UNIVERSITY OF BAMBERG

Classification and Class Search Using Voting Techniques

BY

Markus Wegmann

Faculty of Information Systems and Applied Computer Sciences at the Otto Friedrich University of Bamberg

Bamberg 2025

URN: urn:nbn:de:bvb:473-irb-1065781
DOI: https://doi.org/10.20378/irb-106578

Diese Arbeit hat der Fakultät WIAI der Otto-Friedrich-Universität Bamberg als Dissertation vorgelegen.

Gutachter: Prof. Dr. Andreas Henrich

Gutachter: Prof. Dr. Guido Wirtz

Tag der mündlichen Prüfung: 31.01.2025

Abstract

Beyond the search for content and information, such as texts and documents of all types in information retrieval, class-based search provides information on suitable, superordinate categories, suitable sources, and appropriate generic terms or classifications based on document rankings and their properties.

The starting point and basis of this thesis is existing research on voting techniques in expertise retrieval, which addresses the search for suitable experts on a searched topic based on the relevant and ranked documents of a collection. In this thesis, the concept of expert search is derived, generalized, and referred to as a class search. To find these suitable superordinate classes in response to a query, associated scored and ranked documents with their properties vote for them. This voting process does not correspond to the classic electoral voting systems, but uses voting techniques that aggregate the probative value of voting documents for each class in different ways.

Following theoretic analyses, the properties of conventional and, in the course of the work, also new voting techniques are examined in a larger experimental setup, and their applicability in general scenarios is investigated. In addition to their use at the document level, voting techniques are further investigated at the level of document passages; the relevance of document rankings resulting from passages that vote for their documents is examined.

In addition to similarity measures, the applicability of voting techniques is also examined and evaluated at the level of distance measures. Using the example of hierarchical clustering, the application of voting techniques is related to known clustering techniques, and their systemic behavior is analyzed.

With its theoretical considerations and evaluations of practical scenarios, this work provides a broad overview of voting techniques, their characteristics, differences, and their diverse application scenarios.

Before writing this thesis, the results were published on an ongoing basis. Parts of this thesis that refer to relevant publications are marked as such at the respective points. The following is a list of the publications in chronological order.

- A Henrich and M Wegmann. 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*. Ed. by M Leyer. Vol. 1917. CEUR Workshop Proceedings. CEUR-WS.org, 2017, p. 253. https://ceur-ws.org/Vol-1917/paper34.pdf
- M Wegmann and A Henrich. Search for an Appropriate Journal - In Depth Evaluation of Data Fusion Techniques. In: *Proceedings of the Conference "Lernen, Wissen, Daten, Analysen", LWDA 2018, Mannheim, Germany, August 22-24, 2018*. Ed. by R Gemulla et al. Vol. 2191. CEUR Workshop Proceedings. CEUR-WS.org, 2018, pp. 343–354. http://ceur-ws.org/Vol-2191/paper41.pdf
- A Henrich and M Wegmann. Search and evaluation methods for class level information retrieval: extended use and evaluation of methods applied in expertise retrieval. In: *SAC '21: The 36th ACM/SIGAPP Symposium on Applied Computing, Virtual Event, Republic of Korea, March 22-26, 2021*. Ed. by C Hung et al. ACM, 2021, pp. 681–684. https://doi.org/10.1145/3412841.3442092

Zusammenfassung

Hinausgehend über die Suche nach Inhalten und Informationen, wie z.B. Texte und Dokumente aller Art im Information Retrieval, liefert die klassenbasierte Suche Informationen über geeignete, übergeordnete Kategorien, passende Quellen und geeignete Oberbegriffe bzw. Klassifizierungen auf der Basis von Dokumentenrankings und deren Eigenschaften.

Ausgangspunkt und Grundlage dieser Thesis ist die bestehende Forschung zu Voting-Techniken im Expertise Retrieval, die sich mit der Suche nach geeigneten Experten zu einem Thema auf Basis relevanter und gerankter Dokumente einer Kollektion befasst. Das Konzept der Expertensuche wird in der Thesis abgeleitet, verallgemeinert und als Suche nach Klassen bezeichnet. Um diese geeigneten Klassen als Antwort auf eine Anfrage zu finden, voten die bewerteten und gerankten Dokumente mit ihren Eigenschaften für ihre zugehörige Klasse. Dieser Voting-Prozess entspricht nicht den klassischen Wahlsystemen, sondern verwendet Voting-Techniken, die die einzelnen Relevanzbewertungen der für ihre Klasse abstimmenden Dokumente auf unterschiedliche Weise aggregieren.

Nach theoretischen Analysen werden die Eigenschaften bekannter und auch neu entwickelter Voting-Techniken in einem größeren Szenario evaluiert und ihre Anwendbarkeit in allgemeinen, klassenorientierten Szenarien untersucht. Neben ihrer Anwendung auf Dokumentenebene werden Voting-Techniken weiterhin auf Ebene von Dokumentpassagen evaluiert; die Relevanz von Dokumenten-Rankings, die aus den für ihre zugehörigen Dokumente votenden Passagen resultieren, wird untersucht.

Neben Ähnlichkeitsmaßen wird die Anwendbarkeit von Voting-Techniken auch auf der Basis von Distanzmaßen untersucht und bewertet. Am Beispiel des hierarchischen Clusterings wird die Anwendung von Voting-Techniken auf bekannte Clustering-Techniken adaptiert und deren systemisches Verhalten analysiert.

Mit ihren theoretischen Überlegungen und Auswertungen praktischer Szenarien gibt die Arbeit einen breiten Überblick über

Voting-Techniken, ihre Eigenschaften, Unterschiede und ihre vielfältigen Anwendungsszenarien.

Acknowledgements

I would especially like to thank my doctoral supervisor Prof. Dr. Andreas Henrich for his support, patience, and understanding over the years. Discussions, comments, disagreements, and words of support from him have always motivated me and helped me move forward.

I also thank Prof. Dr. Daniela Nicklas and Prof. Dr. Guido Wirtz for joining the thesis committee and assessing this dissertation.

I additionally would like to thank the members of the PhD round-table, whose many discussions and conversations have considerably broadened my perspective and inspired me over the last years. I particularly would like to mention Felix Engl, Martin Bullin, and Leon Martin, who supported me with tips on presentations, organizational issues, subject-specific questions and proofreading this thesis.

I would like to thank my wife and children for their support for my project and understanding during times when I could not pay the usual attention to them.

Contents

1 Introduction

1.1 Motivation

The starting point for this thesis is the search for experts in research or in enterprises. In particular, in larger organizations, the search for suitable experts who can support tasks with their specialist knowledge arises. An approach to finding experts is the document-based approach, in which documents in a collection are associated with one or more authors as candidates for the requested expertise. Balog et al. give in their survey "Expertise Retrieval" [Bal+12] an overview of document-based approaches, including voting techniques. The basis for the application of voting techniques is the search for an expert topic in a collection of documents. The documents in the resulting ranking then vote for their associated candidate(s) as potential experts. The result is a ranking of suitable experts for the requested topic derived from the votes of the documents[1].

Basis of this thesis: Research in the field of voting techniques in expertise retrieval

1: Balog et al. define expertise retrieval as the linking of humans to expertise areas, while expert search is the direct search for a person with specific expertise [Bal+12]. We use both terms synonymously in this thesis.

A milestone in this domain is established by Macdonald with his work "The voting model for people search" [Mac09], in which he applies and evaluates variants of voting techniques for expertise retrieval.

Building on the application in expertise retrieval, the document-based approach and the use of voting techniques are conceivable in other scenarios:

Further application scenarios for voting techniques:

As an example, the search for suitable providers or suitable companies can be performed by indexing their websites as documents and voting for their associated companies in response to queries. The search for suitable bibliographic collections can be realized based on their indexed documents by having their documents vote for their associated collection in response to a query. Matching threads in forums can be found by having their posts vote for their associated thread in response to a search query.

Suitable companies, bibliographic collections or forums

In addition to searching for the appropriate company, collection, or forum to be found, as mentioned in the examples, the request itself

Classification of the request itself

can be classified according to the search result. A conceivable scenario is a classification of e-mails based on existing messages that are already assigned to categories. Thus, classifications according to properties such as spam or thematic affiliations are possible.

Further analysis of known and implementation of new voting techniques

In addition to the other conceivable scenarios of an application of the document-based approach in conjunction with voting techniques, the thesis examines the properties and behavior of known techniques and develops proposals for additional voting techniques in the course of the thesis. We examine the various factors that influence voting techniques and identify parameters that are of decisive importance for the quality of the rankings.

Evaluation scenario: The search for a journal suitable for the requested topic or for the publication of an article on the topic

The following scenario is chosen as a concrete example and evaluation scenario in this thesis: We are looking for a journal that fits our query thematically, for example, to publish an article in it or to search for suitable articles. For this purpose, we use a collection of scientific papers published by AMiner [Tan+08] in our setup and test scenario in Chapter 5.

1.2 Problem Description

In this thesis, we generalize the principle of expert search and call it the *search for classes*. The search for categories or classes in which collections of documents can be divided is pursued. Exemplary applications of a search described above each represent instances of these classes.

Adoption of the probabilistic approach

Although the generalization of the problem is based on expert search, the basic assumptions remain the same. Smirnova and Balog describe a widely used model of the probabilistic and document-centered approach to voting techniques in expertise retrieval in their article "A User-Oriented Model for Expert Finding" as follows [SB11]:

> *"The key idea behind the model is to simulate how a user may search for experts using a standard document search engine: first, finding documents which are relevant, and then, examining each of these documents for associated persons. By scanning through a number of documents, the user can obtain an idea of which people are more likely to be experts on the query topic."*

This document-centered probabilistic approach separates the query from the candidate by assuming conditional independence between the two. Reference to associated candidates is established only via the matching documents.

Following Smirnova and Balog, we transform this probabilistic approach to expertise retrieval into a general and class-based form:

The search for a suitable class c, matching a query q, can be regarded as a probability $p(c|q)$. According to the Bayes rule, the equation is then formulated as follows:

$$p(c|q) = \frac{p(q|c) \cdot p(c)}{p(q)}$$

or from a ranking perspective:

$$p(c|q) \propto p(q|c) \cdot p(c) \tag{1.1}$$

In addition, considering the explanation of Smirnova and Balog, the relationship between a query q and the class c is the weighted sum of the relevance scores $p(q|d)$ of the documents related to a class:

$$p(q|c) = \sum_{d} p(q|d) \cdot p(d|c) \tag{1.2}$$

Under the assumption of the quoted work that $p(c)$ in equation 1.1 has a uniform probability distribution, equation 1.2 would represent the final ranking in a general class-based scenario.

Ideal behavior not for real collections

There are aspects in real-world scenarios that the model does not properly take into account and that lead to unintended behavior. For example, within collections, classes are not evenly represented by documents and have different numbers of documents. This means that the likelihood $p(q|c)$ of equation 1.2 as a sum of probabilities leads to a bias that potentially values classes with many documents higher in total, and thus favors them.

Model favors classes with many documents.

Summed probabilities are not independent.

The addition of individual document probabilities voting for a class is only meaningful to a limited extent, since they are not independent probabilities. For example, the probabilities of two voting documents d_1 and d_2 for a class c are not mutually exclusive; rather, part of the probative value of document d_2 may already be contained and reflected in the preceding document d_1. We counteract this with damping factors for the summands of d, which are also intended to compensate for the bias caused by classes with many documents.

Associations of documents with classes

The probability that document d is associated with class c is expressed by $p(d|c)$ in equation 1.2. In the example of TREC collections [CVS05; SVC06], these associations are made from the participation or mention of experts in documents. Using the example of classes, documents are declared as associated or not, but

values between 1 and 0 are also possible here.
This is a major difference from the expertise retrieval model, in which this probability is specified as an association with the candidate ca with $p(d|ca)$. In expertise retrieval, a candidate is, according to the prerequisite, involved in the document d with its knowledge and its individual contribution, while in the general case of a class-based search, a document d is usually associated with the class c and is assumed to be a typical example of this class. In this thesis, we investigate to what extent this association of document d with candidate ca can be transferred to a general scenario as an association with a class c.

Collections have certain specific characteristics, such as an even or uneven distribution of documents between classes. Voting techniques react differently and produce results of varying quality, depending on the collection structure. Thus, the results of the evaluations of expertise retrieval can also be based only on the properties of these collections. In this thesis, we highlight and evaluate the general properties and validity of the qualities of voting techniques.

Evaluation for new and individual collections

In addition to the analysis of known techniques and new techniques developed in the course of the thesis and their behavior in different collection structures, the evaluation methodology plays an important role. During research on voting techniques applied in expert search, two collections of TREC tracks were used mainly for evaluation [Bai+07; CVS05], unfortunately involving additional challenges due to ambiguities and uncertainties in the association of documents with experts. Evaluation of class-based search results is not trivial as results are difficult to judge and as there are only view collections that have relevance assessments. In our evaluation scenario in Chapter 5 Search for Classes - Evaluation of the Introduced Voting Techniques, we choose a different approach that makes uncertainties such as noise and inaccuracies negligible due to the high number of evaluated results.

Generalization of the problem as baseline research

The challenges are elaborated and described in detail in Section 1.4.3 Influencing Factors of Voting Techniques on page 8. However, the search for ideal classes is not examined in this thesis according to practical, accompanying conditions such as availability, actuality, popularity, financial or geographical aspects, as different influencing and individual factors are important for each example and instance of a class-based search.

1.3 Thesis Structure

In the remainder of Chapter 1, voting techniques are categorized in the context of information retrieval, and their application in the general scenario of class search is established. The basic approach is explained, and factors that influence the class search are presented. These factors include document properties and their indexing, the search for documents, processing of the result set, and the subsequent voting process for classes based on the documents found. Section 1.5 of this chapter addresses research questions regarding the generalized application of voting techniques and the contribution of the thesis.

Chapter 1: Introduction

Following the introduction, Chapter 2 addresses the theoretical background of voting techniques. Techniques known from expertise retrieval are shown with their algorithmic and functional aspects using a virtual scenario as an example, in which the basic characteristics of each technique are also outlined.

Chapter 2 Voting Techniques: Theoretical Background

The current state of research and the literature relevant to this thesis are summarized in Chapter 3. Aspects of voting techniques applied in expertise retrieval that can be transferred to a general scenario of searching for classes and which are also relevant in this domain are presented and discussed.

Chapter 3 Voting Techniques: Applications from the Literature

Further developed voting techniques are presented in Chapter 4. In the course of evaluation and research, we have modified existing voting techniques and developed new techniques, which we present and describe with their characteristics using the example scenario in Chapter 2.

Chapter 4 Extended Techniques and Interrelationships

In Chapter 5, we evaluate techniques explained and discussed in Chapters 2 and 4 using a scenario based on a bibliographic collection. We develop three different search scenarios and apply newly introduced types of evaluation. After discussing and evaluating the results in Section 5.4 Experimental Results, we evaluate further parameters in Section 5.5 Systemic Behavior of the Applied Techniques that are influential on the success of voting techniques for class-based search.

Chapter 5 Search for Classes - Evaluation of the Introduced Voting Techniques

In Section 5.6 Principle of Inclusion and Exclusion, we consider a further technique for aggregating documents of a class based on the addition of probabilities. The central problem is the transferability of the addition of probabilities to an aggregation of BM25-based scored and ranked documents per class. The performance of the developed technique is evaluated using the previous class search scenario.

Chapter 5 closes with another example from the literature in which the voting-based search for relevant threads in a large travel forum

is discussed. We use this example to test the performance of the newly introduced techniques in comparison to known techniques from the literature.

Chapter 6 Passages as Voters for their Documents

In Chapter 6, the voting scenario is no longer viewed from a class perspective, but from a document perspective: Documents are divided into passages, which then vote for their documents in response to a query. Using various collections, the evaluation shows to what extent a ranking with higher relevances can be achieved with this technique compared to a classic search using BM25.

Section 6.4 Investigations Using Pseudo-Collections uses an artificially generated pseudo-collection to examine the behavior of the voting techniques based on the document structure and the term distribution.

Chapter 7 Voting Techniques Applied to Clustering Algorithms

Based on distance measures and not similarity measures, the transferability of voting techniques to agglomerative clustering is evaluated in Chapter 7. The idea here is that each point distance between two clusters can be seen as a voting candidate, and thus the distances between clusters can be determined for different voting techniques. We evaluate this consideration using example scenarios with classic benchmark data sets.

Chapter 8 Conclusion

In the final chapter, we categorize the evaluations from the previous chapters and give an overview of all voting procedures and their transferability to the universal scenario, the search for classes. The objectives of the thesis and its contribution are summarized, and a further outlook on scenarios and the transferability of voting techniques is given.

1.4 Voting Techniques in the Context of Information Retrieval

Data fusion techniques as the origin of voting techniques

In the area of information retrieval, voting techniques represent a manifestation of data fusion techniques that were originally designed to transform results or rankings from different sources of evidence into a single ranking that effectively represents the relevancies of the results as a common ranking [Bal+12]. In this context of information retrieval and data fusion, the terms *meta-search* or *combining evidence* are also used to manifest the results of different systems in a ranking.

Voting techniques, based on the general principle of data fusion techniques, do not merge the rankings of several sources into a single ranking but consider the documents of a single ranking as individual voters who vote for a class associated with the document.

This thesis discusses and evaluates voting techniques that find their origin in data fusion techniques such as CombSUM or CombMNZ [Wu12], which will be discussed and evaluated in the course of this thesis, among other established and newly proposed techniques.

Application of voting techniques in expertise retrieval

Voting techniques have been explored in many articles, especially in the field of expertise retrieval by Macdonald et al. [Mac09; MO06]: Expertise retrieval deals with the search for experts on a query topic in both companies and academia. Documents related to candidates as potential experts provide their evidence of expertise and vote for their associated candidate(s) in response to a query. This leads to an expert ranking in response to the query.

Generalization of the expert search to the class search

Generalization of combining evidence In this thesis, the principle of combining evidence is generalized, and the notion of expert search is extended and called *class search*. Each document in a collection has, based on its membership, a general association with a class for which it votes during the voting process.

1.4.1 Searching for Classes as an Information Need

The search for classes is based on a collection of documents, each of which can be assigned to one or more classes. A search query is formulated to the collection in order to find suitable and matching classes. Subsequently, a determined document ranking on the basis of similarity measures provides documents that belong to clusters on the basis of their properties or associations, and result in a class ranking.

Example scenarios of a class-based search

As a generalization of the explored voting-based expert search, the search for classes can have many different scenarios:

Search for a suitable forum thread

In the work of Albaham et al., the search for matching forum threads related to a query is evaluated by applying voting techniques using the example of two different forums. Individual messages vote here as documents for their associated thread [AS12].

Search for a suitable service provider

Furthermore, the search for a suitable service provider has already been realized via voting techniques: In their work, Blank et al. describe the search for a suitable IT service provider. Based on a crawl, the individual websites in the index act as voting documents for their associated company [BBH16].

Main evaluation scenario: Search for an appropriate journal

The main evaluation scenario of this thesis is to find a journal that thematically best fits a query and is potentially suitable for publication on the query topic. In this scenario, according to the query, the indexed articles vote for the journal in which they were published. The result is a class or journal ranking that yields the publications that fit best to the query topic.

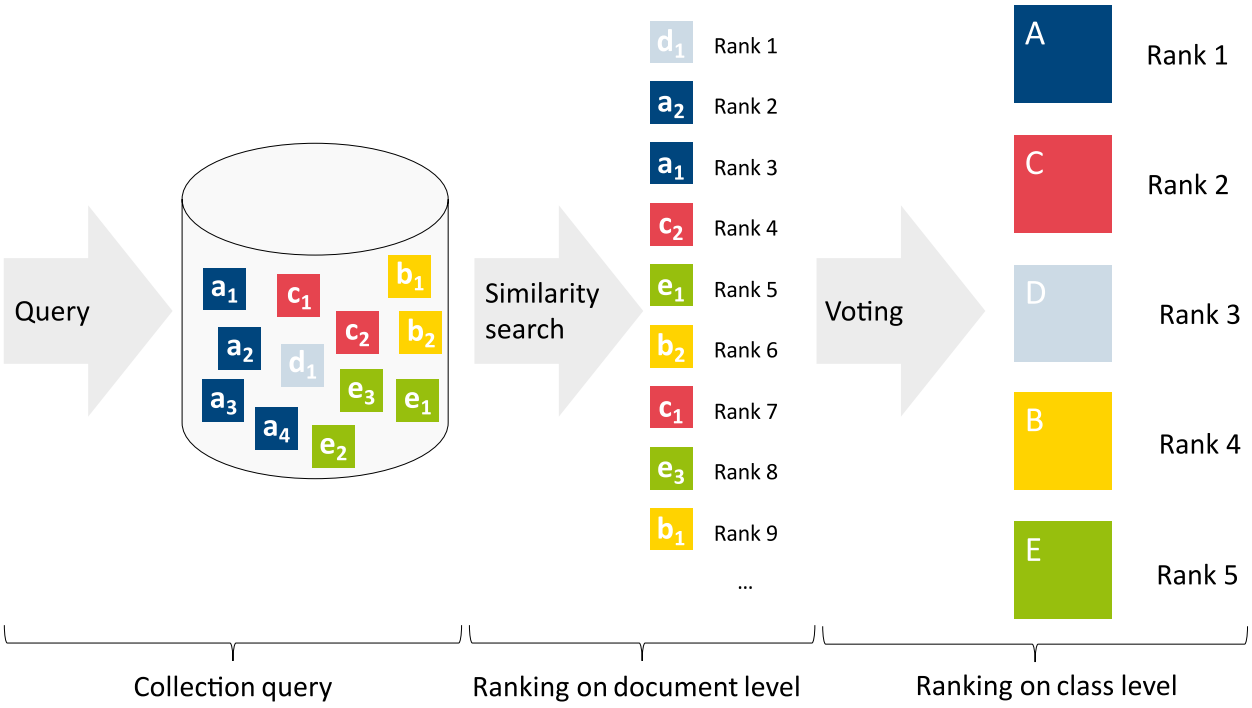

Figure 1.1: Search for appropriate classes: The search for the best matching classes calculates a class ranking. The first step is to *query* the collection, i.e. the indexed documents, in order to find the best matching classes. The color of the documents and the lowercase letters indicate which documents belong to the corresponding classes A to E. The *similarity search* calculates a ranking of documents, which serves as a basis for the application of *voting techniques*. The resulting ranking provides classes that best match the query, sorted in descending order of quality.

1.4.2 Basic Process of Voting Techniques in Information Retrieval

Figure 1.1 shows the basic voting technique process, which we divide into three steps:

The search for classes that best match starts with a query to the indexed documents, the *collection query*. Based on the similarity search, which can be implemented with different retrieval techniques, a document ranking with matching documents from the collection is established, summarized in the figure by the *ranking on document level*. By applying voting techniques, a *ranking on class level* is obtained that returns classes that match the original query in descending quality.

1.4.3 Influencing Factors of Voting Techniques

In this thesis, parameters that control a search for classes and influence its success are investigated and evaluated. Based on Figure 1.1, Figure 1.2 shows the aspects that are influential for an effective search for classes and are discussed and/or evaluated with their adjustments and settings.

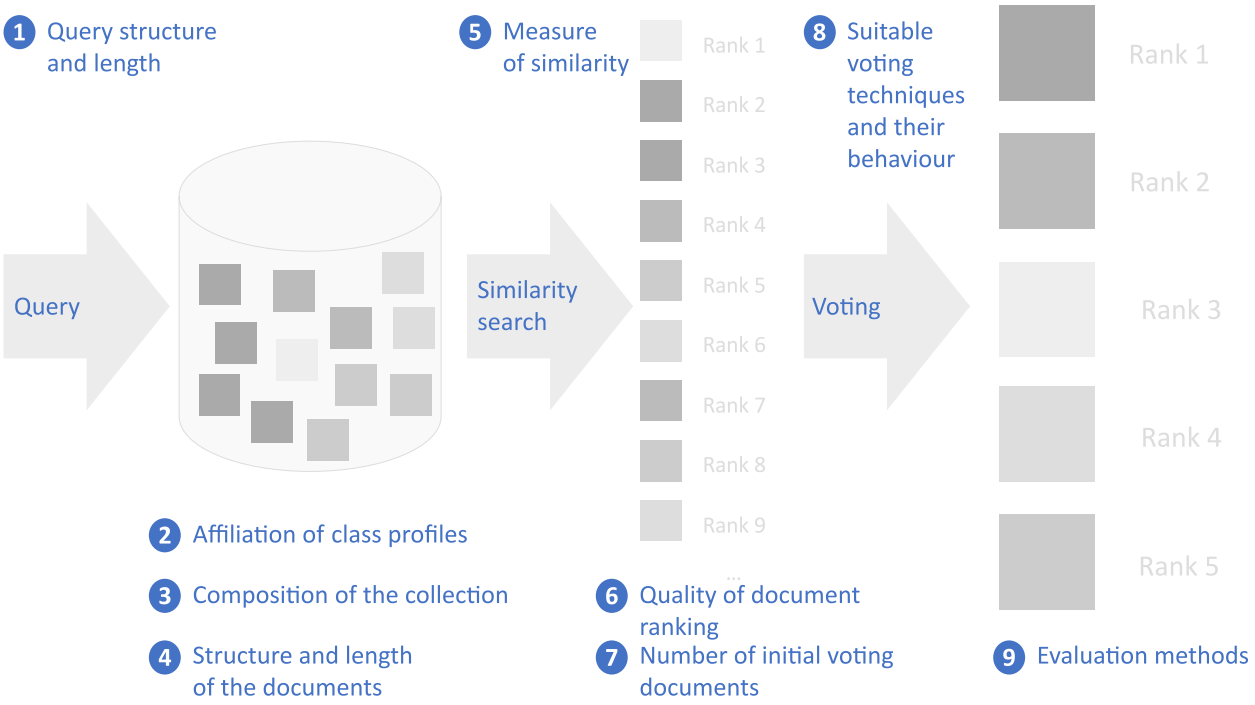

Figure 1.2: Search for appropriate classes: aspects and factors influencing the behavior and performance of voting techniques, numbered in process order and discussed in Section 1.4.3 Influencing Factors of Voting Techniques.

The aspects in Figure 1.2 and discussed in the following paragraphs are numbered in the order of the process steps and do not reflect the order in this thesis or their weighting within it.

Aspect 1 - Query structure and length The more precise the request, the more qualitative the results. In Section 5.5.2 Influence of the Query Length we investigate how the length of queries in the evaluation scenario affects the quality of class results, in particular, which voting techniques are more or less affected.

Aspect 2 - Affiliation of class profiles The elements or documents in the collection can be assigned to either one or more class profiles. Based on the evaluations in this thesis, the relations between classes and documents are one-to-many, but the evidence of expertise can exist with cardinality m:n even provided with various weighted associations, as in the evaluations of the Enterprise TREC tracks on expertise retrieval (see also Section 3.1.1 The Voting Model for People Search in Chapter 3 Voting Techniques: Applications from the Literature).

Aspect 3 - Composition of the collection The composition of the collection as the ratio of large and small classes with respect to the number of documents determines the choice and parameterization of the voting techniques and their success. To this end,

in our evaluations in Section 5.5 Systemic Behavior of the Applied Techniques we give an overview of how the techniques act in relation to the quantitative composition of the collection, i.e. the ratio of small and large classes.

Aspect 4 - Structure and length of the documents The length and structure of the documents in the collection determine the quality of the results at the class level. Based on a collection of scientific papers, in our evaluation scenarios, we investigate how the length of the voting documents influences the ranking at the class level in Chapter 5 Search for Classes - Evaluation of the Introduced Voting Techniques.

Furthermore, Chapter 6 Passages as Voters for their Documents is dedicated to the question to what extent term distributions play a role and whether and in which cases concatenated documents for each class also provide good or even better results via a simple search.

Aspect 5 - Measure of similarity The influence of the similarity measure on the document level is discussed in this thesis, using BM25 with standard parameterization in the evaluations of this thesis. Furthermore, the findings from the literature in the context of expertise retrieval are presented in the thesis in Chapter 3 Voting Techniques: Applications from the Literature.

Aspect 6 - Quality of document ranking This point is related to the previous aspect 5, since the quality of the document ranking results from the similarity search. Basically, there are very few collections that provide assessments for both the document- and class-level results. In Section 3.1.2 The Influence of the Document Ranking in Expert Search, we describe the findings from the literature in the scenario of expertise retrieval research.

Aspect 7 - Number of initial voting documents Not only the composition of the collection in terms of different class profile sizes influences the behavior of voting techniques, but also the number of basically voting documents. A document-level ranking does not have to be completely included in the voting process, but only a limited number of documents can be considered. Truncating the set of voting documents can be done either score-based or quantity-based. Thus, in 5.5.3 Influence of Initially Voting Candidates and 5.7.2 Results for New York Travel Forum Task we consider the influence of the number of initial voting documents on class ranking.

Aspect 8 - Suitable voting techniques and their behavior This section covers the largest part of the thesis and includes evaluations of existing and new voting techniques introduced in different settings and scenarios. In addition to the findings of the literature discussed in Chapter 3 Voting Techniques: Applications from the Literature, in Chapter 5 Search for Classes - Evaluation of the Introduced Voting Techniques the performance and behavior of voting techniques are investigated, also with respect to changes in settings in the previously mentioned aspects.

Aspect 9 - Evaluation methods In Chapter 3 Voting Techniques: Applications from the Literature, we present methods of evaluation in the context of expertise retrieval and from the literature, before developing new variants to assess the relevance of class rankings, starting in Chapter 5 Search for Classes - Evaluation of the Introduced Voting Techniques.

Assessing expert or class rankings is a difficult task. Within the TREC conferences, evaluation methods are used, which we describe in Chapter 3 Voting Techniques: Applications from the Literature. Furthermore, we have developed two new evaluation methods for class rankings via our experimental setup, which we present in Chapter 5 especially in Sections 5.4.2 Rank Distribution of the Original Journal and 5.4.3 Journal Relationships as Measure for Relevance.

1.5 Contribution of this Thesis

Application of voting techniques in general classification scenarios

Generalized application of voting techniques Based on the factors discussed previously, this thesis examines and states the effective application of voting techniques transferred to other domains and classification tasks in information retrieval. Starting from the state of research on expertise retrieval based on voting techniques and supporting documents, this work generalizes the topic and labels it as a *search for classes*. In analogy to a topic-based query that gets a subsequent response with potential experts on this topic, the generalized scenario of a system yielding suitable classes for a request is developed. In the course of developing a search for classes, we go into the underlying principles, but not the optimization, for single practical and individual scenarios. Each practically oriented form of a search on class level has additional individual aspects that must be taken into account in a typed system. The specific aspects which would have to be considered for a production-ready system in a certain application domain are beyond the scope of this thesis.

Suitability of voting techniques depending on collection and document structures

Document and collection structures Voting-based classification techniques are not suitable for all scenarios, as we show using various settings and collections with different structures. The predicted strength of evidence for a query being associated with a class decreases with increasing coherent document structures and more uniform term distributions across documents.

Furthermore, we evaluate to what extent a search on larger documents can be expected to yield better results in the context of a search for matching classes. Three scenarios are designed for this purpose: Searching on titles of documents, their abstracts, and their full text. We show that increasing the quantity of information in the form of supporting document sizes conditionally increases the class-search effectiveness and quality of class-based search results. Following studies of expertise retrieval, we also investigate the effect of cutting off the basic number of voting documents and varying the allowed number of voters per class.

New combinations of score- and ranking-based document values per class as stable indicators

New variants of voting techniques Combining score-based information from the document level and ranking-based information in the scope of the class level can improve voting effectiveness and produce stable and competitive results. This thesis introduces newly developed voting techniques that combine document score and class-based ranking factors of voting documents within each class. The summarization of document scores in combination with a class-scoped, ranking-based damping function leads to a stable indicator that a class is relevant to the requested query. Even in scenarios where the use of voting techniques is not recommended, competitive results are achieved. This thesis evaluates the new techniques in different forms and variations and relates the results to techniques known from the literature.

Inclusion of relationships and relations from metadata as a basis for evaluation

New evaluation techniques In addition to manual assessments and ground truth data, relationships and affinities extracted from document metadata can be used for a meaningful evaluation of voting techniques.

In the main experimental setup of this thesis, items are removed from the original collection and used as requests. By determining the rank of the class known from the removed item, we can perform evaluations in large volumes. If relationships between classes of indexed documents can be established, for example, based on metadata, other returned and ranked results can also be evaluated regarding relevance for the request.

Voting techniques applied in hierarchical clustering yield results that fluently approximate the widely used methods and can help identify searched categorizations.

From similarity to distance measures Voting techniques potentially have a broad and not yet exhausted application potential as

a supplier of decision and classification features. Beyond a generalization within information retrieval, we transfer voting techniques to another domain, hierarchical agglomerative clustering.

Voting techniques can be applied to hierarchical clustering in that points of a cluster serve as voters and the distances between clusters result from these voting results. Depending on the parameterization and setting of the techniques, results of well-known and often applied clustering techniques are approximated, and new in-between solutions can be defined.

2 Voting Techniques: Theoretical Background

This chapter presents voting techniques as they are used in the literature on a theoretical basis. We start with the underlying ranking procedures at the document level, which are discussed in Section 2.1. The two ranking techniques TF/IDF and BM25 are presented with their formulas and functions.

Section 2.1 Ranking on Document Level: Introduction of two applied document ranking techniques

The basic search is followed by voting techniques that further process the results obtained at the document level in Section 2.2. In this section, the presented techniques are evaluated for two reasons: firstly, because they are considered in many papers as standard, and, on the other hand, because they show good performance in expertise retrieval. In Chapter 4 Extended Techniques and Interrelationships, we will also introduce new variants of voting algorithms based on those in this chapter, which also yield good results with respect to our experiments.

Section 2.2 Ranking on Class Level: Introduction and discussion of existing voting techniques

2.1 Ranking on Document Level

This section introduces both scoring functions TF/IDF and BM25 yielding the document scores for the subsequent voting process whose techniques and variants are introduced in Section 2.2.

Both scoring functions are based on a *term frequency* component which expresses the number of occurrences of a query term in a document and an *inverse document frequency* component which determines its weight based on the number of term occurrences throughout the collection.

The *term frequency* component is document specific for each document in the collection, while the *inverse document frequency* results from the composition of the indexed terms and documents as a collection-specific characteristic.

For our experiments, we used Elasticsearch in version 7.0.0 to index the document collection and to obtain scoring at the document

level.[1] All experiments are based on the Porter stemming algorithm of the search engine and applying stopword elimination.[2]

1: Information to Elasticsearch is available at https://www.elastic.co/guide/en/elasticsearch/reference/current/index.html

2: Information to the used stemmer algorithm is available at https://www.elastic.co/guide/en/elasticsearch/reference/current/analysis-stemmer-tokenfilter.html. Stopword elimination is described in https://www.elastic.co/guide/en/elasticsearch/reference/7.0/analysis-stop-analyzer.html

2.1.1 TF/IDF

Our first experiments were based on the Elasticsearch TF/IDF-based document ranking, which takes different document lengths into account via a normalization factor in order to yield comparable results for all documents. The formula is made up of three multiplicatively linked components; the term frequency, the inverse document frequency, and the normalization factor. It is defined as follows:

The *score(d,q)* of a document d given a query q which consists of terms t is computed as:[3]

$$score(d,q) = \sum_{t\ in\ q} \left(tf(t\ in\ d) \cdot idf(t)^2 \cdot norm(d) \right) \tag{2.1}$$

3: The notation is analogous to the Elasticsearch guide and associated blogs. For TF/IDF, the formula can be found at this URL: https://www.elastic.co/guide/en/elasticsearch/guide/2.x/scoring-theory.html

The term frequency tf as the first factor describes the frequency of term t in document d and is defined as

$$tf(t\ in\ d) = \sqrt{frequency} \tag{2.2}$$

The inverse document frequency for term t as second factor across the collection is computed as

$$idf(t) = 1 + \log\left(\frac{numDocs}{docFreq(t) + 1}\right) \tag{2.3}$$

where *numDocs* is the number of all documents in the collection and *docFreq(t)* is the number of documents containing term t.

The document length normalization as the last factor is defined by the term

$$norm(d) = \frac{1}{\sqrt{numTerms}} \tag{2.4}$$

where *numTerms* is the number of terms in document d.

In their book *Introduction to Information Retrieval*, the authors Manning et al. show further variants of the TF/IDF formula with different term weightings [MRS08, sec. 6.4.3].

Our previous studies have shown that applying TF/IDF as a ranking function yields slightly worse rankings than using BM25 for all experiments [WH18]. Since the results were slightly worse, in

this thesis we pursue the following BM25 approach to rank at the document level.

2.1.2 BM25

This technique, as our alternative and further pursued scoring technique at the document level, is to be parameterized for variables k and b. The parameter k regulates the impact of term frequency saturation while the variable b controls the influence of the document length on the scoring of a document.

The results of previous investigations [WH18] show that changing the parameters within the upper and lower meaningful limits did not result in large changes compared to the standard parameterization. In particular, worse results could only be obtained in the case of longer texts such as abstracts when setting $b = 0.1$, which means dampening the document length consideration.

Based on these findings, we use BM25 with the standard parameterization setting $k = 1.2$ and $b = 0.75$.[4] The formula for the BM25 scoring is defined as follows:

4: Further information regarding the BM25 implementation in Elasticsearch is available at https://www.elastic.co/blog/practical-bm25-part-2-the-bm25-algorithm-and-its-variables

The $score(d, q)$ of a document d given a query q which consists of terms t is computed as follows:

$$score(d,q) = \sum_{t \in q} \frac{idf(t) \cdot tf(t \text{ in } d) \cdot (k+1)}{tf(t \text{ in } d) + k \cdot \left(1 - b + b \cdot \frac{|d|}{avgdl}\right)} \tag{2.5}$$

The term frequency tf describes the number of occurrences of term t in document d, $|d|$ represents the document length, and $avgdl$ is the average document length over all documents in the collection.

The inverse document frequency idf for term t is computed as

$$idf(t) = \log\left(1 + \frac{numDocs - docFreq(t) + 0.5}{docFreq(t) + 0.5}\right) \tag{2.6}$$

where $numDocs$ is the number of all documents in the collection and $docFreq(t)$ is the number of documents containing term t.

Based on the probabilistic ranking principle, the paper "The Probabilistic Relevance Framework: BM25 and Beyond"[RZ09] written by Robertson and Zaragoza presents the development and further variations of the BM25 ranking algorithm.

Since we use a standard parameterization for BM25 applying $k = 1.2$ and $b = 0.75$, the expression $score(d, q)$ at the document level refers to this parameterization of BM25 in the following sections of the thesis.

2.2 Ranking on Class Level

Based on the rankings on document level, various approaches, either introduced in expert retrieval or novel, varied combinations, are applied in this thesis to derive a class ranking. The interface between the ranked documents and the voting process includes two parameters. The first parameter determines the total number of ranked documents to be considered and is discussed in the next paragraph. The second parameter comprises three key figures from the document ranking that can be used for the class ranking. These key figures are explained after the first parameter in the following sections. After an introductory discussion of these points, this subsection presents the voting techniques with their properties and formulas and relates them to each other.

The number of voting documents can be limited based on their score or number.

Number of ranked documents to consider As a prerequisite, we define $R(q)$ as the set of documents retrieved for the query q. The document ranking techniques TF/IDF as well as BM25 basically can yield an unrestricted amount of ranked and scored documents. This overall amount of documents returned can be limited by defining a threshold for their score as well as setting a global maximum number of highest-ranking documents to be considered for the following processing step. In this thesis, variants of retrieved document volumes that act as initial voters are considered. We define this number of initial voting documents as $|R(q)|$. Adjusting this number has a major impact on the quality of the results for certain techniques. Taking into account too many initial voting documents can degrade the results [Mac09]. In this work, we mainly set the size $|R(q)|$ to 3,000 documents that are considered by the voting process, based on experiments and evaluations in Section 5.5.3 in which we examine the effects of changing this value.

Three key figures result from the document ranking as potential input for the voting process.

Three key figures for the voting process In general, for voting-based techniques, class ranking is based on different inputs [Mac09]:

- *Number* of documents in the search result associated with a class: This number is to be distinguished from the number of initial voters and is related to the number of voting documents per class. This key figure states in purely quantitative terms that the number of voting documents for a class determines its success and thus its relevance to the query q. This quantitative indicator alone is only a limited expression of class relevance, as the success of any class is potentially dependent on its size.
- *Ranks* of the documents associated with a class: In this case, the global rank achieved for the document at the document level is included in the voting process. As the ranks are implicitly

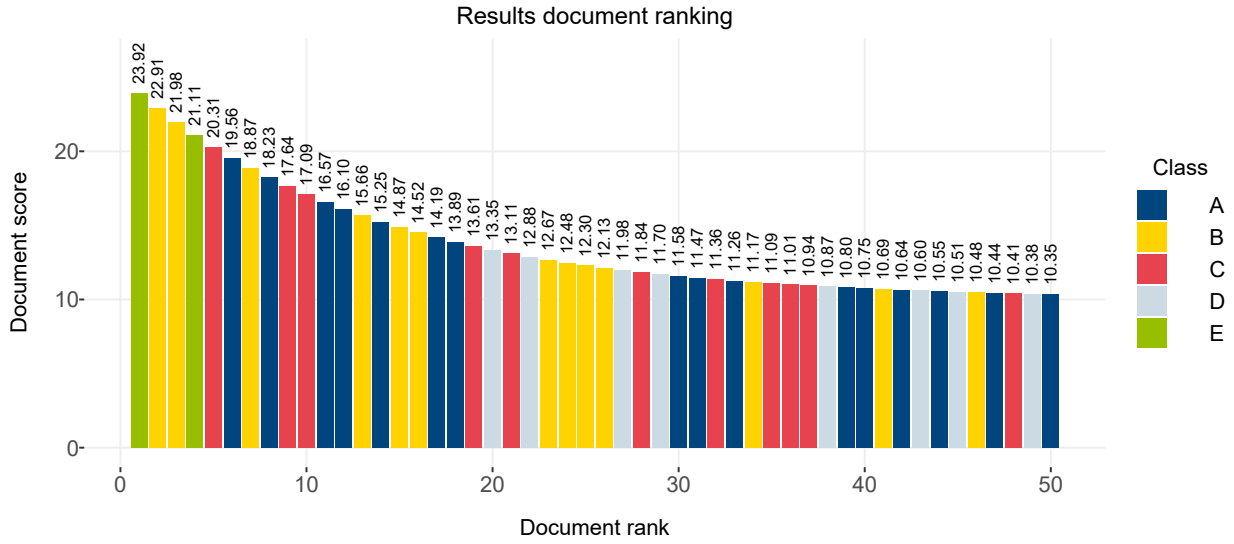

Figure 2.1: Exemplary progression of the score values and the document ranking for $|R(q)| = 50$. All voting documents belong to one of classes A to E. The course of these exemplary score values is characterized by an increasingly flat decline and lies approximately in the range of half of the measurement results resulting from the analyzes in Chapter 5, illustrated in Figure 5.4.

formed from the sorted documents with their scores, this indicator can be seen as an abstraction that neglects differentiated scoring. In this context, classes that receive votes from documents in the top ranks obtain a higher relevance judgment for query q than those classes supported from documents in the bottom ranks.

- *Score values* calculated for documents associated with a class: According to the previous case, the globally achieved document scores are included in the voting process. This indicator is slightly similar to the previous one, but here the results are differentiated based on the applied similarity measure. In particular, steeply or flatly declining score curves of the global ranking can be taken into account when assessing the relevance of classes.

The voting techniques introduced in the following sections use these key figures individually or in combination.

Sample document ranking as a case study

For plausibility and clarification, we use a sample ranking of 50 documents belonging to classes A to E according to their colors. This document ranking is shown in Figure 2.1 and is referenced in the calculation of each class ranking in the following sections. The example is used to demonstrate the properties and differences between voting techniques.

In the following, we define a scoring function $score(c, q)$ that computes a score for class c and a query q. Basic techniques to calculate this score are presented in the following sections.

