

# Search for an Appropriate Journal – in Depth Evaluation of Data Fusion Techniques

Markus Wegmann<sup>1</sup> and Andreas Henrich<sup>2</sup>

<sup>1</sup> [markus.wegmann@live.de](mailto:markus.wegmann@live.de)

<sup>2</sup> Media Informatics Group, University of Bamberg, Bamberg, Germany  
[andreas.henrich@uni-bamberg.de](mailto:andreas.henrich@uni-bamberg.de)  
<https://www.uni-bamberg.de/minf/>

**Abstract.** Based on available or generated metadata, collection objects can be categorized and organized in disjoint sets or classes. In our approach we search for appropriate classes of a categorized collection based on object instance queries. More concretely, our collection consists of scientific papers as object instances which are classified by the journals they are published in. Our textual query searches for the most appropriate classes respectively journals. At LWDA 2017 [1] we introduced a voting-based approach for searching these appropriate journals: Utilizing randomly removed articles from the article collection as query instances we searched for journals as classes, having potentially similar or related articles and topics. To evaluate the relevance, we determined the rank of the requested article's journal, assuming that our request, respectively article title, is a significant example for its journal. A complete automation of search and evaluation enables us to send a huge number of requests against the collection and to evaluate and fine tune the techniques. In this contribution we maintain our base approach of search and evaluation while adding search on abstracts, variations of similarity measures, new voting techniques, and evaluations considering our collection structure regarding the journal/article count distribution.

**Keywords:** Expert Search · Expertise Retrieval · IR Systems · Collection Search.

## 1 Motivation and Related Work

One of the approaches introduced in literature for expertise retrieval [2] is based on relevant documents retrieved by a search query. These documents vote for their associated authors as candidate experts. In this work, we transfer this approach to another domain: Our keyword search on a bibliographic collection yields matching articles which vote for the journals where the articles have been published, as beneficial sources or targets. Our aim is to identify a technique which yields and ranks journals that potentially contain other publications and resources which match the information need of the user. This information need targets journals rather than single articles.

In a first step, we don't evaluate the discussed techniques using manually created test data or test users. Instead, we use article-titles from the collection itself to automatically send these titles as search requests. Since we have the information in which journal a single article has been published, we can measure the position of this respective journal in the result ranking and evaluate the algorithms.

This work is based on research in data fusion techniques and their application in the field of expertise retrieval. Different approaches show, that combining multiple retrieval results using voting models can improve retrieval effectiveness [3]. In their survey, Balog et al. [4] present different approaches used in expertise retrieval including the document-based voting model. Rank- and score-based fusion techniques are listed and evaluated, mostly based on the work of MacDonald et al. [2]. Furthermore, normalization methods are applied for the underlying candidate expert profiles to gain better results. In the mentioned works, it becomes quite significant that the documents in the upper ranks together with their score values have a disproportionately high impact on the quality of the fusion results; exponential variants of fusion techniques can have better results and prove this fact [4].

In the paper at hand, we investigate how such approaches perform in our setting. We present significant extensions to our previous paper at [1]: We maintain our base approach of search and evaluation while adding search on abstracts, variations of similarity measures, new voting techniques, and evaluations considering our collection structure regarding the journal/article count distribution.

It should be mentioned that existing journal recommenders from publishers – like EndNote's manuscript matcher, Elsevier's journal finder, or Springer's journal suggester – are obviously related to our approach. However, these systems apply complex ranking schemes using much more information than our simple approach discussed in this paper. The aim of our paper is to investigate the capability of rather simple voting techniques in the sketched scenario – and similar scenarios such as company search based on their web pages [5] or the search for scholarly collections based on the single collection items [6]. Hence, a comparison with the existing journal recommenders might be an interesting next step but is out of scope for this paper.

