

Information Systems on Bibliographic Metadata for Researchers

vom Fachbereich IV der Universität Trier zur Verleihung des akademischen
Grades Doktor der Naturwissenschaften (Dr. rer. nat.) genehmigte
Dissertation

Christin Katharina Kreutz

January 2022

Supervisor: Prof. Dr.-Ing. Ralf Schenkel

Examiners: Prof. Dr.-Ing. Ralf Schenkel
(Trier University)
Prof. Dr. Guillaume Cabanac
(University of Toulouse)

Abstract

Everyday life of researchers gets intermitted by tasks for which information systems on bibliographic metadata are used. Whether a researcher is checking for novel journals from their area, looking up updates on their colleagues, searching for seminal publications, re-arranging their reading lists or doing organisational work for a conference in which they act as a chair, they require adequate systems to support these types of work. This cumulative dissertation addresses multiple needs of researchers using information systems on bibliographic metadata: *i)* general information search and exploration, *ii)* identification of influential publications, *iii)* scientific paper recommendation, and *iv)* reviewer set recommendation for publications.

The first main component tackles *general information search and exploration*. It introduces *SchenQL*, a domain-specific query language and GUI on bibliographic metadata. It features functionality usually only found with complex-to-learn general purpose languages such as SQL or Cypher. By utilising domain jargon and offering possibly sophisticated domain-specific functions, it supports information search and exploration for domain-experts as well as casual users of digital libraries. A user study showed that users are satisfied with our query language and GUI. Further, *SchenQL* is a suitable alternative compared to SQL for domain-experts as well as non-experts for typical tasks encountered in digital libraries.

The second component tackles *identification of influential publications* with using semantometrics. The work observes citation networks and contents of scientific papers and extracts features from distances between publications (the so-called *semantometrics*) to estimate and predict their influence. Citation counts determine the influence labels for papers. A comparison of different document vector embeddings, distance measures and classifiers produced high accuracy in predicting the future influence or class of papers, i.e., seminal survey or uninfluential, when only observing features which are already known as soon as a paper is published.

The third component covers the area of *scientific paper recommendation* for researchers. The survey observed and describes contemporary literature from January 2019 to October 2021. Here, we introduce novel dimensions to classify paper recommendation approaches and present current datasets as well as evaluation measures. Lastly, we discuss already identified as well as upcoming shortcomings of the mentioned works. We found that a number of former challenges are no longer relevant as systems tend to become more complex and of a hybrid type, but several issues still remain and get rarely highlighted, such as scalability or privacy of approaches. Upcoming problems such as a lack of explainability of results or inadequate evaluations were also

identified.

The fourth and fifth components both investigate *recommendation of reviewer sets* for scientific papers on different levels. First, *RevASIDE* conducts assignment of reviewers to papers as a problem on a paper-based level only. The system produces suitable assignments which are composed of reviewers with expertise, authority, and interest in the field. The reviewer set also is of diverse seniority and reviewers' expertise and experience. *RevASIDE* utilises the expert search task as a preparatory step for the actual reviewer set assignment. We found that more sophisticated document representation methods do not necessarily lead to better overall results, and that our method constructs diverse reviewer sets which experts deem suitable.

Lastly, *DiveRS* introduces a diverse reviewer set recommendation method which not only constructs suitable sets of complementing reviewers for all submissions of a conference but also considers a current program committee. This approach does not assume the program committee to be perfectly composed for the incoming submissions, but instead actively extends the reviewer candidate pool. It strives to compute assignments of reviewers to submissions to cover their topical composition as well as provide diversity in professional background, location, and seniority of reviewer sets. Additionally, it allows for submissions and reviewer candidates to be out of scope of a specific conference. As a result, we showed that the proposed assignments and recommended new reviewers are suitable, diverse and fair.

Keywords

Bibliographic Metadata, Digital Libraries, Semantometrics, Domain-Specific Query Languages, Scientific Recommendation, Paper Recommendation Systems, Reviewer Recommendation Systems, Recommender Systems, Evaluation.

