however both these approaches perform significantly worse than the Baseline.

Table 4 presents the MAP scores for the examined conferences across all folds. VeTo+ outperforms its competitors in all conferences, achieving statistically significant improvements in SIGMOD, TPDL and JCDL. In particular, complying with the results of MRR, it achieves the most notable differences over VeTo in TPDL and JCDL conferences managing 57% and 30% improvements, respectively. VeTo+ achieves significantly larger MAP scores than the Baseline method which in turn outperforms ADT and DOC that underperform in all examined scenarios.

5.4 Discussion on the Configuration of VeTo & VeTo+

In our experiments we have examined a range of parameter values to identify the best performing configuration for each dataset (Section 5.2.1). However, this is a time consuming grid search approach that is not practical for many real applications, especially when using a new dataset. Although there are approximation techniques that could tune these parameters reducing the execution cost (e.g., using simulated annealing [15, 31]), in some cases even this cost may not be acceptable. Identifying this issue, in this section we discuss a set of best practices and guidelines on how to come up with an acceptable configuration for a given dataset. Our guidelines are based on the findings of our experimental section.

Taking into account our findings in Section 5.2.1, we conclude that the Reciprocal Rank Fusion (RRF) performs overall better than other rank aggregation algorithms in all examined datasets. RRF yields good results when its λ parameter is set to 100 (or smaller values in some cases). In general, we observe that VeTo+ achieves the maximum gains when $\alpha \in [0.5, 0.6]$, $\beta \in [0.4, 0.5]$ and $k \geq 2000$. Last but not least, as discussed in Section 5.2.2 more 'focused' similarities based on venues bring considerable gains when the PC members of the examined conference have a large proportion of their work published in venues of different disciplines. It is also worth mentioning that in case 'focused' venue similarities are used, it is preferable to use an aggregator that considers the rank positions of the elements in the input ranked lists (like BC and RRF) and not the actual similarity scores, in order to avoid the bias against `fAPV` similarities as discussed in Section 5.2.3.

6 Related Work

Expertise retrieval consists an interesting field of research in many disciplines like digital libraries, data management, information retrieval, and machine learning. A wide range of problems, ranging from expert finding to expert profiling, belong in this field and there are many related real-time applications (e.g., collaboration recommendation, reviewer recommendation). A detailed review of the field is beyond the scope of the current work. The reader interested could refer to the excellent survey in [13]. In the next we will focus on the variations of the *expert finding* problem.

Finding experts for a given topic in the industry has been a relatively well-studied problem. Initial approaches relied on manually curated databases of skills and knowledge (e.g., [8]), however the interest quickly shifted to approaches that extract employee's expertise from document collections that could be found within corporate intranets or the Web [4, 7]. A common platform to empirically assess such approaches has been developed by the TREC community[11] facilitating the development of various relevant methods [1, 9, 21, 24]. Apart from details about the exact expert finding problems solved by each of the previous methods, our work significantly differs from these works in principle, since it is tailored for academic experts and since it does not rely on document collections because such collections are often available due to the existing paywalls.

Finding experts in academia, where the experts are researchers with knowledge and interests in a given topic, has also been an important field (e.g., [25, 34]). However, most of these methods also rely on scientific text corpora, which are often limited behind paywalls. Motivated by this problem, many researchers turned their focus on approaches that are able to utilise alternative data sources, like the various scholarly knowledge graphs, which have lately become popular and contain rich information about scholars and their publications (e.g., the Open Research Knowledge Graph [14], the OpenAIRE Research Graph [19, 20]).

As an indicative example, the authors of [2] introduce two approaches that can take advantage of the data stored in a scholarly knowledge graph. The one approach, called *DOC*, considers academics to be similar if they share a significant number of common publications. Similarly, the other approach, called *WG*, calculates a similarity score for academics based on the number of co-authors they have in common (i.e., it takes into consideration 'working groups' of academics). Although both methods might sometimes bring interesting results, it is evident that they calculate academic similarities using very limited information about them, thus it is difficult to bring valuable suggestions in the majority of cases.

Another indicative example, is *ADT* [12], which captures the association strength between two academics by considering the paths that connect them to topics. Given a tripartite graph comprising academics, papers and topics and weighted edges between them, ADT first uses a multiplicative scheme to aggregate edge weights into path weights. Then, the association strength of two academics is calculated by summing the weights of the paths that connect them via the topic nodes in the graph. This method highly depends on the weights assigned to the edges of the network, which can be problematic since, in many cases, scholarly knowledge graphs do not include such information.