## 2 Results and Conclusions from our Last Contribution

For our contribution in 2017 we took data from the dblp computer science bibliography<sup>3</sup>. dblp offers bibliographic metadata, links to the electronic editions of publications, and consists of nearly 3,800,000 publications. We restricted our investigations to journal articles, extracted from the offered dump. Based on this collection and our voting and search paradigm, CombMAX, taking only the first ranked article of each journal into account, yielded the best results regarding the journal ranking. For all measured searches, the utilized Votes algorithm yielded

<sup>3</sup> dblp Homepage, <http://dblp.uni-trier.de/>. Last accessed 4<sup>th</sup> Jun 2018

the worst results. Analyzing the voting process based on sums of articles' scores we concluded that an improved CombSUM technique considering only the upper ranking articles might deliver more accurate results than a CombSUM that takes all results into account. Furthermore, to emphasize the first articles' results, we planned to include RR (Reciprocal Rank) as a voting algorithm in our test runs which performed well in [8].

### 3 Collection and Setup

The experiments presented in this paper are based on a dump of 154,771,162 scientific papers offered by AMiner [7, 9]. In contrast to our preceding paper where the utilized collection mostly covered computer science subjects, this collection includes papers from conferences and journals across all sciences.

AMiner is a project from Tsinghua University, led by Jie Tang, which deals with search and mining of academic social networks and offers search/mining services for the academic community. Within this research, publication data from online databases including dblp bibliography, ACM Digital Library, CiteSeer, and others are merged [7]. The complete collection is divided in three compressed files containing the data in json format which can be downloaded from the homepage. Not all records contain the complete metadata; fields like issue or abstract are not filled reliable. For our experiment we parsed an extract of 2,000,000 articles and gained 864,330 articles which had an abstract, corresponding to 38,145 journals. These extracted articles were sent to Elasticsearch where we carried out the experiments.

### 4 Applied Techniques

Our search consists of two parts: first, we are searching over the collection *titles* like we did in case of the dblp collection. In a second step, again we take the removed titles from the collection and use them to search over the *abstracts*. For similarity measures we use Elasticsearch's TF/IDF similarity and in addition three variations of BM25 similarity. These constellations lead to combinations of voting techniques, field searches, and similarity measures shown in table 1.







Throughout all techniques and levels we applied Elasticsearch's built in stop-word elimination and stemming mechanisms on similarity measure level.

#### 4.1 Applied Similarity Measures for document ranking

For flat, title-based article search – i.e. the underlying document ranking – we use Elasticsearch's TF/IDF and BM25 algorithm in three parametrizations.

**Similarity based on TF/IDF:** The  $score(d, q)$  of a document  $d$  given a query  $q$  which consists of terms  $t$  is computed as  $score(d, q) = \sum_{t \in q} (tf(t, d) \cdot idf(t^2)) \cdot$

**Table 1.** Combinations of voting techniques, search fields, and similarity measures resulting in  $6 \cdot 2 \cdot 4 = 48$  possible search constellations

| Colour  | Voting Technique | Search Field | Similarity Measure  |
|---|------------------|--------------|---------------------|
|  | Votes            |              |                     |
|  | CombSUM          |              |                     |
|  | CombSUM TOP 10   | Abstract,    | TF/IDF,             |
|  | CombSUM TOP 5    | Title        | BM25 (3 variations) |
|  | CombMAX          |              |                     |
|  | RR               |              |                     |

$norm(d)$ ). The term frequency  $tf$  describes the frequency of term  $t$  in document  $d$  and is defined as  $tf(t, d) = \sqrt{frequency}$ . The inverse document frequency for  $t$  across the collection is computed as  $idf(t) = 1 + \log\left(\frac{numdocs}{docFreq(t)+1}\right)$  where  $numdocs$  is the number of documents in the collection and  $docFreq(t)$  is the number of documents containing term  $t$ . In addition, a document length normalization is added with  $norm(d) = \frac{1}{\sqrt{numTerms}}$ .

**Similarity based on BM25:** For BM25, the  $score(d, q)$  is computed as  $score(d, q) = \sum_{t \in q} \left( idf(t) \cdot \frac{tf(t, d) \cdot (k+1)}{tf(t, d) + k(1-b + b \cdot \frac{|D|}{avgdl})} \right)$ .  $|D|$  represents the document length, and  $avgdl$  is computed as the average document length over all documents in the collection. The inverse document frequency  $idf$  for term  $t$  is computed as  $idf(t) = \log\left(1 + \frac{numDocs - docFreq(t) + 0.5}{docFreq(t) + 0.5}\right)$  with  $numDocs$  and  $docFreq(t)$  defined as before. In our experiments, we use one standard parameterization for BM25 ( $k = 1.2$  and  $b = 0.75$ ) complemented by two variations. Variation 1:  $k = 3$  and  $b = 0.1$ ; Variation 2:  $k = 3$  and  $b = 1.0$ .