Zusammenfassung

Der Alltag von Forschenden wird durch Aufgaben unterbrochen, für die Informationssysteme auf bibliografischen Metadaten verwendet werden. Ganz gleich, ob Forschende neue Journalen aus ihren Fachgebieten recherchieren, aktuelle Informationen über Kollegen einholen, nach bahnbrechenden Publikationen suchen, ihre Leselisten neu ordnen oder organisatorische Arbeiten für eine Konferenz erledigen, bei denen sie als Vorsitzende fungieren, sie benötigen geeignete Systeme zur Unterstützung dieser Arbeiten. Diese kumulative Dissertation befasst sich mit den vielfältigen Bedürfnissen von Forschenden, die Informationssysteme auf bibliografischen Metadaten nutzen: *i)* allgemeine Informationssuche und -exploration, *ii)* Identifizierung einflussreicher Publikationen, *iii)* Empfehlung wissenschaftlicher Arbeiten und *iv)* Empfehlung von Gutachtermengen für Publikationen.

Die erste Hauptkomponente befasst sich mit *allgemeiner Informationssuche und -exploration*. Sie führt *SchenQL* ein, eine domänenspezifische Abfragesprache und GUI für bibliografische Metadaten. Sie verfügt über Funktionen, die normalerweise nur in komplex zu erlernenden Allzwecksprachen wie SQL oder Cypher zu finden sind. Durch die Verwendung von Fachjargon und die Bereitstellung möglicherweise anspruchsvoller fachlicher Funktionen unterstützt sie die Informationssuche und -exploration sowohl für Fachexperten als auch für Gelegenheitsnutzer digitaler Bibliotheken. Eine Benutzerstudie hat gezeigt, dass die Nutzer mit unserer Abfragesprache und der Benutzeroberfläche zufrieden sind. Darüber hinaus ist *SchenQL* sowohl für Domänenexperten als auch für Nicht-Experten eine geeignete Alternative zu SQL für typische Aufgaben, die in digitalen Bibliotheken anfallen.

Die zweite Komponente befasst sich mit der Identifizierung von einflussreichen Publikationen mit Hilfe der Semantometrie. Die Arbeit betrachtet Zitationsnetzwerke und Inhalte wissenschaftlicher Arbeiten und extrahiert Merkmale aus den Abständen zwischen Publikationen (die sogenannte *Semantometrie*), um deren Einfluss abzuschätzen und vorherzusagen. Die Anzahl der Zitationen bestimmt die Einflusskategorie für die Publikationen. Ein Vergleich verschiedener Dokumentenvektoreinbettungen, Abstandsmaße und Klassifikatoren ergab eine hohe Genauigkeit bei der Vorhersage des zukünftigen Einflusses oder der Klasse von Papieren, d. h. seminal (bahnbrechendes Papier), survey (Übersichtspapier) oder uninfluential (nicht einflussreiches Papier), wenn nur Merkmale berücksichtigt werden, die bereits bekannt sind, sobald ein Papier veröffentlicht wird.

Die dritte Komponente deckt den Bereich der Empfehlung wissenschaftlicher Publikationen für Forscher ab. Das Übersichtspapier betrachtet und beschreibt zeitgenössische Literatur von Januar 2019 bis Oktober 2021. Hier

stellen wir neue Dimensionen zur Klassifizierung von Literaturempfehlungsansätzen vor und präsentieren aktuelle Datensätze sowie Bewertungsmaße. Abschließend diskutieren wir bereits identifizierte sowie derzeit aufkommende Defizite der genannten Arbeiten. Wir haben festgestellt, dass eine Reihe von früheren Herausforderungen nicht mehr relevant ist, da die Systeme immer komplexer und hybrider werden, aber einige Probleme bleiben bestehen und werden nur selten beschrieben, wie z.B. Skalierbarkeit oder Überlegungen bezüglich der Wahrung der Privatsphäre von Nutzern der Ansätze. Es wurden ebenfalls aufkommende Probleme wie die mangelnde Erklärbarkeit von Ergebnissen oder unzureichende Evaluierungen festgestellt.