2.2.1 Votes

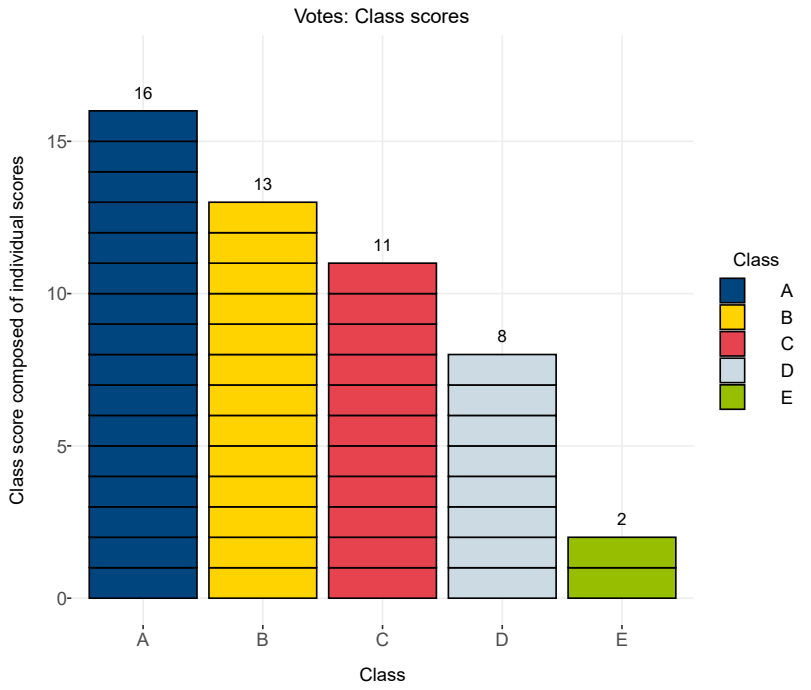

Figure 2.2: Results for the classes after application of Votes: The class ranking is determined by the number of voting documents per class. Class A as the one with the most votes leads the ranking.

Votes corresponds to the quantitative evaluation of voting documents.

Based on the example of electoral voting systems, Votes takes the number of documents found for each class as the score. Documents that were determined relevant in the preceding search each have one vote for their associated class. Based on the composition of a collection, classes that have a relatively large number of documents receive a potential benefit by applying this technique, and the results are biased due to different class sizes. Based on the document ranking in Figure 2.1, Figure 2.2 shows the result of the voting technique: Classes are ranked in descending order of the number of their voting documents, which is calculated using the formula:

$$score_{Votes}(c, q) = |\{d \mid d \in R(q) \wedge d \in c\}| \tag{2.7}$$

By ignoring the quality of each vote and considering only quantities that vote for classes, this technique is not a good performer without any normalization, neither in the literature [Bal+12; Mac09] nor in our experiments [HW21]. In particular, for collections with very high differences in class members or for collections that have small and unequally distributed numbers of voting candidates per class, this technique gives results that do not correspond to their relevance.

Nevertheless, in this work we take it into account because it is the only technique based exclusively on the number of voters and it is converged by an extreme parameterized variant of our newly introduced technique RR^x introduced in Section 4.1.3.

2.2.2 CombSUM

This technique is based on the document score values of the matching results. It is designated as such by Fox and Shaw [FS93] and originally designed to combine the results of multiple retrieval runs. In our scenario, for every class c, CombSUM sums up the scores at the document level of its voting documents.

Summation of document scores per class leads to a bias favoring large classes.

In the case of constant scores, this technique would yield the same class ranking as the Votes technique. When confronted with slow decreasing document scores that form a flat falling curve, CombSUM also tends to prefer classes having a relatively high document count and yields results slightly tending to those of Votes. This can also be seen in Figure 2.3, which shows the same ranking of classes as Votes.

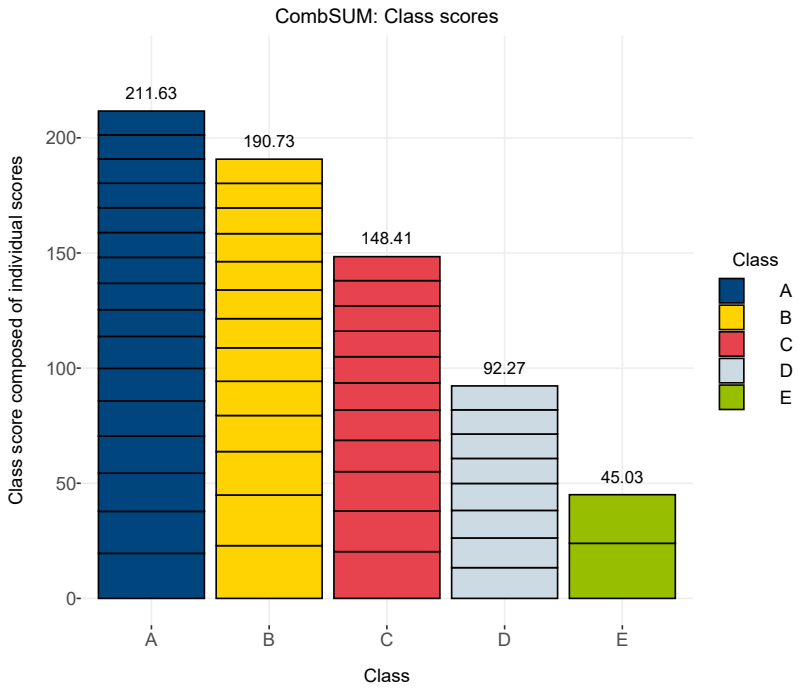

Figure 2.3: Results of classes A to E for the application of CombSUM. Due to the only slightly decreasing score progression, in this exemplary scenario the final result corresponds to that of Votes. The flatter the score values of the voting documents fall, the more quantity-oriented the CombSUM results are.

The formula for class-based summation of document scores is the following:

$$score_{CombSUM}(c,q) = \sum_{d \in R(q) \wedge d \in c} score(d,q) \tag{2.8}$$

Profile normalization as a correction factor for class size differences

To prevent this bias based on class size differences, Macdonald proposes candidate normalization methods in the scenario of the enterprise expert search task [Mac09, sec. 6.4]. The basis of the approaches to normalizing expert profiles is the length of each profile, derived from the *number of tokens* as the number of terms of a complete expert profile or the *number of documents* associated with the expert[5]. Results improved by normalization can be achieved with voting techniques that take into account the number of voting documents, such as Votes. However, the results of techniques that only take the score into account like CombSUM, but implicitly benefit from the number of voters, can also be improved with profile normalization.

5: In Section 3.1.1 we give a more detailed overview of the candidate normalization methods proposed by Macdonald and the evaluated results.

As an anticipation, it is worth mentioning here that during experiments with our collection setup, we also encountered huge class size differences and slow-decreasing score values of the initially voting documents. It turned out that on average the value of $score(d,q)$ at rank 5 was 80% and the value at rank 50 was still 60% of the document with the highest score. Hence, the influence of the lower ranks is still relatively high. Figure 5.4 on page 71 shows the progression of the scores during our experiments with the AMiner collection in quartiles.

Compensation of the bias using class-based cutoff and damping factors

In this thesis, CombSUM is evaluated as a standard technique, but also represents the basis for other variants of voting techniques that compensate for a bias caused by large class differences and flat declining score curves. We do not follow the concept of profile normalization as a penalty for large-sized classes because a second index for length profiles of classes means additional overhead and maintenance. Instead, we counter the bias with two factors. A per-class cut-off of the voting documents and damping factors are used as methods in the following techniques known from the literature or newly introduced in this thesis.

2.2.3 CombMNZ

A combination of Votes from Section 2.2.1 and CombSUM from Section 2.2.2 is represented by CombMNZ, also first presented to our knowledge in the work of Fox and Shaw [FS93]. Per class, this technique multiplies the sum of the scores by the number of voters:

$$score_{CombMNZ}(c,q) = score_{Votes}(c,q) \cdot score_{CombSUM}(c,q) \tag{2.9}$$

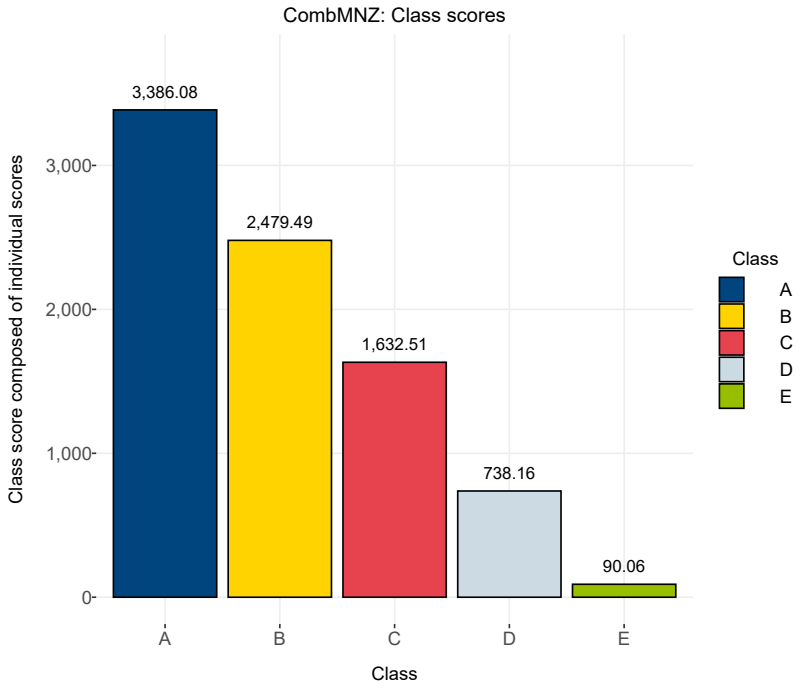

Figure 2.4: Ranking of the classes for CombMNZ: The ranking of the classes is similar to that of CombSUM and Votes, but by multiplying the summed score results with the number of votes for each class, the final class score differences in the exemplary scenario become larger. In contrast to the normalization of class sizes, this multiplication means that classes with many voters are explicitly favored.

In the exemplary example, which is characterized by flatly decreasing scores in the initial document ranking, the result is the same as for CombSUM and Votes, except that the differences between the class scores are clearer here. Figure 2.4 shows the results, which show an even higher bias in favor of larger classes with many documents.

2.2.4 expCombSUM

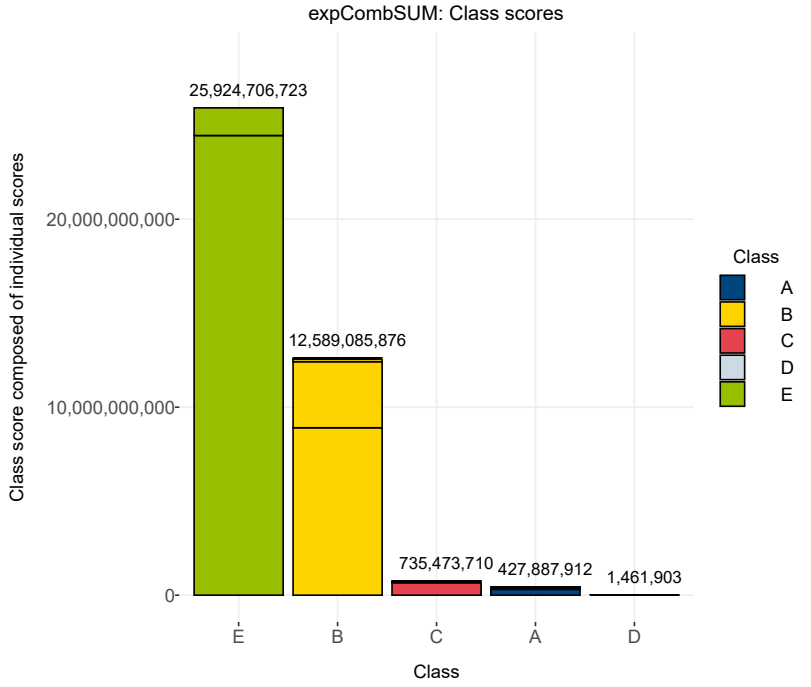

Figure 2.5: Ranking of the classes for expCombSUM: Due to the exponentiation, even weakly decreasing document scores from the global ranking result in high scoring differences for the classes.

Emphasis on classes with documents from the top ranks by exponential transformation of scores

For this technique, each document score serves as the exponent of the Euler number e, which causes a huge amplification and gradation of the document scores. This method is introduced by Ogilvie and Callan substantiated by the following argument [OC03]:

"This transformation is justified when the retrieval system returns the log of the query generation probability because this places the scores back on the probability scale."

This technique gives first-ranked documents greater importance and voting strength, thus expressing their higher probative value. Figure 2.5 shows that classes whose documents lead in the global ranking get a corresponding ranking by using their score as a power of e: Class E marked green reaches position one in the ranking, although it has only two voting documents, which nevertheless occupy the top positions in the scoring.

The formula for expCombSUM to calculate the score of a class c given a query q is:

$$score_{expCombSUM}(c,q) = \sum_{d \in R(q) \wedge d \in c} e^{score(d,q)} \tag{2.10}$$

Applying the documents' scores as exponents to the Euler number emphasizes the upper scores and widens the score spectrum on the class level. Figure 2.6 shows the enormous differences in the document-level scores for the range 18 to 25, also listed in Table 2.1. While the lower resulting scores are still in the millions, the following results quickly reach the single- and double-digit billion range.

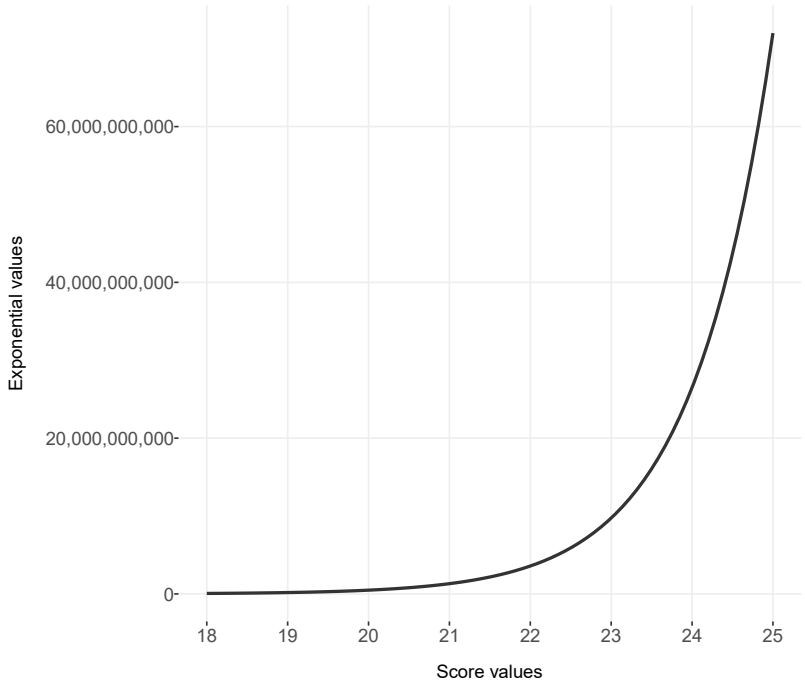

Figure 2.6: The y-axis shows transformed document scores for expCombSUM by exponentiation of e with the original document scores in the cropped range of 18 to 25 on the x-axis. At first glance, the exponential values appear to start at 0, but for a score value of 18, the resulting value is in the range of millions at 65, 659, 969, as shown in Table 2.1. Due to the enormous differences in the transformed scores, results are produced here, in the exemplary scenario, at class level that approximate those of CombMAX (see Section 2.2.7).

This results in a step towards decoupling the success of a class from its size – though still all returned results voting for their associated class are taken into account.

Interestingly, Macdonald concludes in his thesis "The voting model for people search" that expCombSUM also gives very good results even if the document-level ranking function – the weighting model – is not logarithm-based:

"However, we find that expCombSUM and expCombMNZ outperform their non-exponential variants on all weighting models, including BM25, which is not based on logarithms. [Mac09, sec. 6.3.2]

Table 2.1: Integer values from the range of the upper scores of the exemplary example with their resulting scores as an exponent of the Euler number *e*.

Score values	Exponential values (rounded)
18	65,659,969
19	178,482,301
20	485,165,195
21	1,318,815,734
22	3,584,912,846
23	9,744,803,446
24	26,489,122,130
25	72,004,899,337

In our experimental setup in Chapter 5, we evaluate expCombSUM based on the similarity measure BM25.

2.2.5 expCombMNZ

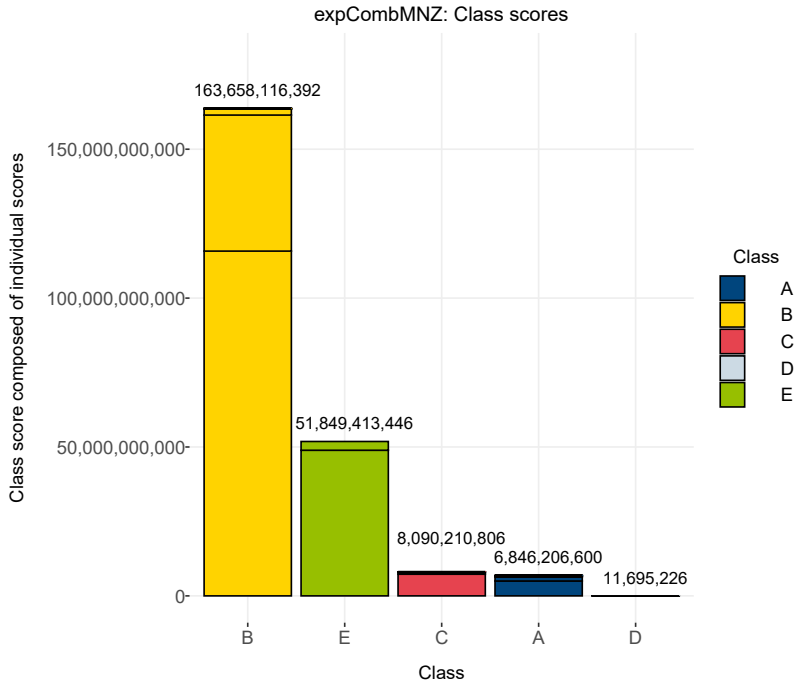

Figure 2.7: Class ranking for expCombMNZ: In the upper scoring range, the class ranking of expCombSUM can still change by including the number of voting documents. In this case, classes B and E swap the two top ranks, caused by the high number of voters for class B. See also Figure 2.5

Simultaneous emphasis on classes with documents from the top ranks and higher number of voting documents

As for the previous technique, the score values of the voting documents are processed as a power of *e*, the sum of which is then further multiplied by the number of voting documents per class. Based on the scores of expCombSUM, the results are multiplied

with the score of Votes:

$$score_{expCombMNZ}(c,q) = score_{Votes}(c,q) \cdot score_{expCombSUM}(c,q) \tag{2.11}$$

By rating a class in combination with the number of its voters, the class ranking results are skewed towards a potential advantage of classes having a higher number of voting documents - especially for high-ranked classes in our scenario, where the multiplication with the number of voting candidates potentially has an impact for the class ranking. Figure 2.7 shows that classes B and E swap their rank from the example with expCombSUM, since B has 13 voting candidates and E has only two.

2.2.6 CombSUM TOP *n*

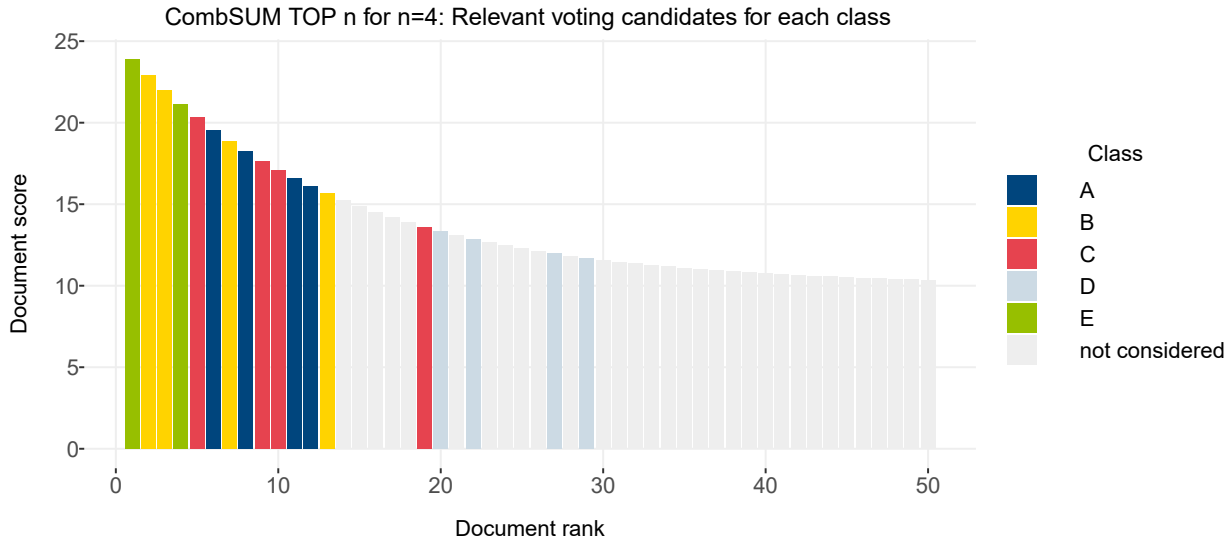

Figure 2.8: CombSUM TOP *n* (here: *n*=4): Original document ranking that color-codes only the relevant voting candidates per class. Other documents are not included in the voting for the class ranking. The relevant scores are not transformed here and are originally taken as sums for each class.

This approach is based on CombSUM and the assumption that the higher ranked documents of each class mean a higher level of significance in terms of their relevance. Therefore, it limits the number of documents that contribute to the score of a class to increase the quality of the voting documents [Juá+10]. At the same time, by setting a cutoff point for potential voters, this technique reduces the influence of class size differences and noise caused by slow-decreasing score curves.

Only the highest top n scores per class are taken into account to compensate for size differences and eliminate interference.

For each class, this aggregation sums up only the top *n* (in evaluations of this thesis, mainly applied: $n \in \{5; 50; 100\}$) scores of the documents for each class. Setting $rank(d,q,c)$ as the rank of d if only documents of c are considered, we yield:

$$score_{CombSUM\ TOP\ n}(c,q) = \sum_{d \in R(q)\ \wedge\ d \in c\ \wedge\ rank(d,q,c) \leq n} score(d,q) \tag{2.12}$$

Figure 2.8 shows the original document ranking and grayed out documents that are ranked above $n = 4$ for each class and therefore are irrelevant to the class ranking.

Setting n to 1 leads to CombSUM TOP 1 as an equivalent to CombMAX explained in Section 2.2.7, whereas CombSUM TOP ∞ has the characteristics of CombSUM described in Section 2.2.2.

Choosing smaller or larger n changes the invariance to initial voters and inclusion or exclusion of relevant or irrelevant document votes.

The different settings for n influence the quality of the voting results. A smaller number of n makes CombSUM TOP n invariant to changes in the number of initial voters and also removes the noise introduced by voting on classes with large numbers of voting documents.

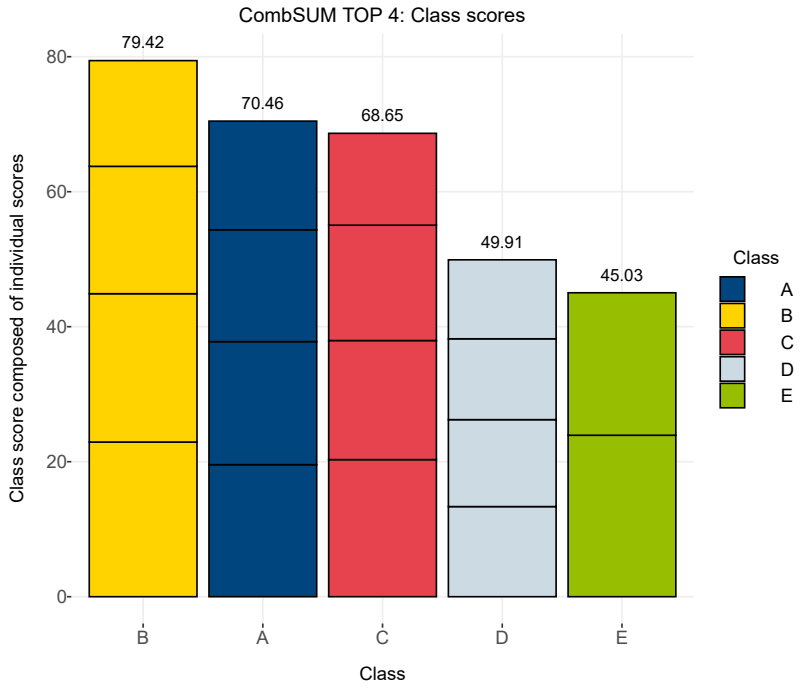

Figure 2.9: Ranking of the classes from the scenario for CombSUM TOP 4:

Figure 2.9 shows the class ranking in the example scenario for CombSUM TOP n with $n = 4$. Here, only the best 4 scoring documents in a class are considered, and in sum, result as the class score. Compared to the CombSUM class ranking in Figure 2.3, the yellow class B is in the first rank and no longer in the second rank. If only the four highest scoring documents are considered on a class basis, their sum represents the highest value in the case of class B. Figure 2.9 also shows that the difference in class scores is not as large as in the case of CombSUM.

This optimization of n is only possible to a certain extent, since variations of n that are too small also exclude important and conclusive voting documents of a class. We will illustrate this fact by the evaluations in Section 5.4 with variants of n or by using the extreme variant CombMAX, which means CombSUM TOP 1. We also refer here to Chapter 3, in which investigations of the dynamic change of n are discussed [CLO10].

2.2.7 CombMAX

Each class scoring is derived from its maximum scoring voting document.

This technique is based on the assumption that the best matching document per class also defines the relevance of this class: CombMAX takes the first result stemming from class c, respectively, the document with the highest ranking, as voting candidate with its score.

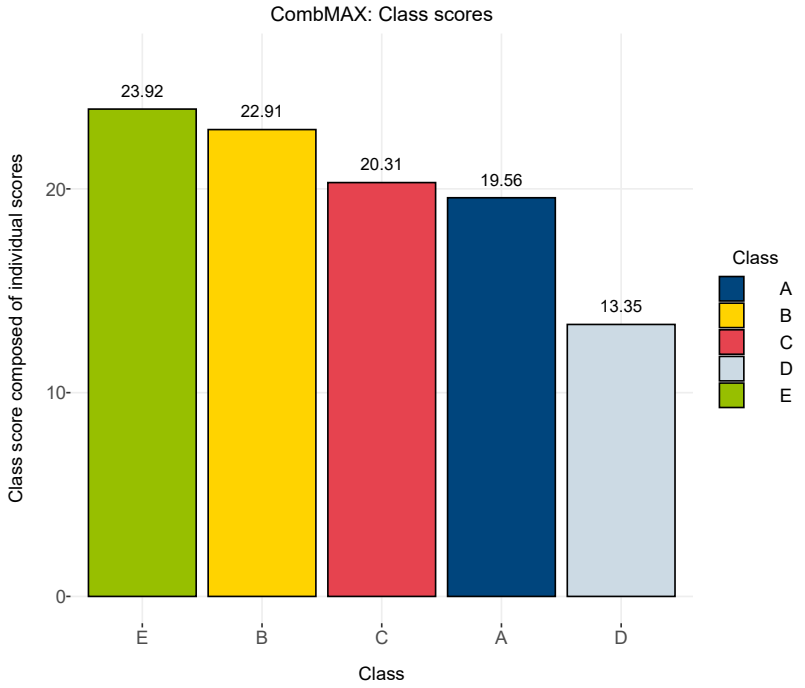

Figure 2.10: Ranking of the classes from the scenario for CombMAX: The score of each class is equal to the score of the highest ranked voting document.

Related to the topic of expert search, Macdonald argues for this method as follows:

"The intuition behind this voting technique is that a candidate who has written (for instance) a document that is very close to the required topic area (i.e. the user query), is more likely to be an expert in the topic area than a candidate who has written some documents that are marginally about the topic area" [Mac09, sec. 4.4.2]

Since the score of a class is derived from its maximum scoring voting document, the formula of CombMAX is defined as follows:

$$score_{CombMAX}(c,q) = \max(\{score(d,q) \mid d \in R(q) \wedge d \in c\}) \tag{2.13}$$

CombMAX has the ultimate invariance to class sizes.

As can be seen in Figure 2.10, the first ranked documents determine the class ranking according to their class-wise occurrence and order. Taking into account only the best-fitting document per class, this technique has the highest invariance with respect to the size of the classes or the number of documents in each class. The bias that potentially arises from large differences in class sizes is eliminated. To what extent this affects or improves the effectiveness of the search results, we examine in Chapter 5 and its sections.

2.2.8 *RR*

RR as a ranking-based technique recalculates the scores based on the harmonic series.

This technique is exclusively based on the global ranks of the voting documents, since it uses the reciprocal rank $RR = \frac{1}{rank}$ and is first mentioned in [Zha+02].

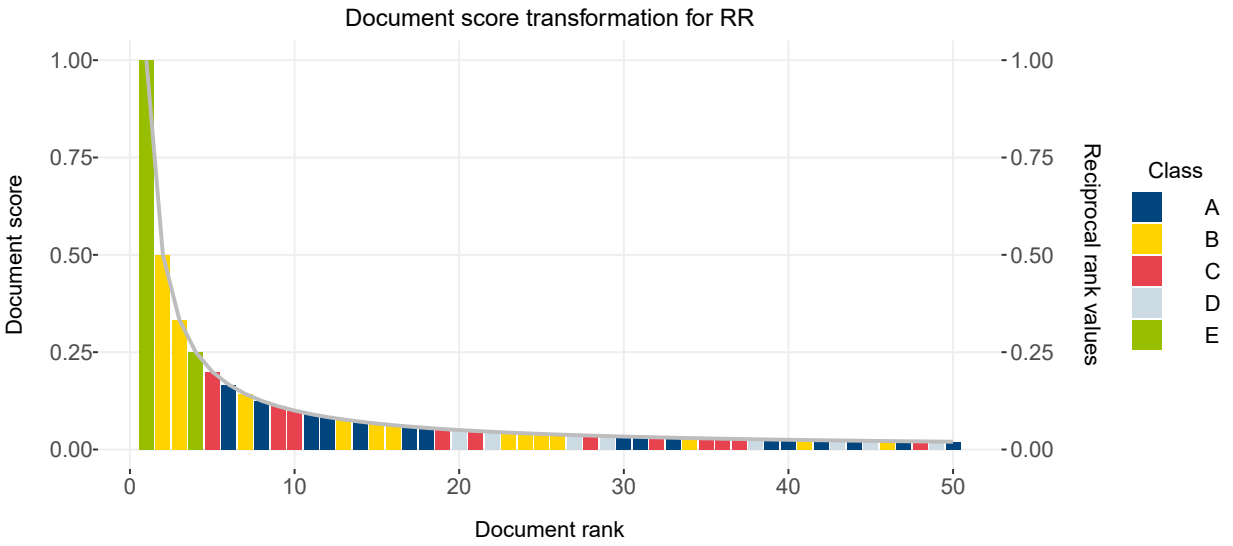

Figure 2.11: Document ranking of the scenario with transformed score values: The score of each document is set to the reciprocal rank value according to its rank.

RR provides an abstraction level on effectiveness measures, since it ignores the progression of the score values and assigns the ranking-based values, calculated on the reciprocal rank, as scores. We note that the scope of the revaluation, which represents a quantization of the score-ordered documents, lies on the first stage, the search on document level. Figure 2.11 shows the initial document ranking,

revalued with reciprocal ranks, which only corresponds to the initial ranking in terms of the order of the documents.

The values of the harmonic series and thus the revalued document scores decrease disproportionately in the front ranks, while the attenuation decreases in intensity towards the back ranks. Like the exponential variants of CombSUM, *RR* emphasizes the global front ranks; Outscoring a class with a document in position 1 in the class ranking is difficult for classes with documents of the back ranks. However, with descending ranks, the influence of *RR* on a class ranking changes; here, the increase of the damping factor becomes smaller and smaller.

The formula for *RR*, adding the ranking-based scores from the document ranking per class is:

$$score_{RR}(c,q) = \sum_{d \in R(q) \wedge d \in c} \left(\frac{1}{rank(d,q)} \right) \tag{2.14}$$

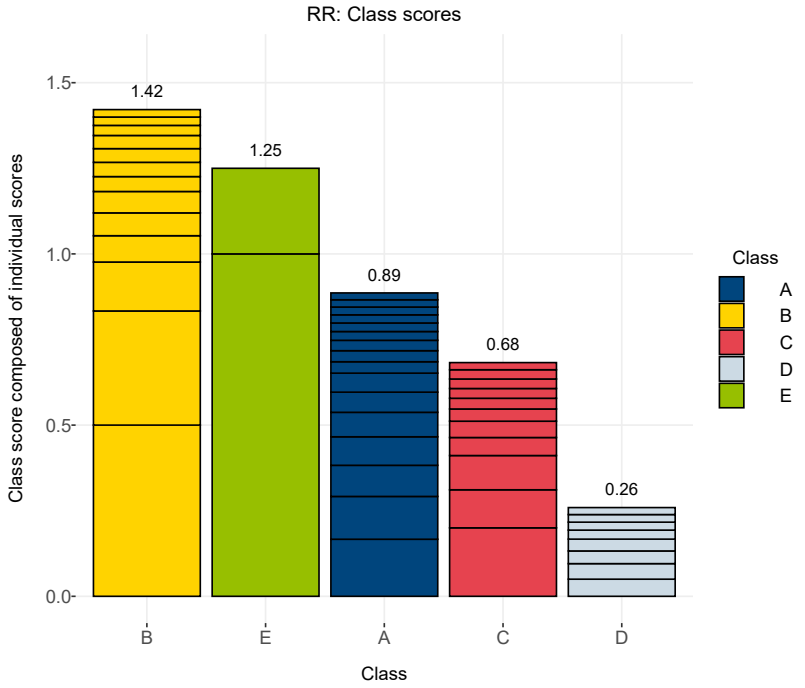

Figure 2.12: Result of the class ranking for *RR*: The score values are strongly damped by the revaluation with values of the harmonic series. Still, with a correspondingly high number of small values for a class, there is the possibility of moving up in the ranking: This is the case for classes B and A in this scenario.

Figure 2.12 shows a similar result to expCombMNZ in terms of ranking, but with swapped ranks for classes A and C. The first two ranked classes B and E are ranked the same in both techniques due to the emphasis on high document scores; for *RR*, class A is able to

prevail in the middle range due to the sum of 16 voting documents compared to class C with 11 documents.

2.2.9 BordaFuse

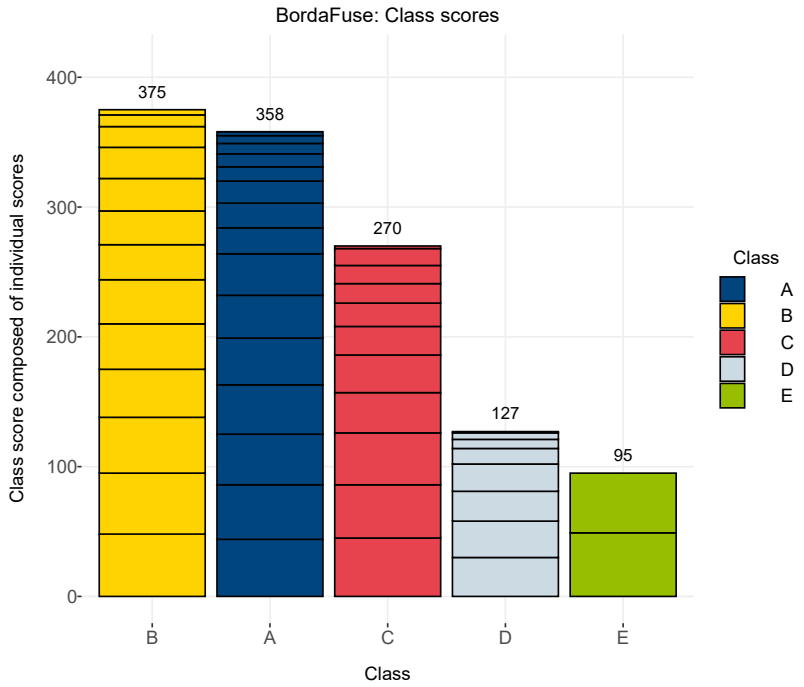

Figure 2.13: Result of the class ranking for BordaFuse: Due to their rather high number of voting documents, classes B and A are ranked on the first two positions, while class E having two high quality documents from the upper document ranks is ranked last.

BordaFuse as a ranking-based technique revalues the scores in a linear decreasing order.

This technique has its origin in the Borda election system, in which the most preferred candidate receives the highest score; the following candidates, in preference order, each receive one point less.

BordaFuse is a technique based solely on the ranking of documents, and each class receives the sum of its voting documents whose score corresponds to their rank in the overall ranking [AM01]. The following summation is made for each class:

$$score_{BordaFuse}(c,q) = \sum_{d \in R(q) \,\wedge\, d \in c} \left|R(q)\right| - rank(d,q) \qquad (2.15)$$

Since BordaFuse is a pure ranking-oriented technique in addition to *RR*, the scores for each rank are determined. While for *RR* there is a value of the harmonic series determined for each document ranked in order, for the document scores of BordaFuse there is a linearly decreasing series. In the example scenario, the score value

for the document at position 1 would start at 49 and fall linearly over discrete values to 0 for the last-ranked document. In the evaluations of this thesis, BordaFuse is not included, but experiences from the literature and in the context of expertise retrieval are discussed in Chapter 3 Voting Techniques: Applications from the Literature.

2.3 Further Voting Techniques

In this section, we present voting techniques that have not been successful in classic expertise retrieval but are worth mentioning or are addressed in other parts of this thesis.

2.3.1 CombMIN

The minimum-score voting document determines the score of each class.

CombMIN as a voting technique is not pursued further, as it has not been proven to be a performant voting technique in the literature and in the scope of expertise retrieval [Mac09]. The minimum score of documents that vote for a class is determined as the score of this class. The formula for this voting technique is introduced in [FS93] and defined as:

$$score_{CombMIN}(c,q) = \min(\{score(d,q) \mid d \in R(q) \wedge d \in c\}) \tag{2.16}$$

The intuition of this method would be that the class for which even the worst-fitting document with respect to the query fits relatively well should be ranked high. However, this intuition ignores the fact that, e.g. experts can be experts in multiple fields. Although its importance as a similarity measure is low, CombMIN has its correspondence in the domain of clustering algorithms as a distance measure and is discussed further in Chapter 7 Voting Techniques Applied to Clustering Algorithms.

2.3.2 CombMED

For each class, the score-based median of the voting documents is used as the assessment.

CombMED is introduced in [FS93] and is based on the median of all document scores that vote for a class c:

$$score_{CombMED}(c,q) = med(\{score(d,q) \mid d \in R(q) \wedge d \in c\}) \tag{2.17}$$

The use of the median does not correspond to the notion that a high score of a class is based on the greatest possible similarity of its documents. CombMED considers all voting documents per class; consequently, many lowest-rated documents disproportionately and negatively affect the result, even if there are documents with high probative value for a class.

2.3.3 CombANZ

The score-based average of voting documents determines the score for each class.

This technique, like CombMIN and CombMED also introduced in [FS93], takes advantage of the scores of documents voting for a class c related to a query q and yields the calculated average value as the class score. The letter sequence "ANZ" represents the average of nonzero values. The formula is calculated as follows:

$$score_{CombANZ}(c,q) = \frac{1}{|d \in R(q) \ \wedge \ d \in c|} \sum_{d \in R(q) \wedge d \in c} score(d,q) \tag{2.18}$$

CombANZ as a voting technique based on average has not proven to be promising in expertise retrieval and will not be considered in the evaluations in Chapter 5 Search for Classes - Evaluation of the Introduced Voting Techniques. In Chapter 7 Voting Techniques Applied to Clustering Algorithms, where we evaluate the application of voting techniques in clustering scenarios referring to distance metrics, we revisit the technique.

2.4 Summary

This chapter presents two techniques for similarity search at the document level in Section 2.1. As a preliminary stage of the voting process, these techniques provide the fundamental key figures for the next stage. These key figures are based on the scores of the documents, their ranking, and the resulting number of documents per class. Each of them has predictive power when voting for the relevance of classes with respect to a query.

Table 2.2: Summary of class rankings resulting from the presented voting techniques in Section 2.2:

	Quantity-	Score-	Ranking-	**Rank**				
Technique	based	based	based	1	2	3	4	5
Votes	x			A	B	C	D	E
CombSUM		x		A	B	C	D	E
CombMNZ	x	x		A	B	C	D	E
expCombSUM		x		E	B	C	A	D
CombMAX		x		E	B	C	A	D
expCombMNZ	x	x		B	E	C	A	D
RR			x	B	E	A	C	D
BordaFuse			x	B	A	C	D	E
CombSUM TOP 4		x		B	A	C	D	E

In Section 2.2, we present voting techniques known from the literature, evolved from data fusion techniques, or developed further from them. They use the three key figures from the document level either individually or in a combined form. Table 2.2 again shows the class rankings from the example scenario in an undifferentiated form, in terms of the ratios within each ranking.

The upper delimited block shows Votes, CombSUM, and CombMNZ, which all give the same result for the class ranking of the example. Apparently, the ranking corresponds in its order to the number of voting documents; see also Table 2.3, even if CombSUM is a score-based method.

Table 2.3: Number of voting documents for classes A to E in the example scenario

Class	#Voting documents
A	16
B	13
C	11
D	8
E	2
Sum:	50

A completely different trend is shown by the ranking results of expCombSUM and CombMAX, which rank class E at position 1 with only two voting documents. The fact that these methods do not primarily consider a quantity, i.e., the number of voting documents, is shown by the ranking of class A on the second-last position.

The third delimited section with expCombMNZ, *RR*, BordaFuse and CombSUM TOP 4 shows similarity to the previous section in that class E can land on a second rank. However, the differences are fluid, as can be seen with BordaFuse and CombSUM TOP 4, which follow Votes, CombSUM, and CombMNZ except for a swap of positions 1 and 2.

After presenting and preliminarily analyzing voting techniques based on the example scenario, the next chapter presents important work from the literature of expertise retrieval and summarizes the collected experience with voting techniques based on selected writings.

3 Voting Techniques: Applications from the Literature

As part of data fusion techniques, voting techniques have become established in expertise retrieval in recent years. A milestone in this research is the work of Macdonald and Ounis [Mac09; MO09], which is presented and summarized in Section 3.1. Exemplary work from expertise retrieval is discussed, the results of which are also of significance for a general application scenario.

Section 3.1 Work by Macdonald et al: Summary of selected research results by Macdonald and Ounis

In Section 3.2, we present a paper written by Cummins, Lalmas, and O'Riordan [CLO10] in which an optimization of the technique CombSUM TOP *n* (introduced and discussed in Section 2.2.6) is considered. The procedure described in the paper is related to an approach in the following Chapter 4 of this thesis, in which we present new techniques.

Section 3.2 Learning Aggregation Functions for Expert Search: Presentation of further work related to the research questions

The aim of the sections in this chapter is not to focus on a complete summary of the works, but rather to highlight the points of the writings that are relevant to this thesis and its research questions. A summarization is given in Section 3.3.

Section 3.3 Summary of the Presented Works

3.1 Work by Macdonald et al.

3.1.1 The Voting Model for People Search

In his thesis "The voting model for people search" [Mac09], Macdonald evaluates the techniques presented in the previous Chapter 2. Essentially, the work is based on two collections, which were the basis for three TREC conferences with their enterprise tracks and expert search tasks from 2005 to 2007, also referred to as EX05, EX06 and EX07 in the following sections.

Evaluations are largely based on two TREC collections

W3C Collection (EX05, EX06) This collection consists of 331,037 documents collected by a crawl of the top-level domain w3.org in 2005. In addition to the actual web presence, email archives, Wikis, code repositories, personal home pages, and other documents are included [CVS05]. In addition, participants are provided with a list of 1,092 potential experts.

CERC Collection (EX07) The CSIRO Enterprise Research Collection (CERC) consists of 370,715 documents crawled in March 2007 from the csiro.au web domain. This collection from the Commonwealth Scientific and Industrial Research Organisation is more characteristic for an enterprise, unlike the W3C Collection. For the expert search task, the topics and their experts were determined by the so-called science communicators, who are responsible for reporting and external communication, as a basis for the evaluation [Bai+07].

Creation of candidate profiles In addition to the voting techniques themselves, the relevance of the ranked candidates is based on the documents assigned to each candidate. For each candidate or expert, the assigned documents define the *candidate profile* or also referred to as the *expert profile*. These assigned documents represent supporting evidence of their competence in the requested topic and have to be created in advance for both the W3C collection and the CERC collection. In Sections 3.4.2.1 and 3.4.2.2, Macdonald distinguishes between two ways of generating these associations of documents with candidates as candidate profiles:

Two different methods to generate candidate profiles

On the one hand, there is *manual candidate profiling*, in which the profiles are created or compiled manually, by third parties or even by the candidates themselves. The other possibility of associating documents with candidates is indicated with *automatic candidate profiling*: In this variant, expert profiles are generated from documents by evaluating the name and e-mail address in the content, sender or recipient addresses of messages or homepages, self-created or visited, as a relation of the candidate to the document.

For both collections, no candidate profiles are available, and so in the thesis based on last name, full name, full name with aliases, and email address, four different variants of profile sets are created via document analysis (Section 6.2.3). The quality of the profiles is determined by complete coverage of all relevant documents for each candidate, with no documents being incorrectly assigned to the candidates. Macdonald speaks here of minimizing false negatives and false positives.