## 4.2 Applied Voting Techniques for journal ranking

Based on the article ranking six approaches introduced in expert search are adopted to derive a journal ranking, respectively, a collection ranking. In general, the voting model can be based on different inputs: the *number* of items in the search result associated with a collection, the *ranks* of the items associated with a collection, and the *score values* calculated for the items associated with a collection [2].

Let  $R(q)$  be the set of articles retrieved for the query  $q$  and  $score(j, q)$  the computed score for journal  $j$  and query  $q$ , we apply six different voting models:

**Votes:** This metric takes the number of found articles for every journal as the score:  $score_{Votes}(j, q) = |\{art \in R(q) \cap art \in j\}|$

**CombSUM:** For every journal, CombSUM sums up the scores of the articles:  $score_{CombSUM}(j, q) = \sum_{art \in R(q) \wedge art \in j} score(art, q)$

**CombSUM TOP n:** For every journal, this aggregation sums up the top  $n$  (in our work:  $n \in \{5; 10\}$ ) scores of the articles for each journal:

$$score_{CombSUM\ TOP\ n}(j, q) = \sum_{art \in R(q) \wedge art \in j \wedge rank(art, j) \leq n} score(art, q).$$

$rank(art, j)$  represents the *rank* of *art* if only articles of *j* are considered.

**CombMAX:** This metric takes the first result stemming from *j*, respectively, the article with the highest ranking, as voting candidate with its score:

$$score_{CombMAX}(j, q) = \max(\{score(art, q) \mid art \in R(q) \wedge art \in j\})$$

**RR:** This algorithm takes the results from the underlying similarity measure in their order and revalues them by applying values of a harmonic series:

$$score_{RR}(j, q) = \sum_{art \in R(q) \wedge art \in j} \frac{1}{rank(art, q)}$$

## 5 Experimental Results

In this section we start presenting the results across all applied voting techniques, initially disregarding the structure of the collection (journals and their associated article count).

**Title Search:** First experimental results are gained by searching with 10,000 randomly removed titles over the remaining 854,330 titles of the collection. Table 2 shows the results for all applied voting techniques based on TF/IDF and BM25 as similarity measure. The values in the 1<sup>st</sup> quartile column specify the rank for which 25% of the queries have ranked the journal from which the query was originally taken at least as good. Similarly, the median and 3<sup>rd</sup> quartile columns state the rank under which 50%, respectively, 75% of the searches include the searched journal. As guessed in our last paper, voting techniques emphasizing

**Table 2.** Results for search over titles, TF/IDF and BM25

| Similarity Measure ▷ | TF/IDF                   |        |                          | BM25                     |        |                          |
|----------------------|--------------------------|--------|--------------------------|--------------------------|--------|--------------------------|
|                      | 1 <sup>st</sup> quartile | median | 3 <sup>rd</sup> quartile | 1 <sup>st</sup> quartile | median | 3 <sup>rd</sup> quartile |
| Votes                | 19                       | 102    | 341                      | 19                       | 102    | 341                      |
| CombSUM              | 13                       | 72     | 275                      | 12                       | 70     | 269                      |
| CombSUM TOP 10       | 3                        | 24     | 149                      | 3                        | 22     | 145                      |
| CombSUM TOP 5        | 3                        | 23     | 134                      | 3                        | 21     | 128                      |
| CombMAX              | 8                        | 49     | 206                      | 7                        | 43     | 194                      |
| RR                   | 5                        | 29     | 127                      | 4                        | 25     | 118                      |

the first ranks of the underlying article search perform better than methods which include all article results like Votes or CombSUM. While CombMAX, considering only the first article result for each journal, performed best in our last paper, the newly utilized CombSUM TOP  $n$  voting techniques involving only the first  $n$  results per journal yield even better results. RR, rescoring the articles by their order with values of a harmonic series, produces nearly similarly good results. For BM25 we applied the standard parameterization. Though BM25

performs slightly better than TF/IDF, this new experiment series confirms our assumption that the underlying similarity algorithm (retrieval model) does not fundamentally change the ranking behaviour of the collection ranking methods.