Die vierte und fünfte Komponente untersuchen beide *Empfehlung von Gutachtermengen* für wissenschaftliche Arbeiten auf verschiedenen Ebenen. Zunächst führen wir mit *RevASIDE* die Zuordnung von Gutachtern zu Arbeiten als Problem nur auf der Ebene der einzelnen Papiere durch. Das System erzeugt geeignete Zuweisungen, die sich aus Gutachtern mit Fachwissen, Autorität und Interesse an dem Fachgebiet zusammensetzen. Die Gutachter haben außerdem ein unterschiedliches Dienstalter, Fachwissen und Erfahrung. *RevASIDE* nutzt die Expertensuche als vorbereitenden Schritt für die eigentliche Gutachterzuweisung. Wir haben festgestellt, dass ausgefeiltere Methoden zur Darstellung von Dokumenten nicht unbedingt zu besseren Gesamtergebnissen führen und dass unsere Methode verschiedene Gutachtermengen erstellt, die von Experten für geeignet gehalten werden.

Letztlich wird mit *DiveRS* eine Methode zur Empfehlung diverser Gutachtergruppen eingeführt, die nicht nur geeignete Gruppen von sich ergänzenden Gutachtern für alle Einreichungen zu einer Konferenz konstruiert, sondern auch ein aktuelles Programmkomitee mit betrachtet. Dieser Ansatz geht nicht davon aus, dass das Programmkomitee perfekt für die eingehenden Einreichungen zusammengestellt ist, sondern erweitert aktiv den Gutachterkandidatenpool. Es wird angestrebt, die Zuordnung von Gutachtern zu den Einreichungen so zu berechnen, dass die thematische Zusammensetzung der Einreichung abgedeckt ist und eine Vielfalt hinsichtlich des beruflichen Hintergrunds, des Standorts und des Dienstalters der Gutachter besteht. Darüber hinaus ist es möglich, dass Einreichungen und Gutachterkandidaten außerhalb des Bereichs einer bestimmten Konferenz liegen. Wir haben gezeigt, dass die vorgeschlagenen Zuordnungen und empfohlenen neuen Gutachter geeignet, vielfältig und fair sind.

Dedicated to my mother

Acknowledgements

No PhD thesis is written without the help and support of others: I would especially like to thank Prof. Dr.-Ing. Ralf Schenkel for his invaluable supervision, advice, guidance, feedback, trust and patience throughout this whole experience. You always encouraged me to formulate and pursue my own research interests, introduced me to researchers from different fields, and guided this journey. I cannot thank you enough for all that you have done over the course of the years.

Naturally, a big thanks goes to Prof. Dr. Guillaume Cabanac for being not only the second examiner for this dissertation but also for providing me additional references.

Gratitude is directed at ERASMUS for funding my participation at ESS-IR 2017, and thanks to DAAD for awarding me a scholarship for Norway as visiting researcher in 2021.

Extra thanks goes out to all of my collaborators, I want to deeply thank you. Prof. Dr. Krisztian Balog and Jun.-Prof. Dr.-Ing. Benjamin Weyers, our interactions inspired and shaped integral parts of this PhD thesis. Your unique views and approaches taught me valuable lessons on scientific writing, user studies and research in general.

Praise should also be directed towards my student collaborators, with a special focus on Michael Wolz and Jascha Knack for contributing so much to SchenQL. Thank you for your hard work.

Special thanks goes to Dr. Ingo Frommholz for his inspiring ideas and going out of his way by contacting a potential host for a research visit.

I offer my thanks to my colleagues Tobias Zeimet and Lorik Dumani for their support, encouragement, suggestions and always having an open ear for me. Not only did we advance our PhD journey together, but we became close friends. Tobias, I cannot recount all the situations in which PhD life in general was just a lot to handle. With your kind words and light-heartedness, each hardship was somehow manageable. Lorik, thanks for the discussions we had about our research. You provided me with new ideas, courage after rejections, an outside perspective and were a reliable collaborator.