It is worth mentioning that both DOC/WG and ADT make use of very limited information compared to the content of popular scholarly knowledge graphs. On the other hand, both VeTo and VeTo+ take advantage of metapath-based analysis techniques, which have been recently developed to exploit complex and latent information that resides in knowledge graphs. This, along with the adoption of advanced rank aggregation techniques, gives them the ability to easily calculate and combine similarities with various semantics.

7 Conclusions

In this work, we study the expert set expansion problem for academic experts, i.e., given a set of known experts to find the n most suitable candidates to expand this set. In this context, we first introduced VeTo [32] and,

[11] `https://trec.nist.gov/`

then, in this work, VeTo+, two expert set expansion approaches for academic experts that exploit information from a given scholarly knowledge graph to estimate similarities between academics, based on their publishing habits. VeTo+ extends VeTo by introducing a flexible weighting scheme for the used similarity measures and a couple of alternative metapath-based similarities and rank aggregation algorithms.

We utilised an extended version of the evaluation framework we introduced in [32] to perform thorough experiments to compare VeTo+ against its predecessor and a set of other competitors. The investigation revealed that VeTo+ improves upon VeTo, in terms of effectiveness, taking advantage of its novel weighting scheme for aggregating the respective similarity measure scores, and allows for more configuration options to better fit in different use cases, compared to its predecessor. Additionally, VeTo+ also considers 'focused' similarities based on venues, an approach that brings noteworthy improvements in the case of TPDL and JCDL conferences. This may be due to the fact that the PC members of TPDL and JCDL have a smaller proportion of their work published in digital library venues compared to those of SIGMOD and VLDB that have a larger proportion of their work published in data management venues. Therefore, in the case of TPDL and JCDL the 'focused' similarities based on venues avoid capturing similarities based on published work on irrelevant topics.

Although VeTo+ outperforms the state-of-the-art for the expert set expansion problem in academia, we believe that there is plenty room for improvements for future research. First of all, our work does not consider the fact that knowledge graphs evolve over time. However, it may be interesting to investigate the effect of temporal changes over the performance of VeTo+ or, even, to propose improved approaches that take into consideration the most recent publishing habits of the scholars as more important than older ones. Another indicative extension could be to investigate the effectiveness of adaptations of VeTo+ to solve similar problems from other domains (e.g., finding similar movie actors) using appropriate knowledge graphs from the respective domains.

Acknowledgements This work was partially funded by the EU H2020 project SmartDataLake (825041). We also acknowledge support of this work by the project "Moving from Big Data Management to Data Science" (MIS 5002437/3) which is implemented under the Action "Re-inforcement of the Research and Innovation Infrastructure", funded by the Operational Programme "Competitiveness, Entrepreneurship and Innovation" (NSRF 2014-2020) and co-financed by Greece and the European Union (European Regional Development Fund). Icons in Figure 1 were collected from www.flaticon.com and were made by Freepik.

A Detailed configurations

In this section we present the exact parameter configurations of the rank aggregation algorithms that found to perform best for each dataset.

Table 5 Parameter configuration of VeTo+ for each rank aggregation algorithm.

	BC	**Sum**	**CombMNZ**	**RRF**
SIGMOD	$\alpha = 0.55$ $\beta = 0.45$ $k = 1000$ $focus = false$	$\alpha = 0.55$ $\beta = 0.45$ $k = 5000$ $focus = false$	$\alpha = 0.65$ $\beta = 0.35$ $k = 500$ $focus = false$	$\alpha = 0.55$ $\beta = 0.45$ $k = 2000$ $focus = false$ $\lambda = 100$
VLDB	$\alpha = 0.55$ $\beta = 0.45$ $k = 1000$ $focus = false$	$\alpha = 0.5$ $\beta = 0.5$ $k = 5000$ $focus = false$	$\alpha = 0.55$ $\beta = 0.45$ $k = 500$ $focus = false$	$\alpha = 0.6$ $\beta = 0.4$ $k = 2000$ $focus = false$ $\lambda = 100$
TPDL	$\alpha = 0.2$ $\beta = 0.8$ $k = 100$ $focus = true$	$\alpha = 0.25$ $\beta = 0.75$ $k = 100$ $focus = true$	$\alpha = 0.8$ $\beta = 0.2$ $k = 100$ $focus = true$	$\alpha = 0.2$ $\beta = 0.8$ $k = 100$ $focus = true$ $\lambda = 100$
JCDL	$\alpha = 0.5$ $\beta = 0.5$ $k = 100$ $focus = true$	$\alpha = 0.85$ $\beta = 0.15$ $k = 500$ $focus = true$	$\alpha = 1$ $\beta = 0$ $k = 500$ $focus = true$	$\alpha = 0.6$ $\beta = 0.4$ $k = 5000$ $focus = true$ $\lambda = 25$