Table 3.1: Overview of the three Expert Search Tasks of the years 2005-2007 (EX05-EX07), taken from the thesis "The voting model for people search" from Macdonald [Mac09]

	EX05	EX06	EX07
Corpus	W3C (331,037 docs)	W3C (331,037 docs)	CERC (370,715 docs)
#Candidates	1,092	1,092	3,475
#Topics	50	49	50
Evaluation Method	Ground Truth	Supporting Documents	Oracle Questionnaires
Mean #Relevant Cand.	30.18	51.48	3.04

During the evaluations, it emerges that profiles based on e-mail addresses and full names are the most accurate and have the smallest average sizes. Ambiguities in the assignment of a document to candidates with the same name can be avoided with a higher probability.

Relevance assessments In Section 3.4.5 of his thesis, Macdonald points out the difficulties in evaluating expert searches: Document rankings are easier to assess than expert relevance ratings. In the evaluations in his thesis, there are three different methods for relevance assessments of experts, also applied in the TREC tracks in 2005-2007.

Three different methods to generate relevance assessments

The *preexisting ground truth* is applied in the 2005 expert search task (EX05). Here, the queries are also, at the same time, the names of the W3C working groups [CVS05], whose composition by experts is known. This ground truth can also be used as a relevance assessment for the evaluation of expert rankings for each topic.

In the 2006 Expert Search Task (EX06), a procedure called *Supporting Evidence* is used as a relevance assessment and a basis for the evaluation[1]: All participating systems submit ranked documents supporting each candidate's expertise in addition to the candidate rankings for each task.

1: Macdonald also refers to the procedure in the following course and in the special case of EX06 as *Supporting Documents*

For evaluation, both the top-ranked candidates of the systems and their top-ranked documents are pooled. Assessors now assess the relevance of experts from the submitted rankings based on the documents from the pool.

Macdonald's third technique to generate a relevance assessment is called *Candidate & Oracle Questionnaires*. Here, the experts themselves are asked about their expertise on each topic. In the context of TREC 07 and its Expert Search Task (EX07), the technique was modified by assigning the so-called *Science Communicators* to form topics and, at the same time, to indicate associated experts.

Evaluation scenarios The first evaluations of the voting techniques are carried out on the three expert search tasks from 2005 to

2007 (EX05-EX07) with their collections. Table 3.1 provides a summary of the collections used with their key data and the evaluation methods used based on the assessments obtained.

In addition, separate runs are performed on the four different candidate profile sets, as well as on various document ranking procedures. The effectiveness measures evaluated are MAP for the entire expert ranking, P@10 for the quality of the first-ranked experts, and MRR for the first-ranked relevant candidate [2].

2: The effectiveness measures MAP, P@k and MRR are described in detail in the book *Introduction to Information Retrieval* [MRS08, sec. 8.4]

All voting results for the effectiveness measures are related to the median of the participating systems from the TREC tasks, and the result of the virtual documents as concatenations is used as a further basis for comparison (called *Virtual Docs* in the following).

Best results for the full name candidate profile set

Evaluation results Regarding the candidate profiles, the evaluation in Section 6.3.1 of the thesis gives the best results for the full name candidate profile set. The sets based on last name and last name + aliases provide too much ambiguous information for the mapping and are therefore very noisy. The candidate profile set based on email addresses, which has the smallest average profile of all four, provides too little information on a candidate and is not convincing in comparison.

The voting techniques are evaluated based on the first test runs in Section 6.3.2. of his thesis. The comparison is made on the basis of the full name profiles, which provide the best results in the previous section. An extract of the evaluation results is presented in Table 3.2, based on the document weighting model BM25 that is also applied as the weighting model in our experiments.

Score-based, exponential variants are among the best performing techniques

Among the winners of the evaluation are expCombSUM and expCombMNZ, which increase the weights of the highest-ranked documents disproportionately: *"Hence, a candidate associated with a few pieces of expertise evidence that are strongly related to the topic (strong votes) is more likely to be expert than a candidate with many weak votes."* [Mac09, sec. 6.3.2]

Macdonald further highlights that the MAP values of CombMAX, CombSUM, CombMNZ and exponential variants such as expCombSUM and expCombMNZ statistically significantly outperform the median results of the TREC runs.

ApprovalVotes[3], as a quantity-based technology, shows good results for the EX05 and EX06 tasks, but is not convincing for the EX07 task. With regard to ranking-based methods, Macdonald states that BordaFuse produces better results than *RR* in all tasks for all measures.

3: In the thesis, the term ApprovalVotes is synonymous with the technique Votes described in Section 2.2.1.

Score-based techniques appear differently, depending on the combination: For EX05 and EX07, for CombMAX, the results are better

Table 3.2: Extract of Macdonald's evaluation results for selected voting techniques for the Expert Search Tasks of the TREC Tracks 2005-2007 (EX05-EX07), based on the full name candidate profile. This extract is limited to the document weighting model BM25 and selected voting techniques. For a better overview, the best results within a column are shown in deep green. Worse results are colored from light green, white to red according to their quality.

Technique	MAP	MRR	P@10
	EX05		
TREC Median	0.1402	0.5067	0.2600
Virtual Docs	0.1070	0.2979	0.1880
ApprovalVotes	0.1707	0.5335	0.2840
RR	0.1723	0.5336	0.2840
BordaFuse	0.1872	0.5625	0.3000
CombMAX	**0.2398**	**0.6053**	0.3340
CombSUM	0.1754	0.5324	0.2860
CombMNZ	0.1738	0.5344	0.2860
expCombSUM	0.2291	0.5763	**0.3360**
expCombMNZ	0.2040	0.5607	0.3180
	EX06		
TREC Median	0.3412	0.8316	0.5082
Virtual Docs	0.3452	0.6407	0.4143
ApprovalVotes	0.5163	0.8986	0.6469
RR	0.5185	0.9014	0.6490
BordaFuse	0.5409	0.9095	0.6490
CombMAX	0.4833	0.8465	0.5939
CombSUM	0.5280	0.9116	0.6469
CombMNZ	0.5240	0.9201	0.6490
expCombSUM	**0.5523**	0.9241	**0.6571**
expCombMNZ	0.5492	**0.9252**	0.6551
	EX07		
TREC Median	0.2468	0.4013	0.1060
Virtual Docs	0.3005	0.3964	0.1120
ApprovalVotes	0.2277	0.3035	0.1020
RR	0.2322	0.3080	0.1060
BordaFuse	0.2736	0.3538	0.1360
CombMAX	0.3616	0.5070	0.1480
CombSUM	0.2721	0.3588	0.1240
CombMNZ	0.2501	0.3241	0.1220
expCombSUM	**0.3809**	**0.5106**	**0.1540**
expCombMNZ	0.3576	0.4642	0.1460

than those of CombSUM and CombMNZ; for EX06, the tendencies are reversed. Regarding CombMNZ, he notes that the additional component, the number of voters per candidate, does not provide any improvements in the tasks.

Techniques that focus on the top of the document ranking do not necessarily produce good P@10 or MRR results. As an example, Macdonald states that ApprovalVotes, considering the complete ranking, gives better values than CombMAX for the two measures in EX06.

The techniques CombANZ, CombMIN, and expCombANZ are left out of the extract shown in Table 3.2 because they do not perform well in each task and are significantly underperforming compared to the TREC medians. In further experiments and evaluations in the thesis of Macdonald they are not considered further.

Quantity-based and partially score-based techniques benefit from normalized profiles.

Candidate-length normalization To counteract the bias that arises for some voting techniques due to size differences in the candidate profiles, Macdonald applies normalizations to the candidate profiles. The basis of the approaches for normalizing the expert profiles is the length of each profile, defined in his thesis as *l_pro*. According to Macdonald's definition, the value of this length is derived either from the *number of tokens* as the number of terms of a complete expert profile or the *number of documents* associated with the expert. Based on *l_pro* and its two variants, two formulas are used for normalization. The first variant $Norm1$ revalues the score for an expert candidate C resulting from the query Q with the reciprocal value of *l_pro*:

$$score_{\text{Norm1}}(C, Q) = score(C, Q) \cdot \frac{1}{l\_pro} \tag{3.1}$$

The second formula as variant $Norm2$ allows a more precise control of the normalization and is defined as:

$$score_{\text{Norm2}}(C, Q) = score(C, Q) \cdot \log_2(1 + c_{\text{pro}} \cdot \frac{avg\_l}{l\_pro}) \tag{3.2}$$

The argument of the dual logarithm consists of two summands. The first summand with its value of 1 ensures that the resulting value of the logarithm does not become less than zero. As the second summand, the average profile length *avg_l* is divided by the length of the candidate profile *l_pro* and multiplied by a factor $c_{\text{pro}} > 0$, known as the hyper parameter. The smaller c_{pro} is set, the higher the impact of normalization.

Four variants, the two definitions of *l_pro* and the two formulas for normalization 3.1 and 3.2, are applied to seven voting techniques from Table 3.2 (*RR* is left out). Based on the existing scenarios of the Expert Search Tasks EX05-EX07, normalized voting techniques are evaluated and assessed [4].

4: As in the previous evaluations, the tests are carried out on four different candidate profile sets, each with four different document ranking procedures, including BM25.

The evaluation results shown in [Mac09] in tables 6.17 - 6.20 confirm that quantity-based techniques, such as ApprovalVotes benefit, and techniques such as CombMNZ or expCombMNZ often also produce improved results, since they also consider the number of voting documents. CombSUM and expCombSUM can benefit here in some cases, since they also implicitly benefit from the number of voters or are biased by large expert profiles. Macdonald states that CombMAX yields the best results in most cases without normalization. In Section 6.4.3 of his thesis, Macdonald states:

"*In conclusion, we have seen that candidate length normalisation is necessary in some settings to improve the retrieval performance of some voting techniques, under certain noisy conditions. In particular, the evaluation showed that normalisation is more useful on the more difficult EX05 and EX07 topics than on the EX06 topics. We conclude that length normalisation is important to take into account in the Voting Model, as it can significantly improve the performance of some voting techniques, particularly when inaccurate or noisy candidate profile sets are applied.*" [Mac09, sec. 6.4.3]

Size of the document ranking In Section 6.5 of his thesis, Macdonald examines the effect of changing the number of initial voting documents. While in the previous experiments the number of voting documents was set at 1,000, in these experiments he evaluates variations from 5 to 2,000 initial voting documents for the TREC tracks EX05-EX07.

The optimal number of initial voting documents is collection-specific and depends on the applied voting technique.

Applying BM25-based document search, an increase in MAP performance can be observed for all techniques up to the range of approximately 50 – 150 voting documents in the three TREC tasks. An additional increasing number can have different effects and depends on the voting technique, the collection structure, and the number of experts searched.

In the EX05 task, performance on a MAP basis increases for some techniques such as expCombSUM and CombMAX, as the number of voting documents increases, while stagnating and decreasing slightly for ApprovalVotes, CombSUM, and BordaFuse. A rather similar picture emerges for the EX07 task.

In the EX06 task, which contains a higher average number of relevant experts, the result is different. There is no noticeable tail-off in MAP values for this task; a deterioration can be observed for

some techniques with 1,000 or more voting documents. For this task, Macdonald states:

> *"The overall trends show that as this task has more complete judgments with a higher number of relevant candidates, voting techniques are able to rank higher more relevant candidates by looking further down the document ranking for even the most tangentially-related evidence of expertise."* [MO09, sec. 6.5]

3.1.2 The Influence of the Document Ranking in Expert Search

In the eponymous paper [MO09], Macdonald and Ounis investigate the impact of document ranking quality on voting techniques and their results for expert ranking.

Prerequisite: Assessments for both document search and expert search are available for the CERC collection.

The two tasks from TREC 2007, for which both assessments for the results of the document search and for the expert search are available, serve as the basis for the evaluation here.

The starting point are the 63 document rankings submitted for the TREC 2007 document search task [Bai+07]. On the basis of these rankings, the expert search is performed in this investigation. The results of both evaluations, respectively, for the document ranking and the expert ranking, are related to draw conclusions on the influence of the document ranking on the expert ranking. Candidate profiles are obtained based on their full names and email addresses in the documents, as described in the previous Section 3.1.1. The techniques Votes, BordaFuse, *RR*, CombMAX, expCombSUM and expCombMNZ are examined in the article.

Correlation analysis between the key performance indicators of the document and expert rankings

TREC document rankings For the mentioned voting techniques, based on the 63 document rankings, the MAP and MRR values for expert search are determined based on the available assessments and set in relation to the MAP, MRR, nDCG, and P@N ($n = 10, 30, 50$) of the document ranking runs.

The aim is to answer the question of a correlation between the two relevance measures from the expert searches and those from the document rankings.

The first result of the study is that there is a basic correlation between the quality of the rankings, but a direct correlation of the corresponding measures on the document and on the expert side cannot always be expected.

When comparing the correlations, it turns out that the two measures of CombMAX have a high correlation with the MRR of the document ranking. This can be explained by the focus on the upper ranks of the document ranking.

For Votes, there is a high correlation with document-based recall, since here the number of voting candidates is included and not their quality.

BordaFuse, expCombMNZ and expCombSUM show correlations with MAP, nDCG, and document-based recall. Surprisingly, the results of *RR* tend to be similar to those of BordaFuse, with no significant correlation to the results for the measures of front rank quality.

In summary, the authors make the following comments.

"However, we do not find any 100% correlations, showing that not every improvement in document search effectiveness can have a positive impact on an expert search engine. The correlations found here do not show that topic relevance document retrieval performance is perfectly related to candidate retrieval performance. This infers that there are some characteristics of the document ranking which are important to the voting techniques that are not being captured by the topical relevance document evaluation measures." [MO09, sec. 4.1]

Virtual document rankings check the dependency of the expert rankings for the entire MAP results spectrum at document level.

Virtual document rankings Since the document rankings from the TREC runs range from 0.28 to 0.45 in terms of their MAP value, the authors follow the question of how the relationship between the quality of the document rankings and the expert rankings is given in a broader range. For this purpose, they generate 400 document rankings based on the collection using the AP simulation algorithm [TS06], which covers the entire range of MAP in terms of assessments. Since no document scores are formed, but only their rankings are formed within this method, only ranking-based voting procedures are considered: Votes, BordaFuse, and *RR*.

Based on the wide range for the MAP results of the document rankings, strong correlations arise here. However, the investigation in this scenario is only of limited value, as the measured document-based performance indicators (MAP, MRR, nDCG, and P@N) also show a high correlation with each other, in contrast to the realistic 63 document rankings.

3.2 Learning Aggregation Functions for Expert Search

To counteract the bias caused by size differences in the expert profiles, normalization can be applied as described in Section 3.1.1.

Another option is to use ranking-based attenuation functions in class scope or at the document level, which use document score weighting to attenuate the influence of voting documents as the number increases. We introduce these functions in Sections 4.1.3, 4.1.4 and 4.1.5 and apply them in Chapter 5 as part of the evaluations.

In their work which is discussed next, Cummins, Lalmas, and O'Riordan [CLO10] pursue an approach based on the voting technique CombSUM TOP *n*: The preliminary considerations in the article "Learning Aggregation Functions for Expert Search" investigate the ideal number of *n*, followed by the evaluation of a scoring function which, according to equation 4.6 on page 64, not only multiplies the document scores by 1 and 0 for including and neglecting them, but also sets a factor in front of each document score as a continuous damping function.

Development of an aggregation function based on a learning model

Basic concept In their paper "Learning Aggregation Functions for Expert Search" [CLO10], the authors attempt to develop an aggregation function based on a learning model that optimally aggregates documents in the expert search task. In experiments, this aggregation function is trained and evaluated using individual query features and, in the second case, features of expert profiles. For their experiments, the authors use the collections W3C and CSIRO from the TREC expert search tracks of the years 2005-2008.

Preceding analysis: Determine an ideal *n* for CombSUM TOP *n* for each query

Manual search for an optimal *n* The starting point of the investigations is the voting technique CombSUM TOP *n* and its parameterization for the best results. In a preliminary analysis, the optimal *n* for the best MAP-based result is determined for each query of the expert search tasks. It turns out that the optimal values of *n* are individually different for each query and can vary significantly.

As the best results are achieved on average for the four runs on the two collections with $n = 5$, this parameterization is used as a reference for further experiments and considerations.

Establishing a correlation of the optimal *n* with query features

Query features After the optimal *n*-values for CombSUM TOP *n* have been determined for each query, the authors attempt to relate these values to individual query features in order to get an input to a prediction function. For this purpose, query features such as $idf_{\min}$, $idf_{\max}$, query length ql, and others are determined

to establish a correlation with the best n for each query. However, it turns out that direct and significant correlations with the optimal number of voting documents for each query cannot be established.

Expert features In this section, the authors examine expert features on the basis of relevant and non-relevant experts for each query. For each topic, the TOP 50 experts and their document features are used in the ranking based on the baseline CombSUM TOP n with $n = 5$. The following three measures are determined as input variables in the learning function for each expert x_i:

- $no\_docs_{x_i}$: Maximum number of documents that can be aggregated for the expert profile x_i
- $top\_exp\_score_{x_i}$: Maximum document score of the expert profile x_i in the document ranking
- $top\_exp\_rank_{x_i}$: The highest document rank of the relevant expert profile x_i in the document ranking

Development of a learning aggregation function based on the features of queries and expert profiles

Based on the fact that for each query using CombSUM TOP n there is an individual n for its best performance, the authors consider finding a continuous weighting function using genetic programming (GP), which determines the revaluation of the document scores for two scenarios: For the *query-based scenario*, the learned aggregation function determines the ideal weights for all documents based on the features of the query; for the *expert-based scenario*, learned function-generated weights are determined based on the properties of the expert profile, and individually revalue the voting documents for each expert.

Weighting aggregation function as a substitution for CombSUM TOP n

For both scenarios, weighting functions are determined, each based on the individual features of the query or the expert profile. These weighting functions are continuous in nature and generate factors between 1 and 0 to flatten the document score according to its ranking in class scope. To generate the dynamic for the voting scores and the two scenarios, the authors analyze the individual query features and the individual expert features.

TREC 2005 collection serves as training collection

Based on a genetic programming (GP) approach, it is evaluated whether the measures determined for both *query-based* and *expert-based* scenarios can be used to train a weighting function that further improves the MAP and P@10 results of expert searches with respect to the baseline CombSUM TOP n with $n = 5$. The aggregation functions are trained on the basis of the TREC 2005 collection, and the evaluation is carried out with the TREC collections of the three following years. MAP and P@10 are used as performance indicators.

Evaluation results For the *query-based scenario*, improvements can only be observed for the TREC 2005 collection; the results for the other collections from 2006 to 2008 are worse than those of the baseline CombSUM TOP n from these years having $n = 5$. The authors assume *"that any combination (even non-linear) of the query-based features is not likely to bring about an improvement in performance"*.

The *expert-based scenario* is different. Here, the aggregation function trained on expert profiles delivers significantly higher MAP and P@10 values for both the training collection and the TREC collections for 2007 and 2008. The results based on the 2006 collection are slightly below the baseline. The reason for this is that in this track, expert profiles with many documents were also more relevant in the assessments.

3.3 Summary of the Presented Works

This chapter highlights researched aspects of expertise retrieval using voting techniques that are also important for the generalized search for classes. A fundamental observation is that voting techniques that emphasize the upper ranks of the document ranking often yield better results and are more invariant to expert profiles of different sizes. This is shown by the results of Macdonald, an example shown in Table 3.2 on page 41 in which the techniques that emphasize these high ranks are outperformers on the basis of MAP, MRR, and P@10. Ranking and score-based voting techniques can significantly increase their performance by applying length-based normalization of expert profiles, only CombMAX yields the best results without normalization [Mac09, sec. 6.4]. Furthermore, the number of initial voting documents affects the quality of the results. An optimum is to be chosen based on the collection structure and the applied voting technique, as Macdonald states in his thesis [Mac09].

Emphasizing the top ranks or global restriction of voting documents can improve performance.

However, the discussed works also show that the global performance of the techniques varies depending on the collection. For example, the global performance range is significantly different in all evaluation measures between the evaluated collections, as can be seen in the results in Table 3.2. To some extent, this behavior can also be individually related to the selected queries or the relevance ratings of the tracks. In addition to the global differences in the result ranges, the performance of individual techniques varies differently, as can be seen in the example of CombMAX or ApprovalVotes in tracks EX05 to EX07. CombMAX delivers only

The structure of a collection has a global influence on the overall quality of the results.

Individually different performance of voting techniques based on different collections

moderate results on track EX06, while it is among the top performers on the other two tracks in Table 3.2. ApprovalVotes delivers results on the mentioned track that are in the middle of the field, but poor results in the other tracks.

Higher relevance of document rankings improves results of voting techniques, but with limited correlation.

In their paper "The influence of the document ranking in expert search" [MO09] Macdonald and Ounis investigate the influence of the quality of the document ranking on the results of voting techniques. To some extent, more relevant document rankings also have a positive influence on the subsequent expert ranking. Depending on the evaluation measure and the processing of the documents as voting candidates, the techniques react differently to the rankings of different qualities.

Based on CombSUM TOP 5 as a high-performing technique, Cummins, Lalmas, and O'Riordan show in their paper [CLO10] that voting results yielded by this technique can be further improved using an individually applied damping function. This function is derived with effort from each expert's profile and applied individually for the aggregation of voting documents.

4 Extended Techniques and Interrelationships

In the first Section 4.1 of this chapter, new voting techniques are presented as modified and extended variants of the techniques presented in Chapter 2. These new techniques are also included in the evaluations in Chapter 5.

Section 4.1 New and Modified Techniques: Introduction of extended and new voting techniques using the example scenario

The second Section 4.2 gives a general overview of all the techniques presented with their characteristics and interrelationships.

Section 4.2 Relationships of the Presented Techniques: Classification of the new voting techniques in relation to the existing ones

4.1 New and Modified Techniques

The modifications and innovations of the techniques discussed in Chapter 2 are outlined below.

4.1.1 sqCombSUM

Based on the idea of expCombSUM and the knowledge of slow decreasing $score(d, q)$ values, this technique is newly introduced in this thesis and amplifies the differences by the exponent 2. The basic idea is to implement a revaluation of the scores using an exponential approach with the scores as a basis and that generates an amplification of the high-ranking documents that is not as extreme as that of expCombSUM.

sqCombSUM squares the document scores as an amplification of high-ranked documents.

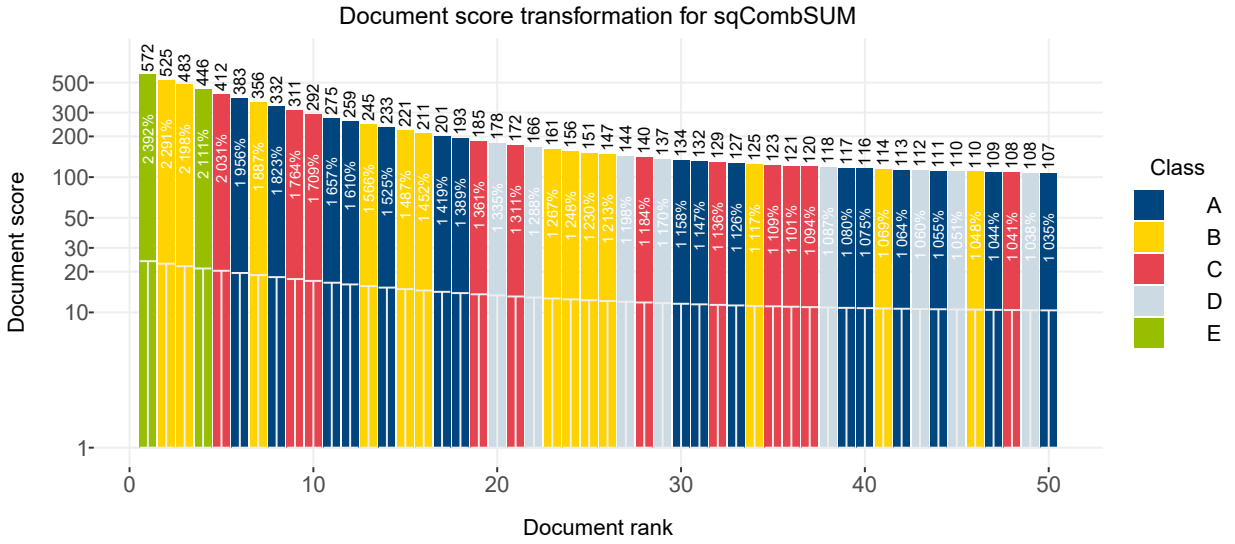

Figure 4.1: Transformed document scores (rounded) for sqCombSUM in logarithmic representation: The top ranks of the global document ranking receive a much greater increase in value and thus greater significance for class voting. The white markers on the bars indicate the original score value, and the white values indicate the percentage increase by which the score increases.

Figure 4.1 shows the ranking of the exemplary scenario in Chapter 2 with the squared document scores in a logarithmic representation: The score of the document ranked first is increased by a factor of almost 24 in this example; for documents in the lower ranks, the value decreases approximately tenfold.

As sqCombSUM uses the square function to amplify the deltas of the document score values, the formula is as follows:

$$score_{sqCombSUM}(c,q) = \sum_{d \in R(q) \wedge d \in c} score(d,q)^2 \qquad (4.1)$$

By summarizing the squared scores without restricting the number of voting candidates per class, the first-ranking documents are emphasized, which causes a positive impact on the search results as we evaluate in Chapter 5. Like CombSUM and expCombSUM, all documents per class are taken into account, which still keeps classes with a high document count potentially to be preferred – in a more attenuated form than in the case of CombSUM.

This observation can also be made using the example scenario and the class ranking in Figure 4.2: Although the blue class A has the highest sum in the unchanged score ranking, candidates of class B in sum get the highest value increases by squaring, which leads to the swapping of the ranking places applying CombSUM and sqCombSUM in the scenario. Despite the increase in the scores for the top-ranked documents, class E (green) is ranked in last place; however, the score difference to the previous position of class D is

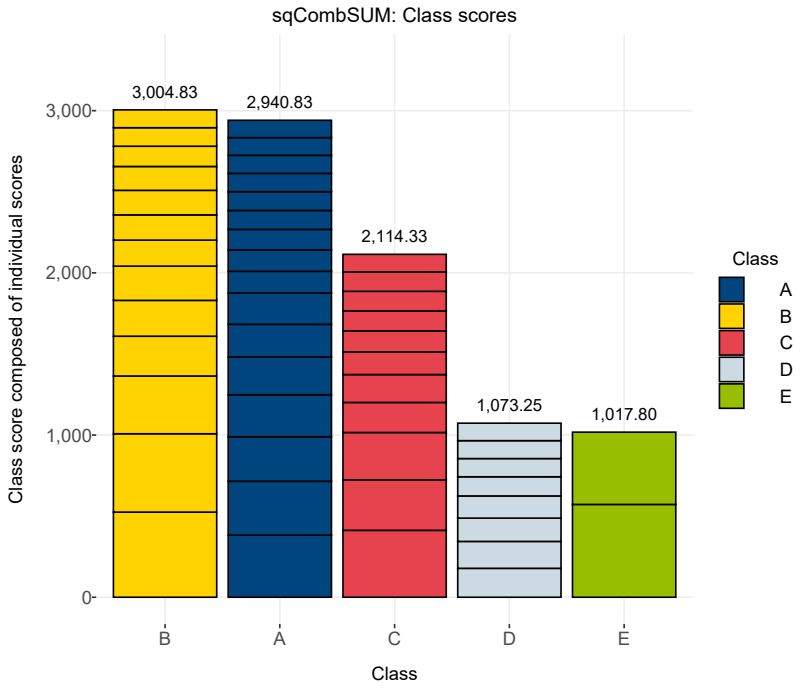

Figure 4.2: Plot of the class ranking and class scores for sqCombSUM on the linear scale: While the blue class A was still in first place for CombSUM (see also Figure 2.3 on page 21), the yellow class B, which was previously in second place, is now in front. The reason for this is the increase in value of the top voting candidates by squaring the scores.

rather marginal here, in contrast to CombSUM (see Figure 2.3 on page 21).

4.1.2 sqCombMNZ

In analogy to expCombMNZ, the number of voting documents is multiplied by the squared scores.

As an equivalent to expCombMNZ and an extension of sqCombSUM, this technique sums the squared scores of the documents for every class and multiplies this sum by $score_{Votes}(c, q)$, giving an advantage to classes with many voting documents. The formula is defined as follows:

$$score_{sqCombMNZ}(c, q) = score_{Votes}(c, q) \cdot score_{sqCombSUM}(c, q) \quad (4.2)$$

Figure 4.3 shows the class ranking results for sqCombMNZ in the example scenario. By including the number of voting documents per class, the class ranking yielded by sqCombSUM as seen in Figure 4.2 is biased towards a result like CombMNZ.

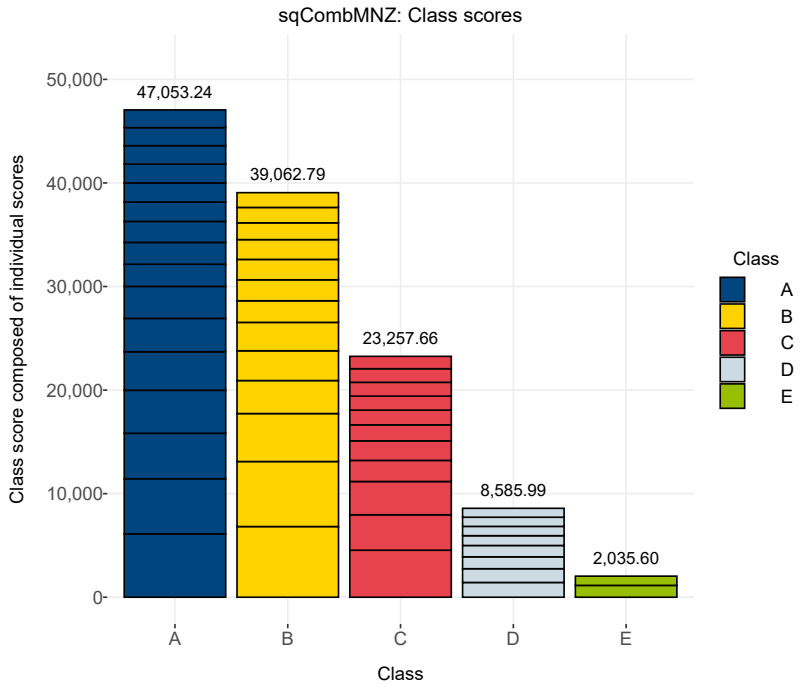

Figure 4.3: Class ranking for sqCombMNZ: According to the order of the classes, the ranking is the same as for CombMNZ. However, the squaring here means an additional strengthening of the higher ranked voting documents and thus a proportionally higher increase of the green or yellow class.

The factor $score_{votes}(c, q)$ prevails over squaring the scores within our example scenario, keeping the ranking the same as compared to the non-squared variant.

4.1.3 RR^x

Based on RR, the exponent x causes the adjustment of the voting characteristics

This voting procedure is based on the RR technique presented in Section 2.2.8. As an extension, we add an exponent x, which allows us to adjust the characteristics. Consequently, the formula for RR^x is defined as follows:

$$score_{RR^x}(c,q) = \sum_{d \in R(q) \wedge d \in c} \left(\frac{1}{rank(d,q)} \right)^x \tag{4.3}$$

$$x \in \mathbb{R}_{\geq 0}$$

The choice of x sets the envelope curve from which the scores of the voting documents are derived. By choosing very small or very large values for x, constellations emerge that approximate the results of two voting techniques presented:

Small values of x produce class ranking results that approximate Votes.

Smaller values for x Choosing small values for x or setting $x \to 0$ causes the envelope curve (grey curve in Figure 4.4) to descend more flatly, increases the score values and leads to scores having smaller differences while approximating each other. As the score differences decrease, it boils down to the fact that the number of voting documents per class becomes decisive. Therefore, the results with this parameterization approximate those of the Votes algorithm: As a limit, the extreme is $score_{RR^0}$, which leads to the mathematical equivalent of formula Votes.

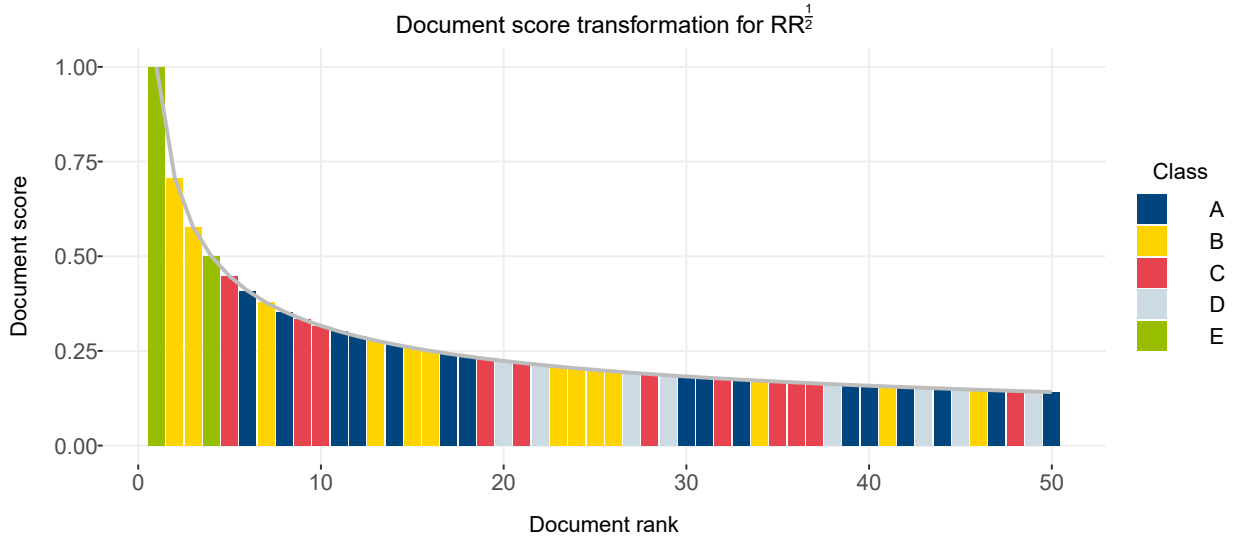

Figure 4.4: Document ranking of the scenario with revalued score values: The transformed scores result from the setting $x = \frac{1}{2}$, which causes a flatter curve and lower attenuation compared to Figure 2.11 on page 30.

Figure 4.4 shows an example of the envelope or revalued scores for $x = \frac{1}{2}$. Here, the attenuation of the scores is lower than in the

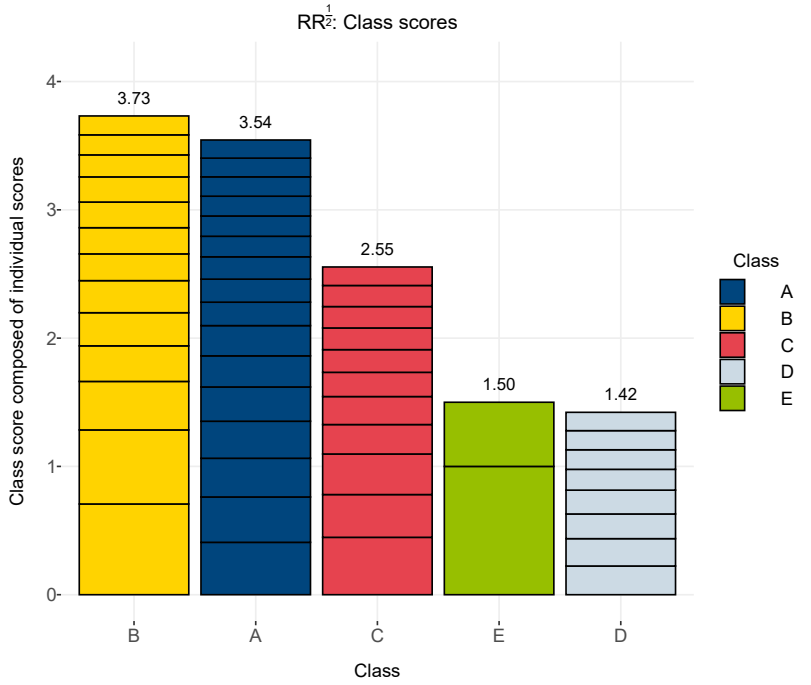

Figure 4.5: Class ranking for RR^x with the parameter $x = \frac{1}{2}$: The influence of class sizes is higher within this parameterization compared to Figure 2.12 on page 31.

original revaluation of RR shown in Figure 2.11 on page 30. Figure 4.5 shows the class ranking based on the revalued scores. Due to the lower attenuation of the document scores, compared to RR, the influence of class sizes, i.e., the number of voting documents, becomes more dominant. While in RR the green class is ranked second with only two but highly ranked documents, here the red and blue classes can prevail with more voting documents.

Table 4.1: Top five RR^x-based document score values for different variations of x. As x increases, the score totals for the top five ranks decrease while the ratios between the rank-based score values increase.

Technique ▹ Rank ▽	$RR^{\frac{1}{2}}$	$RR^{\frac{3}{4}}$	RR	$RR^{\frac{3}{2}}$	RR^2
1	1.00	1.00	1.00	1.00	1.00
2	0.71	0.59	0.50	0.35	0.25
3	0.58	0.44	0.33	0.19	0.11
4	0.50	0.35	0.25	0.13	0.06
5	0.45	0.30	0.20	0.09	0.04
Sum	**3.23**	**2.69**	**2.28**	**1.76**	**1.46**

Starting from $x = \frac{1}{2}$, Table 4.1 shows the development of the document scores for higher values for x. Only the top-ranked documents of the upper five ranks are considered in the table. As x increases, the sum of document scores decreases, while the ratios

of score values increase for each parameterization. Even in the range of $\frac{1}{2}$ to 2 for x shown, the changes have a fundamental effect on the class ranking.

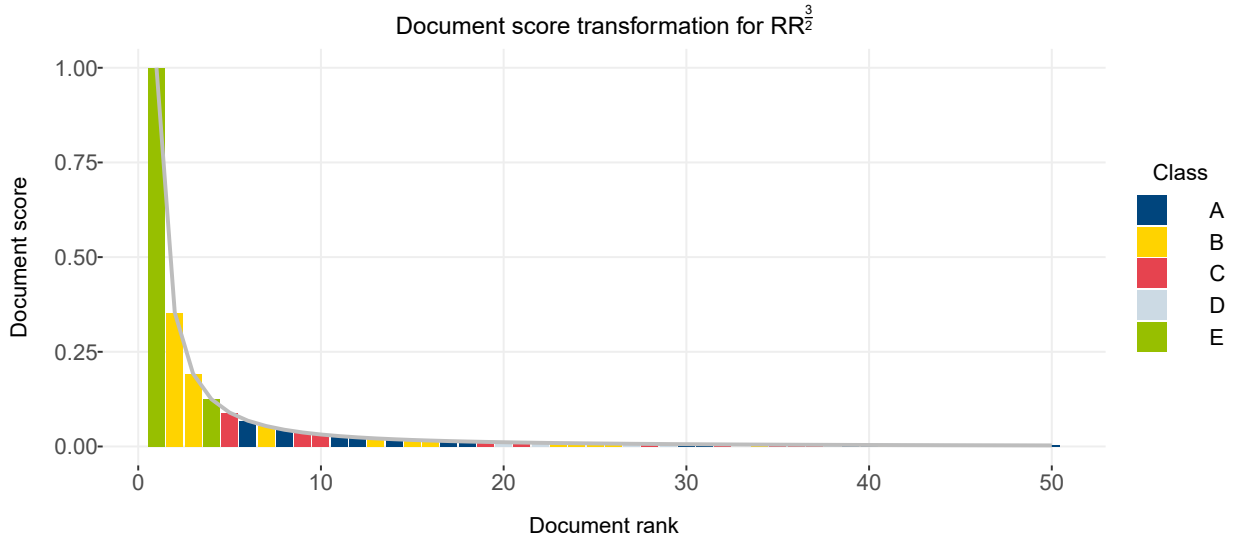

Figure 4.6: Document ranking of the example scenario with revalued score values by setting $x = \frac{3}{2}$: The envelope shows higher attenuation and decrease of the document scores.

Higher values for x This effect is demonstrated by an example with the parametrization of $x = \frac{3}{2}$. Figure 4.6 shows the revalued scores of the documents with the corresponding envelope. The attenuation is stronger here with the effect that the votes of higher-ranked documents get a higher influence. Here, the second-ranked document receives a score of $\frac{1}{2}^{\frac{3}{2}} = 0.35$, while the third-ranked document receives a score of $\frac{1}{3}^{\frac{3}{2}} = 0.19$, as can be seen in Table 4.1. Due to these more decreasing scores and the higher attenuation, it becomes increasingly difficult for low-ranked documents to place their associated class on the higher ranks of the class ranking by means of their number. This leads to the fact that the results of RR^x are close to those of CombMAX with higher values of x. The class ranking in Figure 4.7 shows this tendency: Classes E and B, in green and yellow, are in the lead here, as in CombMAX (compare Figure 2.10 on page 29), however, in contrast to CombMAX, the blue class A can still position itself as the third place due to its high number of voting documents. As the exponent x increases, this effect also disappears and the result of CombMAX is obtained.

High values of x produce class ranking results that approximate CombMAX.

As the exponent increases, RR^x more intensively prefers classes that are referenced by high-ranked documents from the document ranking: This strategy is also followed when applying expCombSUM and has proven to be promising in various results from the literature on expertise retrieval [MO06].

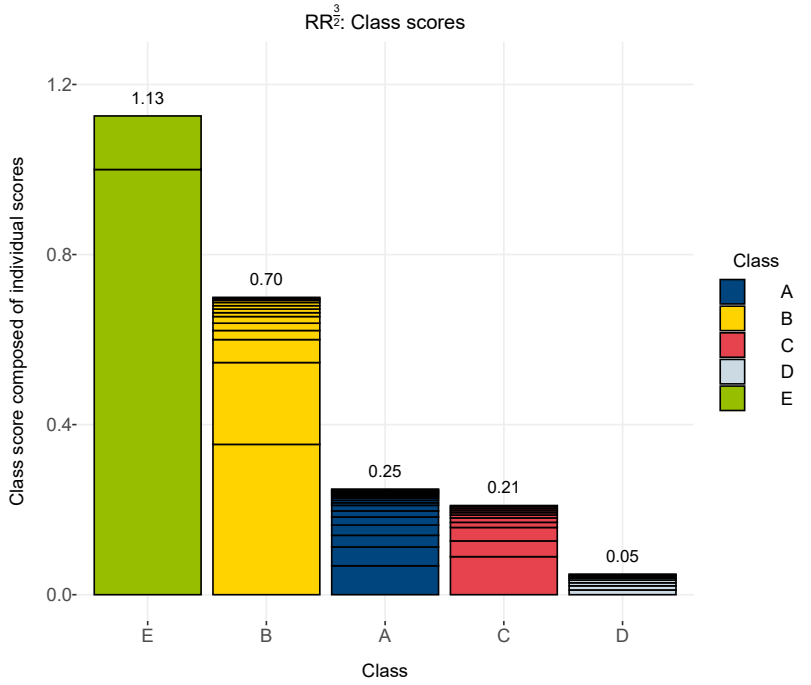

Figure 4.7: Class ranking for RR^x with the parameter $x = \frac{3}{2}$: Due to exponential high attenuation, the influence of highly ranked documents increases and the results tend to those of CombMAX.

Our evaluations in Chapter 5 cover parameterizations of x where

$$x \in \{\frac{1}{2}, \frac{3}{4}, 1, \frac{3}{2}, 2\}.$$

4.1.4 CombSUM RR^x

CombSUM RR^x as a combination of summing up the document scores multiplied with a class scoped, ranking based damping factor.

A further newly introduced technique is a combination of summed document scores and reciprocal rank valuation per class.

The rationale is that if there are multiple documents of a class c in the ranking, the highest ranked document $d_{c,1}$ reveals a fit to the query for c which can be quantified by $score(d_{c,1}, q)$ — according to the CombMAX technique.

For the second-ranked document of this class – $d_{c,2}$ of c – parts of its score might already be reflected in the score of the preceding document. Therefore, for subsequent documents of c in the ranking a discounting function is used. This function is implemented by multiplying $score(d_{c,r}, q)$ by the class-scoped reciprocal rank $\frac{1}{r}$ which is provided with an adjustable exponent x.