Thanks to all reviewers, editors and proofreaders for their input. My gratefulness is directed towards all study participants who took the time to complete questionnaires and tasks, as well as give valuable comments.

Thanks to the students I supervised for all those inspiring discussions, in particular I want to thank Martin Vu, Peter Königstein and Martin Blum.

Lastly, thanks to my parents, family and close friends for their support, patience and encouragement. A special thanks goes out to Henry, Cara and Daniel who were there to support me at every step of the way.

Contents

Preamble

Introduction	2
1.1 Information Search and Exploration	2
1.2 Identification of Influential Publications	4
1.3 Scientific Paper Recommendation	6
1.4 Reviewer Set Recommendation for Publications	7
1.5 Structure	11
Summarised Findings in Current Scientific Landscape	12
2.1 Information Search and Exploration	12
2.2 Identification of Influential Publications	13
2.3 Scientific Paper Recommendation	14
2.4 Reviewer Set Recommendation for Publications	15
Relations between Papers	18
Future Work	21
Bibliography	28

Publications

SchenQL: in-depth analysis of a query language for bibliographic metadata	30
5.1 Introduction	32
5.2 Related Work	34
5.3 SchenQL: QL and GUI	35
5.3.1 Data Model	35
5.3.2 Building Blocks	36
5.3.3 Syntax	37

5.3.4	Implementation	38
5.4	Evaluation	44
5.4.1	Benchmarks: Database Engines and Target Language as well as SchenQL Compiler Evaluation	45
5.4.2	Qualitative Study: Interviews	55
5.4.3	Quantitative Study: SchenQL CLI vs. SQL, GUI and UEQ	56
5.5	Conclusion and Future Work	66
	Bibliography	68

Evaluating Semantometrics from Computer Science Publications **73**

6.1	Introduction	76
6.2	Semantometrics and Related Work	78
6.2.1	Semantometrics	78
6.2.2	Related Work	80
6.3	SUSdblp Dataset	84
6.3.1	Introduction	84
6.3.2	Contained Data	85
6.3.3	Number of References and Citations	86
6.3.4	Publication Years	87
6.3.5	Sub-fields and Topic Distributions in Publications	90
6.3.6	Description and Discussion	91
6.4	Methodology	93
6.4.1	Document Vector Representations	94
6.4.2	Distance Measures	94
6.4.3	Classification Algorithms	95
6.4.4	Implementation	95
6.5	Evaluation of the Approach	96
6.5.1	Single Features	97
6.5.2	Multiple Features	99
6.5.3	Combination	101
6.5.4	Truly Uninfluential Publications	102
6.5.5	Information Available at Publication Time	102
6.5.6	Discussion	103
6.6	Evaluation of the Dataset	105
6.6.1	Robustness of Dataset	105
6.6.2	Classification for Different Years	106
6.6.3	Discussion	107
6.7	Alternative Approaches for Classification	107
6.7.1	Classification on Number of References and Citations	108

6.7.2	Classification on Document Vector Representations . . .	109
6.7.3	Discussion	112
6.8	Conclusion and Future Work	113
6.A	Evaluation of the Approach: Single Features	115
6.B	Evaluation of the Approach: All Features	116
6.C	Evaluation of the Approach: 33 Features	116
6.D	Evaluation of the Approach: Combination	117
6.E	Evaluation of the Dataset: Classification for Different Years .	118
6.F	Evaluation of the Dataset: Abstract Length Bias	119
6.G	Other Approaches: All Dimensions of Document Vector Rep- resentations of Publications	119
6.G.1	Single Dimensions of Document Vector Representations of Publications	119
6.G.2	All Dimensions of Document Vector Representations of Publications	121
6.G.3	All Dimensions of Document Vector Representations of Referenced Papers	121
6.G.4	All Dimensions of Document Vector Representations of Citing Publications	122
	Bibliography	124