**Consideration of Abstracts:** In contrast to the dblp dataset, the AMiner collection contains abstracts which we include in our search experiments. Table 3 shows the results for getting the associated journal. Please note that we search with 10,000 titles as queries in a collection only consisting of abstracts in this scenario. Regarding the Votes and CombSUM aggregations, rankings, and there-

**Table 3.** Results for search over abstracts, TF/IDF and BM25

| Similarity Measure $\triangleright$ | TF/IDF                    |                          |        | BM25                     |                          |        |                          |
|-------------------------------------|---------------------------|--------------------------|--------|--------------------------|--------------------------|--------|--------------------------|
|                                     | Voting Technique $\nabla$ | 1 <sup>st</sup> quartile | median | 3 <sup>rd</sup> quartile | 1 <sup>st</sup> quartile | median | 3 <sup>rd</sup> quartile |
| Votes                               |                           | 39                       | 177    | 451                      | 39                       | 177    | 451                      |
| CombSUM                             |                           | 21                       | 113    | 355                      | 20                       | 111    | 348                      |
| CombSUM TOP 10                      |                           | 3                        | 20     | 113                      | 2                        | 17     | 103                      |
| CombSUM TOP 5                       |                           | 3                        | 20     | 106                      | 3                        | 17     | 95                       |
| CombMAX                             |                           | 9                        | 50     | 208                      | 7                        | 42     | 189                      |
| RR                                  |                           | 5                        | 29     | 122                      | 4                        | 25     | 113                      |

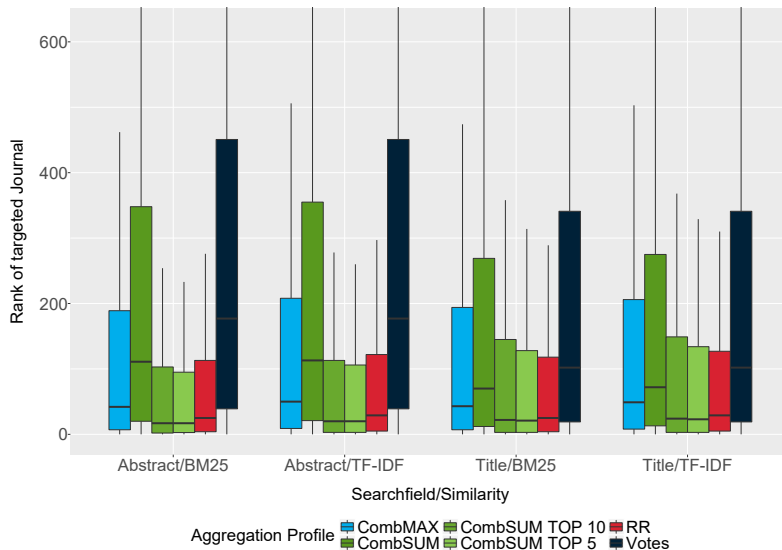
fore the result quality decrease significantly. An noteworthy enhancement can be observed regarding the 3<sup>rd</sup> quartile of CombSUM Top 10 and CombSUM Top 5. Again, in case of abstracts, the choice of the underlying similarity measure between TF/IDF and BM25 does not change the ranking considerably.

Figure 1 shows the overall results. In case of abstracts, aggregation techniques like CombSUM and Votes show significantly worse results, whereas CombSUM TOP 10 and CombSUM TOP 5 yield better results than RR which performs better for the search on titles.

**Impact of Journal Size:** In a second step the interest goes to how the voting techniques perform dependent on the article distribution over the journals collection. Keeping the introduced search fields, similarity measures, and voting techniques, we divided our test queries regarding the article count of the respective journal of origin. We examined the classifications shown in table 4.