For every class, this aggregation sums up its voting documents' scores, whereby each document score is multiplied with its corresponding, class-based reciprocal rank raised to the power of x:

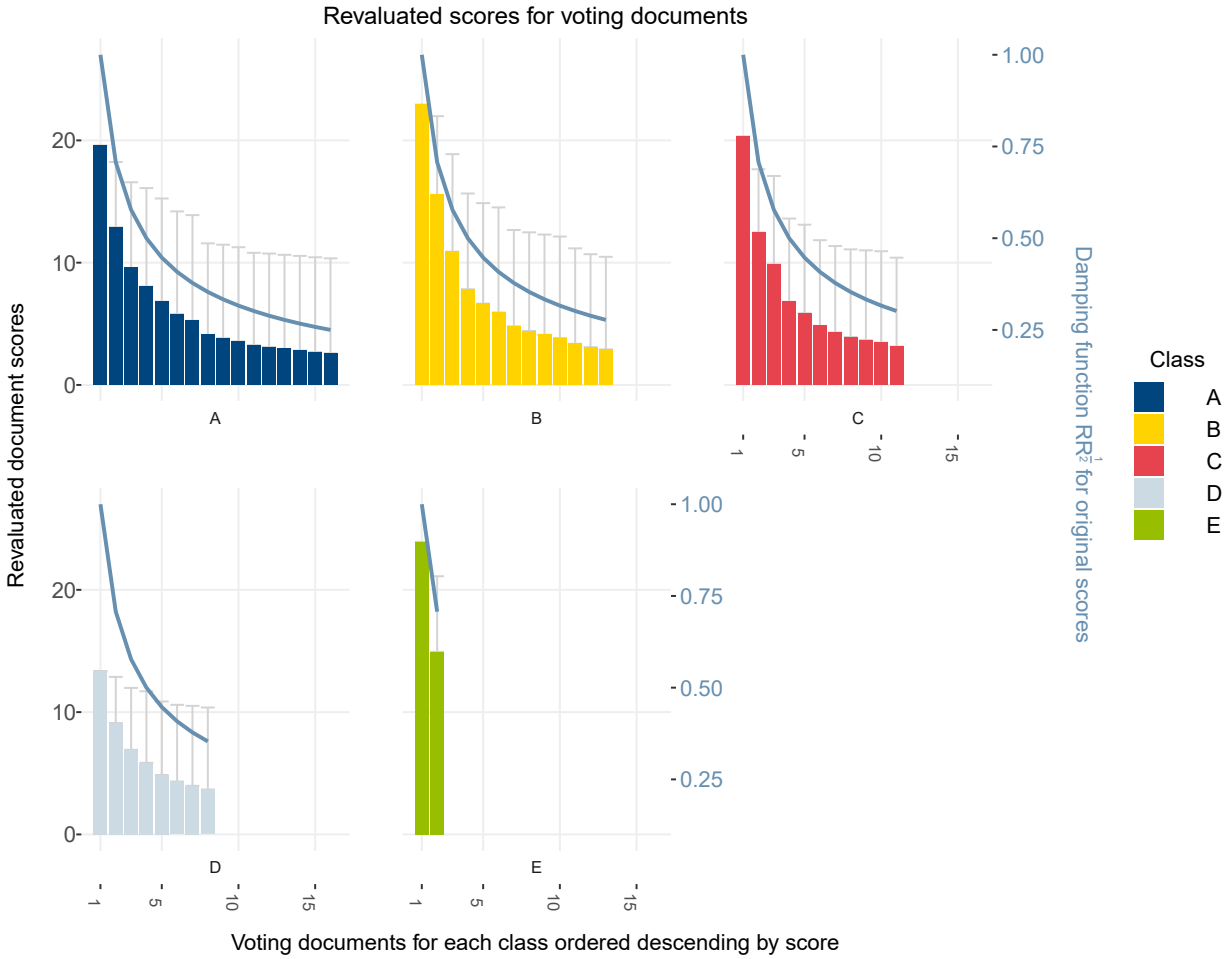

Figure 4.8: Revaluation of the document scores from the sample scenario for CombSUM RR^x, setting $x = \frac{1}{2}$: Recalculation is done in the context of each class. The figure shows the voting documents of each class with their original scores in gray and their recalculated values using the factor $\left(\frac{1}{rank(d,q,c)}\right)^{\frac{1}{2}}$.

$$score_{CombSUM\ RR^x}(c,q) = \sum_{d \in R(q)\ \wedge\ d \in c} score(d,q) \cdot \left(\frac{1}{rank(d,q,c)}\right)^x \qquad (4.4)$$

The evaluations in Chapter 5 consider x set to values of

$$x \in \{\frac{1}{2}, 1, 2\}.$$

This damping factor x can be used to map scenarios that approximate CombSUM or CombMAX via its extreme parameterization:

$x \to 0$ approximates CombSUM,
$x \to \infty$ yields results converging CombMAX.

CombSUM is approximated setting small values for $x \to 0$ which results in a multiplication of the document scores with a value converging 1.

CombMAX behavior is achieved with high exponents or $x \to \infty$. According to Table 4.1, the high attenuation of the following ranks means that the first and highest document score in a class has by far the highest weight and is decisive for the result of a class.

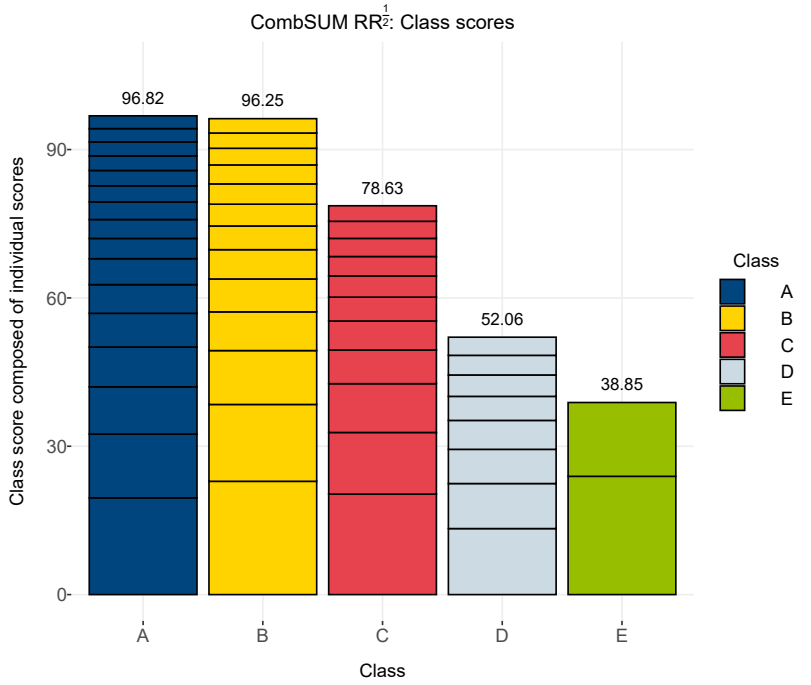

Figure 4.9: Class ranking for CombSUM RR^x setting $x = \frac{1}{2}$

The main goal of this technique is to achieve a certain degree of independence from class sizes and to correctly add the scores of voting documents in a class according to their probative value. CombMAX as one extreme of CombSUM RR^x is independent of class sizes, only the maximum voting document of a class is taken into account. CombSUM as the other extreme applies a simple addition of score values per class, which introduces a bias in favor of larger classes. Furthermore, the simple sum of the scoring documents per class does not reflect the correct relevance of a class.

Figure 4.9 shows the class ranking of the example scenario for CombSUM $RR^{\frac{1}{2}}$. Basically, the result with its order of classes still corresponds to CombSUM (see Figure 2.3 on page 21), but here the scores of blue class A and second-placed yellow class B are very close together due to attenuation by factor $RR^{\frac{1}{2}}$. Furthermore, the ratios of the scores of the first rank (class A) and the last rank (class E) are lower than those of the CombSUM class ranking. Therefore, the first indications of decoupling the success in the class ranking from the class size can already be observed by applying CombSUM RR^x with $x = \frac{1}{2}$. CombSUM RR^x can represent scenarios and a trade-off between CombSUM and CombMAX and is evaluated with the parameters mentioned in Chapter 5.

4.1.5 sqCombSUM RR^x

Scores from the document are squared followed by a class ranking-based attenuation.

This technique follows the intention of CombSUM RR^x with the difference in considering sqCombSUM described in 4.1.1. The technique is based on the squared scores of the global document ranking, whose values are provided with a class-based attenuation in a second step. This creates a separate context in which amplification and attenuation are applied. At the global document level, we leave it to squaring the scores, resulting in a moderate differentiation of the global-ranked documents and emphasizing the high-scored documents. Then, in the scope of the class, we vary the ranking-based attenuation, using different values for x to decouple the success of a class from its size.

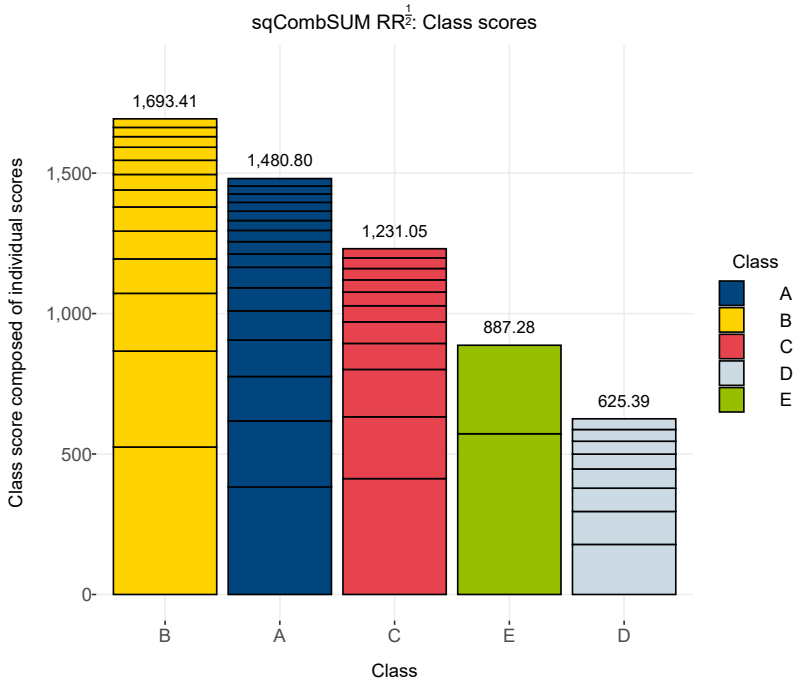

Figure 4.10: Class ranking for sqCombSUM RR^x setting $x = \frac{1}{2}$

For every class, the squared scores of the articles, multiplied by their reciprocal rank raised to the power of x, are summed up:

$$score_{sqCombSUM\ RR^x}(c,q) = \sum_{d \in R(q)\ \wedge\ d \in c} score(d,q)^2 \cdot \left(\frac{1}{rank(d,q,c)}\right)^x \tag{4.5}$$

In our experiments x is set to the values

$$x \in \{\frac{1}{2}, 1, 2\}.$$

Depending on the exponent x, the upper ranked documents and their squared scores for each class are emphasized differently.

Figure 4.10 shows the ranking of the classes with their scores for sqCombSUM $RR^{\frac{1}{2}}$. In contrast to CombSUM $RR^{\frac{1}{2}}$, the yellow class B, as in the case of sqCombSUM, can benefit from the squaring of the document scores and achieves the highest score and rank 1. However, the class-based attenuation makes the difference here in that the blue class A ranks second by a clearer margin. When looking at the lower ranks, it is noticeable that the green class E can outperform the light blue class D through this combination of squaring the document scores and subsequent class-based attenuation: Class E achieves rank 4 ahead of Class D – in contrast to the scenarios of CombSUM $RR^{\frac{1}{2}}$ and sqCombSUM.

4.2 Relationships of the Presented Techniques

The techniques presented result in 21 evaluation cases that arise from the parameterizations in the next chapter.

In the preceding sections, voting techniques are introduced as well-known or new algorithms to rank classes. This leads to 21 evaluated techniques in the next Chapter 5, based on different parameterizations in selected techniques.

Figure 4.11 shows a complete overview of the techniques presented, classified, and related to input based on quantity, ranking, or score. On the basis of these three input parameters, the techniques presented provide different results for class rankings, as shown in the previous chapters. The quality of the three parameters is also determined by the composition and structure of a collection. The number of classes in the index and their size ratios are of crucial importance for the results of the class ranking.

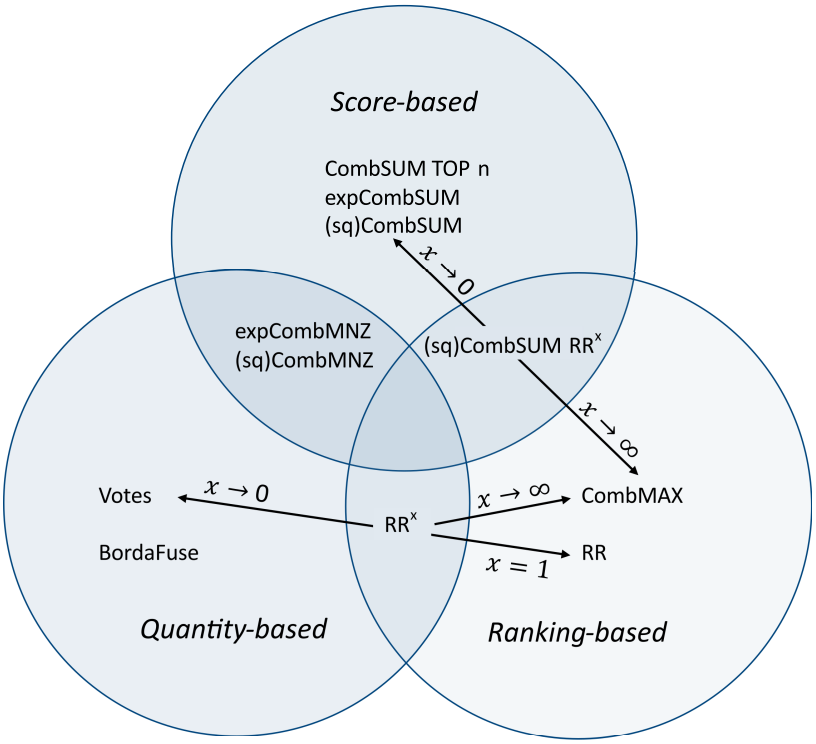

Figure 4.11: The voting techniques discussed in Sections 2.2 and 4.1 are qualified based on the key figures utilized and displayed with their relationships.

Interpolation between techniques To address this, we introduce the exponent x for the techniques RR^x, CombSUM RR^x and sqCombSUM RR^x and evaluate to what extent voting techniques can be adapted for optimized results.[1] These techniques are located at the intersections between quantity- and ranking-based classifications and ranking- and score-based classifications.

1: An optimization of the techniques using the exponent x is done by performing a pragmatic grid search.

RR^x: Approximation between quantity- and ranking-based techniques

The RR^x technique approximates both Votes and CombMAX by adjusting the exponent x. This allows us to evaluate the extent to which the settings between quantity-based methods and techniques that focus on the first rank of each class provide optimized results.

sqCombSUM RR^x and CombSUM RR^x: Approximation between scoring and ranking-based techniques

Just like quantity-based techniques, score-based methods can also lead to a bias that favors large classes with many voting documents, especially in the case of slowly decreasing score values. Likewise, it is known from the literature that the quality of the voting results for a class is only increased to a limited extent by the number of its voting documents [CLO10]. For this purpose, we also evaluate sqCombSUM RR^x and CombSUM RR^x, which approximate the techniques CombSUM and CombMAX by adjusting the exponent x.

In Figure 4.11 the techniques CombMNZ, expCombMNZ, and sqCombMNZ are to be found at the intersection between score-based and quantity-based voting, since they include both input measures.

CombSUM TOP n: Approximation between CombSUM and CombMAX

An approximation from CombSUM to CombMAX is achieved by varying the parameter n at CombSUM TOP n. This variation can also be interpreted as a damping function for CombSUM at the class level in the form of a step function that results in the redefined function for CombSUM TOP n from 2.12 as:

$$score_{CombSUM\ TOP\ n}(c,q) = \sum_{d \in R(q)\ \wedge\ d \in c} score(d,q) \cdot f\Big(r(d,q,c)\Big) \tag{4.6}$$

where $f\Big(r(d,q,c)\Big)$ represents the ranking-based function of a voting document in class-level scope:

$$f\Big(r(d,q,c)\Big) = \begin{cases} 1 : r(d,q,c) \leq n \\ 0 : r(d,q,c) > n \end{cases}$$

5 Search for Classes - Evaluation of the Introduced Voting Techniques

In this chapter, we evaluate the search and voting techniques presented in the previous chapters 2 and 4 by incorporating the findings of the literature discussed in Chapter 3.[1] As a basis, we use a bibliographic collection, which we have compiled and adapted to the requirements for the application of voting techniques and further evaluations of our search scenarios.

1: The setup, approach and evaluations of Sections 5.1 to 5.5 are based on three conference papers published as part of the research on this work [HW17; HW21; WH18].

In Section 5.1 we present the basic intention of the search with its underlying collection and the automated search process. Three different variants of the basic search scenario are introduced.

Section 5.1 Search Scenarios: Basic description of the evaluation scenario

The following Section 5.2 presents two evaluation methodologies and their measures used in the evaluation.

Section 5.2 Evaluation Methods: Presentation of two relevance assessment approaches

Section 5.3 describes the setup of the evaluation, the collection with its characteristics and composition, queries, and any inadequacies. As a preliminary investigation, we also analyze the scoring behavior at the document level.

Section 5.3 Experimental Setup: Setup for the following evaluations and collection characteristics

After presenting the scenarios including our evaluation approach and the specific setup, a detailed evaluation of the presented and selected voting techniques is provided in Section 5.4.

Section 5.4 Experimental Results: Results of the experiments based on the two relevance assessment approaches

In Section 5.5, we relate the performance of the techniques to their systemic behavior. The extent to which voting techniques systematically prefer certain class sizes is examined. In addition, we look at the influence of the length in terms of the requests. Finally, we look at the influence of the number of initial voting documents.

Section 5.5 Systemic Behavior of the Applied Techniques: Behavior of the voting technique related to class sizes, query lengths, and initial voting documents

Following the detailed evaluation of the different aspects of our scenario, we move on and discuss a voting technique in Section 5.6 that is based on the addition of dependent probabilities and is

Section 5.6 Principle of Inclusion and Exclusion: Aggregation according to the principle of adding probabilities

more complex to implement. We evaluate this methodology based on our scenario from previous sections.

Section 5.7 Online Forum Thread Retrieval: Forum search for a relevant thread based on an example from the literature

In the last Section 5.7 of this chapter, we evaluate the newly introduced methods from this thesis on a scenario already introduced in the literature.

5.1 Search Scenarios

The search of appropriate journals as an instance of the search on class level

The basic intention for all introduced search scenarios is to search for journals that potentially contain publications of interest to the user's information needs. In a reinterpreted form, the search scenarios also represent the search for a journal that is most suitable for the user to publish on a requested topic. With respect to this environment, the search for appropriate journals is equivalent to the search for classes that represent them. The articles published in the journals represent voting documents, and each article votes for the journal in which it is published.

Three search scenarios based on article titles, abstracts and full text

The experiments and evaluations in the following sections are based on the AMiner collection. The AMiner collection is a bibliographic collection that contains articles with their title, abstract, and full text [Tan+08]. This leads to three scenarios working with this collection where the search is done over the following articles' components:

- Titles - Only articles' titles are indexed and serve exclusively as search basis.
- Abstracts - In this scenario, only articles' abstracts are searched. The titles, main articles, and references are excluded in this scenario.
- Full Text - Full text articles are indexed (mostly also including the title, the abstract, and also references).

To automate the search and subsequent evaluation process, we removed 10,000 articles from the collection and used their titles as search queries for each of the three scenarios[2]. This is done under the assumption that the title of a removed article and its content represent the thematic domain of its journal. By automatically sending these 10,000 requests to the collection, we can assess existing and newly introduced voting techniques for a large number of requests. Figure 5.1 illustrates the evaluation setup and shows the removal of articles whose titles serve as queries for the search. The index serves with three different variants for the journal search.

2: For our setup only journals containing more than two articles were considered.

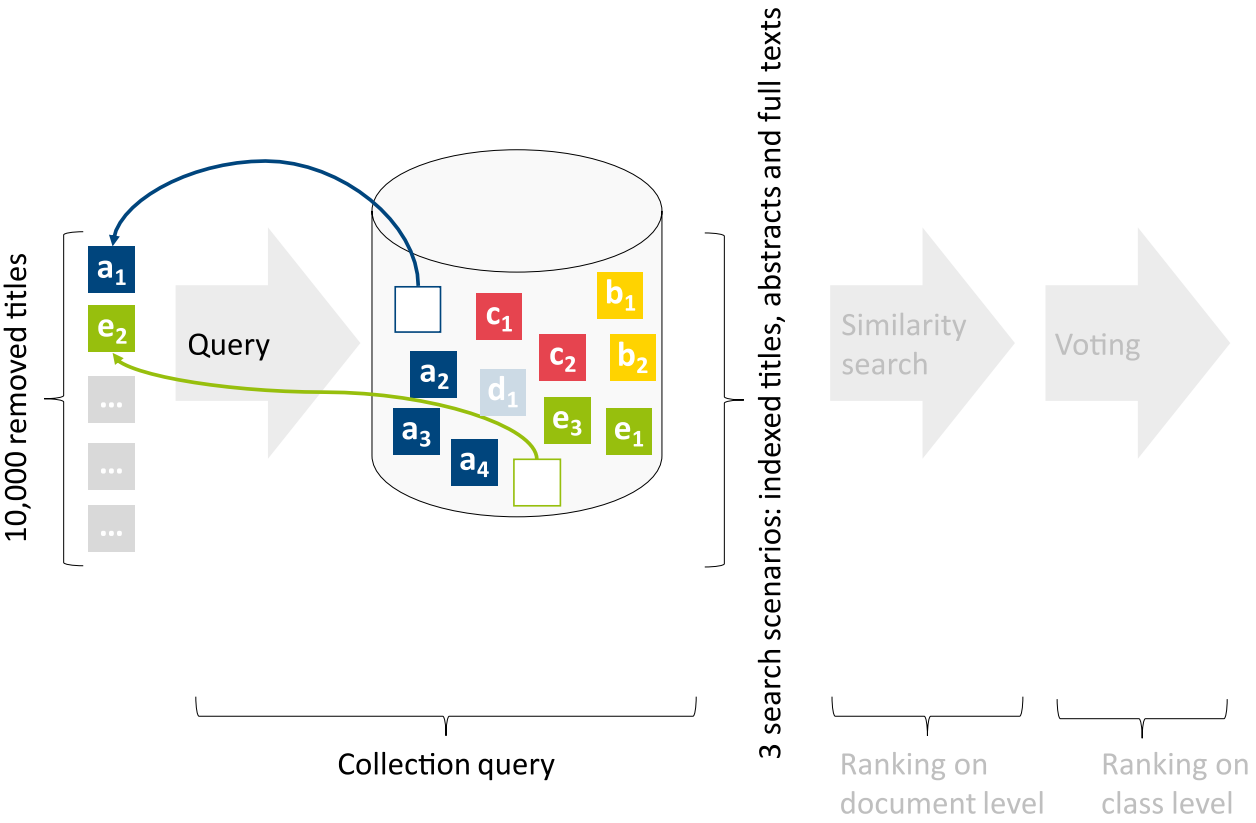

Figure 5.1: The setup for the evaluation is shown based on Figure 1.1 on page 8. Of all the articles originally intended for the search, 10,000 items are randomly excluded from the index and their titles are used as requests. The index itself is configured for three variants for the evaluation scenarios: Only titles are indexed, abstracts of articles, or articles are completely indexed as full texts, including their entire content including references.

5.2 Evaluation Methods

Regarding the expert search task, basically three evaluation strategies can be applied [MO08]: pre-existing ground truth, candidate questionnaires, and supporting evidence. The ground truth aims to identify potential candidates by their membership in research groups. The latter two require manual assessment: candidate surveys state their expertise for certain topics or not, and supporting evidence is gained by judging initially given supporting documents for each candidate.

5.2.1 Two Evaluation Scenarios

Our two evaluation approaches – technically inspired by the idea of [CS02] – use the given relationship between articles and journals in which they have been published, and potentially in addition journal relationships which result from common authors between journals as ground truth data and can be fully automated. This results in two evaluation scenarios illustrated in Figure 5.2 and explained in the next two sections, 5.2.2 and 5.2.3:

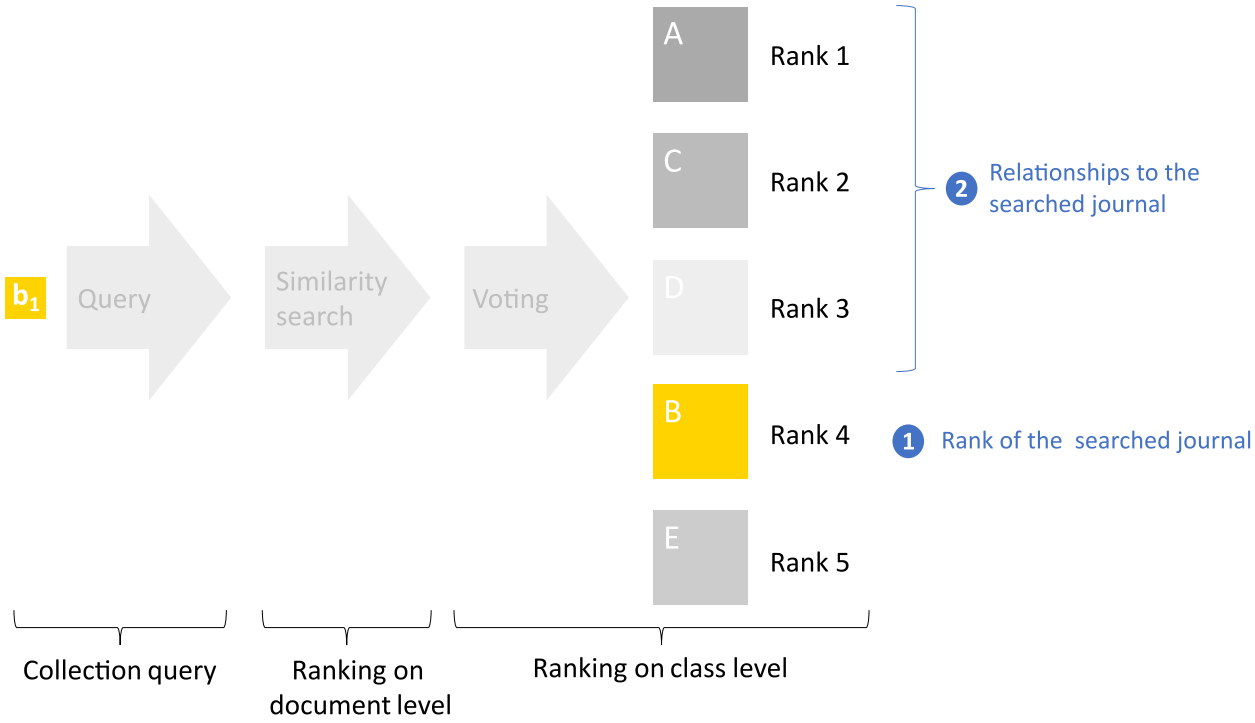

Figure 5.2: Two evaluation scenarios are illustrated using the example of query b_1, taken from the yellow Journal B. Evaluation scenario 1 is based on the rank of the searched journal – the journal the query is taken from – in the overall class ranking and is described in Section 5.2.2. Evaluation scenario 2 examines the question of how the journals in the upper ranks are related to the searched journal on the basis of common authors. Scenario 2 is described in Section 5.2.3.

5.2.2 Ranking of the Searched Journal

1st Evaluation measure: Rank of the journal from which the query was taken from

For a first evaluation measure, our ground truth is based on the information in which journal a request (that is, title of a removed article) has been published. Knowing this, we can measure the position of the associated journal in the ranking of the results and evaluate the applied voting techniques. An advantage of this approach is that we can use a huge number of queries to evaluate our ranking methods. Employing this evaluation technique, we assume that the request as the title of the removed article is a typical and well-representing example of its journal.

5.2.3 Determination of Journal Relationships

2nd Evaluation measure: Grades of relationship between the collection's journals

One problem of the first evaluation measure is that it does not consider the fact that journals can be thematically related. If the journal originally containing the query article is on rank three, the journals on ranks one and two might nevertheless be good matches.

In preparation for this second evaluation measure, we therefore calculated grades of relationship between the collection's journals

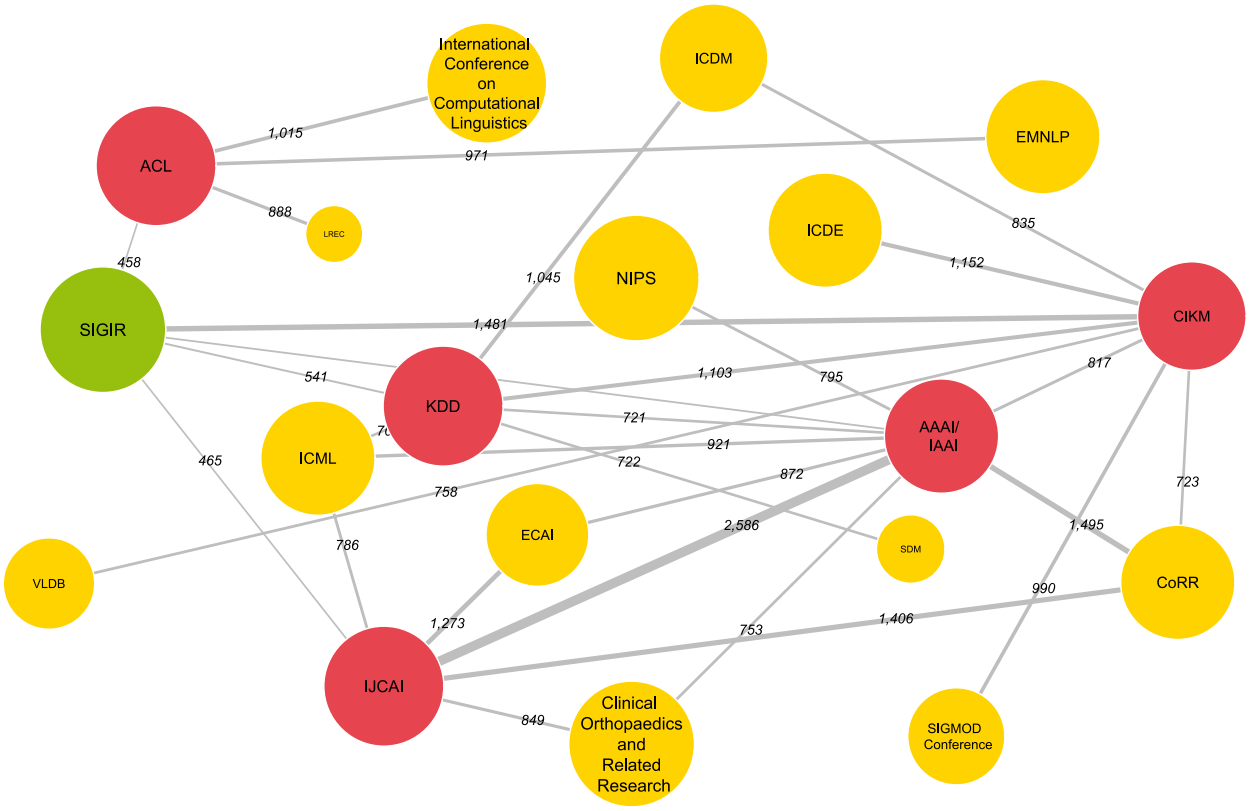

Figure 5.3: 1st- and 2nd-grade relationships for the journal SIGIR displayed in green. Only grades above 450 for the 1st (in red) and grades above 700 for the 2nd grade (filled in yellow) are shown for reasons of clarity. The size of the circles is derived from the number of articles stored in the test collection for each journal. (For an explanation why *Clinical Orthopaedics* is related to SIGIR see Section 5.3.2)

Relationships between journals as thematic overlap or accordance

contained in the experimental setup based on the number of common authors, assuming that this figure corresponds to a thematic accordance or overlap. To calculate this measure, we parsed the entire AMiner collection [19] and extracted those authors who contributed to the journals contained in our setup (see Section 5.3). Since the authorships are defined for each article and each author has their own ID, the relationships can be clearly established on this basis. As a result, by examining all 1,916 journals and their common authors, we gained 484,991 relationships between these journals. 75% of these are based on five or fewer common authors, the most intense relationship is based on 18,176 common authors.

The number of common authors is calculated on the basis of all available journals and their articles throughout the AMiner collection and is not limited to the small excerpt of the collection in this evaluation. We do not normalize the number of common authors at this point, as the number of outliers between large and small journals should be smoothed due to the high number of queries considered.

Following and inspired by the example of expertise graphs [Bal+12],

undirected edges expressing the number of common authors establish the relationship between journals as nodes. Using this information as a second ground truth, we can evaluate the relevance of the journals on all ranks by examining the relationship of these journals to the journal from which our request was taken.

The grade of relationship corresponds to the number of common authors.

In the following, the expression "grade of relationship" is synonymous with the number of common authors between two journals.

Figure 5.3 exemplary shows the relations of the SIGIR-journal, whereby for reasons of clarity only 1st level- and 2nd level-grades of relationship are displayed (each having a grade over 450 and 700).

5.3 Experimental Setup

5.3.1 AMiner Collection

Our experiments are based on the dump of 154, 771, 162 scientific papers offered by AMiner[3]. In this collection, publication data from online databases including the dblp bibliography, the ACM Digital library, CiteSeer, and others are merged [Tan+08].

3: The complete data sets are offered at https://www.aminer.org/oag2019. The number of papers offered has increased since the download for this thesis.

Table 5.1: Collection structure and voting setup for the adjusted AMiner collection - only articles whose title, abstract and full text were available were considered for the evaluations.

Adjusted AMiner Collection	
#Journals	1,916
#Relationships	484,991
#Articles	115,292
#Queries	10,000

For our experiments, we extracted and obtained 125,292 articles which bring along title information, their abstract, and also a downloadable full text PDF. The size of the test collection compared to the total collection size is small due to the fact that at the time of retrieval in the year 2019 by far not all articles were completely available with abstract and their complete PDF document. Furthermore, we only considered articles from journals that contain three or more articles. The article collection in our setup corresponds to 1,916 journals as shown in Table 5.1. According to Figure 5.1 we randomly removed 10,000 articles and used their titles to search the remaining collection. In this way, we obtained 10,000 test queries for which we know the presumed best-fitting journal—the one in which the article was published. About 50% (997 journals) of the searched collection's journals include 11 or fewer articles, about 75%

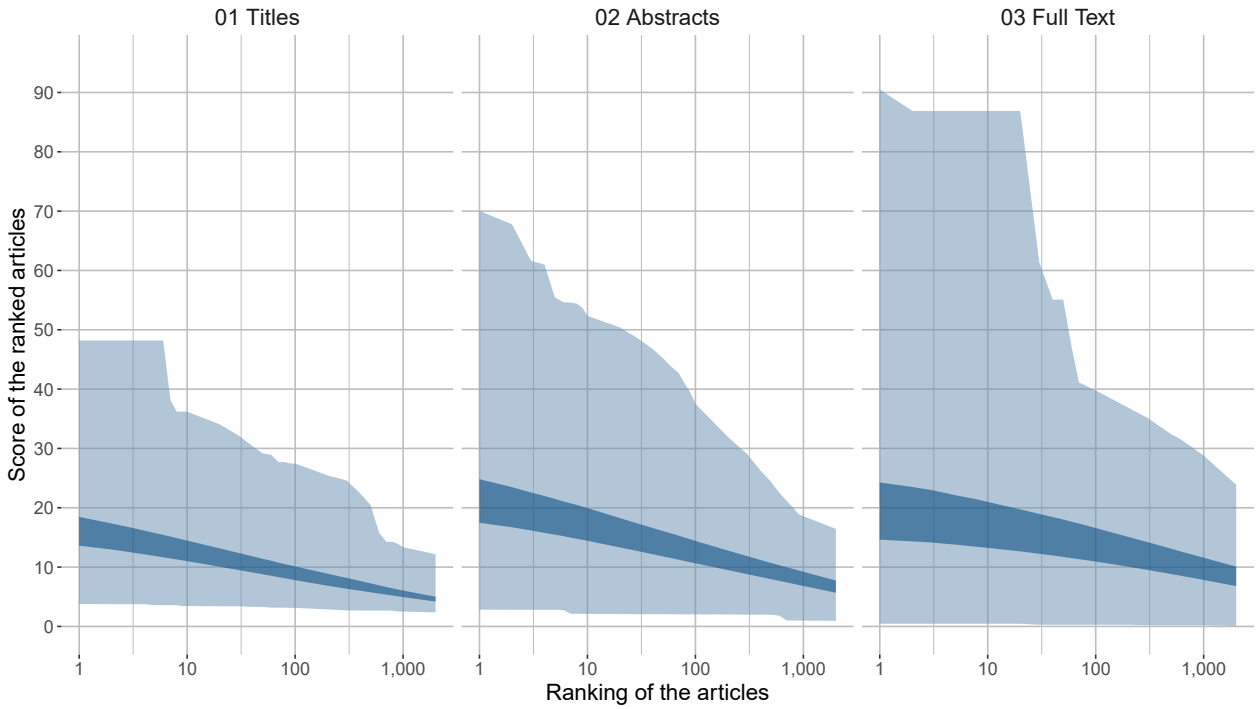

Figure 5.4: Score distributions for the 10,000 queries plotted in quartiles of the article rankings for three search scenarios. The lower boundary to the dark ribbon describes the distribution of the 1st quartile, the dark ribbon holds 50% of the ranked documents, and the upper area holds the upper 25% of scores on each rank.

(1.447 journals) have 29 or fewer articles, while the largest journal holds 4,483 articles.[4]

4: The extracted journals, articles and queries can be found at: https://doi.org/10.48564/unibafd-3xt8e-zg557

5.3.2 Properties of the Collection

Slight inaccuracies in the base material are neglected.

While working with the AMiner collection, we discovered some inaccuracies that are caused by multiple assignments and ambiguities. Probably caused by ambiguities, articles are assigned to the journal *Clinical Orthopaedics and Related Research* (CORR) instead of *Computing Research Repository* (CoRR). In addition, some articles are mentioned twice for the same journal, though having a different article ID.

Since these inaccuracies affect only very small numbers of articles, we neglect this effect in our experiments.

5.3.3 Scoring Properties of Voting Articles

Search on document level yields a flat gradient of the document scores.

Articles that vote for their journals are ranked according to the score computed by applying the BM25 formula. Since numerous voting techniques are based on the score values of the voting documents, we will first examine the progression of these score values over the 10,000 queries for a better insight.

Our first empirical measures for the voting articles' scores evaluate ranks from 1 to 9,000. For the first nine ranks, we look at every rank, then every 10th, every 100th and finally every 1,000th rank.

Figure 5.4 shows the score distributions for the ranked articles up to rank 9,000. The titles scenario (01) and the full text scenario (03) show plateaus for the highest score values in the top ranks. Regarding the titles scenario, this is based on the fact that one article has been published multiple times in one or more journals, resulting in a constant high score[5]. On the level on abstract search this is not noticeable because this article is not the high-scorer in this scenario.

5: The title of the article assigned multiple times is "Scale-Out Beyond Map-Reduce CISL Team Members"

The search over full text articles shows a plateau because of 9 articles which are gathered together in one PDF document as a proceedings overview. If a removed title is executed as a query, the other articles in the summarized document implicitly vote for the correct journal, as the requested query title is still available in their PDF texts.

The majority of the scores descend rather flat, the average score on rank 100 is still 70% of the top-ranked article score.

5.4 Experimental Results

In this section, we consider three effectiveness measures for assessing the quality of the class ranking. We start with two measures considering the rank of the original journal from which the query was taken in Sections 5.4.1 and 5.4.2. The third measure considers the relationships between the journals to give a broader picture and is discussed in 5.4.3.

5.4.1 Using MRR as Measure

MRR of the journal from which the query was taken

Table 5.2 shows the results using the mean reciprocal rank (MRR) as a measure of the effectiveness of the aggregation techniques applied in all scenarios. The rank of the first relevant answer in our case is the rank of the journal from which the query was taken and labeled $rank_{searched\ journal}$ in the equation 5.1. If the journal searched for is not found in the results, we calculate its rank as a fictitious last rank and set it to the number of journals in the index as $rank_{searched\ journal} = 1,916$.

In our setup, the MRR is computed with $|q| = 10,000$, which corresponds to the number of removed article titles that serve as queries against the index.

$$MRR = \frac{1}{|q|} \sum_{i=1}^{|q|} \frac{1}{rank_{searched\ journal\ (i)}} \tag{5.1}$$

Green cells in Table 5.2 including high values indicate good performance in yielding the searched journal early in the results, whereas shades of light green over white to red show worse effectiveness. Color gradations refer to the values within a single scenario (column) and are not applied across all scenarios.

The applied techniques are shown in rows and start with variants of CombSUM, whose number of top voting candidates considered per class becomes increasingly limited with descending rows. The next two variants of expCombSUM are followed by new variants of CombSUM with a valuation of RR^x. The results of the techniques using the squared score follow next. The ranking-based and exponential variants RR^x are surrounded by Votes and CombMAX, which are approximated according to the exponent set to RR^x.

Best results are observed by taking into account or emphasizing upper ranks at class level. An adjustment of RR^x also provides competitive results.

Performance of the applied techniques Techniques considering or emphasizing the first ranks in the scope of the class, such as CombSUM TOP 5, CombSUM *RR*, sqCombSUM *RR*, and the

Table 5.2: MRR for the search over titles, abstracts and full text having 3,000 initial voting articles

Scenario ▹ / Technique ▽	Titles	Abstracts	Full Text
CombSUM	0.27	0.29	0.30
CombSUM TOP 100	0.28	0.30	0.31
CombSUM TOP 50	0.31	0.33	0.34
CombSUM TOP 5	0.34	0.36	0.36
expCombSUM	0.28	0.28	0.33
expCombMNZ	0.32	0.33	0.35
CombSUM $RR^{\frac{1}{2}}$	0.29	0.31	0.31
CombSUM RR	0.34	0.35	0.35
CombSUM RR^2	0.32	0.33	0.35
sqCombMNZ	0.27	0.29	0.30
sqCombSUM	0.30	0.31	0.31
sqCombSUM $RR^{\frac{1}{2}}$	0.33	0.34	0.34
sqCombSUM RR	0.34	0.36	0.36
sqCombSUM RR^2	0.30	0.32	0.35
Votes	0.25	0.27	0.28
$RR^{\frac{1}{2}}$	0.31	0.33	0.33
$RR^{\frac{3}{4}}$	0.34	0.35	0.36
RR	0.31	0.33	0.34
$RR^{\frac{3}{2}}$	0.27	0.28	0.29
RR^2	0.26	0.27	0.28
CombMAX	0.25	0.27	0.27

purely ranking-based technique $RR^{\frac{3}{4}}$ achieve the best rankings on average. These techniques perform better in all three scenarios.

CombSUM as an addition of all voting scores per class does not yield good results, a gradual limitation of the scores to be considered per class up to CombSUM TOP 5 improves performance for all scenarios. The exponential variant expCombSUM also does not perform well. As described in Section 2.2.4, document scores serve as an exponent of the Euler number e, leading to a high influence of the global top scoring documents. Due to the large expansion of the score spectrum, results can arise here that tend toward CombMAX. Multiplication of the results of expCombSUM by the number of voters per class yields improvements in the case of expCombMNZ.

Worst results are based either solely on the pure number of voters or the best voter at class level.

Votes and CombMAX yield the worst results in all scenarios, so we can state that techniques that only consider the number of voters or the best voter per class are not suitable to determine the journal searched.

Performance of the scenarios Regarding the three scenarios that are displayed in the columns of Table 5.2, the search for abstracts yields better MRR values than the search over titles throughout all techniques except expCombSUM. These summaries and introductions to the topics of the articles result in more information to obtain better search results.

More information through abstracts improves the results for the mean reciprocal rank.

While the information content of the abstracts can increase the effectiveness of search, the search over full text documents provides improvements for all techniques to a limited extent. The best results from the search over abstracts are not exceeded when searching using full texts: Very well-performing techniques from the abstracts scenario, such as CombSUM TOP 5, CombSUM *RR*, and sqCombSUM *RR* yield the same results when searching via full texts. The results of expCombSUM, expCombMNZ, and sqCombSUM RR^2 benefit more from the third scenario. Results of other techniques improve only slightly or remain the same. In summary, the search for complete texts provides only limited improvements compared to the search over abstracts. The cause of this limited improvement can be noise caused by indexing complete documents, including references, as well as artifacts that arise during extraction, such as formulas and expressions. It should be noted that the lowest and highest MRR values remain unchanged in both scenarios.

Results for the search over full text documents are only partially better

5.4.2 Rank Distribution of the Original Journal

This section further considers the rank of the searched journal, but looks at its distribution in the results over the 10,000 requests for each applied technique. The evaluation results are shown in Table 5.3.

The values for the 1st quartile, the median and the 3rd quartile represent the rank r for which 25%, 50%, respectively 75% of the 10,000 queries rank the journal we are looking for at this rank r or better. Queries for which the desired journal does not have a voting article within the considered part of $R(q)$ are counted as the rank 1,916 which corresponds to the number of journals in the index. As an example, the value 52 for CombSUM in the titles scenario for the 3rd quartile means that for 75% of the queries the journal searched for was ranked 52 or better among the 1,916 journals in the results.