**Scientific Paper Recommendation Systems: a Literature Re-
view of recent Publications** **133**

7.1	Introduction	136
7.2	Problem Statement	137
7.3	Literature Review	138
7.3.1	Scope	138
7.3.2	Meta analysis	140
7.3.3	Categorisation	140
7.3.4	Paper Recommendation Systems	144
7.3.5	Other relevant Work	153
7.4	Datasets	155
7.4.1	dblp based datasets	158
7.4.2	SPRD based datasets	158
7.4.3	CiteULike based datasets	158
7.4.4	ACM based datasets	159
7.4.5	Scopus based datasets	159
7.4.6	AMiner based datasets	159
7.4.7	AAN based datasets	159
7.4.8	Sowiport based datasets	160
7.4.9	CiteSeerX based datasets	160

7.4.10	Patents based datasets	160
7.4.11	Hep-TH based datasets	160
7.4.12	MAG based datasets	161
7.4.13	Others	161
7.5	Evaluation	161
7.5.1	Relevance and Assessment	161
7.5.2	Evaluation Measures	164
7.5.3	Evaluation Types	166
7.6	Open Challenges and Objectives	167
7.6.1	Challenges Highlighted in Previous Works	167
7.6.2	Emerging Challenges	173
7.6.3	Discussion	177
7.7	Conclusion	178
	Bibliography	180

RevASIDE: Evaluation of Assignments of Suitable Reviewer

Sets for Publications from Fixed Candidate Pools 196

8.1	Introduction	199
8.2	Related Work	201
8.2.1	Team Formation	201
8.2.2	Expert Search	202
8.2.3	Reviewer Set Recommendation	203
8.3	Aspects	205
8.3.1	Aspect 1: No Conflicts of Interests	205
8.3.2	Aspect 2: Disjoint Publications	205
8.3.3	Aspect 3: Expertise	205
8.3.4	Aspect 4: Authority	206
8.3.5	Aspect 5: Diverse Expertise (Diversity)	206
8.3.6	Aspect 6: Current Interest	206
8.3.7	Aspect 7: Diverse Experience (Seniority)	206
8.4	Approach	207
8.4.1	Step 1: Expert Search	207
8.4.2	Step 2: Reviewer Set Assignment	209
8.4.3	Run Time Analysis	213
8.5	Datasets	214
8.5.1	Data Acquisition	215
8.5.2	Document Representations	216
8.5.3	MOL'17, BTW'17 and ECIR'17	217
8.5.4	Discussion and Challenges	217
8.6	Evaluation: Preface	218
8.6.1	Hypotheses	219

8.7	Evaluation: Step 1 - Expert Search Task	219
8.7.1	Setting	221
8.7.2	Results (Analysis of H_1 and H_2)	222
8.8	Evaluation Step 2 - Reviewer Set Assignment Task	224
8.8.1	Setting	224
8.8.2	Quantitative Evaluation	225
8.8.3	Qualitative Evaluation (Analysis of H_7)	232
8.9	Conclusion and Future Work	237
	Bibliography	239

Diverse Reviewer Suggestion for Extending Conference Program Committees **243**

9.1	Introduction	245
9.2	Related Work	247
9.3	Problem Setting	250
9.3.1	Problem Statement	250
9.3.2	Notation	250
9.4	Method	251
9.4.1	Modelling Diversity	251
9.4.2	Algorithm	251
9.4.3	Practical Issues and Effects of Parameters	256
9.5	Experimental Setup	257
9.5.1	Datasets	257
9.5.2	Parameter Settings	258
9.5.3	Established Measures	258
9.5.4	Novel Measures	259
9.6	Experiments	260
9.6.1	Part 1: Reviewer Assignment	260
9.6.2	Part 2: Reviewer Suggestion	263
9.7	Conclusion	264
	Bibliography	265