**Table 4.** Classifications of 10,000 requests by journal article count

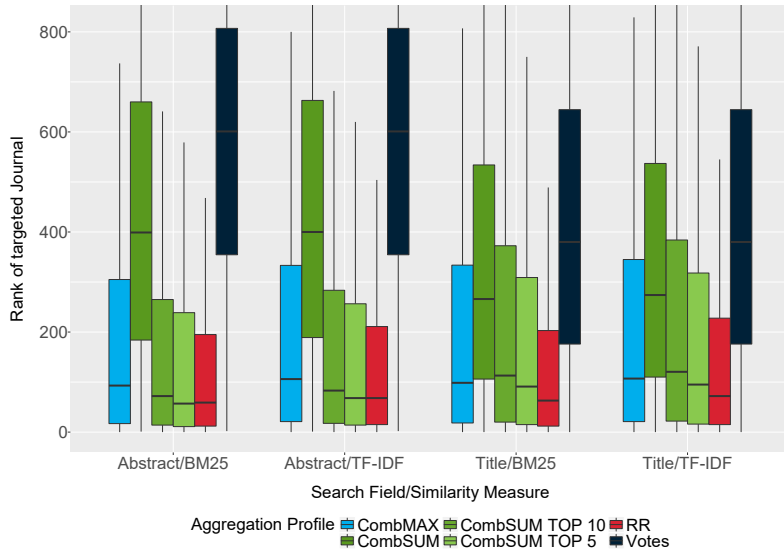
| Number of Requests | Articles Available for the Journal |             |
|--------------------|------------------------------------|-------------|
|                    | Lower Bound                        | Upper Bound |
| 5,203              | 2                                  | 99          |
| 3,440              | 100                                | 499         |
| 680                | 500                                | 999         |
| 677                | 1,000                              | 4,999       |



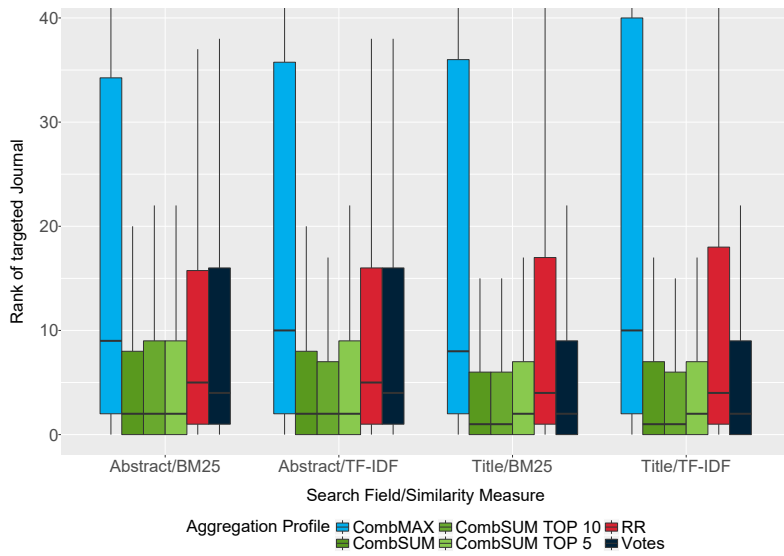
**Fig. 1.** Box plots for the ranks of the journals from which the respective query title was taken

Figure 2 shows the performance of requests having up to 100 articles in their corresponding journal. Across all utilized ranking techniques, results are inferior to the statistics regarding the entire collection (Fig. 1). Requests having 100 up to 499 corresponding articles achieve ranking results comparable to the overall statistics. Regarding increasing values for the number of associated articles, it turns out that the ranking results get better for all techniques. Further, considering Fig. 3, CombSUM using the complete sum of the scores and Votes using the number of results for each journal, perform better than CombMAX and RR which have a strong emphasize on the first result.

**System Behaviour and Average Rank Distribution:** In a next step we investigate if any of the introduced techniques fundamentally favours certain requests dependent on their corresponding journal’s article count. In order to compare probability of relevance with probability of retrieval, we divided the collection into 40 bins. At an amount of 864,330 articles, each bin gains a capacity of 21,608. The 38,145 journals from the collection are ordered by their number of articles and passed through in ascending order. As long as the cumulated sum of the journals’ articles does not exceed  $n \cdot 21,608$ , a journal gets associated with the current  $n^{th}$  bin. Fig. 4 shows the distribution of the 38,145 journals over the bins. We compared the number of queries taken from each bin (corresponds to probability of relevance) and the number of Top 1 results stemming from each bin (corresponds to probability of retrieval) for our 10,000 title queries. Fig. 5 shows these two distributions over the bins for the RR voting technique with underlying