Green values of the cells from saturated to light green indicate results having the best or rather good ranking properties within a quartile over all techniques, whereas worse results are shown in shadings from white to read corresponding to their results. This time, the coloring in the table is based on a quartile level and goes across all scenarios. This cross-scenario coloring is intended to highlight the differences in performance between the scenarios, in

Table 5.3: Empirical quartiles for the rank of the searched journal for search over titles, abstracts, and full text articles; the cells are relatively colored per quartile, across scenarios.

Scenario ▹	**Titles**			**Abstracts**			**Full Text**		
Technique ▽	1st q.	med.	3rd q.	1st q.	med.	3rd q.	1st q.	med.	3rd q.
CombSUM	3	11	52	2	9	45	2	9	42
CombSUM TOP 100	2	11	52	2	9	45	2	9	42
CombSUM TOP 50	2	11	52	2	9	45	2	7	42
CombSUM TOP 5	2	6	38	2	6	30	2	5	31
expCombSUM	3	9	45	2	8	36	2	7	36
expCombMNZ	2	7	41	2	6	32	2	6	33
CombSUM $RR^{\frac{1}{2}}$	2	9	47	2	8	40	2	8	39
CombSUM RR	2	7	42	2	6	34	2	6	35
CombSUM RR^2	2	7	40	2	6	31	2	6	32
sqCombMNZ	3	11	51	2	9	44	2	8	41
sqCombSUM	2	9	45	2	8	38	2	8	38
sqCombSUM $RR^{\frac{1}{2}}$	2	7	40	2	6	33	2	6	35
sqCombSUM RR	2	6	38	2	6	31	2	6	32
sqCombSUM RR^2	2	9	45	2	7	34	2	6	33
Votes	3	13	63	3	11	52	2	10	47
$RR^{\frac{1}{2}}$	2	8	44	2	7	35	2	7	33
$RR^{\frac{3}{4}}$	2	7	41	2	6	32	2	6	32
RR	2	8	41	2	7	33	2	6	33
$RR^{\frac{3}{2}}$	3	10	46	2	8	36	2	8	36
RR^2	3	11	49	3	9	38	2	9	40
CombMAX	3	12	56	3	10	43	3	10	45

particular their median and 3rd quartile.

1st quartile: Only view changes between the scenarios

Differences between the scenarios Apart from a few exceptions, no changes in the scenarios can be identified within the 1st quartile. Only poor performers from the previous evaluation, such as CombSUM, expCombSUM, sqCombMNZ, or Votes, can benefit from searching the abstracts or the full text.

Median: Improvements for the results of search over abstracts and partial slight improvements for full text search results

The title-based values shown in the median improve for almost all techniques when searching over abstracts. Only the best two results yielded by CombSUM TOP 5 and sqCombSUM *RR* for the titles scenario remain the same in the abstracts scenario. In the full text scenario, there are slight improvements for the median values in a few cases by one rank. The best result is achieved in the median by CombSUM TOP 5 having a value of 5 - 50% of all requests yield the journal searched for at rank 5 or better when applying this technique in the full text scenario.

3rd quartile: Improvements in higher ranges for the results abstract search, diffuse but mostly not significant changes in the full text search.

The 3rd quartile shows differences in higher ranges between scenarios. Switching from search over titles to abstracts yields improvements from at least 7 ranks to a maximum of 13 ranks in the case of CombMAX. Based on the consistently positive changes in the

results of the abstract search, the search using full text does not provide a clear trend. In particular, well-performing techniques deteriorate slightly in the third quartile, while techniques such as CombSUM and its variants, which include many voting candidates per class, but also Votes in the third quartile of the full text scenario, benefit.

Observed behavior of the scenarios tend to correspond to those from the MRR evaluation in 5.4.1

In summary, it can be stated that for all voting techniques shown in Table 5.3 the search over titles performs the worst. Searching over abstracts generally yields improved results, while performance increases can only be observed to a limited extent when searching in the full text scenario, corresponding to our observation from the MRR analysis in Section 5.4.1. Performance differences in the scenarios are obvious for the median and the 3rd quartile; the first quartile often has similar performance results. This is consistent with the intuition that more detailed texts should improve the recall, but not necessarily the precision of the front ranks.

Due to related measures, the performance of the techniques tends towards the results of the previous evaluation.

Performance of the applied techniques The best-performing techniques correspond mostly to those of the previous evaluation. CombSUM TOP 5, CombSUM RR^x, sqCombSUM RR^x (both having x set to 1 or 2) and, as a ranking-based technique, $RR^{\frac{3}{4}}$ produce the best results. The technique expCombMNZ again outperforms expCombSUM in the results and is also one of the best performers in this evaluation in addition to the techniques mentioned above. Votes and CombMAX, only based on the number of voting articles or the 1st ranked article of a journal, yield worse results. The results of the 1st quartile for CombMAX show that this technique cannot even benefit from the higher amount of information provided by abstracts or full texts. The results for variants of CombSUM considering the TOP n candidates confirm that only a few top-ranked articles should be included per journal, in our case the top 5 articles.

5.4.3 Journal Relationships as Measure for Relevance

Evaluating the relevance of other high ranked journals

This evaluation expands on previous ones from Sections 5.4.1 and 5.4.2. While the previous two studies consider the ranking of the journal searched, in this section the relevance of other top-ranked journals is analyzed. If, for example, the journal we are looking for only appears on rank 4, are the results in the first ranks really not relevant, or do these ranks also contain relevant results? This next evaluation technique attempts to consider this question.

Relevance of the results increases with higher relationship to the journal searched for.

We assume that a large number of common authors indicates a similar or related topic for conferences. Hence, in this evaluation,

Table 5.4: Number of authors in common with the desired journal for the journals at position 1. The relative coloring refers to each quartile across all scenarios. Green-colored cells indicate good performance within a quartile with high grades of relationship; descending performance is indicated by white to red coloring.

Scenario ▸	**Titles**			**Abstracts**			**Full Text**		
Technique ▽	1st q.	med.	3rd q.	1st q.	med.	3rd q.	1st q.	med.	3rd q.
CombSUM	627	108	22	697	122	24	697	117	25
CombSUM TOP 100	728	125	24	883	141	26	849	134	26
CombSUM TOP 50	919	145	27	971	156	29	944	144	26
CombSUM TOP 5	944	168	27	971	169	30	971	172	27
expCombSUM	499	78	11	499	80	13	737	124	20
expCombMNZ	766	143	23	769	142	24	944	148	27
CombSUM $RR^{\frac{1}{2}}$	737	128	26	846	140	27	802	127	27
CombSUM RR	1,015	175	31	1,015	176	32	1,004	148	30
CombSUM RR^2	745	142	21	795	145	25	971	170	27
sqCombMNZ	652	112	23	704	125	25	704	117	25
sqCombSUM	728	125	25	802	135	26	778	127	26
sqCombSUM $RR^{\frac{1}{2}}$	971	162	30	1,005	166	31	971	142	29
sqCombSUM RR	971	173	30	1,020	182	33	1,015	173	30
sqCombSUM RR^2	655	113	17	697	125	20	892	155	24
Votes	520	95	21	652	114	23	649	113	23
$RR^{\frac{1}{2}}$	795	135	26	919	145	28	944	142	28
$RR^{\frac{3}{4}}$	971	162	27	1,015	173	29	1,015	169	30
RR	656	93	12	730	115	16	795	117	16
$RR^{\frac{3}{2}}$	416	65	9	458	72	11	471	72	10
RR^2	415	65	9	445	71	11	465	71	10
CombMAX	415	65	9	445	71	11	459	71	10

we investigate how many common authors exist between the top-ranked journals and the desired journal, the one from which the query article was taken. If the desired journal is on the rank that we examine, the number of authors for this journal is used. Otherwise, the number of common authors. The considerations presented in Section 5.2.3 and Figure 5.3 suggest that this might be reasonable, especially when we can consider the results over 10,000 queries: In this case, outliers and problems with the relationships between big and small journals should be smoothed by the big number of queries considered.

Table 5.4 shows the grades of relationship (number of authors in common with the desired journal as described in 5.2.3) for each technique applied in the scenarios and the journal with the highest rank. On the vertical axis, all applied techniques are listed. On the horizontal axis, for each scenario, the quartiles are shown. The 1st quartile shows the minimum grade of relationship between the searched journal for 25% of the queries with the 1st ranked journal, the median indicates the grade for 50% of the top ranked journals. 75% of the journals have a grade equal to or better (higher) than the

figure shown in the 3rd quartile. As an example, the number 944 for CombSUM TOP 5 in the 1st quartile for title-based search means that for 2,500 of the 10,000 queries, the journal at rank 1 has 944 or more authors in common with the original and searched journal.

Performance of the scenarios For all applied voting techniques and in all quartiles, the search over abstracts performs equal and mostly better than the search for titles, except expCombMNZ which has a negligible deterioration in the median. The best results or maximum values of the three quartiles from the abstract-based search are not exceeded in the full text scenario. Here, the results of CombSUM *RR* and sqCombSUM *RR* deteriorate slightly in all three quartiles. CombSUM RR^2 and sqCombSUM RR^2 yield rather worse results in the titles and abstracts scenario, but can gain considerably in the full text search. Similarly, expCombSUM, which also yields rather worse grades of relationship for titles and abstracts, improves significantly.

The search on abstracts provides improved results. Results of the top performers from the abstract search are not exceeded in the titles scenario.

Performance of the applied techniques In accordance with tables 5.2 and 5.3, in Table 5.4 voting techniques that emphasize the first voting documents yield the best results. Like in the preceding evaluations, the techniques CombSUM *RR*, sqCombSUM *RR*, and $RR^{\frac{3}{4}}$ show the best performing results throughout all scenarios. CombSUM TOP 5 shows slightly worse results, which are still among the best. These variants seem to find a good trade-off between considering multiple articles per journal and damping the effect of additional articles. The results of the RR^x techniques are very sensitive to changes in the exponent x from $RR^{\frac{3}{4}}$ up to 2 and approximate those of CombMAX, which shows the worst performance in this evaluation. The technique expCombSUM performs worse for search on titles and abstracts, but shows a rapidly increasing quality for search over full texts. This effect can also be observed in MRR-based evaluation, Section 5.4.1.

Evaluation based on the relationships shows comparable results in terms of performance for the voting techniques.

Viewing the rankings As part of this evaluation, not only the first rank results as shown in Table 5.4 were evaluated, but also the following ranks. Table 5.5 shows an overview of the first five positions when applying CombSUM TOP 5 and CombSUM *RR* as examples of good performing techniques and Votes as a bad performer for the grades of relationship. The first row with position 1 corresponds to the results of the previous Table 5.4 for the three voting techniques, the following rows show the grades of relationship of the subsequent ranks up to position 5.

For all techniques the grades of relationship decrease with lower ranks.

Table 5.5: Grades of relationship for the first five positions based on the example of CombSUM TOP 5, CombSUM *RR* and Votes. The coloring of the cells is applied across scenarios and rankings for each quartile.

Scenario ▹		**Titles**			**Abstracts**			**Full Text**	
Position ▽	1st q.	median	3rd q.	1st q.	median	3rd q.	1st q.	median	3rd q.
CombSUM TOP 5									
1	944	168	27	971	169	30	971	172	27
2	465	105	18	472	109	21	445	100	19
3	340	79	15	372	84	17	328	80	16
4	285	68	13	289	71	14	262	66	13
5	224	56	12	250	64	13	235	56	12
CombSUM *RR*									
1	1,015	175	31	1,015	176	32	1,004	148	30
2	505	116	22	541	124	24	510	112	21
3	365	83	18	387	93	20	364	86	17
4	284	71	16	306	79	17	284	72	15
5	246	64	13	252	67	15	244	65	14
Votes									
1	520	95	21	652	114	23	649	113	23
2	364	81	17	414	94	18	458	93	18
3	290	69	15	349	79	17	332	74	15
4	241	59	13	284	71	15	264	65	14
5	225	52	11	257	64	14	236	60	13

Generally, Table 5.5 shows that the application of the grade of relationships determined represents a meaningful measure of effectiveness and a quality indicator. For all techniques, the grades of relationship within each quartile decrease with descending position. In position 1, the grade of relationship for the well-performing techniques CombSUM TOP 5 and CombSUM *RR* is still much higher than for Votes as an underperforming technique. With descending position, at the latest with position 5, these differences become smaller, especially when comparing CombSUM TOP 5 and Votes.

5.4.4 Summary of the Results

Best results for techniques that emphasize the top rankings at class level

Best performing techniques The three evaluations of Sections 5.4.1, 5.2.2 and 5.4.3 result in CombSUM TOP 5, CombSUM *RR* and sqCombSUM *RR* as very good performing techniques. The basic principle of score processing or valuation at the class level proves its effectiveness here – be it by taking into account the top ranked documents per class or by the revaluation of scores using a damping function.

$RR^{\frac{3}{4}}$ as a ranking-based technique is also a provider of very good results in the evaluations. The technique RR^{x}, which approximates

between Votes and CombMAX using values of x, finds the best parameterization with $x = \frac{3}{4}$ in the evaluations. It can be inferred that this method, which considers only the ranking of documents rather than their scores, is effective in this context with its collection-specific parameterization, and that the value for x may not easily be applicable to other scenarios for achieving good results.

Best performing scenario Although some techniques can increase their results when searching over full texts, the best resulting MRR value from searching via abstracts cannot be surpassed. This is probably due to noise associated with extracting PDF documents and including references from each original document. Presumably the document ranking in the full text scenario is only improved in its recall, which can be assumed from the example of the only slightly changing values of CombMAX when switching from abstracts to documents. The techniques expCombSUM, CombSUM RR^2 and sqCombSUM RR^2 can significantly improve their results in the full text scenario of the evaluations. The latter two are among the good performing techniques predominantly in this scenario.

Searching over abstracts increases the quality, the full text scenario only provides improved results to a limited extent.

Grades of relationship In addition to the MRR measure and the rank of the searched journal, the class rankings are also evaluated using the grades of relationship. The trends of the previous evaluations are confirmed by the results. The measure becomes problematic when looking at the lower ranks, as the differences between the results of the techniques are no longer clearly apparent.

The grades of relationship measure is not indicative for lower ranks

5.5 Systemic Behavior of the Applied Techniques

In this section, we look at the behavior of voting techniques related to collection-based class structures, the length of requests in terms, and finally related to the number of initial voting documents.

The following Section 5.5.1 first examines the systemic behavior of the voting techniques with regard to different class sizes. We examine the performance development with respect to the lengths of the requests in terms in Section 5.5.2. In Section 5.5.3, we investigate to what extent the change in the number of initial voting documents influences the voting techniques and their performance.

5.5.1 Impact of Class Sizes on Voting Techniques

Evaluating the impact of journal size differences on the voting process for each technique

The adjusted AMiner collection of this setup is based on journals that exhibit a wide variation in their number of articles. One assumption could be that a technique that adds up the scores over many articles tends to prefer larger journals that contribute more articles. These big sized journals then tend to have a higher probability of yielding good performance results like having many authors in common with the journal in question. In this section, we examine the composition of the journal collection and evaluate the impact of journal sizes on voting techniques and their results. As a result, we get to know if and which of the applied techniques systematically favor journals dependent on their number of articles.

5.5.1.1 Setup for the Investigation

10 bins with an approximate capacity of $\frac{1}{10}$ of the articles in the search index

For the evaluation setup we created ten bins, each having a capacity of $\frac{1}{10}$ of the collection's article count. By sorting the collection's journals according to their article count in ascending order, each bin is filled with journals until the number of articles in these journals reaches the capacity of the bin. This leads to the bin structure shown in Table 5.6.

Ascending sorting of the journals according to their size and subsequent, step-by-step filling of the 10 bins with corresponding articles

Table 5.6: Structure of the bins (journals by size): Slight variations regarding the number of articles arise due to the fact that journals have to be assigned to one bin as a whole.

Bin	# Articles	# Journals	Bin	# Articles	# Journals
1	11,515	1,338	6	11,549	20
2	11,538	314	7	11,315	13
3	11,479	122	8	10,935	8
4	11,477	60	9	10,201	5
5	11,589	32	10	13,694	4
			Overall	**115,292**	**1,916**

Every bin is attempted to hold the same number of articles, restricted by journals with their sizes which have to be assigned to one bin as a whole. The overall result reflects the base data of our collection, 115,292 articles (125,292 minus the 10,000 randomly chosen query articles) in 1,916 journals.

5.5.1.2 Baseline: Origin of the Requested Journals

Figure 5.5 shows for the various techniques the number of rank 1 results originating from the respective bins. The faceted diagrams show the ascending sorted bins on the horizontal and the number of journals included in each bin on rank 1 on the vertical axis. As far as possible, all diagrams reflect logical groupings of voting techniques. In a non-biased or ideal case, having the desired journal on rank 1 for each query, each distribution would represent the given baseline which reflects the number of queries taken from each bin.

5.5.1.3 Systematic Behavior of the Applied Techniques

Techniques that focus very strongly on the first ranks approximate the baseline.

Some techniques approximate the baseline. In particular, these are techniques that emphasize the top voting candidate(s) per class, such as CombMAX and sqCombSUM RR^2, or techniques that extremely attenuate or amplify the initial document scores, such as CombSUM RR^2 or expCombSUM.

More quantitative oriented techniques clearly prefer the large classes for the first rank.

Other techniques like Votes or CombSUM clearly favor large journals.

Both extremes do not perform well with respect to the measures considered in the previous evaluations. This can be seen considering the line colors in Figure 5.5. Here, green saturated lines represent good performance going over light green, white and red lines to bad performing techniques. For each category, the values of the bin number 5 are relatively close, whereas the spread grows for the bins on the left and right sides.

5.5.1.4 Correlation of the Bin Distributions and the Performances

High performers from the previous evaluations neglect small journals and prefer large ones to a certain extent.

The best performing techniques such as CombSUM TOP 5, CombSUM RR, sqCombSUM RR, or $RR^{\frac{3}{4}}$ from previous evaluations show a distribution over the bins by discriminating small journals and preferring large journals in a more or less moderate way. Regarding bin 1, they return less than $\frac{1}{10}$ of the requested journals on the top rank. This lack is compensated by returning more top 1 ranks resulting out of the higher bins.

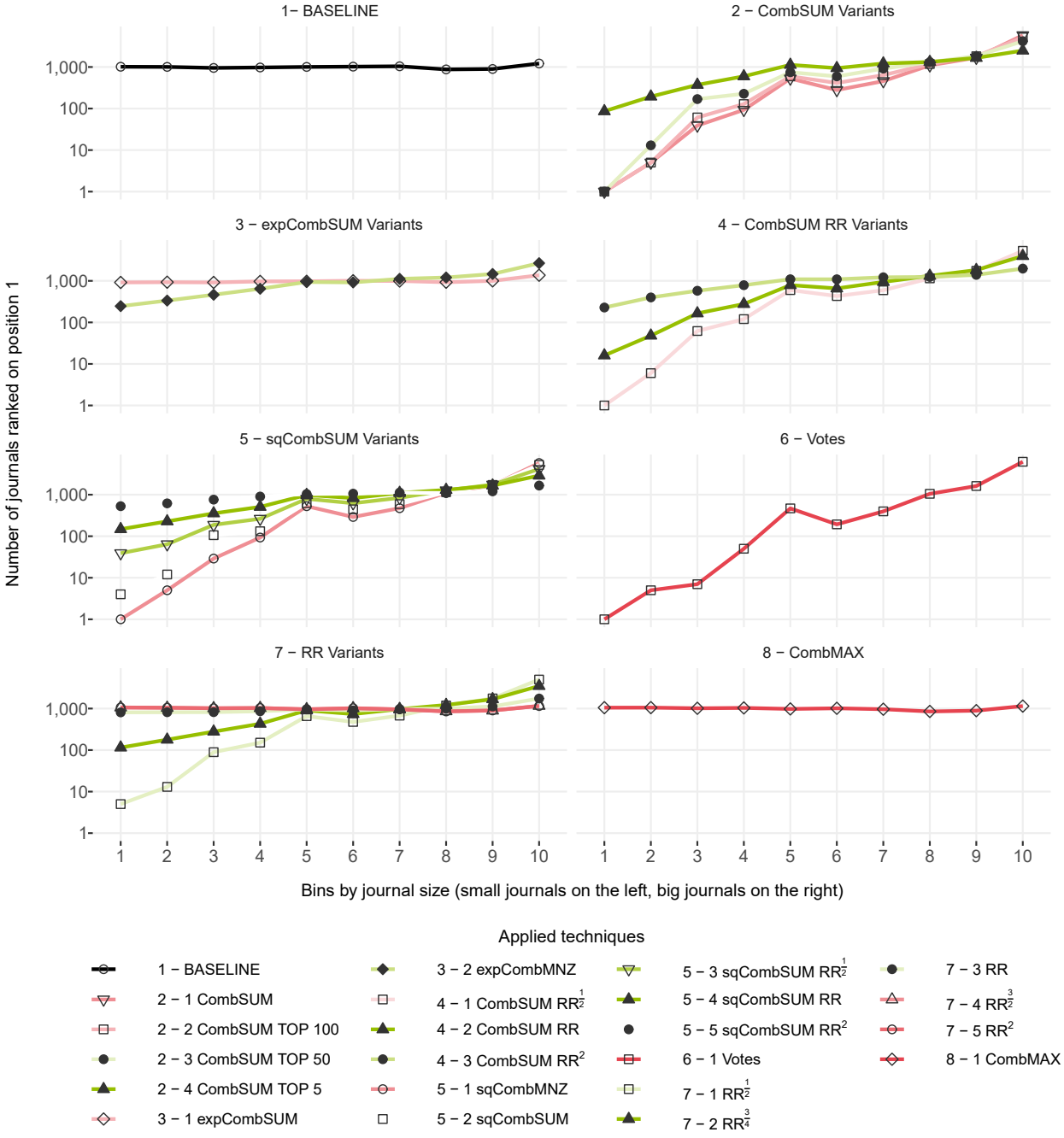

Figure 5.5: Top 1 ranked journals per bin for all applied voting techniques in the search over titles scenario. BASELINE represents the origin of the 10,000 requested articles. The colors of the other lines correspond to the values from the MRR-based evaluation shown in Table 5.2 on page 74; performance increases from dark red moving over white and light green to the best performing techniques in saturated green.

Other techniques such as CombSUM, CombSUM TOP 100 and sqCombSUM $RR^{\frac{1}{2}}$ whose MRR performances are lower show these tendencies in a pronounced form. For bin 1 they return less than 10 journals in rank 1, for high bins, their number of top 1 results exceed those of the techniques mentioned before.

5.5.1.5 Summarizing the Influence of Class Sizes

Techniques unaffected by class sizes The CombMAX, expCombSUM and RR^x techniques with higher exponents are close to the baseline. This means that the size structure of returned journals at position 1 corresponds to the size structure of journals whose articles were extracted. These techniques do not yield good results in the previous evaluations in this chapter. They have a strong connection to document ranking in that they map it rather strictly to a class ranking. An aggregation of the voting documents per class does not occur at all or only with little effect. Although the curves show an ideal, independent shape with regard to differences in class size, they provide only partially relevant results in the evaluations. The curve also shows that the equal distribution of the removed queries across the bins in terms of quantity is also evenly distributed in thematic terms, as the curves of the rank-oriented voting techniques are close to the baseline.

Techniques strongly affected by class sizes Techniques that directly or indirectly include the number of voting documents, such as Votes, RR^x with very small values for x and CombSUM clearly favor large journals and deliver for BIN 1 among 10 journals on rank 1. These techniques also did not deliver good results in the previous evaluations from this chapter.

Techniques slightly affected by class sizes The best-performing techniques from previous evaluations in this chapter are shown in Figure 5.5 in saturated green and can also be recognized by the icon "▲". These include CombSUM TOP 5, CombSUM *RR*, sqCombSUM *RR* and $RR^{\frac{3}{4}}$. They all have a roughly similar curve which shows that large journals are favored and smaller journals from the left bins are disadvantaged. This can be observed in a more extreme form with CombSUM *RR* yielding fewer journals from bin 1 than the other best performing techniques.

In the AMiner collection scenario, it seems advantageous to slightly neglect small journals and slightly favor large ones. With dampening at class level, it is possible with these techniques to reconcile the results of journals with few and many potential voting documents, despite the different and considerable journal sizes.

5.5.2 Influence of the Query Length

This section examines how the results from the evaluations of the voting techniques in this chapter relate to the length of the queries in number of terms. The results of each technique evaluated are set in relation to the number of terms in the queries.

Quality of the document ranking as a dependency of *query clarity* and its informative content

The query creates a document ranking, which is used to calculate the class ranking and thus determines its quality. There are various approaches in research to estimate the expected quality of a document ranking based on the properties of the query, even in advance. In their work "Query performance prediction", He and Ounis incorporate the *query clarity*, which is inversely proportional to *query ambiguity* [HO06]. Based on the work of [CZC02], the simplified query clarity score is calculated as the sum of the Kullback-Leibler divergence of the query model from the collection model. Furthermore, the paper proposes and evaluates definitions for calculating the distribution of the informative amount based on *idf* -properties of the terms from the query. One result of the work of He and Ounis is that the number of terms in a query on its own is not decisive for its performance. In this evaluation, however, we assume that with the high number of 10,000 queries, the informative content and clarity increase on average with the length.

The quality of the results increases for all techniques as the number of query terms increases.

In the context of this evaluation, we look at the statistics on the number of terms of the 10, 000 queries after eliminating the stop words. The distribution of the query length in terms is shown in Figure 5.6. The most frequent query length is six terms, with a number of 1, 852 queries. Since the selection of the titles and thus the queries are randomly generated, the distribution corresponds proportionally approximately to the term distribution of the indexed titles or the corresponding search scenario.

Figure 5.7 shows the results for the MRR measure corresponding to the evaluation of Section 5.4.1 on page 73 for the different query length classes. The techniques are summarized here in logical groups in one diagram each; the vertical axis shows the mean reciprocal rank for the number of query terms indicated on the horizontal axis.

The general trend is that the quality of the results increases as the number of query terms increases. All techniques benefit equally from the higher content of the information in longer queries. The results improve mostly in parallel within a group, but also between groups. From a query length of 9 terms, the performance improvement stagnates and even decreases slightly.

This evaluation also shows the dependence of the techniques on the quality of the document ranking (see also Section 3.1.2 on page 45), which improves due to the higher information content.

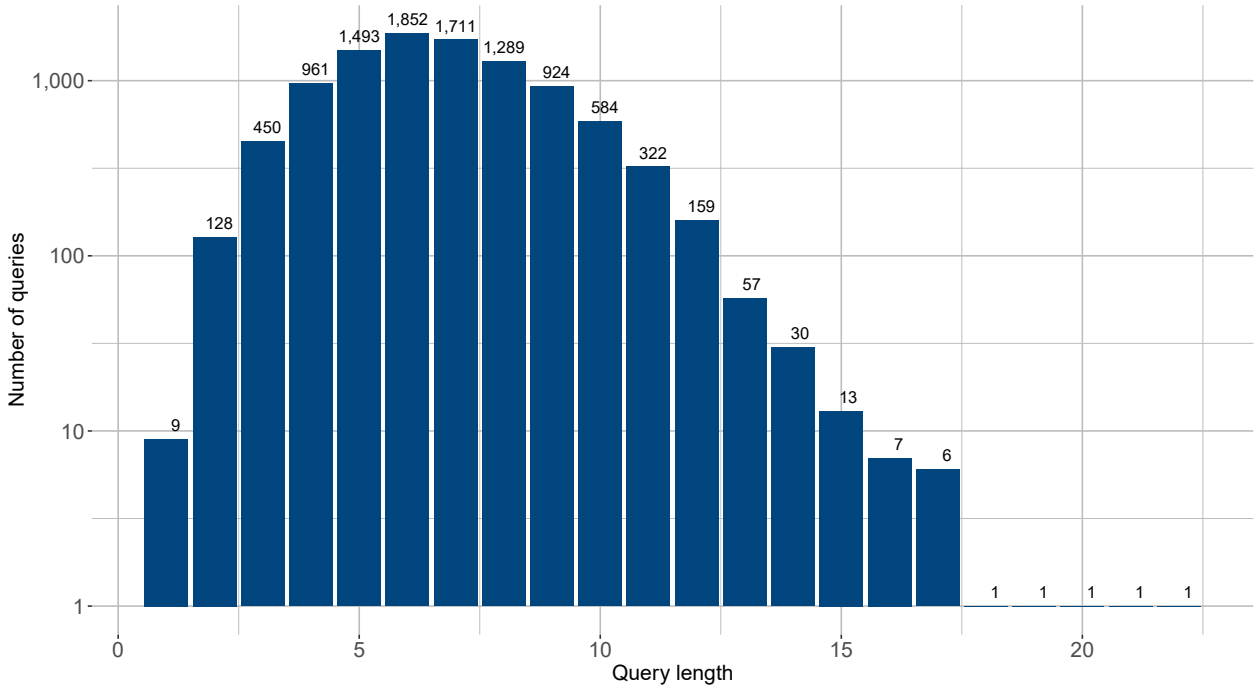

Figure 5.6: Query length distribution after stopword elimination for the AMiner setup from Table 5.1 having 10,000 queries.

5.5.3 Influence of Initially Voting Candidates

Evaluation of 3,000 and 10,000 initial voting candidates in a modified setup

In addition to the composition and structure of the collection, the number of initial voting documents – defined as $|R(q)|$ in Section 2.2 – is a relevant factor for the quality of the voting results. This influencing factor has already been well researched in the domain of expertise retrieval [AS12; MO08]. We build on these research results and investigate the influence of the variation in the number of global and initial voting candidates on the applied voting techniques using the AMiner collection. In this evaluation, the AMiner collection is composed of a higher number of articles and journals, since we do not consider the full texts and are therefore not dependent on their existence.[6]

Having a modified setup with 1,115,976 articles corresponding to 26,270 journals as shown in Table 5.7, we applied the same techniques discussed in the sections before and changed the number of initially voting candidates. Table 5.8 shows the results of the search over titles that initially have $|R(q)| = 3,000$ and $|R(q)| = 10,000$ articles as the maximum of potential voting candidates. For plausibility reasons, some extreme variants of RR^x and CombSUM RR^x are also shown.

6: The extracted journals, articles and queries can be found at: https://doi.org/10.48564/unibafd-3xt8e-zg557

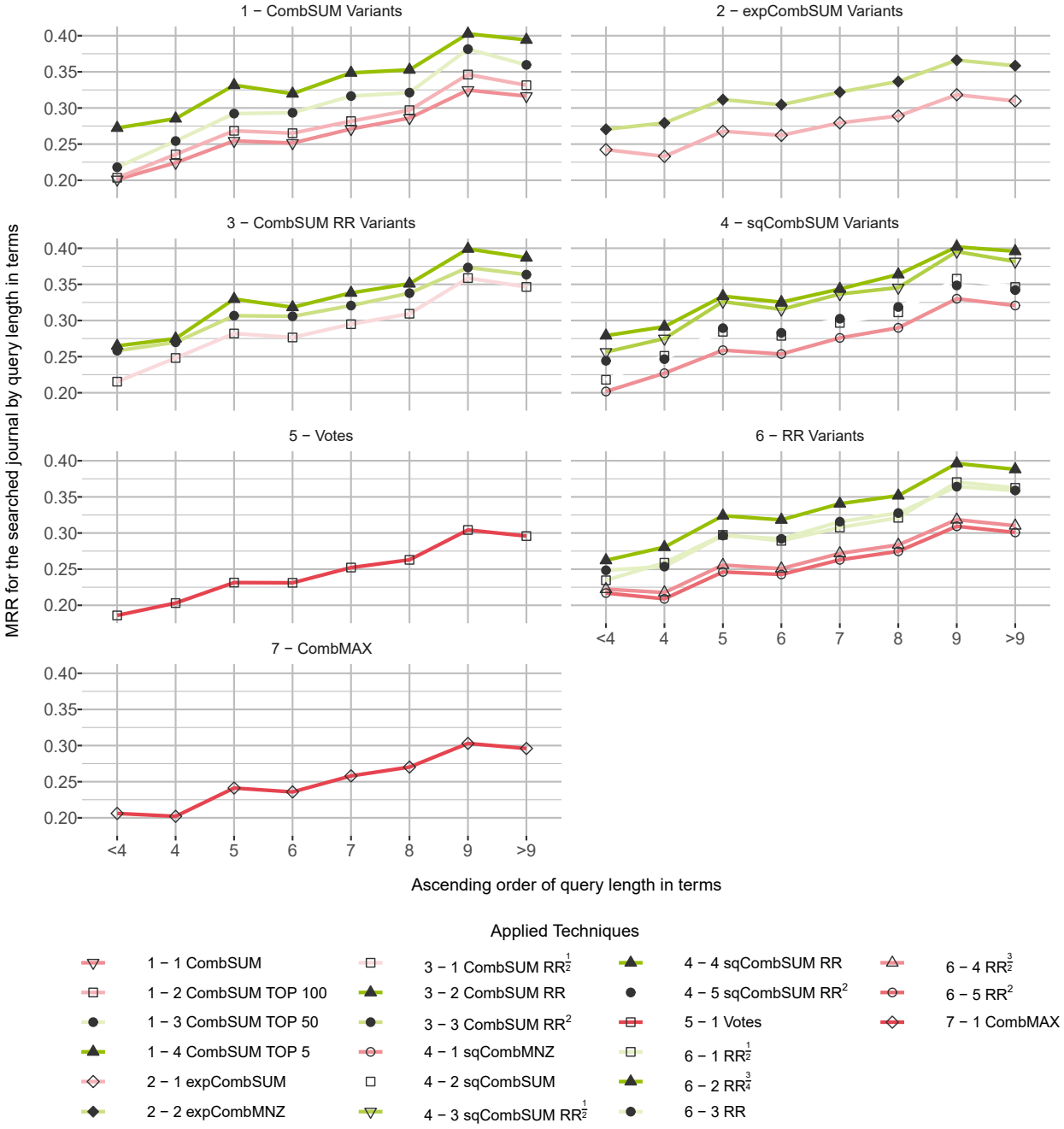

Figure 5.7: Performance of the techniques in the search over titles scenario dependent on the query term length. The horizontal axis is divided according to the length of the queries, and the vertical one shows the mean reciprocal rank for the respective length of the queries. The colors of the lines correspond to those shown in Table 5.2 in Section 5.4.1 for each technique. Global performance increases from dark red moving over white and light green to the best performing techniques in saturated green

As already stated before, the values for the 1st quartile, the median, and the 3rd quartile represent the rank r for which 25%, 50%, and 75% of the queries will rank the journal we are looking for at this rank r or better. Queries for which the desired journal does not have a voting article within the considered part of $R(q)$ are counted

Table 5.7: Modified extract of the AMiner collection and its structure for the evaluation of different numbers of voting candidates

Adjusted AMiner Collection	
#Journals	26,270
#Articles	1,115,976
#Queries	10,000

as rank 26,270.

Global view of the results Compared to Table 5.3 on page 76, the results shown in Table 5.8 for the rank of the searched journal are generally worse, as this setup covers a much larger collection with more than ten times the number of journals.

Effect on the results due to differences in the size of the collections

The results for the relative global performance of the techniques tend to be similar to those of the previous evaluations. In addition to CombSUM TOP 5, techniques like CombSUM *RR* and sqCombSUM *RR* provide the best results. Among the ranking-oriented techniques, $RR^{\frac{1}{2}}$ and $RR^{\frac{3}{4}}$ deliver the best results, which also corresponds to the previous trends. The approximation of RR^x with greater exponents to CombMAX and lower values of x to Votes is clearly visible here. Furthermore, the variation of x in CombSUM RR^x indicates the approximation to CombSUM and CombMAX as described in Section 4.1.4.

Relative differences in performance confirm the previous evaluations

Comparison of the scenarios If up to 10,000 initial voters are taken into account, the number of documents returned as voting candidates increases, but the results returned in the top 3,000 ranks remain the same compared to the smaller voting scenario. The main effect of this is that large classes may receive further growth. Furthermore, newly added candidates can vote for classes that would not have been considered by the top 3,000 ranked documents. The fact that these new voting candidates lead to a different, worse result, particularly in quantitative terms, can be observed especially for Votes and CombSUM. A possible higher recall at the document level can, depending on the course of the scores, automatically lead to a preference for larger journals, and thus to poorer results at the class level regarding quantitatively oriented techniques.

High number of voting candidates negatively impacts performance for quantitative techniques.

The situation is different for techniques that focus on the top global ranks. With CombMAX, taking into account only the document with the highest score per class, the results are almost the same for both scenarios. Similarly, the results for the best performers change only slightly with a higher number of voters in the first two quartiles. In the 3rd quartile, they even benefit from the higher number of global voting articles. This effect can be observed with

Techniques emphasizing the top ranks are influenced only to a small extent

Table 5.8: Results for selected voting techniques having 3,000 and 10,000 articles as initial voters in the search over titles scenario.

# Voting Articles ▸	3,000 Voting Articles			10,000 Voting Articles		
Technique ▽	1st q.	med.	3rd q.	1st q.	med.	3rd q.
CombSUM	8	46	338	11	63	370
CombSUM TOP 500	8	46	338	11	63	370
CombSUM TOP 100	8	46	338	11	63	370
CombSUM TOP 50	8	46	338	10	63	370
CombSUM TOP 10	5	45	338	5	41	370
CombSUM TOP 5	5	37	338	5	35	350
expCombSUM	9	59	409	9	59	398
expCombMNZ	7	45	359	7	45	345
CombSUM $RR^{\frac{1}{4}}$	7	44	333	10	58	356
CombSUM $RR^{\frac{1}{2}}$	6	41	330	8	52	340
CombSUM RR	5	36	328	5	39	314
CombSUM RR^2	5	39	333	6	41	336
CombSUM RR^4	9	59	407	9	60	417
sqCombMNZ	7	45	337	11	61	365
sqCombSUM	6	39	330	8	49	320
sqCombSUM $RR^{\frac{1}{2}}$	5	36	334	6	39	316
sqCombSUM RR	5	36	341	5	37	324
sqCombSUM RR^2	7	48	370	7	49	370
Votes	9	56	406	15	82	476
$RR^{\frac{1}{16}}$	8	52	353	14	76	430
$RR^{\frac{1}{8}}$	8	49	342	12	70	408
$RR^{\frac{1}{4}}$	7	42	326	10	59	348
$RR^{\frac{1}{2}}$	5	36	335	6	40	321
$RR^{\frac{3}{4}}$	6	40	355	6	40	331
RR	7	47	372	7	46	353
$RR^{\frac{3}{2}}$	9	56	398	9	55	388
RR^2	10	61	418	10	61	415
RR^4	10	66	451	10	66	451
RR^{10}	10	68	461	10	68	461
CombMAX	10	68	462	10	68	462

CombSUM RR, sqCombSUM RR, and also with $RR^{\frac{3}{4}}$, which significantly increase their performance in the 3rd quartile.

5.6 Principle of Inclusion and Exclusion

Our evaluations applying voting techniques have shown the need for a damping factor to reduce the impact of posterior voting candidates for a class. Just adding the scores as a combination of evidence will produce a misleading result. In order to determine a damping factor from the perspective of probability theory, in this section we refer to the problem description in Section 1.2 and build on the approach of adding probabilities from formula 1.2. Our considerations aim to process the voting scores of a class according to the principle of adding depending probabilities.

5.6.1 Determining an Aggregation Function Based on the Addition of Probabilities

Class-wise addition of document probabilities for representing typical examples of their associated class

The scores determined during the search at the document level correspond to the probabilities that the documents found will represent their associated class in the context of the query. These probabilities are not disjoint, as their occurrence is not independent. Based on this assumption, a scenario for a class c having probability $P(c)$ as a candidate and having two voters v_1 and v_2 would have an optimal aggregation of probabilities P expressed by

$$P(c) = P(v_1 \cup v_2) = P(v_1) + P(v_2) - P(v_1 \cap v_2) \tag{5.2}$$

Extending this exemplary representation of two voters for class c to n voting elements v_i leads to the inclusion-exclusion principle or the sieve formula [Gal11; Hen08] adapted in formula 5.3.

$$P\left(\bigcup_{i=1}^{n} v_i\right) = \sum_{k=1}^{n}\left((-1)^{k+1} \cdot \sum_{I \subseteq 1\ldots n, |I|=k} P(\bigcap_{i \in I} v_i)\right) \tag{5.3}$$

Transfer of the principle of inclusion and exclusion to the addition of document probabilities

The formula expresses the methodical approach of first adding all single probabilities, then subtracting the intersections of all tuple combinations followed by adding all the intersections of triple combinations, and so on until the number of n combinations is reached. The set I is a subset of elements from 1 to n in each step. The number of elements of I is determined with each step by assigning $|I| = k$ in ascending order. All combinations of v_i, the number of which is determined by $|I|$, are added or subtracted in each step – depending on the term $(-1)^{k+1}$.

For a class c having n voting candidates v_i, the sum of their probabilities $P(v_i)$ is expressed in a schematic presentation like this:

$$
\begin{aligned}
P(c) = P(v_1) + P(v_2) + \ldots + P(v_n) \\
-P(v_1 \cap v_2) - P(v_1 \cap v_3) \ldots \text{all pairwise combinations} \\
+P(v_1 \cap v_2 \cap v_3) \ldots \text{all triple combinations} \\
\ldots \\
+(-1)^{n-1} \cdot P(v_1 \cap v_2 \cap v_3 \ldots \cap v_n)
\end{aligned}
$$

Definition of intersections based on the term frequency

Determining the intersections In order to apply these considerations to a voting scenario, we have to transfer the expression $P(v_1 \cap v_2)$ in formula 5.2 to our scoring formula at the document level.

In Section 2.1.2 on page 17 the BM25-based scoring of a document is defined as the result of a multiplicative composition of the inverse document frequency idf and the term frequency tf. The idf is a term-based factor that results from the term distribution over the entire collection. Since it is applied equally to all documents, we leave this component out of these considerations.

As the second factor, the frequency of the term tf is based on documents and is the result of the number of occurrences of the term t in the document d. Having one term t and j documents d we define an intersection I as

$$I(t, d_1, ..., d_j) = \min_{i=1;j} (tf(t \text{ in } d_i)) \tag{5.4}$$

Based on the bag-of-words model, for all documents d, this definition implicitly generates the smallest common corpus for the term t to find the intersection for.

Applying the systematic of the adapted sieve formula 5.3 and the definition of the intersection in formula 5.4 to our experimental setup and search scenario, we obtain the scoring formula 5.5 for the search for journals. This formula defines the score of a journal based on the set of its voting documents $V_{Journal}$, whose individual documents are defined by d_u. The query q is made up of the terms $t_1, ..., t_n$, of which the number n is defined by $|q|$.

$$score_{Journal} = \sum_{i=1}^{|q|} idf(t_i) \cdot \sum_{k=1}^{|V_{Journal}|} (-1)^{(k+1)} \cdot \sum_{\substack{U \subseteq V_{Journal}, \\ |U|=k}} \min_{d_u \in U} \left(tf(t_i \text{ in } d_u) \right) \tag{5.5}$$

The first factor in equation 5.5 iterates over all terms t_i included in a query q that has the number of terms $|q|$ and gets his value from the term-specific idf.

The second factor iterates from one to the number of voting documents $|V_{Journal}|$ and potentiates (-1) in order to add or subtract intersections which are determined in the last factor according to the definition in equation 5.4.

Example calculation A calculation using a simple example with a query q consisting of a single term is intended to illustrate formula 5.5. As a result of the query q with $|q| = 1$, three documents d_1, d_2 and d_3 vote for a journal in the example, resulting in $|V_{Journal}| = 3$. The values of the term frequencies tf for the query term t in the documents are shown in Table 5.9 with the fictitious values 2,3 and 5. The value of $idf(t)$ is only shown as a factor symbol for the explanation in the example.

Table 5.9: Term frequency values $tf(t)$ for the query term t for the documents d_1, d_2 and d_3

	$tf(t\ in\ d_1)$	$tf(t\ in\ d_2)$	$tf(t\ in\ d_3)$
Query term t	2	3	5

The calculation of the journal score then results in the following calculation 5.6 as follows:

$$\begin{aligned} & score_{Journal} \\ & = idf(t) \cdot \Big((1) \cdot (2+3+5) + (-1) \cdot (2+2+3) + (1) \cdot (2) \Big) \\ & = idf(t) \cdot (2+3+5-2-2-3+2) \\ & = idf(t) \cdot 5 \end{aligned} \tag{5.6}$$

This example shows that the highest occurring value, 5 in the example, prevails for the term frequency in the score calculation and is decisive for the calculation of the journal score.

Performance of the technique In this thesis, we use this formula restricting $|V_{Journal}|$ to a maximum of five voting articles as an approach to obtain an optimized CombSUM TOP 5 technique. According to the MRR-based evaluation in Section 5.4.1, this constellation produces an MRR value of 0.19 which represents a disappointing result and cannot compete with the voting techniques and results in Section 5.4.1 and shown in Table 5.2.

Tests with the AMiner collection from Section 5.3 within the title-based scenario a deliver poor results.