Appendix

Curriculum Vitae (as of Jan 2022)	269
Complete List of Publications (as of Jan 2022)	271

List of Tables

5.1	SchenQL base concepts Publications (PU), persons (PE), conferences (C), journals (J) and institutions (I) with their respective literals (L), specialisations (S), filters (F) and standard return values (V, relevant for the CLI).	38
5.2	Overview of queries evaluated in the benchmark. <i>A</i> , <i>B</i> are unique names of different authors, <i>J</i> is a journal acronym, <i>P</i> is a publication title, <i>K</i> is a keyword, <i>F</i> is a forename and <i>T</i> is a term.	47
5.3	Evaluation query categories, queries and figures.	49
5.4	Categories (C) of system response times in ms and associated user experience.	49
5.5	Overview of SchenQL queries derived from Table 5.2 evaluated in the benchmark. <i>A</i> , <i>B</i> are unique names of different authors, <i>J</i> is a journal acronym, <i>P</i> is a publication title, <i>K</i> is a keyword, <i>F</i> is a forename and <i>T</i> is a term.	54
5.6	Templates of all queries used in the qualitative evaluations. <i>A</i> are unique names of different authors, <i>C</i> is the acronym of a conference and <i>Y</i> is a year.	57
5.7	SchenQL and SQL formulations of queries utilised in our evaluations.	58
5.8	Correctness (CORR) in percent, assessed average difficulty (DIFF) and average time in minutes for the four queries for SQL and the SchenQL CLI.	60
5.9	Correctness (CORR) in percent, assessed average difficulty (DIFF) and average time in minutes for the four queries for the GUI.	64
5.10	Description of the six dimensions measured by the UEQ to capture users' impressions of interactive products [33].	65
6.1	Numeric description of the SUSdblp dataset.	86

6.2	Average publication years for papers from P , X and Y as well as average distances in years between publications from P and X as well as P and Y for the three classes.	88
6.3	Best classifiers dependent on distance measures for all document vector representations. For the classification algorithm achieving the highest accuracy per combination, the single feature (F), the corresponding F1 score as well as accuracies for the three classes c_0 , c_1 and c_2 are displayed.	98
6.4	Best classifiers dependent on distance measures for all document vector representations. For the classification algorithm achieving the highest accuracy per combination, the corresponding F1 score as well as accuracies for the three classes c_0 , c_1 and c_2 are displayed.	100
6.5	Classification accuracy and F1 scores for features of groups B and D derived from all different vector representations from the SUSdblp dataset for the best performing distance measure and classification algorithm.	103
6.6	Classification accuracy and F1 scores for all dimensions from the different one-vector representations of publications with their citing and referenced papers from the SUSdblp dataset for the best performing classification algorithm.	110
6.7	Classification accuracy and F1 scores for concatenated on-vector representations of references and publication from the SUSdblp dataset for the best performing classification algorithm.	111
6.8	Observation of different distance measures for all respective document vector representations. For the classification algorithm achieving the highest accuracy, the single feature (F), the corresponding F1 score as well as accuracies for the three classes c_0 , c_1 and c_2 are displayed.	115
6.9	Best classifiers dependent on distance measures with different D2V and BERT vector representations. For the classification algorithm achieving the highest accuracy, the corresponding F1 score as well as accuracies for the three classes c_0 , c_1 and c_2 are displayed.	116
6.10	Best classifier dependent on document vector representation and distance measure. For the classification algorithm achieving the highest accuracy, the corresponding F1 score as well as accuracies for the three classes c_0 , c_1 and c_2 are displayed.	117
6.11	Best accuracies dependent on the number of combined sets of features derived from distances between document vector representations.	118