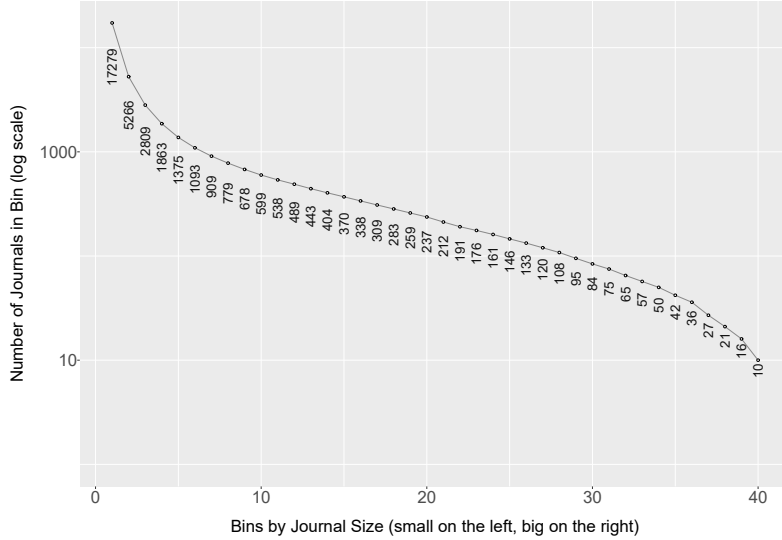


**Fig. 2.** Box plots for the ranks of the journals from which the respective query title was taken for requests having up to 99 articles for their associated journal



**Fig. 3.** Box plots for the ranks of the journals from which the respective query title was taken for requests having 1,000 to 5,000 articles for their associated journal





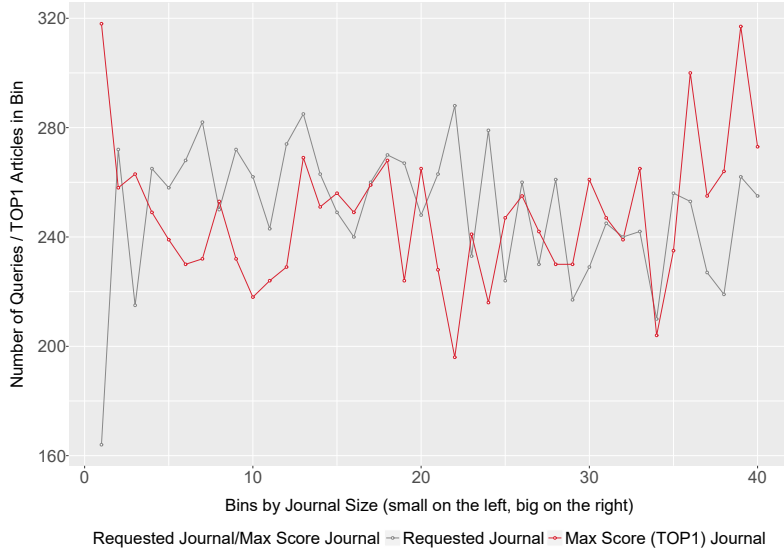
**Fig. 4.** Journal distribution over the calculated bins

TF/IDF similarity measure. Regarding RR, CombMAX, and the CombSUM TOP  $n$  techniques, no basic and systematic deviations of favourizations can be observed.