References

1. Balog, K., Azzopardi, L., de Rijke, M.: Formal models for expert finding in enterprise corpora. In: SIGIR 2006: Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Seattle, Washington, USA, August 6-11, 2006, pp. 43–50 (2006)
2. Balog, K., de Rijke, M.: Finding similar experts. In: Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '07, p. 821–822. Association for Computing Machinery, New York, NY, USA (2007). DOI 10.1145/1277741.1277926. URL https://doi.org/10.1145/1277741.1277926
3. Beitzel, S.M., Jensen, E.C., Chowdhury, A., Grossman, D., Frieder, O., Goharian, N.: Fusion of effective retrieval strategies in the same information retrieval system. Journal of the American Society for Information Science and Technology **55**(10), 859–868 (2004)
4. Cao, Y., Liu, J., Bao, S., Li, H.: Research on expert search at enterprise track of TREC 2005. In: Proceedings of the Fourteenth Text REtrieval Conference, TREC 2005, Gaithersburg, Maryland, USA, November 15-18, 2005 (2005). URL http://trec.nist.gov/pubs/trec14/papers/microsoft-asia.ent.pdf
5. Chen, H.H., Gou, L., Zhang, X., Giles, C.L.: Collabseer: A search engine for collaboration discovery. In: Proceedings of the 11th Annual International ACM/IEEE Joint Conference on Digital Libraries, JCDL '11, p. 231–240. Association for Computing Machinery, New York, NY, USA (2011). DOI 10.1145/1998076.1998121. URL https://doi.org/10.1145/1998076.1998121
6. Cormack, G.V., Clarke, C.L., Buettcher, S.: Reciprocal rank fusion outperforms condorcet and individual rank learning methods. In: Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval, pp. 758–759 (2009)
7. Craswell, N., Hawking, D., Vercoustre, A.M., Wilkins, P.: P@noptic expert: Searching for experts not just for documents. In: Ausweb Poster Proceedings, Queensland, Australia, vol. 15, p. 17 (2001)
8. Davenport, T.H., Prusak, L.: Working knowledge: how organizations manage what they know. Ubiquity **2000**(August), 6 (2000)
9. Fang, H., Zhai, C.: Probabilistic models for expert finding. In: Advances in Information Retrieval, 29th European Conference on IR Research, ECIR 2007, Rome, Italy, April 2-5, 2007, Proceedings, pp. 418–430 (2007)
10. Fox, E.A., Shaw, J.A.: Combination of multiple searches. NIST special publication SP **243** (1994)
11. Gollapalli, S.D., Mitra, P., Giles, C.L.: Similar researcher search in academic environments. In: Proceedings of the 12th ACM/IEEE-CS Joint Conference on Digital Libraries, JCDL '12, p. 167–170. Association for Computing Machinery, New York, NY, USA (2012). DOI 10.1145/2232817.2232849. URL https://doi.org/10.1145/2232817.2232849
12. Gollapalli, S.D., Mitra, P., Giles, C.L.: Ranking experts using author-document-topic graphs. In: 13th ACM/IEEE-CS Joint Conference on Digital Libraries, JCDL '13, Indianapolis, IN, USA, July 22 - 26, 2013, pp. 87–96 (2013). DOI 10.1145/2467696.2467707
13. Gonçalves, R., Dorneles, C.F.: Automated expertise retrieval: A taxonomy-based survey and open issues. ACM Comput. Surv. **52**(5), 96:1–96:30 (2019)
14. Jaradeh, M.Y., Oelen, A., Farfar, K.E., Prinz, M., D'Souza, J., Kismihók, G., Stocker, M., Auer, S.: Open research knowledge graph: Next generation infrastructure for semantic scholarly knowledge. In: Proceedings of the 10th International Conference on Knowledge Capture, K-CAP 2019, Marina Del Rey, CA, USA, November 19-21, 2019, pp. 243–246 (2019). DOI 10.1145/3360901.3364435
15. Kirkpatrick, S., Gelatt, C.D., Vecchi, M.P.: Optimization by simulated annealing. science **220**(4598), 671–680 (1983)
16. Lillis, D., Toolan, F., Collier, R., Dunnion, J.: Extending probabilistic data fusion using sliding windows. In: European Conference on Information Retrieval, pp. 358–369. Springer (2008)
17. Lillis, D., Zhang, L., Toolan, F., Collier, R.W., Leonard, D., Dunnion, J.: Estimating probabilities for effective data fusion. In: Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval, pp. 347–354 (2010)
18. Lu, J., Chen, J., Zhang, C.: Helsinki Multi-Model Data Repository. http://udbms.cs.helsinki.fi/?dataset (2016)
19. Manghi, P., Atzori, C., Bardi, A., Shirrwagen, J., Dimitropoulos, H., La Bruzzo, S., Foufoulas, I., Löhden, A., Bäcker, A., Mannocci, A., et al.: Openaire research graph dump (2019)
20. Manghi, P., Bardi, A., Atzori, C., Baglioni, M., Manola, N., Schirrwagen, J., Principe, P.: The openaire research graph data model (2019). DOI 10.5281/zenodo.2643199. URL https://doi.org/10.5281/zenodo.2643199
21. Petkova, D., Croft, W.B.: Hierarchical language models for expert finding in enterprise corpora. International Journal on Artificial Intelligence Tools **17**(1), 5–18 (2008)
22. Salatino, A.A., Osborne, F., Thanapalasingam, T., Motta, E.: The cso classifier: Ontology-driven detection of research topics in scholarly articles. In: International Conference on Theory and Practice of Digital Libraries, pp. 296–311. Springer (2019)
23. Salatino, A.A., Thanapalasingam, T., Mannocci, A., Osborne, F., Motta, E.: The computer science ontology: a large-scale taxonomy of research areas. In: International Semantic Web Conference, pp. 187–205. Springer (2018)
24. Serdyukov, P., Hiemstra, D.: Modeling documents as mixtures of persons for expert finding. In: Advances in Information Retrieval , 30th European Conference on IR Research, ECIR 2008, Glasgow, UK, March 30-April 3, 2008. Proceedings, pp. 309–320 (2008). DOI 10.1007/978-3-540-78646-7_29. URL https://doi.org/10.1007/978-3-540-78646-7_29
25. Sfyris, G.A., Fragkos, N., Doulkeridis, C.: Profile-based selection of expert groups. In: International Conference on Theory and Practice of Digital Libraries, pp. 81–93. Springer (2016)
26. Shi, C., Kong, X., Huang, Y., Philip, S.Y., Wu, B.: Hetesim: A general framework for relevance measure in heterogeneous networks. IEEE Transactions on Knowledge and Data Engineering **26**(10), 2479–2492 (2014)
27. Shi, C., Li, Y., Philip, S.Y., Wu, B.: Constrained-meta-path-based ranking in heterogeneous information network. Knowledge and Information Systems **49**(2), 719–747 (2016)
28. Shi, C., Li, Y., Zhang, J., Sun, Y., Yu, P.S.: A survey of heterogeneous information network analysis. IEEE Trans. Knowl. Data Eng. **29**(1), 17–37 (2017). DOI 10.1109/TKDE.2016.2598561. URL https://doi.org/10.1109/TKDE.2016.2598561