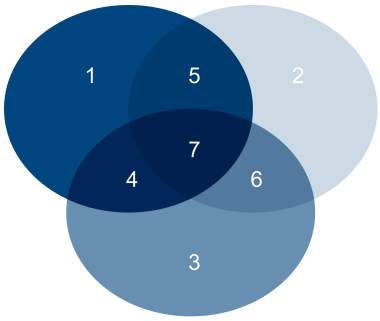

Figure 5.8: Three amounts and their intersections; every subset is numerated.

5.6.2 Adjusting the Aggregation Function

Regulation of inclusion and exclusion using the factor α

Calculating the damping factor α The preceding considerations and definitions completely eliminate repetitive intersections between probabilities and amounts. This is based on the fact that the term frequencies of a term in voting documents do not overlap, but smaller values for frequencies are included in higher ones. The idea is now to gradually reduce the complete elimination of lower term frequency values by using a damping factor.

To soften this complete removal of intersections, we implement a damping factor α. Figure 5.8 shows three sets and their intersections. The idea is not to completely remove multi-counted intersections by applying the inclusion and exclusion principle, but to partially keep the proportions included by using a damping factor α.

Table 5.10: Schematic addition of the three circles' amounts from Figure 5.8 by applying the principle of inclusion and exclusion, parameterized with factor α.

1st Step: Complete addition of all circles' cardinalities										
All subareas of the upper left ellipse:	Σ	|1|			|4|	|5|		|7|		
All subareas of the upper right ellipse:	$+\Sigma$		|2|			|5|	|6|	|7|		
All subareas of the lower ellipse:	$+\Sigma$			|3|	|4|		|6|	|7|		
2nd Step: Reduction of circles' intersections										
Intersection of the upper left and lower ellipse:	$-\alpha \cdot (\Sigma$				|4|			|7|	)	
Intersection of the upper left and right ellipse:	$-\alpha \cdot (\Sigma$					|5|		|7|	)	
Intersection of the upper right and lower ellipse:	$-\alpha \cdot (\Sigma$						|6|	|7|	)	
3rd Step: Addition of the central amount										
Intersection of all three ellipses:	$+\alpha \cdot ($							|7|	)	

Table 5.10 schematically shows the method for adding cardinalities of three amounts, including a damping factor α. If α was chosen to 0.7, the cardinalities of sections four, five, and six would still go into the sum with factor 1.3. Additionally, the central intersection between all three amounts numbered by seven would contribute with a factor of 1.6.

This factor α can vary between 0 and 1. Setting $\alpha = 1$ would have the effect of the first and initial approach in equation 5.5. The other extreme, setting $\alpha = 0$, would end with the result of applying CombSUM TOP n, in the case of the example shown in the schematic addition of Table 5.10, it would be CombSUM TOP 3.

Including factor α and considering the definition in 5.4 leads to the calculation of the score expressed in formula 5.7.

$$score_{Journal} = \sum_{i=1}^{|q|} idf(t_i) \cdot \left(\sum_{d_v \in V_{Journal}} tf(t_i \ in \ d_v) - \alpha \cdot \left(\sum_{k=2}^{|V_{Journal}|} (-1)^{(k)} \sum_{\substack{U \subseteq V_{Journal}, \\ |U|=k}} \min_{d_u \in U} (tf(t_i \ in \ d_u)) \right) \right) \tag{5.7}$$

The first factor represents the inverse document frequency idf of the term t_i from the query q. The numerator i iterates from 1 to the number of query terms $|q|$.

The second factor in the outer brackets consists of two components: First, the term frequencies tf for each voting document d_v of the entire set of voting documents $V_{Journal}$ are added without including alpha. The second component, the subtrahend, is controlled in intensity via α and incorporates the principle of inclusion and exclusion as already defined, but starting with $k = 2$, since the initial and complete term frequencies tf have already been added before.

Example calculation of the adjusted formula With reference to the example scenario of Section 5.6.1 and its exemplary calculation of the journal score, the calculation of the journal score that includes α would be performed as in the following equation 5.8:

$$\begin{aligned} & score_{Journal} \\ & = idf(t) \cdot \Big((2+3+5) - \alpha \cdot \big((1) \cdot (2+2+3) + (-1) \cdot (2) \big) \Big) \\ & = idf(t) \cdot \Big(10 - \alpha \cdot \big((2+2+3) - (2) \big) \Big) \\ & = idf(t) \cdot \Big(10 - 5 \cdot \alpha \Big) \end{aligned} \tag{5.8}$$

Table 5.11: Results for the MRR applying the sieve formula having a maximum of five voting candidates with α as a damping factor. To show marginal differences, the results are shown with three digits after the point.

α	MRR (Titles)
0.0	0.342
0.1	**0.344**
0.2	**0.344**
0.3	0.343
0.4	0.339
0.5	0.332
0.6	0.321
0.7	0.300
0.8	0.273
0.9	0.237
1.0	0.193

MRR: Variations of alpha do not provide significant improvements in the results.

Performance of the technique applying α Applying this formula, having the previous setup with up to five voting documents, to our journal search, we get results for the MRR as shown in Table 5.11. By omitting the reduction and addition procedures and setting α to 0, we obtain the result of CombSUM TOP 5, while setting it to 1 we get the worst results for the MRR results. It should be noted that setting α from 1.0 to 0.5 rapidly enhances the MRR values, while setting α from 0.5 to 0.0 only slightly enhances them. A maximum of 0.344 is approximated for the values of α having 0.1 and 0.2, which are slightly better than the result of $\alpha = 0$, which means CombSUM TOP 5.

Grades of relationship to the searched journal: Slight improvements for small values of α.

Taking into account the relationships, according to the evaluation in Section 5.4.3 on page 77, Table 5.12 shows the grades of the relationship of the first-ranked journals to the searched journal using the sieve formula damped by α. As stated previously, each quartile shows the minimum grade of relationship for 25%, 50% and 75% of the ranked journals.

The trends of the MRR-based results in Table 5.11 are confirmed here. Slight improvements are achieved with $\alpha \in \{0.1, 0.2, 0.3\}$ in all quartiles, in the third quartile performance improves further with $\alpha = 0.4$ and $\alpha = 0.5$, while it decreases in the 1st quartile and median. This suggests that the average recall for these values of α increases.

As α increases towards 1.0, the degrees of relationships decrease rapidly in all quartiles.

Influence of the sieve formula and α on a preference for small or large journals

The last part of this section evaluating the sieve formula examines the systemic behavior of the voting algorithm. Analogously to Section 5.5 on page 82, the extent to which variations of alpha

Table 5.12: Number of authors in common with the desired journal for the journals on position 1 for the search over titles. The left column shows ascending values for α as damping factor. The coloring refers to the context of each single column.

Scenario ▹		Titles	
α ▽	1st quart.	median	3rd quart.
0.0	944	171	27
0.1	**971**	175	28
0.2	**955**	176	30
0.3	**971**	176	31
0.4	919	176	**32**
0.5	884	175	**32**
0.6	846	165	31
0.7	725	145	28
0.8	652	126	24
1.0	359	76	16

systematically lead to preferences for large or small journals is examined. The diagram 5.9 shows the influence of differences in journal sizes on the voting technique evaluated. On the horizontal axis, the journals are sorted into bins ascending by size, according to the scheme of Figure 5.5. The vertical axis shows the number of journals ranked in the first rank for applying different values for α. All variants have tendencies that correspond to the voting technique CombSUM TOP 5 on which they are based. The colors correspond to the performances arising from Table 5.11.

5.6.3 Summary

It remains to be seen that the complete removal of overlapping term frequency scores worsens the results. Applying equation 5.7 and choosing a small value for the damping factor α can lead to improved results that possibly surpass the results obtained by CombSUM TOP 5.

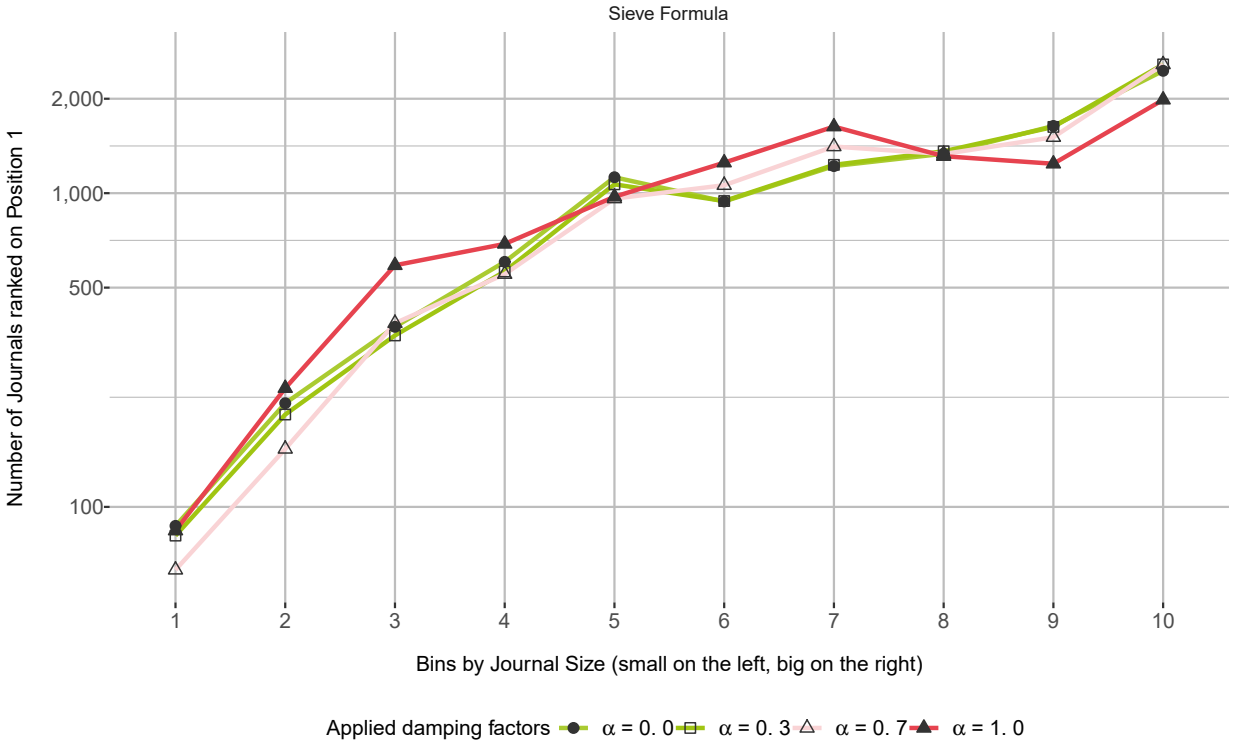

Figure 5.9: Number of journals ranked on position 1: Results for applying the principle of inclusion and exclusion in the search over titles scenario following equation 5.7 with different values for α

5.7 Online Forum Thread Retrieval

Evaluation of the newly proposed techniques on an existing scenario: forum messages that vote for their associated thread

During our research and the search for comparable scenarios, we found a paper written by Albaham et al. having the title "Adapting Voting Techniques for Online Forum Thread Retrieval" and discussing the search for appropriate threads in online forums regarding a certain information need [AS12].

5.7.1 The New York Travel Forum Task

One of the paper's two investigation scenarios is related to the search for relevant threads in a New York travel forum.[7] In this scenario, all thread messages returned for a query act as voters for their associated thread. Many of the voting techniques that are the subject of this thesis are also applied by the authors in this paper. Our aim is to evaluate how these techniques perform in comparison to our newly proposed techniques and, furthermore, to determine how these results reconcile our evaluation results regarding the AMiner collection. 25 queries were obtained by extracting keywords from frequently searched topics [BM10] and manually judged in order to establish an evaluation basis.

7: Threads and messages were taken from the Tripadvisor New York Travel Forum: https://www.tripadvisor.com/ShowForum-g28953-i4-New_York.html

Differences in the setup of the experimental settings

Regarding the experimental settings, there are differences between our AMiner setup and the baseline of the configuration of the paper. First, all the evaluation results in the considered paper are based on manual judgments, whereas our ground truth in this thesis is established by the relationships of titles, their articles, and belonging journals. For comparability reasons, we continue to use BM25 for the similarity search, while the effectiveness measure of the paper is based on the query language model [PC17]. The results shown in the work of Albaham et al. do not correspond directly to the values resulting from our experiments with the collection.

The excerpt from the New York Travel Forum contains 83,072 threads, which include 590,021 messages, the initial messages included. The evaluation setup is designed with 25 queries and 4,478 query relevance judgements as shown in Table 5.13.

Table 5.13: Collection structure and setup for the New York travel forum retrieval task

New York Travel forum	
#Threads	83,072
#Messages	590,021
#Queries	25
#Qrels	4,478

Regarding the structure of the collection, about 50% of the threads have a message count of 5 or fewer messages, about 75% of the threads include 9 or fewer messages. The highest message count for one thread is 1,217.

While working with the collection, we noticed that 21 threads having the same key and content are included twice. These double entries are not part of the relevance ratings; for comparability reasons, we do not make any corrections at this point.

5.7.2 Results for New York Travel Forum Task

Number of initially voting messages Table 5.14 shows the mean average precision (MAP)[8] of the techniques applied for three variations of the initial voting messages in the columns.[9] Due to the predominantly small number of messages in the threads, the results for CombSUM, CombSUM TOP 100 and CombSUM TOP 50 are equal for each variation of the initial voting messages. A raising number of voting candidates mostly affects and worsens the performance of techniques which take all or many items into account. Examples for this observation are, particularly, CombSUM, even in its limited variants, and Votes. Less effects are to be

8: MAP values are based on the definition in the book *Introduction to Information Retrieval* [MRS08].

9: The standard tool trec_eval is used to determine the values programmatically and is available under https://github.com/usnistgov/trec_eval

Table 5.14: Mean average precision (MAP) results for the online thread retrieval task for three variations of initially voting messages. The coloring refers to MAP values within a column. Green, white and red tones are related to the performance of the voting techniques among themselves and not across the variants of the initial voters.

Number of Voting Messages ▹ / **Technique ▽**	**1,000**	**5,000**	**10,000**
CombSUM	0.220	0.195	0.171
CombSUM TOP 100	0.220	0.195	0.171
CombSUM TOP 50	0.220	0.195	0.171
CombSUM TOP 5	0.223	0.212	0.209
expCombSUM	0.185	0.194	0.195
expCombMNZ	0.217	0.237	0.239
CombSUM $RR^{\frac{1}{2}}$	0.225	0.214	0.198
CombSUM RR	0.237	0.243	0.236
CombSUM RR^2	**0.241**	**0.262**	**0.265**
sqCombMNZ	0.221	0.196	0.173
sqCombSUM	0.236	0.237	0.224
sqCombSUM $RR^{\frac{1}{2}}$	**0.243**	**0.258**	**0.256**
sqCombSUM RR	**0.240**	**0.262**	**0.268**
sqCombSUM RR^2	0.215	0.230	0.233
Votes	0.178	0.141	0.111
$RR^{\frac{1}{16}}$	0.218	0.174	0.136
$RR^{\frac{1}{2}}$	0.215	0.235	0.238
$RR^{\frac{3}{4}}$	0.196	0.211	0.214
RR	0.187	0.198	0.199
$RR^{\frac{3}{2}}$	0.178	0.185	0.186
RR^2	0.175	0.181	0.181
CombMAX	0.170	0.175	0.176

seen for CombMAX and techniques that emphasize the first ranks. Some techniques even benefit from a high number of voters, in particular CombSUM RR^2, sqCombSUM $RR^{\frac{1}{2}}$, and sqCombSUM RR.

Techniques that emphasize the top ranks at class level achieve the best results.

Performance of the applied techniques Best results for the mean average precision are obtained by techniques that combine score values with a class-scoped and ranking-based damping factor. These newly introduced techniques in this thesis include CombSUM RR^2 and the exponential variants sqCombSUM $RR^{\frac{1}{2}}$ and sqCombSUM RR. Not only do they show the best results within the scenarios, but their performance is relatively robust to changes in the number of initial voters. These observations state the results of the investigations and experiments with the AMiner collection.

As in previous experimental results, the variants RR^x approximate the techniques Votes and CombMAX. $RR^{\frac{1}{2}}$ provides the best results for this group and can even benefit from an increasing number of initial voting messages. The effect of a very low exponent for RR^x approximating the worst results of Votes becomes obvious for a high number of initial voting messages. For 1,000 voters, it is remarkable that $RR^{\frac{1}{16}}$ gives the best results in this category of RR^x. Only an increase in initial voters shows the real approximation for $RR^{\frac{1}{16}}$ at 5,000 and 10,000, thus a noticeable deterioration.

In relation to the best results of Albaham et al., new techniques from this thesis deliver better results

In addition to BordaFuse and expCombMNZ, CombSUM performs best for the MAP measure with 1,000 initial voting messages in Albaham et al.'s paper. Although the results from Table 5.14 as absolute numbers do not match those of the paper due to the different setup, by looking at the results in relative terms it can be concluded that the CombSUM RR^x and sqCombSUM RR^x variants can outperform those techniques based on MAP. In Table 5.14, CombSUM and expCombMNZ deliver good results, but are not among the top performers in our analyzes.

6 Passages as Voters for their Documents

In this chapter, we investigate how passages that vote for their documents can improve the results of document retrieval. This analysis focuses in particular on long documents whose relevance is to be determined. The idea is to split these documents into passages and then apply voting techniques for document retrieval: Every passage relevant to the query votes for its document.

Basis of this chapter: Extracted passages that vote for their associated documents

We begin with Section 6.1 with a brief overview of passage retrieval and its areas of application.

Section 6.1 Passage Retrieval as an Object of Research in Information Retrieval provides an exemplary overview.

In the following Section 6.2 we first reverse the train of thought and discuss what results can be expected for BM25-based retrieval with the contents of previously voting articles from the AMiner collection in their concatenated form.

Section 6.2 Virtual Documents: Initial consideration of the three reverse scenarios, the concatenation of AMiner titles, abstracts, articles, and BM25-based document search on that basis

We then check the effectiveness of passage voting in Section 6.3 using selected collections as an example. Passages extracted from collections' documents vote for their associated documents using the voting techniques presented. We compare these results with a flat document search based on BM25.

Section 6.3 Passages as Voters: Exemplary Studies: Extracted passages that vote for their documents, using selected test collections from conferences and literature

In Section 6.4 we develop a test environment consisting of virtual documents to experimentally substantiate the results of previous evaluations. Since the voting results cannot be evaluated qualitatively when using a pseudo-collection, we analyze the systemic and quantitative behavior of the voting techniques based on the structure and term distribution of the pseudo-collection.

Section 6.4 Investigations Using Pseudo-Collections: Pseudo-collections with a predefined document and term structure provide information on the quantitative behavior of voting techniques.

6.1 Passage Retrieval as an Object of Research in Information Retrieval

Distribution of the searched terms in long documents as a relevance factor

Relevance assessment of long documents The passage retrieval approach addresses the challenges in information retrieval that arise, in particular, with long documents. From a *technical perspective*, BM25 takes into account the frequency of term occurrence in combination with moderate document length normalization; see formula 2.5 on page 17. However, the distributions of searched terms within long texts, which can be decisive for the relevance of the document, are not taken into account by this formula. If the searched terms are not equally distributed but concentrated in a section of a long document, this can be decisive for the relevance of the document. A document having a section with a high concentration of query terms and therefore of high importance to the user can be more relevant than a document with an even distribution of query terms.

Combination of document-based and passage-based relevances

A case illustrating how the relevance of entire documents and their individual passages is combined is discussed by Callan in his paper "Passage-Level Evidence in Document Retrieval" [Cal94]. Within its experimental setup, the relevance of rankings can be increased by combining passage-based and document-based relevances.

Return of passages from the user's perspective

Just as the relevance analysis of passages of long texts can be advantageous from a technical point of view, there are also advantages from the *user's perspective*: The user's need for information might be better served by the return of suitable passages from a very long document than by returning the whole document, as Salton, Allan, and Buckley state in their paper "Approaches to Passage Retrieval in Full Text Information Systems" [SAB93].

Generation of passages based on different features

Generation of passages In his paper "Passage-Level Evidence in Document Retrieval" [Cal94], Callan defines three different ways of extracting passages: In the *discourse*-based extraction, passages are generated on the basis of sentences, paragraphs, and sections. *Semantic* division is achieved on the basis of logical sections resulting from the content structures. *Window*-based extraction of passages divides long texts into sections based on a fixed number of terms. For the generation of window-based passages, Callan also considers overlapping sliding windows. The rationale behind this approach is the risk-reduction of splitting small blocks of related text or information into two different passages, potentially reducing their relevance.

Use of passage retrieval in the recent past

Recent approaches Until recently, approaches based on passage retrieval have been used to improve document and information

retrieval. With the publication of the MS MARCO dataset, exemplary tasks in the field of question answering are proposed, which are solved using passages from the Web [Ngu+16].

Aggregation of passage scores to assess document relevance

The approaches that aggregate passage scores to assess the relevance of the entire document are pursued by Dai and Callan in their article "Deeper Text Understanding for IR with Contextual Neural Language Modeling" [DC19]. Using BERT (Bidirectional Encoder Representations from Transformers), passages obtained by using a 150 word sliding window with a progression of 75 words are evaluated in terms of their relevance. The relevance of the passage of each document is then aggregated using three techniques. The *BERT-MaxP* technique is based on the maximum passage score of a document as its relevance. *BERT-FirstP* takes the relevance of a document from its first passage, and *BERT-SumP* adds all the passage scores of a document [DC19; LNY21].

This last approach comes close to this chapter's concept of establishing a document ranking using passages and their scores to create a document ranking via voting techniques. The aforementioned *BERT-MaxP* then corresponds to the technique CombMAX and *BERT-SumP* corresponds to the voting technique CombSUM.

6.2 Virtual Documents

Considering AMiner documents as passages of virtual documents representing virtual journals

In this section we go back to the setup of the AMiner collection from Section 5.3 and address the question of what results can be expected from a concatenation of journal articles and a subsequent BM25-based document search.

We concatenate the titles, abstracts, and complete texts of the articles into three experimental setups. For each variant, we get 1,916 entries in the index, corresponding to the number of journals in the AMiner collection. According to the evaluations from Chapter 5, we use the 10,000 queries to determine the MRR value of the searched journal (Section 6.2.1) and the grades of relationship of the top ranked journals (Section 6.2.2). In this way, we check to what extent a BM25-based search with its document length normalization over long concatenated documents with large differences in lengths is applicable for the search for appropriate journals.

6.2.1 MRR for the Searched Journal

Result quality for concatenated titles is best for virtual documents.

Table 6.1 shows the results for the mean reciprocal rank (MRR), which is the rank of the journal for which we are searching in each of our 10,000 requests. In contrast to the voting scenarios discussed

Table 6.1: MRR for the BM25-based search over concatenated titles, abstracts and full texts for each journal. Note the coloring which only refers to this setup of concatenated AMiner-articles and excludes the previous evaluations from Section 5.4.

	MRR of the searched journal
Titles	0.23
Abstracts	0.22
Full Text	0.19

and previous evaluations, performance decreases as information increases from concatenated titles over abstracts to full texts.

Applied voting techniques yield better results compared to the search over virtual documents.

Basically, the results of these three scenarios cannot compete with those of Section 5.4.1 Using MRR as Measure. Here, the best value for the MRR reached 0.34 in the titles scenario (CombSUM TOP 5, CombSUM *RR*, sqCombSUM *RR*, $RR^{\frac{3}{4}}$). A value of 0.36 for the MRR was reached within the abstract and full text scenarios.
All the results of the three concatenated setups cannot reach the performance of the voting scenarios, whose worst MRR value is 0.25 for CombMAX and Votes in the titles scenario (see also Table 5.2 on page 74).

In the domain of expertise retrieval, Macdonald substantiates the worse results for virtual documents in his thesis as follows:

"*However, in the expert search scenario, the (virtual) documents are very large [...], so there is a high chance that many query terms will occur in a large fraction of the virtual documents, and hence removing these terms would result in no documents being retrieved. Moreover, in such a scenario, the weighting model will struggle to differentiate between an informative term and a non-informative term, because the term specificity, measured by the number of profiles a query term occurs in, will be similar for many terms, as many documents about varying topics will exist in the profiles of many candidates.*" [Mac09, sec. 6.3.2].

Based on this reasoning from Macdonald, the declining performance of the title, abstract, and full text scenarios can also be justified. As the length and information content of the concatenated virtual documents increases, the specificity of the journal profiles decreases and converges. As the specificity decreases, the probative value of the virtual documents for each journal decreases, and thus the quality of the search results.

Table 6.2: Number of authors in common (grades of relationship) with the desired journal for the results on positions 1 to 5. Note that the coloring refers to the performance of this section's setup and is not related to the values from the previous evaluation in Section 5.4.3. The relative coloring relates to each quartile across all scenarios.

Scenario ▸	Titles			Abstracts			Full Text		
Position ▽	1st q.	med.	3rd q.	1st q.	med.	3rd q.	1st q.	med.	3rd q.
1	219	39	6	203	38	6	147	24	4
2	183	33	5	177	31	5	118	21	3
3	162	28	4	156	27	4	104	19	3
4	142	27	4	136	26	4	101	18	3
5	123	23	4	127	23	4	90	16	3

6.2.2 Journal Relationships on the Upper Ranks

All returned grades of relationship are lower than those returned by voting techniques. Searching over concatenated titles provides the best results for virtual document scenarios.

In analogy to Section 5.4.3 Journal Relationships as Measure for Relevance, Table 6.2 shows the relationships of journals ranked at positions 1 to 5 with the journal we are looking for. For every scenario, the 1st quartile shows the minimum relationship as the number of common authors for 25% of the results, the median shows the minimum relationship for 50% of the returned journals, and 75% of the results have a relationship equal or better than shown in the 3rd quartile.

These results confirm those in the previous Table 6.1. The search via titles delivers better results than the other scenarios in which the search via full texts performs the worst. The best results are below the values determined by the voting-based search that yielded a maximum grade of relationship of 1,015 (CombSUM *RR*) and a worst result of 415 (CombMAX) for the title-based 1st quartile (see also Table 5.4 on page 78).

Grades of relationship as a meaningful measure

The presentation in Table 6.2 shows that the evaluation of the results and the ranks using degrees of relationship is meaningful. All positions of journals are determined independently on the basis of the search and shown in total, based on quartiles, with their derived relationships. In all rows and within the quartiles, there are decreasing values that prove this measure.

6.2.3 Systemic Behavior and Influence of the Collection Structure

In contrast to voting techniques, the search over concatenated titles returns more small journals on the first rank.

Following the intention of Section 5.5 Systemic Behavior of the Applied Techniques, we examine to what extent the search over concatenated documents favors large or small journals. Figure 6.1 shows the bin structure as designed in Table 5.6 in combination with the results at position 1 of the virtual document search based on concatenated titles.

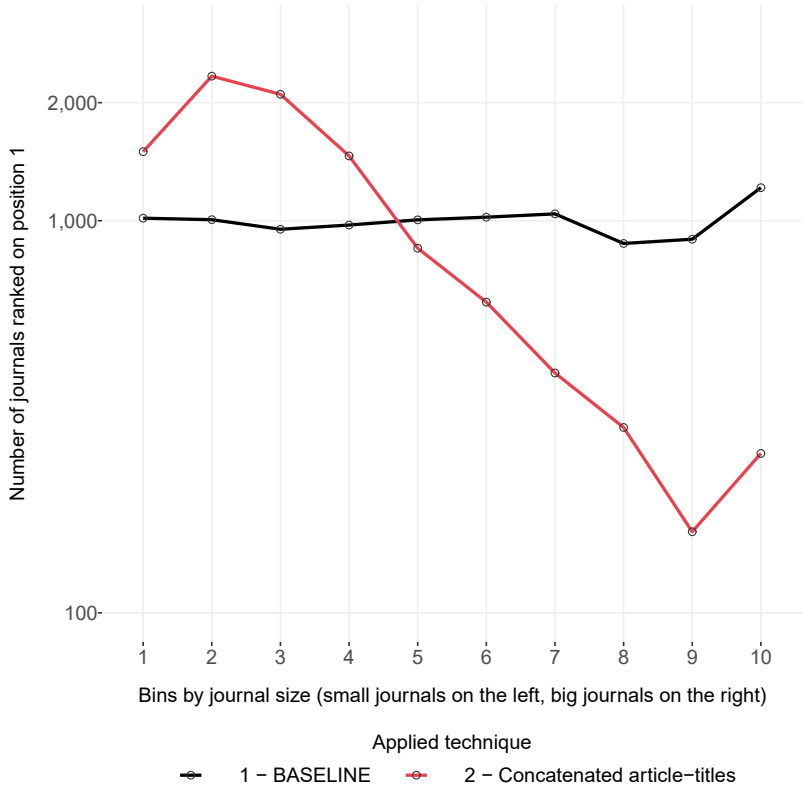

Figure 6.1: Top 1 ranked journals per bin for the search over concatenated titles (virtual documents). BASELINE represents the origin of the 10,000 requested articles.

Contrary to all results in the voting scenario, not large journals with many articles tend to be preferred here, but concatenated small ones more often achieve rank 1 above the baseline.

One reason for this is the fact that large journals provide broader thematic content and are not as focused on specific topics. Furthermore, document length normalization leads to a saturation of the score values for high-term occurrences, which has a more intense effect for longer journals or their concatenated article-titles.

6.3 Passages as Voters: Exemplary Studies

This section evaluates the extent to which the use of voting techniques can achieve better results than traditional document retrieval. For this purpose, we use collections and divide their documents into passages. Based on the query, the passages vote for their associated documents, and as a result, we obtain a document ranking. We compare these results with those based on BM25 document retrieval.

Following the considerations of Section 6.1, we select the passages in Section 6.3.1 INEX 2009 Collection *discourse*-based, resulting from the structuring markup in the source documents.

In the following experiment in Section 6.3.2 Robust04 Collection the passages are created *window*-based. In addition, following the literature [Cal94; DC19], we select passages with different lengths and overlaps in this section to compare the results.

6.3.1 INEX 2009 Collection

For our first experiment, we utilized a dump of English Wikipedia articles taken in October 2008.[1] This dump contains 2,666,190 articles with their structuring markup. A preliminary version of this collection is described in [SSK07]. In this experimental case, we used semantically related sections for the generation of passages. These sections are predefined by the markup like and <template> layers. Other undefined or unmarked passages are overtaken as body text as one passage. After removing the tags, we

1: The INEX 2009 collection without annotation tags is available at https://www.mpi-inf.mpg.de/departments/databases-and-information-systems/software/inex/

On average few passages per document, semantically defined by markup

Table 6.3: Collection structure and passage voting setup for the INEX 2009 Collection

INEX 2009 Collection	
#Articles	2,666,190
#Passages	10,338,753
Avg. passage length (terms)	113
#Queries/Topics	107
#Qrels	39,031

obtained 10,338,753 passages for the index to search, resulting in an average ratio of 3.88 passages per article. Queries and relevance judgments are taken from the INEX 2010 Ad Hoc Track [Arv+10].[2]

2: Queries and relevance judgements from the INEX 2010 Ad Hoc Track are available at https://inex.mpi-inf.mpg.de/data/documentcollection.html

Table 6.4 shows the results of the extracted passages voting for their associated articles. In accordance with the article retrieval analyses from the INEX 2010 Ad Hoc Track, we use the P@10 measure here.[3] Good results for P@10 are shown in green, whereas colors that fade from white to red mark worse values. Since the

3: P@10 values are based on the definition in the book *Introduction to Information Retrieval* [MRS08].

Table 6.4: P@10 results for extracted passages voting for their Wikipedia articles

Evaluation Measure ▷ **Technique ▽**	P@10
Article Search	0.529
Voting Passages	
CombSUM	0.396
CombSUM TOP 100	0.396
CombSUM TOP 50	0.396
CombSUM TOP 5	0.454
expCombSUM	0.489
expCombMNZ	**0.512**
CombSUM $RR^{\frac{1}{2}}$	0.435
CombSUM RR	0.475
CombSUM RR^2	**0.527**
sqCombMNZ	0.398
sqCombSUM	0.460
sqCombSUM $RR^{\frac{1}{2}}$	**0.515**
sqCombSUM RR	**0.521**
sqCombSUM RR^2	**0.517**
Votes	0.315
$RR^{\frac{1}{16}}$	0.346
$RR^{\frac{1}{2}}$	**0.514**
$RR^{\frac{3}{4}}$	0.496
RR	0.475
$RR^{\frac{3}{2}}$	0.467
RR^2	0.469
CombMAX	0.467

average value for the number of passages resulting from one article is lower than 4, the results for the variants CombSUM, CombSUM TOP 100, and CombSUM TOP 50 are the same.

The application of voting techniques does not produce results that exceed the P@10 values of a document search performed with BM25 similarity search. In view of the potentially low number of voting passages, the Votes technique as the worst performer in this scenario is unsuitable. When applied, many articles obtain the same integer-based rating values, but are necessarily distributed in the ranks. On average, this setup has only a few semantically structured passages per document. However, the CombMAX example shows that not just one passage is of value in scoring the document in response to the query. Including more relevant passages of a document by applying damping improves performance, as seen in the RR^x and CombSUM variants revalued by RR^x.

Strong competitive results are obtained by applying CombSUM

RR^2 and the variants of sqCombSUM RR^x. Variations of RR^x can compete with the basic search, but do not pass it. It is interesting that $RR^{\frac{1}{2}}$ yields the best results for these variants before a low exponent $x = \frac{1}{16}$ approximates the results of Votes.

6.3.2 Robust04 Collection

For a next experiment using passage voting for the associated document, we used a well-known corpus from the TREC 2004 Robust Track [Voo04]. This corpus contains 528,155 articles from Financial Times, Federal Register, FBIS, and LA Times[4] The composition and structure of this collection are shown in Table 6.5. For our experiments, we used the 250 queries that come with qrels for our evaluation.

4: Further information to the TREC 2004 Robust Track is available at https://trec.nist.gov/data/robust/04.guidelines.html

Table 6.5: Composition of sources and amount of documents for the TREC 2004 Robust Track collection

Source	#Docs
Financial Times	210,158
Federal Register 94	55,630
FBIS	130,471
LA Times	131,896
Total	528,155

Generation of passages with static length

In contrast to the previous evaluation, the passages in this collection are statically defined as windows of a fixed number of terms. Furthermore, in the following examples, different step sizes are defined for the extraction of passages, which may differ from the length of the window.

6.3.2.1 Consideration of Alternative Values for the *idf*

For our first experiment, we generate document passages that have a length of 150 words and a sliding window with a progression of 75 words. This results in 3, 384, 229 passages and an average number of 6.4 passages per document.

To investigate the extent to which changes in inverse document frequency (*idf*) values caused by the generation of passages affect the evaluation scenario, we extend it.

Changes in *idf* values resulting from the generation of passages

Change of *idf* values for passages Table 6.6 exemplary shows the *idf* values and their document parameters for determination for two query terms taken from the 250 topics. The upper half of the table shows the values related to *idf* for the original collection, the lower half for the generated passages. The term "Country" occurs

very frequently in the collection with 229,735 instances, while the term "polyandry" occurs only once in the collection. Considering the development of the *idf* values for the passages, it becomes visible that the term "Country" can increase its value by factor 2.20, while the factor for the higher *idf* for "Polyandry" is only increased by factor 1.11 from its collection-based value to its passage-based value.

Table 6.6: Two exemplary chosen query terms and their related *idf* values, including parameters for determination. The upper half of the table shows the values for the original Robust04 collection. The lower half shows the parameters and the resulting *idf* values based on generated passages that have a length of 150 words and a sliding window of 75 words.

	"Country"	**"Polyandry"**
#Collection documents	528,155	
#Collection documents containing term	229.735	1
idf value (documents)	**0.83**	**12.77**
#Passages	3,384,229	
#Passages containing term	543.039	2
idf value (passages)	**1.83**	**14.12**
Increasing factor for *idf* value	**2.20**	**1.11**

This example demonstrates that frequently occurring terms can increase their *idf* values by a higher factor than rare distributed terms when generating passages: The number of documents resulting from the generation of passages increases by a factor of 6.41 in the example. The number of passage documents containing "Polyandry" doubles, while the number of passage documents with "Country" increases by a factor of 2.36. This relatively small increase in the number of documents containing "Country" results in a higher increase in the *idf* values for this term.

Applying the collection-based *idf* values of the query terms for each topic in a second scenario

Alternative application of the document-based *idf* values In order to evaluate the described effect on the *idf* values, we evaluate if the collection or original document-determined *idf* values have a higher informative value with regard to term significance.

Based on the formulas 2.5 and 2.6 on page 17, we evaluate whether determining the value $idf(t)$ alternatively based on the original distribution of t over the articles leads to better results in the passage voting. The intention is to determine whether the original distribution of the query terms across the collection documents gives a more realistic representation, and thus leads to a better ranking of the voting passages.

Technical realization In addition to the values of *idf* based on the passage index, we also determine the values based on the original collection for each query term. The article-based values for

Table 6.7: Robust04 Collection: nDCG@20-based results for passages voting for their associated documents. The first line shows the result for the BM25-based article search. The left column of the following lines displays the outcomes utilizing the original *idf* which results from indexing the extracted passages. The right column shows results for usage of the article-based *idf*.

Evaluation Measure ▷ / **Technique ▽**	**nDCG@20**	
Article Search	**0.41**	
Voting Passages	**Passage-based** *idf*	**Article-based** *idf*
CombSUM	0.232	0.239
CombSUM TOP 100	0.232	0.239
CombSUM TOP 50	0.232	0.240
CombSUM TOP 5	0.332	0.337
expCombSUM	**0.399**	**0.400**
expCombMNZ	0.398	0.396
CombSUM $RR^{\frac{1}{2}}$	0.275	0.284
CombSUM RR	0.333	0.338
CombSUM RR^2	0.391	0.396
sqCombMNZ	0.233	0.241
sqCombSUM	0.301	0.307
sqCombSUM $RR^{\frac{1}{2}}$	0.347	0.352
sqCombSUM RR	0.382	0.384
sqCombSUM RR^2	**0.401**	**0.400**
Votes	0.159	0.159
$RR^{\frac{1}{2}}$	0.311	0.311
$RR^{\frac{3}{4}}$	0.347	0.347
RR	0.355	0.355
$RR^{\frac{3}{2}}$	0.360	0.360
RR^2	0.361	0.361
CombMAX	0.391	0.390

the *idf* of a query term correspond to the value resulting from the indexing of the articles of the original collection. After determining the passage and article-based *idf* values for each topic and its query terms, we index the passages. For this evaluation scenario, Elasticsearch has been configured so that document scores are calculated only on the term frequency-based components from formula 2.5 and then manually multiplied by the respective components *idf* defined in formula 2.6 for the two scenarios.

Results Table 6.7 shows the results of this experimental setup. In accordance with the experiments from [DC19], we adopt the effectiveness measure nDCG@20.[5] The top line shows the result for the BM25-based article search that has a value for nDCG@20 = 0.41. The applied voting techniques cannot exceed this result, but certain techniques can come close to approximating it.

Passage voting results do not outperform BM25-based searches, but can compete for some voting techniques.

5: nDCG@20 values are based on the definition in the book *Introduction to Information Retrieval* [MRS08].

Results of the two *idf*-based scenarios differ only marginally

Differences between the *idf* based scenarios The results in Table 6.7 show that applying the article-based *idf* of the query terms does not make about any fundamental change. The results of Votes are the same in both columns, as the score of a passage is not relevant to the technique. The changes in the scores are only marginal and in some cases occur only in the fourth decimal place, which is not shown here.

Best results for techniques with extreme emphasis on the top ranks

Performance of the voting techniques The techniques expCombSUM and sqCombSUM RR^2 provide the best results for the nDCG@20 measure. The emphasis on the high-ranked passages by using their scores as exponents or squaring their scores has a positive effect. This is also shown by the very good results for CombSUM RR^2 and CombMAX. As in the previous evaluation, based on the INEX collection, Votes delivers worse results. The same applies to CombSUM, whose values change to a positive level for the variant CombSUM TOP 5.

6.3.2.2 Variation of Window Lengths and Overlaps

In the previous evaluation in Section 6.3.2.1, we apply a window size of 150 terms in length and a step size of 75 terms to generate the passages. In this section, we evaluate, based on the information structure of the news texts, whether shorter window sizes provide passages of higher relevancy or whether longer window sizes are more relevant if they do not segment and split relevant and related content. Therefore, in an additional evaluation configuration, we investigate the impact of variations in window lengths and overlaps on the results of passage voting. To achieve this, we create window sizes of 50, 100, and 200 terms, each with different step sizes for the window. Table 6.8 shows the different setups in the first four rows. The resulting number of passages is shown below the rows determining the window sizes and step sizes; the fourth row shows the passage-document ratio which results from the average resulting number of passages per document.

Relation of voting technology performances is consistent in the scenarios and confirms the previous evaluation.

Performance of the voting techniques As in the previous evaluation, the results confirm techniques that emphasize the upper ranks, either at the document or at the class level. In all scenarios, expCombSUM and sqCombSUM RR^2 deliver very good or even best results. With a higher exponent x, the results of CombSUM RR^x and RR^x, which approximates CombMAX, also improve.

Passage lengths of 200 terms yield the best results

Performance of the scenarios A window length of 50 terms and the same step size deliver the worst results across all voting techniques used. A length of 100 terms improves the results, but for most techniques only if the windows overlap. An equal step size of

Table 6.8: nDCG@20 results for different variations of voting passages for the Robust04 collection: The term length or window size includes sizes of 50, 100 or 200 terms including different step sizes. The coloring of the cells refers to the entire results of the table and is to be regarded as scenario-independent.

Window size (terms)	50	100	100	200	200	200
Step size (terms)	50	50	100	50	100	200
# Passages	5,707,727	5,185,722	2,985,085	4,254,210	2,495,195	1,621,808
Passage-document ratio	10.8	9.8	5.7	8.1	4.7	3.1
CombSUM	0.227	0.249	0.223	0.258	0.238	0.217
CombSUM TOP 100	0.227	0.249	0.223	0.258	0.238	0.217
CombSUM TOP 50	0.227	0.250	0.223	0.259	0.238	0.217
CombSUM TOP 5	0.309	0.351	0.303	0.363	0.338	0.284
expCombSUM	0.348	0.389	0.374	**0.406**	**0.406**	**0.401**
expCombMNZ	0.364	0.395	0.382	0.395	0.401	0.398
CombSUM $RR^{\frac{1}{2}}$	0.258	0.281	0.261	0.295	0.280	0.262
CombSUM RR	0.307	0.334	0.315	0.346	0.334	0.321
CombSUM RR^2	0.367	0.395	0.384	0.391	0.392	0.394
sqCombMNZ	0.230	0.251	0.226	0.259	0.240	0.221
sqCombSUM	0.278	0.297	0.289	0.308	0.302	0.301
sqCombSUM $RR^{\frac{1}{2}}$	0.320	0.341	0.335	0.350	0.348	0.350
sqCombSUM RR	0.357	0.383	0.371	0.384	0.385	0.387
sqCombSUM RR^2	0.369	0.396	0.389	**0.400**	**0.408**	**0.405**
Votes	0.182	0.198	0.163	0.200	0.167	0.141
$RR^{\frac{1}{2}}$	0.327	0.348	0.341	0.349	0.352	0.358
$RR^{\frac{3}{4}}$	0.349	0.377	0.367	0.383	0.393	0.388
RR	0.349	0.383	0.370	0.392	0.398	0.394
$RR^{\frac{3}{2}}$	0.345	0.385	0.369	0.396	0.398	0.395
RR^2	0.344	0.383	0.368	0.398	0.398	0.393
CombMAX	0.340	0.382	0.368	0.398	0.396	0.392

100 terms dampens the quality of the results. The best results are achieved with the largest intended window size of 200 terms for most techniques, in particular, with a step size of 50 or 100 terms.

6.3.3 Summary: Passages as Voters

Essentially, a scenario in which passages vote for their documents, with the low average number per document of passages evaluated here, seems only suitable to a limited extent. Aggregations cannot express their strengths in the INEX 2009 and Robust04 scenarios (6.3.1,6.3.2) and yield worse results than the basic BM25-based document search, which best reflects the relevance of the documents with its factors idf, tf and integrated document length normalization. In these scenarios, direct and indirect quantity-based methods such as Votes and CombSUM perform very unsatisfactorily and are not suitable because evidence-based aggregation is of limited effectiveness here.

However, techniques that already deliver very good results in the evaluations of Chapter 5 and whose resulting values are close to the BM25-based results should be noted. CombSUM RR^2 and sqCombSUM RR^2 deliver very good results alongside expCombSUM, which come close to the BM25-based document search.

6.4 Investigations Using Pseudo-Collections

Pseudo-collections are composed of virtual documents that contain passages and term distributions.