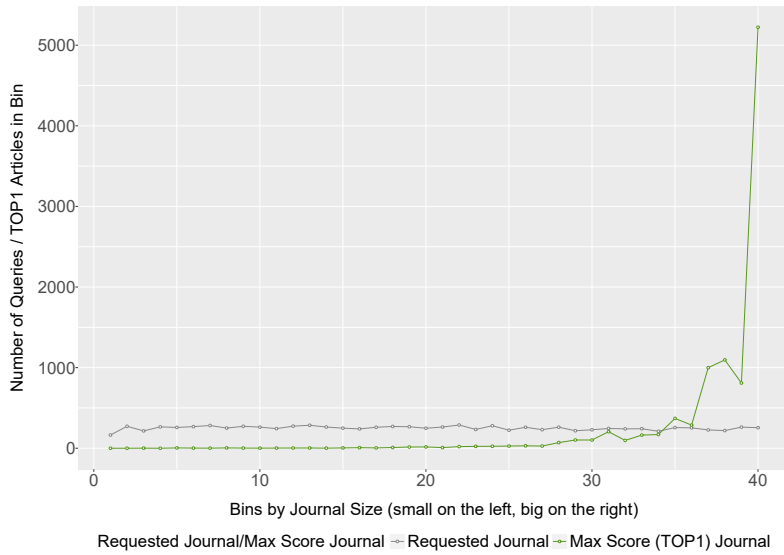
CombSUM and Votes show a significant tendency yielding journals with a high amount of articles as the Top 1 result. Fig. 6 shows an exemplary chart for CombSUM in combination with TF/IDF. Whereas the grey graph as the requested journals' article count is equal to Fig. 5, CombSUM almost exclusively yields journals having a high number of articles as top results. The same effect occurs applying the Votes technique.

**Average Rank Distribution:** Based on the described bin distributions we also look at the query results and their average scoring in each bin. Fig. 7 shows the average ranks for all applied voting techniques using TF/IDF similarity. Led by RR, voting techniques emphasizing the upper ranks yield the best average ranking results in lower bins (small journals). For the last ten bins, this tendency is reversed and the aggregating voting techniques like CombSUM and Votes are predominant.

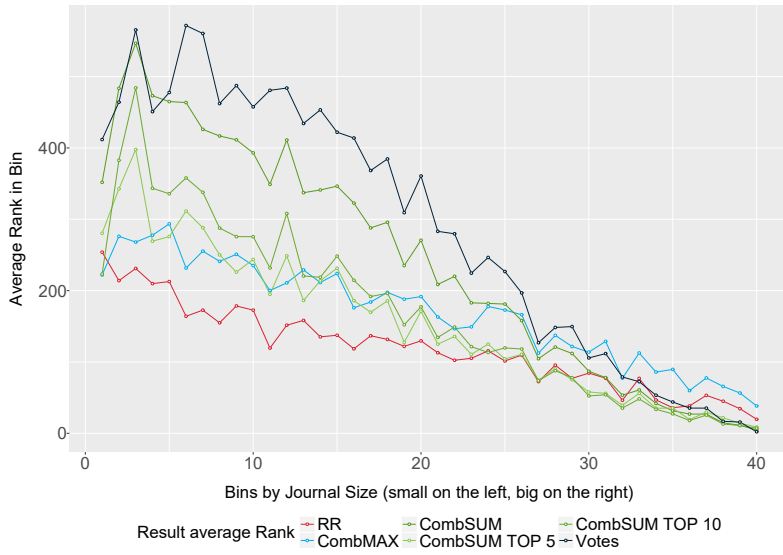
**BM25 Parameter Change:** In our experiment applying BM25 we also changed the parameter values for  $k$  and  $b$  as shown in section 4.1. Notably worse results could only be gained in case of abstracts, setting  $b = 0.1$ , dampening the document length normalization. Figure 8 shows the overall results considering all applied techniques.



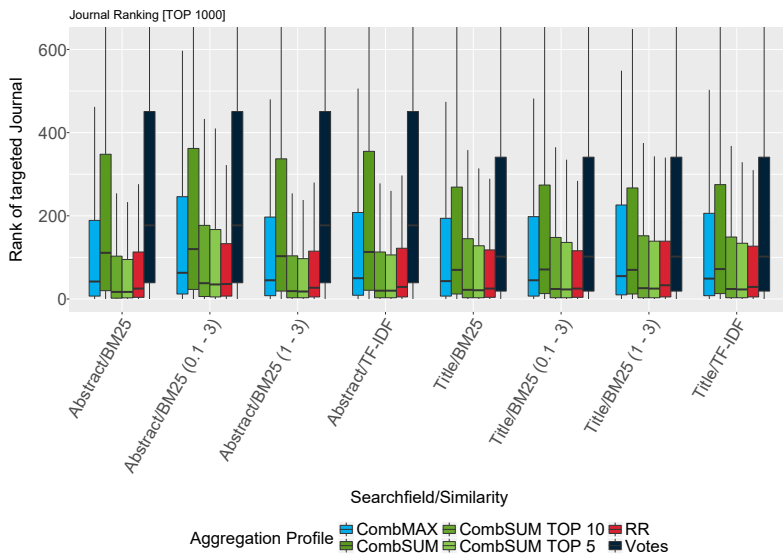
**Fig. 5.** Number of queries taken from each bin (Requested Journal) vs. number of Top 1 results stemming from each bin (Max Score). Exemplary chart for RR voting technique and TF/IDF similarity measure.