29. Sun, Y., Han, J., Yan, X., Yu, P.S., Wu, T.: Pathsim: Meta path-based top-k similarity search in heterogeneous information networks. Proceedings of the VLDB Endowment **4**(11), 992–1003 (2011)
30. Tang, J., Zhang, J., Yao, L., Li, J., Zhang, L., Su, Z.: ArnetMiner: Extraction and Mining of Academic Social Networks. In: Proceedings of the 14th ACM SIGKDD, pp. 990–998. ACM (2008)
31. Tsallis, C., Stariolo, D.A.: Generalized simulated annealing. Physica A: Statistical Mechanics and its Applications **233**(1), 395–406 (1996). DOI https://doi.org/10.1016/S0378-4371(96)00271-3
32. Vergoulis, T., Chatzopoulos, S., Dalamagas, T., Tryfonopoulos, C.: Veto: Expert set expansion in academia. In: International Conference on Theory and Practice of Digital Libraries, pp. 48–61. Springer (2020)
33. Xiong, Y., Zhu, Y., Yu, P.S.: Top-k similarity join in heterogeneous information networks. IEEE Transactions on Knowledge and Data Engineering **27**(6), 1710–1723 (2015)
34. Yang, K., Kuo, T., Lee, H., Ho, J.: A reviewer recommendation system based on collaborative intelligence. In: 2009 IEEE/WIC/ACM International Conference on Web Intelligence, WI 2009, Milan, Italy, 15-18 September 2009, Main Conference Proceedings, pp. 564–567 (2009). DOI 10.1109/WI-IAT.2009.94. URL `https://doi.org/10.1109/WI-IAT.2009.94`