We start from the hypothesis that the effectiveness of voting techniques strongly depends on the distribution of search terms and topics throughout the collection based on articles or passages. Especially with long documents in which only one or a few passages are relevant, a basic search over complete documents with document length normalization does not yield best relevance rankings. To make the different systemic behavior of voting techniques and flat BM25-based search plausible, we have created a routine for generating pseudo-collections of documents having passages and different term distribution structures. A subsequent search and evaluation demonstrate the systemic behavior of the applied voting techniques.

6.4.1 Experimental Setup

According to the previous Section 6.3, we continue research on the passages that vote for their associated documents in this section. Using an artificially generated collection and a comprehensively structured scenario, we examine the systemic behavior of the applied voting techniques. The pseudo-collection consists of documents with a static static length, each having a fixed number of passages. We assume a query with a single term. This term is distributed differently across the documents and their passages.

Table 6.9 shows the structure of the pseudo-collection. The query term is included in the collection 871 times in total and is distributed differently across the 50 documents with their 500 passages.

Table 6.9: Composition of documents and passages for the pseudo-collection

Structure of the pseudo-collection	
#Documents	50
Document length (terms)	2,000
Passage length (terms)	200
#Passages per document	10

Each document has a different term burstiness, regardless of the frequency of the query term.

For each document, the term burstiness changes as a parameter for the frequency of its recurrence in each document and passage.

Figure 6.2 shows the distribution of the query term in the 50 documents from two perspectives. The upper bar chart shows the frequency of the query term for each of the 50 documents on the vertical axis. The documents are sorted in ascending order on the horizontal axis according to this frequency. The colors of each bar

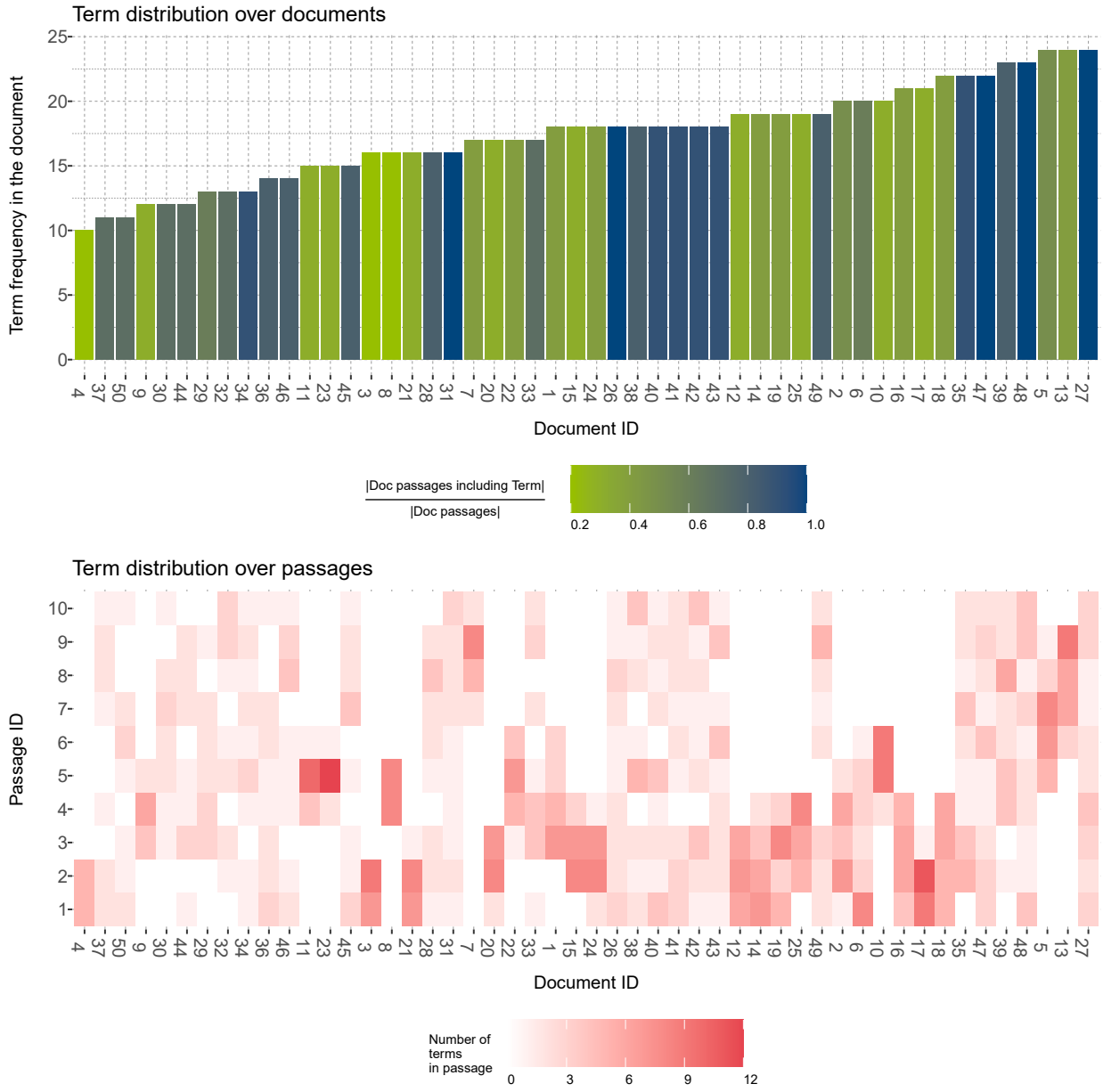

Figure 6.2: Virtual Documents with their query term distributions: The upper bar chart shows the documents with their number of query terms. Colors from green to dark blue indicate a more concentrated to more uniform distribution of the query term across the passages. The lower heatmap breaks down the number of terms for each passage. The white areas show the absence of the query term, while the red areas with their intensity indicate the number of terms for each passage.

are the result of the distribution of the query term in each document. If the search term is distributed relatively evenly across the passages of the document, the quotient of passages that contain this term and the total number of passages in a document is approximately 1. We define this quotient as the term passage distribution:

$$Term\,passage\,distribution(TPD) = \frac{\#Doc\ passages\ including\ Term}{\#Doc\ passages} \tag{6.1}$$

Documents or bars whose colors tend towards dark blue show an even distribution of the query term, while lighter green bars indicate a concentration of the query term on a few passages.

The second diagram in Figure 6.2 shows on the horizontal axis the documents in the same sorting in relation to the 10 passages, shown on the vertical axis. The heatmap shows the concentration of the query term for each document and its passages with a more or less intense shade of red. For passages that do not contain the term, a white area is displayed within the heat map. Depending on the frequency of occurrence, the tone for each passage is displayed in a light or dark red color.

After having created a collection with the given structure, we send a request to the collection with the query term whose distribution over the documents and passages we have defined. For each of the 50 documents, we compare the results of the search on all documents using BM25 with those of the passage voting using selected voting techniques.

6.4.2 Experimental Results

For our search over the pseudo-collection we applied the following voting techniques:

- CombSUM
- CombMAX
- CombSUM RR^x having different variants of x
- Votes

Figure 6.3 shows a comparison of the results for the voting-based application of CombSUM and a flat document search using BM25. The horizontal axis displays the rank for a document resulting from its voting passages, whereas the vertical axis represents the ranking results for the flat BM25 based search over the whole documents. The canvas contains all 50 documents arranged according to their ranking in both search techniques.

Case-related comparison of the results of passage voting with those of the document search

Documents near the bisecting line show almost identical ranking behavior for document search and voting passages. Documents lying above the bisector are rated better by the voting passages, while documents further to the right of the bisector yield higher ranks when applying document-based search.

The color of the documents corresponds to the color shown in Figure 6.2 and is the result of the passage distribution of the term. Documents colored darker blue have a relatively even distribution of the query term.

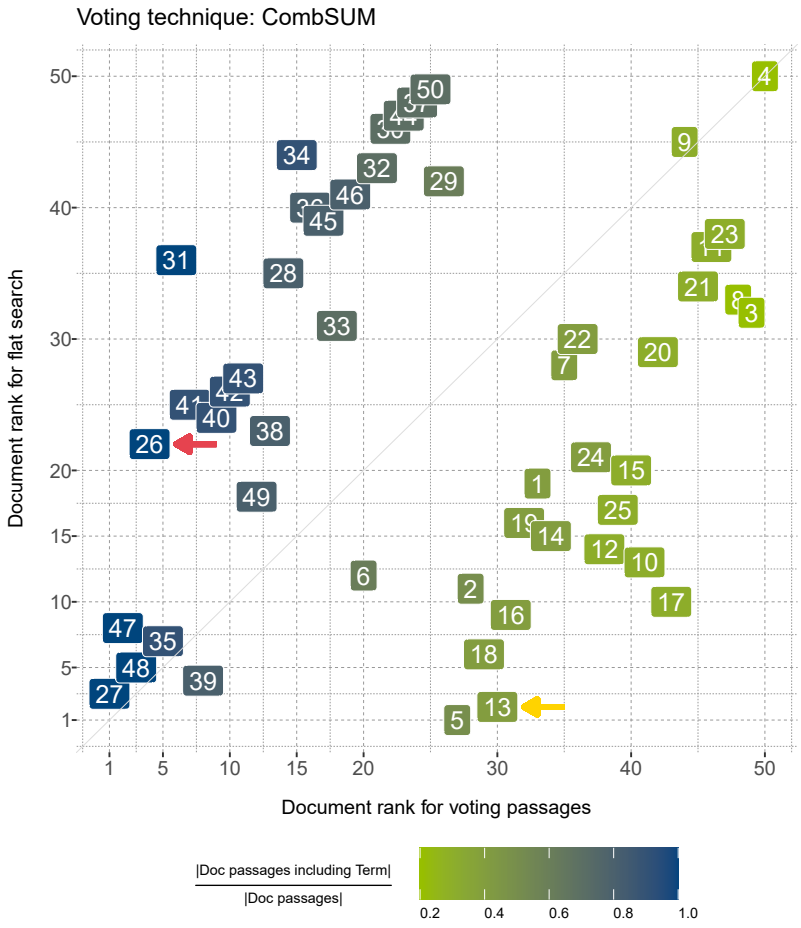

Figure 6.3: Comparison of the document ranking for the BM25-based flat document search and passages that vote for their associated document applying CombSUM.

Documents with a higher TDP of the query term benefit from the use of CombSUM.

CombSUM Figure 6.3 shows that documents having a relatively uniform distribution of the query term reach the upper ranks when using CombSUM. In contrast, in document-based search, documents with the highest number of hits are ranked in the top ranks, regardless of the distribution of the query term. While the distribution of terms plays a crucial role in the passage voting, the BM25-based search evaluates the entire document. For example, a document with ID 26 (see ←) that incorporates the query term 18 times has position #4 for passage voting, while it only has position #22 for the flat document search. Looking at the document that has ID 13 (see ←), we can observe a very low ranking for the passage voting, but for the BM25-based scoring it ranks #2. In

addition to the number of terms requested in the document, their distribution in the passages also plays a decisive role for the results of CombSUM.

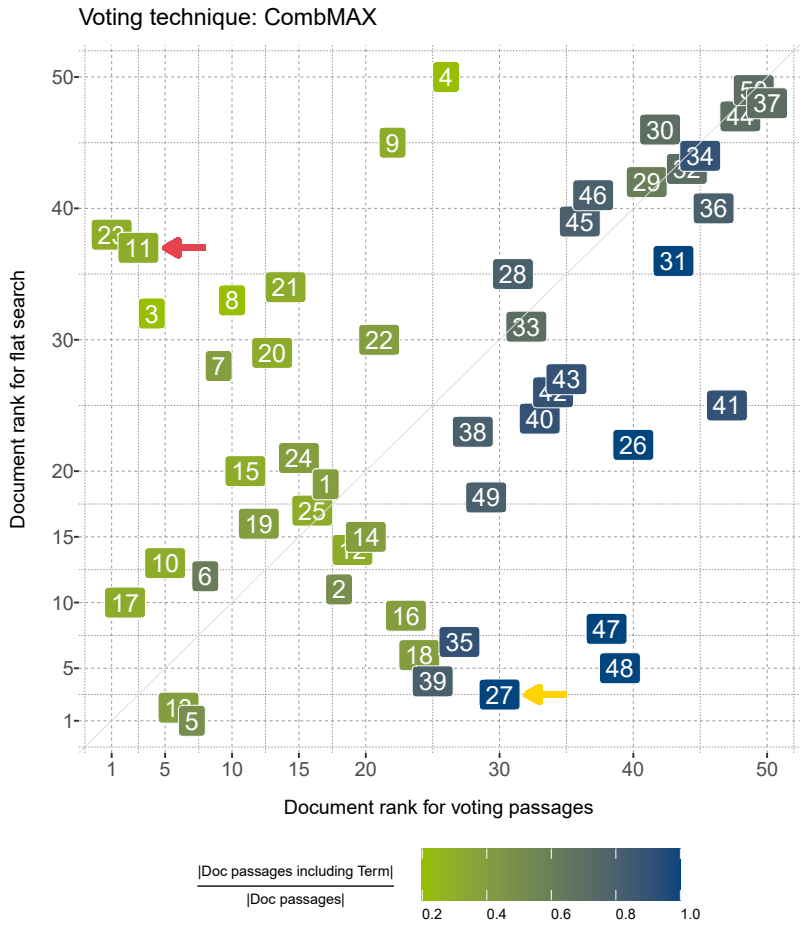

Figure 6.4: Comparison of the document ranking for the BM25-based document search and passages that vote applying CombMAX.

Documents with a lower TDP of the query term benefit from the use of CombMAX.

CombMAX Figure 6.4 shows the relationship between BM25-based document rankings and the rankings produced by passage voting using CombMAX. Applying a voting technique that considers only the best-scored passage tends to yield the opposite result: CombMAX prefers documents with a high term density and low distribution over the passages. The document with ID 11 (see ←) has only 15 term occurrences, but is ranked #3 for CombMAX due to its dense term distribution, while it is ranked #37 in the document search. In contrast, in document 27 (see ←) the query term is frequently present, which causes the document to be ranked at position #3 in the document search, but the relatively uniform distribution causes a ranking at position #30 in the CombMAX example.

Document ranking using CombMAX represents a combination of a high number of searched terms and a dense occurrence of

them. Positive if both factors interact; otherwise, there happens to be a trade-off between the two, and the document is ranked accordingly.

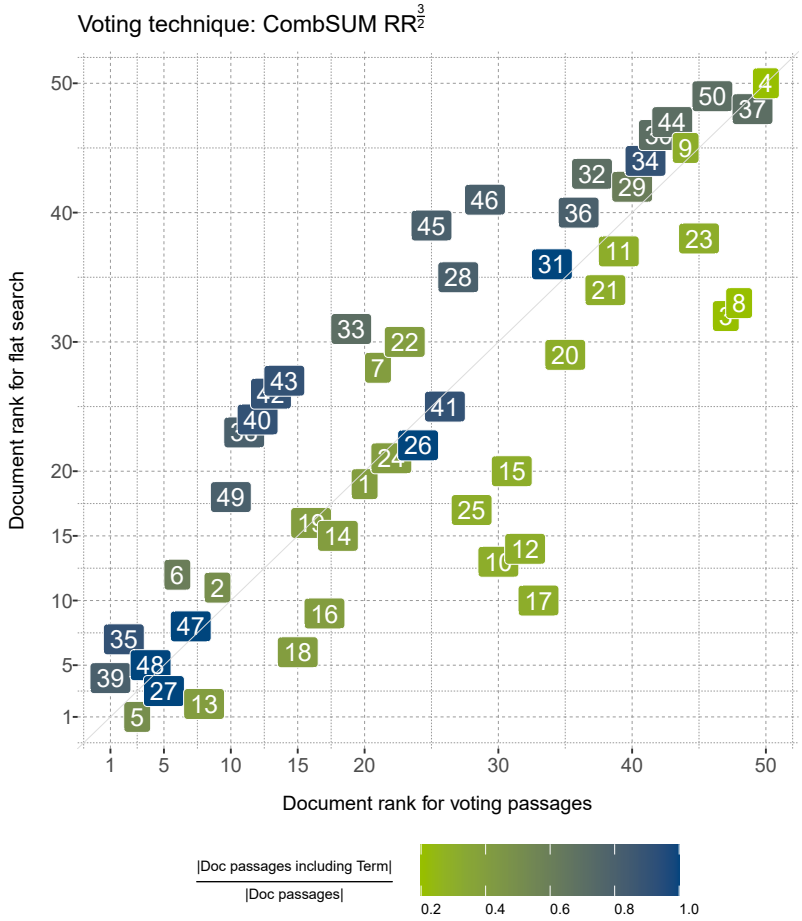

Figure 6.5: Comparison of the document ranking for the flat search and passages that vote applying CombSUM $RR^{\frac{3}{2}}$. Results from previous examples of CombSUM and CombMAX lying opposite to and further away from the angle bisector move towards it here and cross it to the other side.

Depending on the value of x, the results range between those of CombSUM and CombMAX.

CombSUM $RR^{\frac{3}{2}}$ Figure 6.5 shows the results of CombSUM $RR^{\frac{3}{2}}$ compared to the BM25-based document search, corresponding to the two previous diagrams. According to Figure 4.11 on page 63, the results of CombSUM RR^x with a small x approximate those of CombSUM, while a large value for x approximates the results of CombMAX. In the case of $x = \frac{3}{2}$, Figure 6.5 shows how the results with high and low TPD, which are found mostly on opposite sides of the angle bisector in the previous examples, are moving toward it and even crossing it to the other side.

In the comparison shown for the BM25-based document search, the results swap the sides of the angle bisector with increasing exponent. With a low exponent x (results approximate those of

CombSUM), documents with a high TPD are still to the left of the dividing line; as the exponent increases, they migrate to the right and swap positions with documents with a low TPD.

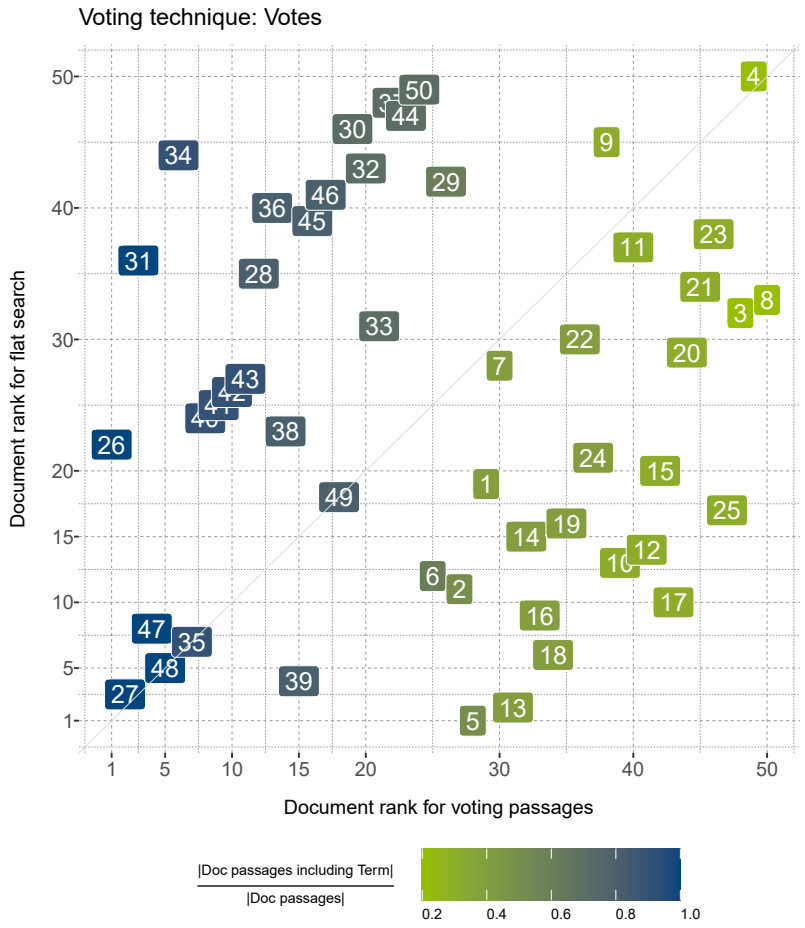

Figure 6.6: Comparison of the document ranking for the flat search and passages that vote applying Votes.

Votes In the context of passage voting, Votes does not qualitatively evaluate the number and frequency of the query term, but uses the number of passages containing the query term as an evaluation criterion. In the examples from Section 6.3, Votes has been shown to be unsuitable for use in passage voting. This is particularly true if the documents are only divided into a few passages and if the query terms are distributed coherently across documents and passages. The results when using Votes tend to be similar to those of CombSUM (Figure 6.3) and are shown in Figure 6.6. The scattering of documents in this figure is even more diffuse than in the CombSUM example and therefore does not show any connection or correlation with the document-based results.

Ranking as a function of the number of terms and their distribution Figure 6.7 shows the document rankings for the voting

Figure 6.7: Results of the rankings for the flat document-based search (BM25) and selected voting techniques as a function of the number of terms in the document and term distributions over passages. The horizontal axis shows the number of terms in the virtual documents, the vertical axis their distribution over passages, expressed as the TDP value. Green shades indicate good results in the ranking, worse results are shown in the transition from white to red.

techniques discussed as a function of their term frequencies and their distributions. The horizontal axis shows the number of terms per document, while the vertical axis shows their term distribution as the TPD ratio. The colors define the ranking of the documents, with a saturated green indicating a good ranking and a variation from white to red indicating a worse ranking.

The document search, based on BM25, provides rankings that focus exclusively on the number of existing query terms and disregard

the distribution within the documents. This results in a coloring based entirely on the term distribution from green to red on the left. The example of CombSUM shows that documents that perform worse in document ranking can also be ranked in the top positions. For example, the document with ID 31 (see →) is ranked at the top, while it is ranked lower in the flat document search. In contrast, the document with ID 13 (see ↑), which was highly rated in the flat search, is rated lower.

As before, Votes shows a similar picture, with the good ratings focusing on documents with a high TPD ratio.

CombSUM $RR^{\frac{3}{2}}$ shows a balanced scoring for documents, taking into account both the number of terms in the document and the term passage distribution. The document with ID 13 (see ↑) scores high due to the number of occurrences of the query term, but also documents with a lower number of term occurrences of less than 20 with a correspondingly high TDP get positions on the higher ranks.

In the case of CombMAX, documents with a low term passage distribution are highly rated. For example, a document with ID 39 (see ←) is rated low regardless of the high occurrence of the query term. The even distribution of the query term worsens the rating by CombMAX.

6.4.3 Summary: Investigations Using Pseudo-Collections

The scenario shown is a simple virtual scenario. If several query terms with different distributions are considered, the results overlap and are no longer as clear and easy to visualize. However, this analysis can show a basic trend. The coherent distribution vs. the incoherent distribution of terms within the documents or their passages contain arguments for or against the use of voting techniques. In the case of coherently structured documents, variants of CombSUM RR^x as a voting technique can also produce good results, such as the flat BM25-based document search.

7 Voting Techniques Applied to Clustering Algorithms

From similarity measures to distance measures

This chapter investigates the application of voting techniques based on distance measures in the context of hierarchical agglomerative clustering (HAC). This variant of clustering starts with the calculation of a distance matrix of all points, which are then merged step by step. It has been researched and discussed from the 1950s to the present; earlier examples include contributions by Anderberg [And73], Sneath and Sokal [SS73] and Everitt [Eve74].

The basic idea in this thesis is that the distances from the points of one cluster to the points of a second cluster are considered as voting candidates. The distance between both clusters is calculated by applying voting techniques.

Section 7.1 Correspondence of Voting Techniques to Agglomerative Clustering Algorithms:

Three best known methods in the context of agglomerative clustering and their logical equivalents in the domain of voting techniques are presented in Section 7.1

Section 7.2 Extended Transfer of Voting Techniques to Agglomerative Clustering: Transfer of voting techniques based on a generic approach

Section 7.2 extends this scenario and introduces additional voting techniques that are transferred to agglomerative clustering. Starting from a generic approach, we develop techniques based on CombSUM TOP n and CombSUM RR^x.

Section: 7.3 Clustering Data Sets Using Voting Techniques: Evaluation based on two-dimensional geometric data sets

In Section 7.3, the methods from Section 7.2 are applied and evaluated based on known geometric data sets.

Section 7.4 Summary of Applying Voting Techniques to Agglomerative Clustering: Summary and visualization of relationships

The chapter concludes with a summary of voting techniques applied in the domain of agglomerative clustering and how they relate to each other in Section 7.4.

7.1 Correspondence of Voting Techniques to Agglomerative Clustering Algorithms

Euclidean distance as distance metric

The agglomerate variant of hierarchical clustering is based on distance metrics that define dissimilarities or distances between partial clusters. By employing a chosen distance metric, the distance between temporary clusters can be measured on the basis of linkage criteria to merge the closest two in a clustering step. In our work, we apply the Euclidean distance as a metric that defines the distance d between two points a and b in a n-dimensional space as

$$d(a,b) = \sqrt{(a_1 - b_1)^2 + (a_2 - b_2)^2 + \ldots + (a_n - b_n)^2}. \tag{7.1}$$

Individual distances act as voting candidates

In order to establish correspondence with voting techniques, individual distances from the points of one cluster to the points of another cluster are considered voting candidates. The distance between two clusters results from the applied voting technique, which determines a total distance from the individual distances.

When applying linkage or voting techniques to merge partial clusters, it is insignificant whether partial clusters consist of one single point or multiple points. Three main linkage criteria (single-, complete- and average-linkage) are common techniques used in hierarchical clustering and considered in this work.

The basic scenario in the following examples is made up of a number of clusters that are successively merged. In each step of the clustering process, the two clusters that are to be merged next are searched for as a pair. When each pair is checked for merging, the points of the two clusters as a pair serve as voters who vote in fusing the clusters.

The following sections first establish logical correspondences between three existing linkage criteria and voting techniques, followed by an extension of this scenario transferring additional voting techniques to agglomerative clustering in Section 7.2.

7.1.1 Single-Linkage

Single-linkage corresponds with CombMIN

Single-linkage determines the distance between the cluster instances as the minimum distance between the associated points of each cluster. The distance D between the cluster A and B is defined as the minimum distance value d between two points a and b:

$$D(A,B) = \min_{a \in A, b \in B} \{d(a,b)\} \tag{7.2}$$

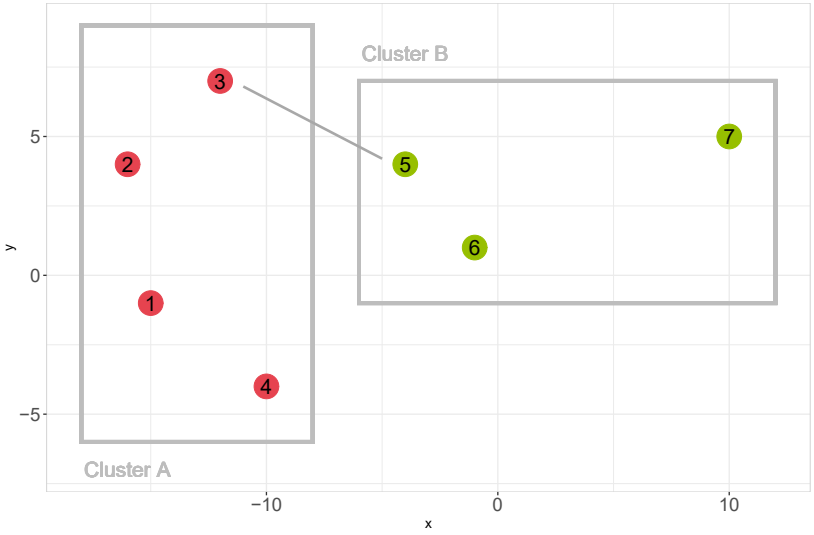

Figure 7.1: Single-linkage: The distance between cluster A and cluster B is determined by the smallest distance between the clusters' points.

With respect to Figure 7.1 the distance of clusters A and B would result from the distance of the points 5 and 3.

A typical characteristic of single-linkage is the chaining effect, in which points along a chain can be connected to each other, even if they belong to different clusters. Single-linkage is therefore sensitive to slight outliers.

In his book *Cluster Analysis*, Everitt comments on the clustering characteristics of this method:

"*Possibly chaining has been regarded as a defect because the majority of users are looking for homogeneous, compact clusters, although in general there is no reason to believe that these are the only types of of structure present in their data. However, because of the chaining effect single linkage may fail to resolve relatively distinct clusters if a small number of intermediate points are present between the clusters*" [Eve74, sec. 3.5]

As we are turning from previous considerations of similarity measures to distance measures in this domain, single-linkage corresponds to CombMIN as voting technique (see also Section 2.3.1 on page 33).

7.1.2 Complete-Linkage

Complete-linkage corresponds with CombMAX

In this procedure, the maximum distance between two points of clusters A and B is used to determine the distance as a linkage criterion.

Regarding Figure 7.2, the distance from both clusters would result from points 7 and 2, which are the points of each cluster having the

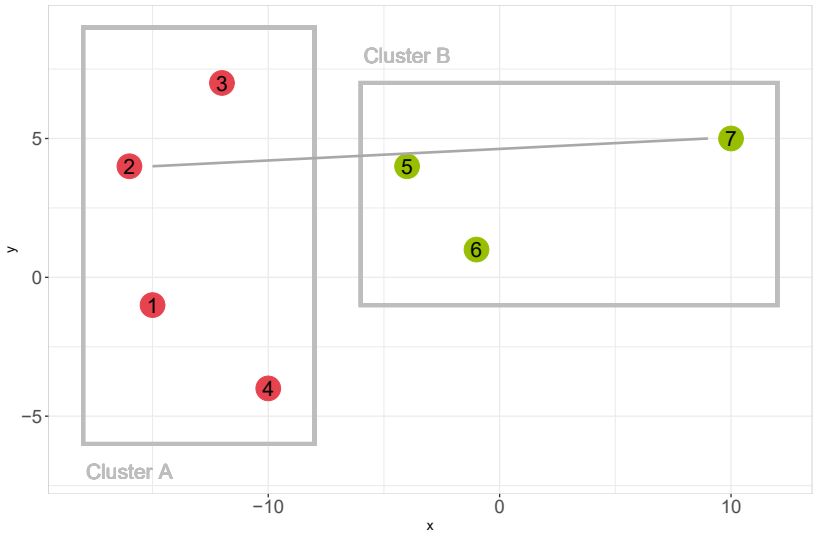

Figure 7.2: Complete-linkage: The distance between cluster A and cluster B is determined by the highest distance between the clusters' points.

maximum distance. Expressed in a formula, complete-linkage defines the distance D between two clusters A and B as the maximum distance value d of points a and b:

$$D(A, B) = \max_{a \in A, b \in B} \{d(a, b)\} \tag{7.3}$$

Complete-linkage tends to create compact and spherical clusters, since it seeks to reduce the maximum distances within the clusters. In their book *Numerical Taxonomy: The Principles and Practice of Numerical Classification*, Sneath and Sokal write about the behavior of complete-linkage:

"*The Method will generally lead to tight, hyperspherical, discrete clusters that join others with difficulty and at relative low overall similarity values*" [SS73, sec. 5.5]

Taking into account only the maximum distance, we adopt CombMAX as the corresponding voting technique. CombMAX is introduced in Section 2.2.7 on page 29.

7.1.3 Average-Linkage

Average-linkage corresponds with CombANZ

This linkage criterion is based on the average distances between all combinations of points in the two clusters and is expressed as distance D between cluster A and B:

$$D(A, B) = \frac{1}{|A| \cdot |B|} \sum_{a \in A, b \in B} d(a, b) \tag{7.4}$$

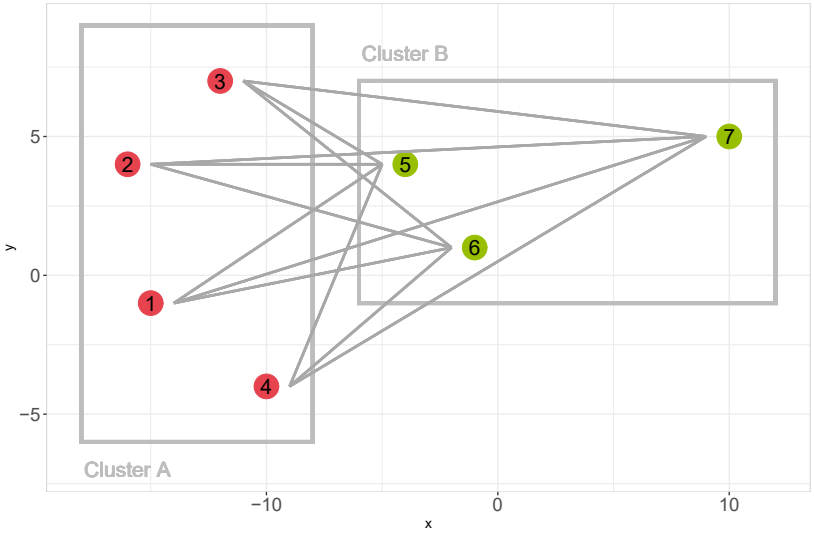

Figure 7.3: Average-linkage: The distance value of two clusters is determined by taking the average of all distance combinations from the two clusters' points.

In the domain of voting techniques, average-linkage has its equivalent in CombANZ which is defined in Section 2.3.3 on page 34.

7.2 Extended Transfer of Voting Techniques to Agglomerative Clustering

Linkage criteria must be invariant to the number of contributing points and distances.

In addition to the standard techniques discussed with correspondences to voting techniques, this section discusses further voting techniques applied to hierarchical clustering. As a prerequisite, we have to define that the results of an applied technique may not be based on the number of voting candidates. Calculated distances would be misrepresented if they were based on the number of cluster points. With this restriction, the use of Votes in this domain is not applicable.

7.2.1 Using the Weighted Average as a Generic Approach

A general approach to the distance-based application of voting techniques in agglomerative clustering is provided by the weighted average formula as an extension of formula 7.4.

Applying the approach according to the formula 7.5, each distance $d(a_i, b_j)$ is multiplied by an individual weighting factor w_{ij}. The sum of these products is then divided by the sum of the applied weights, resulting in the distance $D(A, B)$ for the (temporary) clusters A and B.

$$D(A,B) = \frac{1}{\sum_{i\in\{1...|A|\},j\in\{1...|B|\}} w_{ij}} \sum_{a_i\in A, b_j\in B} d(a_i,b_j)\cdot w_{ij} \qquad (7.5)$$

If all weights w_{ij} were set to the value 1, the results would be like those of the average-linkage or CombANZ formula.

Re-implementing CombSUM to CombANZ

The variants of CombSUM such as CombSUM RR^x or CombSUM TOP n are subsequently implemented as CombANZ RR^x and CombANZ TOP n. The average must be calculated at this point to achieve independence from the number of voting distances, which would otherwise be implicitly given by the summation of their values.

Sorting the distances according to their absolute value

To determine a weighting according to the reciprocal rank RR corresponding to a ranking, the distances must be sorted according to their absolute value before applying the formula – either in ascending or descending order. In the following, we refer to these variants as "**increasing**" or "**decreasing**".

Informal procedure for CombANZ RR^x The steps for implementing CombANZ RR^x are described below in an informal procedure:

Step 1: Sort all distances $d(a_i,b_j)$ between points of two clusters A and B in ascending or descending order. The result is a ranking of distances, starting with either the largest or the smallest distance.

Step 2: Multiply each distance $d(a_i,b_j)$ by the weight w_{ij}. In the case of RR^x, this weight is based on the rank in which the distance was sorted and is $(\frac{1}{rank(d(a_i,b_j))})^x$.

Step 3: Calculate the sum of all weighted distances.

Step 4: Divide the resulting sum by the sum of all weights. The result is the distance between clusters A and B calculated on the basis of CombANZ RR^x.

7.2.2 Transferring further Voting Techniques

To get an overview, we consider both score-based and ranking combined methods. As a score-based method, we evaluate CombANZ TOP n and for a combined application, we look at variants of CombANZ RR^x. Each of these techniques can be applied in two different ways: CombANZ TOP n can be implemented starting with the n smallest distances or the n highest distances between two temporary clusters. Consequently, CombANZ RR^x can be applied

by starting the potentiation with the highest or lowest measured distances. In summary, the following techniques are evaluated:

- **CombANZ TOP n (increasing)** – This technique considers the top n smallest distances in increasing order. Setting $n = 1$ results in the application of single-linkage or CombMIN, while the highest number of n leads to an average-linkage that expresses CombANZ.
- **CombANZ TOP n (decreasing)** – Here the top n highest distances are considered in decreasing order. Setting $n = 1$ yields the same results as applying complete-linkage or CombMAX. By increasing the values for n, we approximate the average-linkage that stands for CombANZ.
- **CombANZ RR^x (increasing)** – This technique considers all voting distances. Sorted in increasing order, the distances of each cluster are weighted by their reciprocal rank potentiated with x. Applying exponents which converge to 0 we obtain results which correspond to average-linkage (CombANZ) whereas high values for x yield results approximating single-linkage which corresponds to CombMIN.
- **CombANZ RR^x (decreasing)** – This technique considers all voting distances. Sorted in descending order, the distances of each cluster are weighted by their reciprocal rank potentiated with x. Applying exponents which converge to 0 we obtain results which correspond to average-linkage (CombANZ) whereas high values for x yield results approximating complete-linkage which corresponds to CombMAX.

We apply these techniques to two-dimensional data sets introduced in the next section.

7.3 Clustering Data Sets Using Voting Techniques

7.3.1 Noisy Blobs Data Set

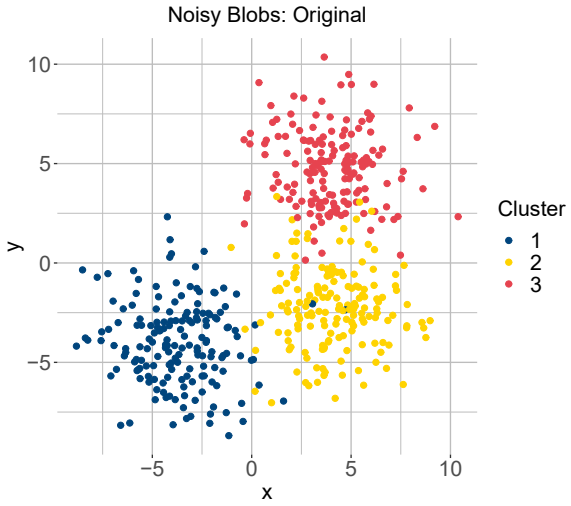

Figure 7.4: Original of the Noisy Blobs data set having three clusters and consisting of 500 observations.

This data set is taken from scikit learn[1] and represents three clusters as isotropic Gaussian blobs. Figure 7.4 shows the original data set that has three clusters and consists of 500 points.

1: Shape data is created using scikit learn which is documented under https://scikit-learn.org/stable/modules/generated/sklearn.datasets.make_blobs.html

Given the number of observations $n = 500$, the maximum possible value of the voting distances can be expressed as

$$\max_{n=500} |Voting\ distances| = \left(\frac{500}{2}\right)^2 = 62,500 \tag{7.6}$$

CombANZ TOP n (increasing) Figure 7.5 shows the results of the increasing variant of CombANZ TOP n. For all results, the output of *three* clusters was also adopted according to the original. The plot on the upper left shows CombANZ TOP 1, which in the increasing variant is synonymous with single-linkage. The chaining effect, typical for single-linkage [MRS08] can be observed here, which results in one large cluster in blue and two negligible clusters in the output (yellow and red). This effect continues for increasing n (plot 2 having $n = 1,000$ and plot 3 having $n = 12,000$) yielding varying small clusters, before two large clusters emerge at approximately $n = 13,000$ (lower left plot 4). Having a value for n exceeding about 13,500 voting distances, a final stage is reached where a stable identification of the three clusters is established and the result corresponds to CombANZ or average-linkage.

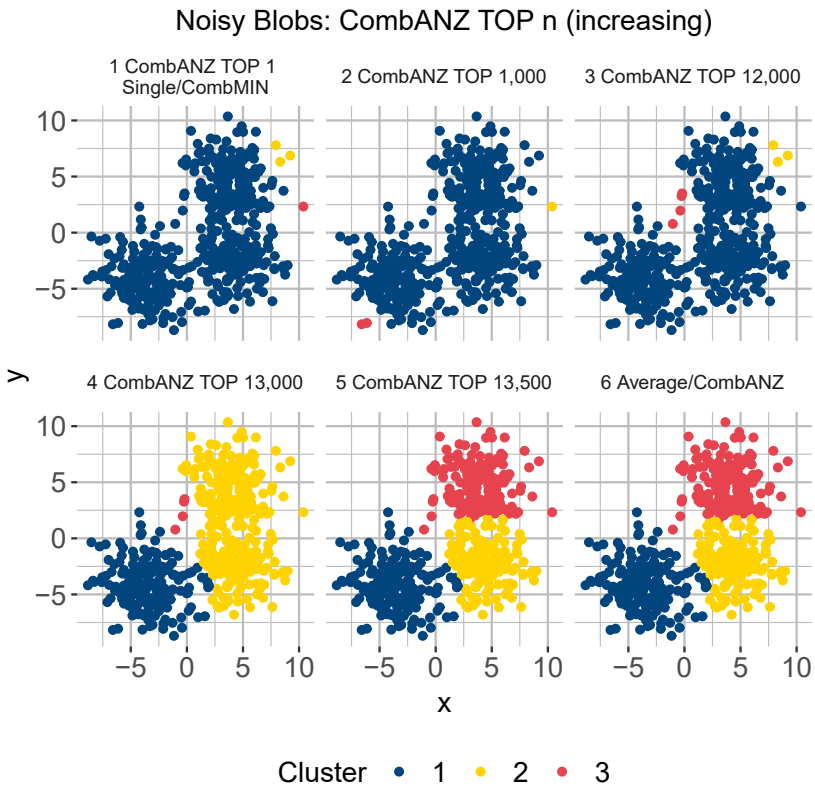

Figure 7.5: Increasing CombANZ TOP n techniques applied to Noisy Blobs data set. A stable identification of the three clusters is reached with high values for n, approximately $n = 13,500$ shown in plot 5.

CombANZ TOP n (decreasing) Figure 7.6 shows the results for the decreasing variant of CombANZ TOP n. In the decreasing variant, CombANZ TOP 1 corresponds to CombMAX or complete-linkage. All three clusters have already formed roughly here (plot 1). As n progresses, small differences arise (plots 2-4), but by $n = 5,000$ at the latest, no more deviations from the approximated CombANZ or average-linkage variant can be observed (plots 5 and 6).

Regarding the Noisy Blob scenario, the two techniques CombANZ TOP n and CombANZ RR^x in the descending variants show comparable results. The first mentioned technique converges from CombMAX to CombANZ, whereas the second technique, beginning with $x = 0$ - converges in the opposite direction between those. Starting from $n = 1$, which stands for CombMAX, there emerge slight shifts between the three identified clusters when increasing n.

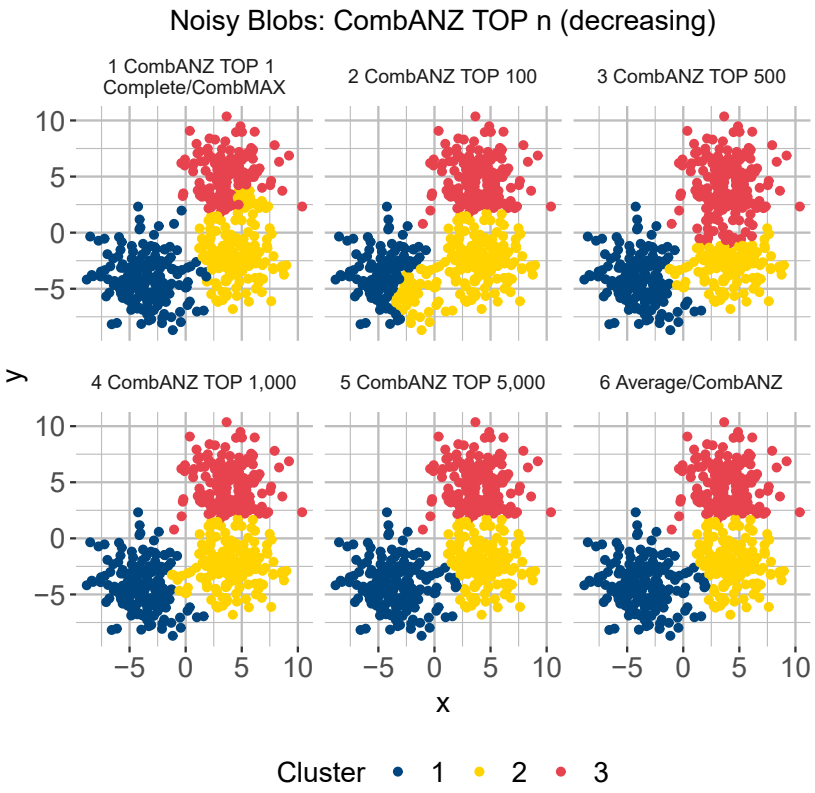

Figure 7.6: The decreasing variant of CombANZ TOP *n* shows detection of three clusters throughout all variations of *n*. A stable status is reached when *n* exceeds the limit by about 5,000 (plot 5).