**Fig. 6.** Number of queries taken from each bin (Requested Journal) vs. number of Top 1 results stemming from each bin (Max Score). Exemplary presentation for CombSUM voting technique and TF/IDF similarity measure.



**Fig. 7.** Distribution of the average rank of the correct journal over the calculated bins for title search and TF/IDF similarity measure



**Fig. 8.** Box plots for the ranks of the journals from which the respective query title was taken considering variations of BM25 parametrization

## 6 Conclusion and Future Work

The search experiments over the AMiner collection confirm the results from our dblp study: Summing up all found articles as voting candidates for their journal by adding their scores or number does not perform well.

CombMAX as the best performing technique in our last study is now outperformed by RR, CumbSUM TOP 10 and TOP 5. Regardless of the applied underlying similarity measure it turns out that the leading, top ranked articles as voting candidates provide the best results: Whereas RR yields the best ranking results regarding journals with few articles, the CumbSUM TOP  $n$  aggregations perform best when requesting journals with a high amount of articles.

Regarding the journals' article count, modifying RR and CombSUM TOP  $n$  by applying a correction factor or taking dynamically more or less voting candidates into account could be an interesting alternative.

Contrary to our expectations, the search over abstracts does not yield significantly better results across all techniques. Voting techniques emphasizing the first articles as voters yield slightly better rankings regarding the median and 3<sup>rd</sup> quartile. As a modification, we plan to extract the top words out of an article's abstract and utilize them as metadata for our search experiments.

## References

1. Henrich, A., Wegmann, M.: Searching an Appropriate Journal for your Paper – an Approach Inspired by Expert Search and Data Fusion. In: Lernen, Wissen, Daten, Analysen (LWDA) Conference Proceedings, Rostock, Germany, September 11-13, 2017.
2. Macdonald, C., Ounis, I.: Searching for Expertise: Experiments with the Voting Model. In: THE COMPUTER JOURNAL, Vol. 52 No. 7, pp. 729 - 748. Published by Oxford University Press on behalf of The British Computer Society 2008.
3. Lee, Joon Ho: Analyses of multiple evidence combination. In: Proc. 20th annual international ACM SIGIR conf. on Research and development in information retrieval (SIGIR '97). ACM, New York, NY, USA, pp. 267-276.
4. Balog, K., Fang, Y., de Rijke, M., Serdyukov, P., Si, L.: Expertise Retrieval. In: Foundations and Trends in Information Retrieval, Vol. 6, pp. 180-184, 2012.
5. Blank, D., Boosz, S., Henrich, A.: IT company atlas upper Franconia: a practical application of expert search techniques. In: Proc. of the 31st ACM Symposium on Applied Computing (SAC '16). ACM, New York, NY, USA, pp. 1048-1053.
6. Gradl, T., Henrich, A.: A novel approach for a reusable federation of research data within the arts and humanities. In: Digital Humanities 2014 – Book of Abstracts. EPFL - UNIL. Lausanne, Switzerland, pp. 382-384.
7. Tang, J., Zhang, J., Yao, L., Li, J., Zhang, L., Su, Z.: ArnetMiner: Extraction and Mining of Academic Social Networks. Proc. ACM SIGKDD International Conf. on Knowledge Discovery and Data Mining (SIGKDD'2008). pp. 990-998.
8. Macdonald, C., Ounis, I.: Voting for candidates: Adapting data fusion techniques for an expert search task. In: Proc. of the ACM International Conf. on Information and Knowledge Management, CIKM '06, New York, NY, USA, pp. 387-396, 2006.
9. Open Academic Graph Homepage, <https://aminer.org/open-academic-graph>. Last accessed 4<sup>th</sup> Jun 2018