7.3.2 Factoextra Multishapes

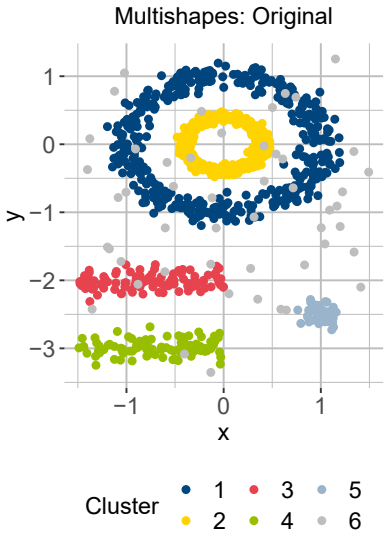

Figure 7.7: Original of the Multishapes data set having five clusters and per definition an extra cluster which defines the noise. The data set consists of 1, 100 observations.

As a second scenario to apply the voting techniques introduced in

Section 7.2, we choose the Multishapes data set from the R package factoextra[2].

2: Shape data is one component of the R package factoextra which is documented under https://www.rdocumentation.org/packages/factoextra/versions/1.0.3/topics/multishapes

It consists of five geometric shapes that are enriched with noise. Having 1, 100 observations, the maximum number of voters is 302, 500 according to equation 7.6.

The original cluster is shown in Figure 7.7. In our investigations, we neglect the additional 6^{th} cluster that defines the noise. Keeping the noisy points, we specify 5 clusters as a result.

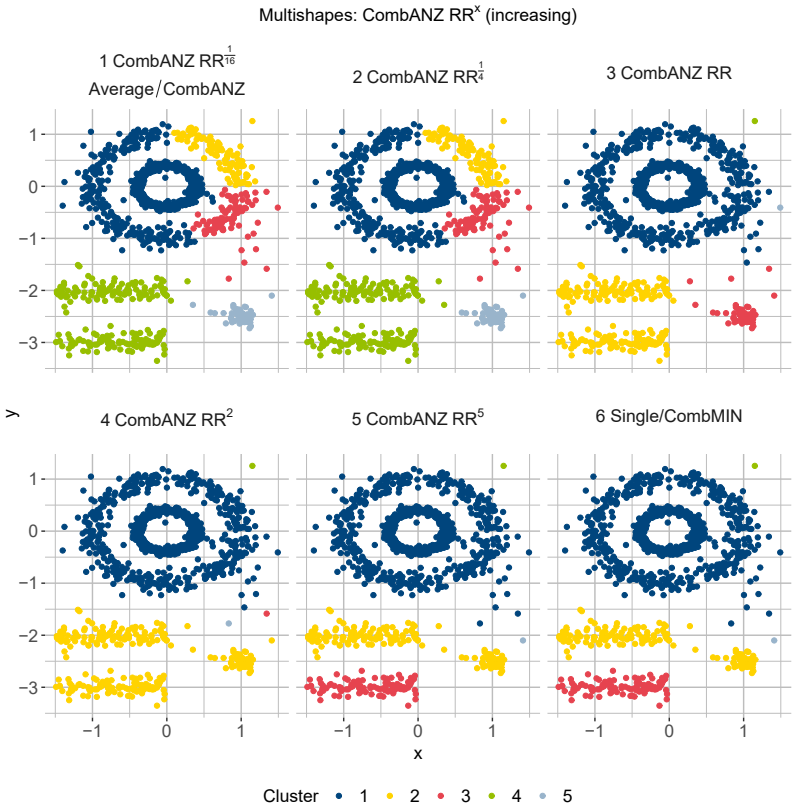

Figure 7.8: Results of clustering after applying CombANZ RR^x with different values for x having the distances arranged in increasing order. Changes in the lower geometries arise for values of x between 1 and 5.

CombANZ RR^x (increasing) Figure 7.8 shows the clusterings for technique CombANZ RR^x when sorting the distances of (temporary) clusters in ascending order. Applying the parameterization having $x = \frac{1}{16}$, the first and smallest voting distance is kept with its value, the other distance values are slightly damped according to their ranking. The plot 1 in Figure 7.8 corresponds to the result of average-linkage or CombANZ due to only slight dampening. With $x = \frac{1}{4}$ we still get the same result in plot 2, with further increasing values of x, the results of the lower three geometries – the two horizontal clusters and the point-shaped cluster – change. With $x = 5$, the distance values following the smallest distance are dampened

to such an extent that the result is the same as with single-linkage or CombMIN, as can be observed in the plots 5 and 6 of Figure 7.8.

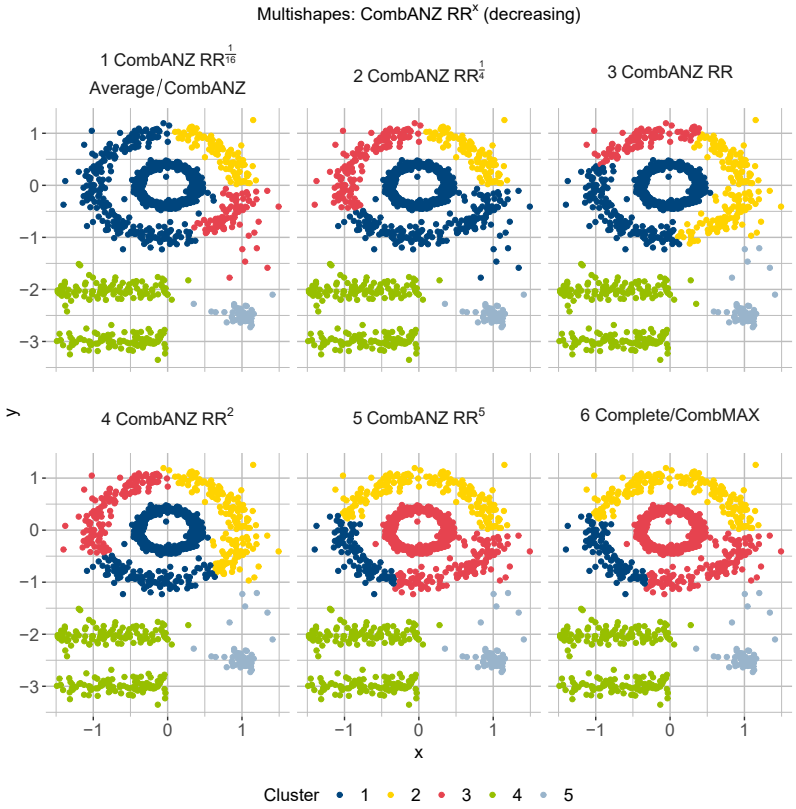

Figure 7.9: Results of clustering after applying CombANZ RR^x with different values for x having the distances arranged in descending order. The lower horizontal geometries – in the original two clusters – cannot be separated across all parameterizations of x. The circles cannot be separated either and are structured in different variations across three clusters (blue, red and yellow).

CombANZ RR^x (decreasing) Figure 7.9 shows the application of CombANZ RR^x in descending order of distance. With the very small exponent $x = \frac{1}{16}$ in plot 1, the clustering result still corresponds to the standard technique average-linkage or the CombANZ voting technique. The two circles are structured into three clusters here, which does not change as x progresses. Similarly, the two horizontal clusters cannot be separated across all values of x, but remain combined in a green cluster. With $x = 5$ at the latest, the result of CombANZ RR^x in Plot 5 no longer changes and provides the same result as complete-linkage or CombMAX in Plot 6.

CombANZ TOP n (increasing) The technique CombANZ RR^x in its decreasing variant yields results that lie between average-linkage and, with increasing values of x, complete-linkage. Figure 7.10 with CombANZ TOP n (increasing) provides results that lie

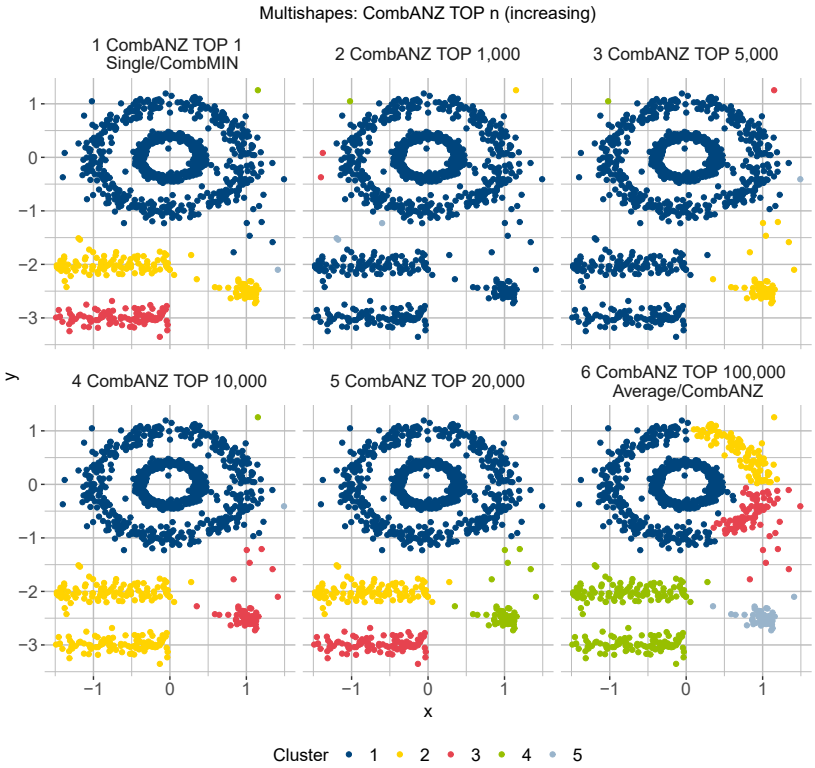

Figure 7.10: Results of using CombANZ TOP n for different values of n when sorting the distances in increasing order. For $n = 1$ the results are the same as for single-linkage or CombMIN (plot 1), for high n the results are the same as for average-linkage or CombANZ (plot 6). The lower three geometries can be correctly classified into clusters with a parameterization of $n = 20,000$, shown in plot 5 (yellow, red and green).

between single-linkage and, with increasing n, average-linkage. In plots 1 to 5, the two round geometries are returned only as one single cluster in blue, the lower classifications alternate with different values for n. CombANZ TOP n with having $n = 20,000$ is the only example that returns three clear clusters for the lower geometries in yellow, red, and green, as shown in the plot 5. As n increases further, results are achieved that correspond to those of average-linkage or CombANZ, shown in plot 6.

7.4 Summary of Applying Voting Techniques to Agglomerative Clustering

This chapter presents approaches to transfer voting techniques to agglomerative clustering. Based on the weighted average, voting techniques that are value- and/or rank-based can be transferred to distance-based agglomerative clustering methods. Using two

examples, we demonstrate the results that lie between the results of the three linkage techniques presented, single-, complete-, and average-linkage, due to their parameterization.

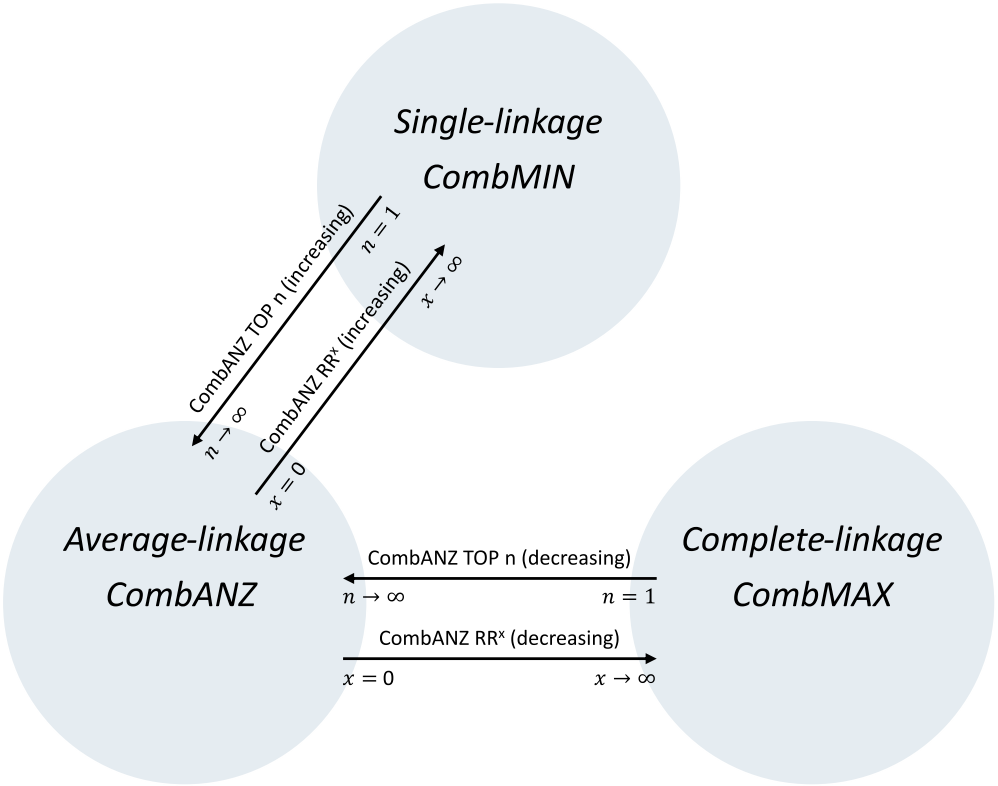

Figure 7.11: Voting techniques applied to agglomerative clustering with their parameterization: Applying variants of n or x, results are obtained that lie between those of the three known clustering techniques.

Small changes in the parametrization of n or x can cause fundamental changes in clustering results.

Figure 7.11 shows, how the presented implementations of voting techniques deliver results through their parameterization that are between the three known clustering methods. The results of the example data sets examined show that a parameterization that changes for n or x in small steps produces either no, small, or even fundamental changes in the results from step to step. No stepwise correspondence of the results based on parameterization does emerge; small changes can already fundamentally change the cluster properties. The clustering results shown in Figure 7.5 show an example of this effect: In the CombANZ TOP n (increasing) variant, two large and one negligible cluster can still be recognized at 13,000 voters (plot 4), while within an increase of $n = 500$ the clustering fundamentally changes to three large clusters (plot 5). This is due to changes in the distance calculations in the first clustering steps, which have a fundamental effect on following earlier temporary clusters, and thus the resulting three clusters.

Although the clustering results do not change uniformly according to parameterization, variants of the voting techniques that lie between the three clustering techniques considered can provide

better or more meaningful results. An example of this can be seen for the Multishapes example in Figure 7.10. Using CombANZ TOP n (increasing) with $n = 1$ (single-linkage or CombMIN), three large clusters are formed as a result. As n increases, one ($n = 1,000$), two ($n = 5,000$) and three ($n = 10,000$) clusters are formed, respectively, while $n = 20,000$ produces the most visually meaningful result for geometric groups having four clusters, shown in the graph 5 (yellow, red, green, and blue).

8 Conclusion

In this chapter, the results of previous chapters are summarized and reviewed with reference to the contribution of the thesis, described in Section 1.5.

The conclusion begins with a look at the applied evaluation strategy, on which the results are based. In Section 8.1 we summarize the advantages and disadvantages of the evaluation techniques applied from Chapters 5 and 6, with their differences from the evaluation techniques in expertise retrieval.

Section 8.1 Evaluation of Voting Scenarios: Summary and comparative analysis of the evaluation techniques

In Section 8.2 we summarize the factors that influence the performance of voting techniques and give a final assessment of the methods that yield stable and performant results.

Section 8.2 Influencing Factors for the Successful Application of Voting Techniques: Summary of influencing factors and highlighting stable and high-performing voting techniques

In Section 8.3 we summarize the exemplary studies on passage-based voting from Chapter 6 and give an assessment of the application of this technique.

Section 8.3 Passage Voting: Summary and assessment

The application of voting techniques in the domain of agglomerative clustering is summarized in Section 8.4.

Section 8.4 Voting Techniques in the Domain of Agglomerative Clustering

A summarizing outlook with further tasks and possible applications in the domain of voting techniques is given in Section 8.5.

Section 8.5 Summary and Outlook: Further tasks and applications

8.1 Evaluation of Voting Scenarios

In the domain of expertise retrieval, it has been stated that the search for experts almost focuses on the highest ranked results, is precision oriented, and therefore MRR is well suited as an evaluation measure [JKA16][Bal+12, sec. 4.1]. In correspondence to the scenario of expertise retrieval, we further assume in our evaluations that the classes listed in the middle- or lower-end of the ranking are generally of little interest to the user.

Class-based search as precision-oriented search focusing on the upper ranks

Establishing a test environment is the foundation for evaluation. This encompasses not only the collection, but also the queries employed and the relevance assessments of the ranked experts or classes.

Queries are equally distributed – thematically and across class sizes.

Query generation The basis for our evaluations are queries, also called topics in the context of TREC tracks, which were randomly extracted from the AMiner collection (see Section 5.1 Search Scenarios and Figure 5.1 on page 67). The evaluated queries are extracted evenly across class sizes (see Section 5.5 Systemic Behavior of the Applied Techniques and the baseline in Figure 5.5 on page 84). We also assume that the queries are equally weighted in terms of the topic distribution of the collection. With regard to the length of queries, all voting techniques benefit almost equally from longer queries. It should be noted at this point that the distribution of the query lengths in terms between 1 and 20 comes close to a normal distribution with a slight bias towards shorter lengths.

Compensation for ambiguities and uncharacteristic examples through a high number of requests

These queries from the evaluations in Chapter 5 Search for Classes - Evaluation of the Introduced Voting Techniques are not created manually and are therefore not potentially influenced by thematic preferences. Nevertheless, there may be inaccuracies, which in our experience are compensated for by the high number of requests: There may arise ambiguities in query terms. There are also a small number of queries that are not thematically profiled and are more general in nature. Another concern with respect to the evaluation in Chapter 5 is that we assume that the queries represent a typical example of the journal from which they are taken. We also accept this point and assume that exceptions are compensated for by the high number of queries.

No prior creation of class profiles required.

Generation of profiles In his thesis "The voting model for people search", Macdonald describes the automatic generation of expert profiles based on the references to candidates in the documents (see also the summary in Section 3.1.1) and the associated challenges [Mac09]. In contrast, the evaluation setup in Chapter 5 uses the given relationships of documents to classes, in this case of articles to journals. A generation of class profiles in advance is not required in the scenario of this thesis.

Relevance assessments Another factor influencing the evaluation results is the quality of the assessments. In general, the results of an expert ranking are not easy to assess [Mac09, sec. 3.4.5]: Manual assessments are difficult to realize in large numbers and can be incomplete due to insufficient information from the assessors and non-existent and incomplete assessments from potential experts. In their paper "Evaluation of Retrieval Algorithms for Expertise Search", Jayasinghe, Karimi, and Ayre highlight similar problems: Judges lack domain knowledge or knowledge of the person they are supposed to assess in terms of their expertise. Furthermore, the authors point out that there is a lack of motivation within an organization to classify and evaluate direct colleagues despite anonymization of the survey [JKA16].

Evaluation based on preexisting ground truth

In this thesis, we present two evaluation measures in Sections 5.2.2 Ranking of the Searched Journal and 5.4.3 Journal Relationships as Measure for Relevance that are not based on manual assessments, but on the given association of articles with journals, and thus on preexisting ground truth. The journal from which the query was taken is considered a suitable candidate in the form of a class in the scenario of this evaluation. We also look at the best-ranked journals or classes and their relationship to the journal searched for. The fact that in exceptional cases a journal title extracted as a query is not typical for this journal can be neglected due to the high number of queries. The second measure of evaluation, the relationship to the journal searched for, also proved to be significant for the top ranks. In the lower ranks, this measure no longer provides such meaningful results because of the low values and resulting low differences for the grades of relationship.

Relationships of journals only as an evaluation criterion for upper ranks

8.2 Influencing Factors for the Successful Application of Voting Techniques

The analyses in this thesis are based on custom scenarios and collections, as well as collections from the literature. Various collection-dependent structures are critical for the successful use of voting techniques.

8.2.1 Class Sizes and their Ratios

Votes and CombMAX as extreme cases

Influence of class sizes Influenced by the number of its voting documents and thus indirectly affected by its size, each of the evaluated voting techniques yields different results for a class and prefers large class sizes more or less. The limits in this context are variable, extending from CombMAX, unaffected by the potential number of voting documents per class, to Votes, which strongly yields results depending on class sizes. In the case of CombMAX, a single best voting candidate per class is often not sufficient to provide a relevant class ranking. Votes, on the other hand, often delivers too many hits per class, which are summed up with the same amount regardless of their score and deliver diffuse results. The ranking-based technique RR^x with individual parameterizations for x can provide a transition between CombMAX and Votes (see also Figure 4.11 on page 63 showing the relationships between voting techniques). This often produces qualitative rankings that yield very competitive results in the evaluations, but only because it approximates between the two techniques and provides an optimum between the two approaches. A general statement on the best parameterization cannot be made at this point. Figure 5.5 on

RR^x can produce optimized results in between the two techniques.

page 84 shows the systemic behavior in relation to class sizes for all voting techniques and their parameterizations evaluated.

Emphasis on scores of highly ranked documents

Compensation of class size ratios As the variability in class sizes within a collection increases, there is a greater need for the capability to mitigate the impact of larger classes. An extremely uneven distribution of potential voters among classes can cause a bias, which skews a class ranking towards large classes.

Techniques that emphasize the score of highly ranked documents at the document level mitigate the influence of large classes, in favor of the probative value of relevant, top-ranked documents – regardless of their class size. These techniques include expCombSUM, sqCombSUM, and, as discussed in the preceding paragraph, RR^x.

Limiting the number of voting documents at document level

Another way of compensating for the influence of large classes is to limit the number of initial voting documents. As this potentially affects large classes, this measure also reduces the influence and thus the bias that is caused. In this thesis, a restriction improved the performance of the affected techniques to a limited extent. Results in the example of the AMiner collection are shown in Section 5.5.3 Influence of Initially Voting Candidates on page 87, but in the literature this limitation only contributed to a limited extent to improving the results [Mac09].

Limiting the number of voting documents at class level

Limiting the voting documents at the *class level*, rather than document, is done by the CombSUM TOP *n* techniques and also leads to a reduction in the influence of large classes and thus to an improvement in the results, as stated in our evaluations. The results of CombSUM TOP 5, which are among the best results in our evaluations, show that this approach is appropriate.

Dampening the scores at class level as a stable and performant approach

The stable and often best results in our evaluations are provided by techniques that implement a damping function based on the harmonic series at the class level. These include CombSUM RR^x and sqCombSUM RR^x. Classes with few voting documents have a corresponding result against large classes whose influence is reduced by the damping function. However, a residual influence of the class size remains with sufficiently highly scored and valid documents. The major difference from the known voting techniques, which revalue the scores at the entire document level, is the processing at the class level. In line with the consideration of Section 5.6 Principle of Inclusion and Exclusion, the dependent probabilities for relevance within a class are not added together, but are added after this revaluation. This damping factor based on the harmonic series and its RR^x variants gives very good results in our evaluations.

8.2.2 Document Lengths and Information Value

Information content on query and document side Our evaluations based on the AMiner collection in Chapter 5 Search for Classes - Evaluation of the Introduced Voting Techniques are based on three scenarios: Searching over titles, abstracts and whole articles. Almost without exception, the search over abstracts provides better results for the applied voting techniques than the search over titles. Searching over the entire extracted articles can only increase the results to a limited extent, but this may also be due to the noise caused by extraction of the PDF documents and the inclusion of references. We make this observation both on the basis of the MRR (Table 5.2 on page 74) and for the journals in the first five ranks with their relationships to the searched journal (Table 5.4 on page 78).

Higher information content at document level improves the class ranking of all voting techniques.

We show a further correlation of the voting performance with the query lengths in terms in Section 5.5.2 Influence of the Query Length. All voting techniques benefit almost equally from longer queries and yield better values for the MRR (see Figure 5.7 on page 88). Due to the higher information content, document rankings with higher relevance are likely to be created, which also supports the relevance of the class rankings.

Longer query term lengths improve the results of the voting techniques equally.

8.3 Passage Voting

In Chapter 6 Passages as Voters for their Documents we look at passages that vote for their associated documents. In the first experimental Section 6.2 Virtual Documents, we first turn the scenario around and initiate a flat BM25-based search over very long virtual documents, concatenated from article titles, abstracts, and full texts from the AMiner scenario in Chapter 5. This BM25-based search over virtual documents yields far worse results for the searched journal than the voting-based search of the Chapter 5. Shorter concatenated documents, composed of smaller journals, are preferred here by document search, as can be seen in Figure 6.1 on page 108. The term specificity is likely to be higher for these shorter virtual documents, leading to preferred and thus better results for the smaller journals.

The inverse scenario of searching over concatenated AMiner articles confirms the choice of voting techniques.

In the experiments based on the collections INEX 2009 and Robust04 of Sections 6.3.1 INEX 2009 Collection and 6.3.2 Robust04 Collection, we have comparatively few passages per document. Here, voting passages work less well and do not reach the results of the BM25-based search using the original, complete documents at all or with difficulty. In case of shorter collection documents

In the passage scenario of shorter articles, selected voting techniques also yield good results.

and thus few voting passages per document, the aggregation model seems not convincing. Voting techniques such as CombSUM or Votes – explicitly or implicitly depending on the number of voters – yield worse results. Techniques such as expCombSUM, CombSUM RR^x or sqCombSUM RR^x with a higher value for x, emphasizing highly ranked passages and ranking their associated documents more relevantly, appear to be suitable to a limited extent in the two collection scenarios.

In summary, voting-based aggregation via passages in the INEX 2009 and Robust04 collections is only suitable to a limited extent. This is probably due to the following points: Documents are commonly shorter, and the number of passages per document does not vary much in both scenarios. In the case of the INEX 2009 collection, the passages are divided due to the markup, and thus also semantically, which explains the acceptable results of techniques such as expCombSUM, expCombMNZ, CombSUM RR^2 and sqCombSUM RR^2. All techniques emphasize the upper ranks of the passage ranking. In the case of the Robust04 collection, these techniques still work acceptably for larger window sizes with 200 terms. However, the principle of aggregation is not as successful with both collections as it is in the voting scenario of the AMiner collection. Here, the passages as articles are distributed very unevenly across the journals and also have a much sharper thematic focus.

8.4 Voting Techniques in the Domain of Agglomerative Clustering

Voting techniques as an approximation between known clustering techniques

In Chapter 7 Voting Techniques Applied to Clustering Algorithms, we use a generic approach, the weighted average, to show that aggregation techniques can also be applied to distance measures in the area of agglomerative clustering. In this scenario, previous considerations on the addition of probabilities and on aggregation of similarity values change.

The distance between two (temporary) clusters is calculated on the basis of the weighted average, and the influence of the distances considered is increased or decreased by the voting process, which determines the weighting factors. CombANZ TOP n and CombANZ RR^x can yield results that range between known techniques such as single-linkage or complete-linkage and thus more or less reflect their typical properties in the calculation of clusters.

8.5 Summary and Outlook

In this thesis, we investigate the characteristics of known and new introduced voting techniques and demonstrate a scenario for generalized transferability, which we test with custom evaluation approaches.

CombSUM RR^x and sqCombSUM RR^x as promising approaches

The new techniques presented in Chapter 4 Extended Techniques and Interrelationships show that approximations can be made between known techniques and that optimized results can be obtained. Based on probabilistic considerations and subsequent evaluations, we introduce CombSUM RR^x and sqCombSUM RR^x techniques that provide good results in all scenarios of this thesis and a promising approach for future use cases. For both techniques, the factor RR^x of the harmonic series is not only a method to approximate between different voting techniques, but also models a general, estimated addition of probabilities when overlaps between the probability values to be added are not known. Improvements are conceivable according to the approach described by Cummins, Lalmas, and O'Riordan [CLO10] and summarized in Section 3.2. In this scenario, with effort and to a certain extent, the weighting factors for the voting documents of experts are individually determined based on the expert profiles.

Application of voting techniques with individual adjustments and enhancements to other scenarios

As described in the beginning Section 1.1 Motivation of this thesis, the use case for voting techniques in the domain of information retrieval is twofold:

Search for classes On the one hand, the search for a suitable class provides the basis for further investigation of its elements. As described in Section 1.1, instances of suitable classes can be appropriate companies, bibliographic collections, and other artifacts whose documents vote for their associated categories on the basis of a query. This thesis examines the basic characteristics of voting techniques; in a productive scenario, individual improvements and enhancements are necessary with respect to specific aspects.

Classification task On the other hand, the voting result can also serve as a classification of the request itself, on the basis of which further tasks or decisions are taken. This includes, for example, the categorization of emails, messages, and documents.

Application of voting techniques in other domains

The application of voting techniques in other domains should be examined, as they often already exist in a related form. In Chapter 7, we discuss the transfer of voting techniques to agglomerative clustering. Furthermore, the application in ensemble learning should be examined in perspective. As Wang defines in his paper "Some

fundamental issues in ensemble methods", *"An ensemble in the context of machine learning can be broadly defined as a machine learning system that is constructed with a set of individual models working in parallel and whose outputs are combined with a decision fusion strategy to produce a single answer for a given problem."* [Wano8]. An investigation into the extent to which the application and transfer of voting techniques is suitable for individual cases of machine learning and other fusion scenarios will possibly provide further application scenarios in the future.

List of Figures

List of Tables

References

[AS12] AT Albaham and N Salim. Adapting Voting Techniques for Online Forum Thread Retrieval. In: *Advanced Machine Learning Technologies and Applications - First International Conference, AMLTA 2012, Cairo, Egypt, December 8-10, 2012. Proceedings*. Ed. by AE Hassanien et al. Vol. 322. Communications in Computer and Information Science. Springer, 2012, pp. 439–448. https://doi.org/10.1007/978-3-642-35326-0 (pages 7, 87, 98, 99, 101).

[And73] MR Anderberg. *Cluster Analysis for Applications*. Probability and Mathematical Statistics 19. New York [u.a.]: Academic Press, 1973 (page 127).

[Arv+10] P Arvola et al. Overview of the INEX 2010 Ad Hoc Track. In: *Comparative Evaluation of Focused Retrieval - 9th International Workshop of the Inititative for the Evaluation of XML Retrieval, INEX 2010, Vugh, The Netherlands, December 13-15, 2010, Revised Selected Papers*. Ed. by S Geva et al. Vol. 6932. Lecture Notes in Computer Science. Springer, 2010, pp. 1–32. https://doi.org/10.1007/978-3-642-23577-1 (page 109).

[AM01] JA Aslam and MH Montague. Models for Metasearch. In: *SIGIR 2001: Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, September 9-13, 2001, New Orleans, Louisiana, USA*. Ed. by WB Croft et al. ACM, 2001, pp. 275–284. https://doi.org/10.1145/383952.384007 (page 32).

[Bai+07] P Bailey et al. Overview of the TREC 2007 Enterprise Track. In: *Proceedings of The Sixteenth Text REtrieval Conference, TREC 2007, Gaithersburg, Maryland, USA, November 5-9, 2007*. Ed. by EM Voorhees and LP Buckland. Vol. 500-274. NIST Special Publication. National Institute of Standards and Technology (NIST), 2007. http://trec.nist.gov/pubs/trec16/papers/ENT.OVERVIEW16.pdf (pages 4, 38, 45).

[Bal+12] K Balog et al. Expertise Retrieval. In: *Foundations and Trends in Information Retrieval* 6(2-3):(2012), 127–256 (pages 1, 6, 20, 69, 143).

[BM10] S Bhatia and P Mitra. Adopting Inference Networks for Online Thread Retrieval. In: *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2010, Atlanta, Georgia, USA, July 11-15, 2010*. Ed. by M Fox and D Poole. AAAI Press, 2010. https://doi.org/10.1609/aaai.v24i1.7521 (page 98).

[BBH16] D Blank, S Boosz, and A Henrich. IT company atlas upper Franconia: a practical application of expert search techniques. In: *Proceedings of the 31st Annual ACM Symposium on Applied Computing, Pisa, Italy, April 4-8, 2016*. Ed. by S Ossowski. ACM, 2016, pp. 1048–1053. https://doi.org/10.1145/2851613.2851695 (page 7).

[Cal94] JP Callan. Passage-Level Evidence in Document Retrieval. In: *Proceedings of the 17th Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval. Dublin, Ireland, 3-6 July 1994 (Special Issue of the SIGIR Forum)*. Ed. by WB Croft and CJ van Rijsbergen. ACM/Springer, 1994, pp. 302–310. https://doi.org/10.1007/978-1-4471-2099-5_31 (pages 104, 109).

[CS02] A Chowdhury and I Soboroff. Automatic evaluation of world wide web search services. In: *SIGIR 2002: Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, August 11-15, 2002, Tampere, Finland*. Ed. by K Järvelin et al. ACM, 2002, pp. 421–422. https://doi.org/10.1145/564376.564474 (page 67).

[CVS05] N Craswell, AP de Vries, and I Soboroff. Overview of the TREC 2005 Enterprise Track. In: *Proceedings of the Fourteenth Text REtrieval Conference, TREC 2005, Gaithersburg, Maryland, USA, November 15-18, 2005*. Ed. by EM Voorhees and LP Buckland. Vol. 500-266. NIST Special Publication. National Institute of Standards and Technology (NIST), 2005. http://trec.nist.gov/pubs/trec14/papers/ENTERPRISE.OVERVIEW.pdf (pages 3, 4, 38, 39).

[CZC02] S Cronen-Townsend, Y Zhou, and WB Croft. Predicting query performance. In: *SIGIR 2002: Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, August 11-15, 2002, Tampere, Finland*. Ed. by K Järvelin et al. ACM, 2002, pp. 299–306. https://doi.org/10.1145/564376.564429 (page 86).

[CLO10] R Cummins, M Lalmas, and C O'Riordan. Learning Aggregation Functions for Expert Search. In: *ECAI 2010 - 19th European Conference on Artificial Intelligence, Lisbon, Portugal, August 16-20, 2010, Proceedings*. Ed. by H Coelho, R Studer, and MJ Wooldridge. Vol. 215. Frontiers in Artificial Intelligence and Applications. IOS Press, 2010, pp. 535–540. https://doi.org/10.3233/978-1-60750-606-5-535 (pages 29, 37, 47, 50, 64, 149).

[DC19] Z Dai and J Callan. Deeper Text Understanding for IR with Contextual Neural Language Modeling. In: *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2019, Paris, France, July 21-25, 2019*. Ed. by B Piwowarski et al. ACM, 2019, pp. 985–988. https://doi.org/10.1145/3331184.3331303 (pages 105, 109, 113).

[Eve74] B Everitt. *Cluster Analysis*. London: Heinemann, 1974 (pages 127, 129).

[FS93] EA Fox and JA Shaw. Combination of Multiple Searches. In: *Proceedings of The Second Text REtrieval Conference, TREC 1993, Gaithersburg, Maryland, USA, August 31 - September 2, 1993*. Ed. by DK Harman. Vol. 500-215. NIST Special Publication. National Institute of Standards and Technology (NIST), 1993, pp. 243–252. http://trec.nist.gov/pubs/trec2/papers/txt/23.txt (pages 21, 22, 33, 34).

[Gal11] J Gallier. *Discrete Mathematics*. Springer New York, NY, 2011 (page 91).

[HO06] B He and I Ounis. Query performance prediction. In: *Inf. Syst.* 31(7):(2006), 585–594. https://doi.org/10.1016/j.is.2005.11.003 (page 86).

[HW17] A Henrich and M Wegmann. 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*. Ed. by M Leyer. Vol. 1917. CEUR Workshop Proceedings. CEUR-WS.org, 2017, p. 253. https://ceur-ws.org/Vol-1917/paper34.pdf (pages iv, 65).

[HW21] A Henrich and M Wegmann. Search and evaluation methods for class level information retrieval: extended use and evaluation of methods applied in expertise retrieval. In: *SAC '21: The 36th ACM/SIGAPP Symposium on Applied Computing, Virtual Event, Republic of Korea, March 22-26, 2021*. Ed. by C Hung et al. ACM, 2021, pp. 681–684. https://doi.org/10.1145/3412841.3442092 (pages iv, 20, 65).

[Hen08] N Henze. *Stochastik für Einsteiger*. Berlin Heidelberg New York: Springer-Verlag, 2008 (page 91).

[JKA16] GK Jayasinghe, S Karimi, and M Ayre. Evaluation of Retrieval Algorithms for Expertise Search. In: *Proceedings of the 21st Australasian Document Computing Symposium, ADCS 2016, Caulfield, VIC, Australia, December 5-7, 2016*. Ed. by S Karimi and MJ Carman.

ACM, 2016, pp. 85–88. http://dl.acm.org/citation.cfm?id=3015035 (pages 143, 144).

[Juá+10] A Juárez-González et al. Selecting the N-Top Retrieval Result Lists for an Effective Data Fusion. In: *Computational Linguistics and Intelligent Text Processing, 11th International Conference, CICLing 2010, Iasi, Romania, March 21-27, 2010. Proceedings*. Ed. by AF Gelbukh. Vol. 6008. Lecture Notes in Computer Science. Springer, 2010, pp. 580–589. https://doi.org/10.1007/978-3-642-12116-6_49 (page 27).

[LNY21] J Lin, RF Nogueira, and A Yates. *Pretrained Transformers for Text Ranking: BERT and Beyond*. Synthesis Lectures on Human Language Technologies. Morgan & Claypool Publishers, 2021. https://doi.org/10.2200/S01123ED1V01Y202108HLT053 (page 105).

[Mac09] C Macdonald. The voting model for people search. PhD thesis. University of Glasgow, UK, 2009. http://theses.gla.ac.uk/609/ (pages 1, 7, 18, 20, 22, 25, 29, 33, 37–44, 49, 106, 144, 146).

[MO06] C Macdonald and I Ounis. Voting for candidates: adapting data fusion techniques for an expert search task. In: *Proceedings of the 2006 ACM CIKM International Conference on Information and Knowledge Management, Arlington, Virginia, USA, November 6-11, 2006*. Ed. by PS Yu et al. ACM, 2006, pp. 387–396. https://doi.org/10.1145/1183614.1183671 (pages 7, 57).

[MO08] C Macdonald and I Ounis. Expert Search Evaluation by Supporting Documents. In: *Advances in Information Retrieval , 30th European Conference on IR Research, ECIR 2008, Glasgow, UK, March 30-April 3, 2008. Proceedings*. Ed. by C Macdonald et al. Vol. 4956. Lecture Notes in Computer Science. Springer, 2008, pp. 555–563. https://doi.org/10.1007/978-3-540-78646-7_55 (pages 67, 87).

[MO09] C Macdonald and I Ounis. The influence of the document ranking in expert search. In: *Proceedings of the 18th ACM Conference on Information and Knowledge Management, CIKM 2009, Hong Kong, China, November 2-6, 2009*. Ed. by DW Cheung et al. ACM, 2009, pp. 1983–1986. https://doi.org/10.1145/1645953.1646282 (pages 37, 44–46, 50).

[MRS08] C Manning, P Raghavan, and H Schütze. *Introduction to Information Retrieval*. Cambridge University Press, 2008. https://www-nlp.stanford.edu/IR-book/ (pages 16, 40, 99, 109, 113, 134).

[Ngu+16] T Nguyen et al. MS MARCO: A Human Generated MAchine Reading COmprehension Dataset. In: *CoRR* abs/1611.09268:(2016). arXiv: 1611.09268. http://arxiv.org/abs/1611.09268 (page 105).

[19] *Open Academic Graph*. 2019. https://www.aminer.org/oag2019 (page 69).

[OC03] P Ogilvie and JP Callan. Combining document representations for known-item search. In: *SIGIR 2003: Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, July 28 - August 1, 2003, Toronto, Canada*. Ed. by CLA Clarke et al. ACM, 2003, pp. 143–150. https://doi.org/10.1145/860435.860463 (page 24).

[PC17] JM Ponte and WB Croft. A Language Modeling Approach to Information Retrieval. In: *SIGIR Forum* 51(2):(2017), 202–208. https://doi.org/10.1145/3130348.3130368 (page 99).

[RZ09] SE Robertson and H Zaragoza. The Probabilistic Relevance Framework: BM25 and Beyond. In: *Found. Trends Inf. Retr.* 3(4):(2009), 333–389. https://doi.org/10.1561/1500000019 (page 17).

[SAB93] G Salton, J Allan, and C Buckley. Approaches to Passage Retrieval in Full Text Information Systems. In: *Proceedings of the 16th Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval. Pittsburgh, PA, USA, June 27 - July 1, 1993*. Ed. by RR Korfhage, EM Rasmussen, and P Willett. ACM, 1993, pp. 49–58. https://doi.org/10.1145/160688.160693 (page 104).

[SSK07] R Schenkel, FM Suchanek, and G Kasneci. YAWN: A Semantically Annotated

Wikipedia XML Corpus. In: *Datenbanksysteme in Business, Technologie und Web (BTW 2007), 12. Fachtagung des GI-Fachbereichs "Datenbanken und Informationssysteme" (DBIS), Proceedings, 7.-9. März 2007, Aachen, Germany*. Ed. by A Kemper et al. Vol. P-103. LNI. GI, 2007, pp. 277–291. https://dl.gi.de/handle/20.500.12116/31804 (page 109).

[SB11] E Smirnova and K Balog. A User-Oriented Model for Expert Finding. In: *Advances in Information Retrieval - 33rd European Conference on IR Research, ECIR 2011, Dublin, Ireland, April 18-21, 2011. Proceedings*. Ed. by PD Clough et al. Vol. 6611. Lecture Notes in Computer Science. Springer, 2011, pp. 580–592. https://doi.org/10.1007/978-3-642-20161-5_58 (pages 2, 3).

[SS73] PHA Sneath and RR Sokal. *Numerical Taxonomy: The Principles and Practice of Numerical Classification*. San Francisco, London: Freeman, 1973 (pages 127, 130).

[SVC06] I Soboroff, AP de Vries, and N Craswell. Overview of the TREC 2006 Enterprise Track. In: *Proceedings of the Fifteenth Text REtrieval Conference, TREC 2006, Gaithersburg, Maryland, USA, November 14-17, 2006*. Ed. by EM Voorhees and LP Buckland. Vol. 500-272. NIST Special Publication. National Institute of Standards and Technology (NIST), 2006. http://trec.nist.gov/pubs/trec15/papers/ENT06.OVERVIEW.pdf (page 3).

[Tan+08] J Tang et al. ArnetMiner: extraction and mining of academic social networks. In: *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Las Vegas, Nevada, USA, August 24-27, 2008*. Ed. by Y Li, B Liu, and S Sarawagi. ACM, 2008, pp. 990–998. https://doi.org/10.1145/1401890.1402008 (pages 2, 66, 70).

[TS06] A Turpin and F Scholer. User performance versus precision measures for simple search tasks. 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*. Ed. by EN Efthimiadis et al. ACM, 2006, pp. 11–18. https://doi.org/10.1145/1148170.1148176 (page 46).

[Voo04] EM Voorhees. Overview of the TREC 2004 Robust Track. In: *Proceedings of the Thirteenth Text REtrieval Conference, TREC 2004, Gaithersburg, Maryland, USA, November 16-19, 2004*. Ed. by EM Voorhees and LP Buckland. Vol. 500-261. NIST Special Publication. National Institute of Standards and Technology (NIST), 2004. http://trec.nist.gov/pubs/trec13/papers/ROBUST.OVERVIEW.pdf (page 111).

[Wan08] W Wang. Some fundamental issues in ensemble methods. In: *Proceedings of the International Joint Conference on Neural Networks, IJCNN 2008, part of the IEEE World Congress on Computational Intelligence, WCCI 2008, Hong Kong, China, June 1-6, 2008*. IEEE, 2008, pp. 2243–2250. https://doi.org/10.1109/IJCNN.2008.4634108 (pages 149, 150).

[WH18] M Wegmann and A Henrich. Search for an Appropriate Journal - In Depth Evaluation of Data Fusion Techniques. In: *Proceedings of the Conference "Lernen, Wissen, Daten, Analysen", LWDA 2018, Mannheim, Germany, August 22-24, 2018*. Ed. by R Gemulla et al. Vol. 2191. CEUR Workshop Proceedings. CEUR-WS.org, 2018, pp. 343–354. http://ceur-ws.org/Vol-2191/paper41.pdf (pages iv, 16, 17, 65).

[Wu12] S Wu. *Data Fusion in Information Retrieval*. Vol. 13. Adaptation, Learning, and Optimization. Springer, 2012. https://doi.org/10.1007/978-3-642-28866-1 (page 7).

[Zha+02] M Zhang et al. THU TREC 2002: Novelty Track Experiments. In: *Proceedings of The Eleventh Text REtrieval Conference, TREC 2002, Gaithersburg, Maryland, USA, November 19-22, 2002*. Ed. by EM Voorhees and LP Buckland. Vol. 500-251. NIST Special Publication. National Institute of Standards and Technology (NIST), 2002. http://trec.nist.gov/pubs/trec11/papers/tsinghuau.novelty2.pdf (page 30).