6.12	Classification accuracy and F1 scores using all features derived from Doc2Vec vector representations for the best performing classification algorithm up until or from and after different years as well as the accuracies and number of publication per class.	118
6.13	Ten most probable words per topic for best performing topics in the single feature classification based on features of the publication alone, in decreasing probability.	120
6.14	Classification accuracy and F1 scores for single dimensions DIM from different vector representations of publications P from the SUSdblp dataset for the best performing classification algorithm.	120
6.15	Classification accuracy and F1 scores for all dimensions from the different vector representations of publications P from the SUSdblp dataset for the best performing classification algorithm.	121
6.16	Classification accuracy and F1 scores for all dimensions from the different vector representations of referenced papers X from the SUSdblp dataset for the best performing classification algorithm.	122
6.17	Classification accuracy and F1 scores for all dimensions from the different vector representations of citing papers Y from the SUSdblp dataset for the best performing classification algorithm.	123
7.1	Top most common venues where relevant papers were published together with their type and number of papers (#p). . .	140
7.2	Indications as what type of paper recommendation system works describe themselves with indication if the description is a common used label (c).	142
7.3	Indications whether works utilise the specific data or methods. Papers describing the same approach without extension of the methodology (e.g. only describing more details or an evaluation) are regarded in combination with each other. . . .	145
7.4	Overview of datasets utilised in most recent related work with (unofficial) names, public availability of the possibly modified dataset which was used (A?), and a list of papers it was used in. Datasets are grouped by their underlying data source if possible.	156

7.5	Description of private datasets utilised in most recent related work with (unofficial) names. Datasets are grouped by their underlying data source if possible. We used the following abbreviations: user(s) <i>u</i> , paper(s) <i>p</i> , interaction(s) <i>i</i> , author(s) <i>a</i> , venue(s) <i>v</i> , reference(s) <i>r</i> , citation(s) <i>c</i> , term(s) <i>t</i>	157
7.6	Indications whether approaches utilise the specified relevancy definitions, target values of evaluations and evaluation measures.	162
7.7	Common evaluation measures and percentage of observed evaluations of paper recommendation systems in which they were applied. Percentages are rounded to two decimal places.	164
7.8	Overview of rare existing measures used in evaluations of observed approaches.	165
7.9	For all observed works with user studies we list their number of participants (#p) and their composition. NA indicates that #p or compositions were not described in a specific user study.	167
7.10	Overview of research groups with multiple papers.	168
7.11	Percentage of the 64 considered papers with different numbers of authors (#). Publications with 1 and 10 authors were encountered only once (1.56% each).	169
8.1	Observed properties expertise (E), authority (A), diversity (D), interest (I) and seniority (S) in related work (• indicates a paper covers this aspect but might define it differently from us) as well as indication if the approach is targeting the whole venue (wv?) or can be fully automated (fa?).	203
8.2	Voting techniques VT and accompanying formulas for reviewer <i>R</i> and manuscript <i>M</i>	208
8.3	Mean average precision@10 (MAP), precision@10 (P@10) and nDCG@10 (nDCG) for all combinations of voting techniques (VT) and document vector representations of manuscripts from BTW'17 (upper half) and ECIR'17 (lower half). Best combination in BTW'17: tf-idf + <i>SUM</i> (short <i>b</i> ₁). Best combinations in ECIR'17: tf-idf + <i>MNZ</i> (short <i>e</i> ₁), DBOW + <i>Votes</i> _{δ=0.5} (short <i>e</i> ₂). Column SD gives information on whether or not MAP (m), P@10 and nDCG (n) significantly differ between the different DVs. If ✓, all three measures are significantly different.	220
8.4	Significant differences between the groups in <i>SC</i> as well as the five quantifiable aspects by datasets MOL'17 (m), BTW'17 (b) and ECIR'17 (e).	225

8.5	Configuration (conf), DV, VT, RL_{top} cut-off value k , utilised content type and threshold t resulting in the highest average scores and corresponding values for A , S , I , D as well as E per dataset and result type.	226
8.6	DV, VT, content type CT (rc symbolises content restricted to research sections), cut-off k , threshold t , RT and scores SC for the highest values for each of the different aspects A , S , I , D and E for all three datasets.	228
8.7	Average cores SC as well as values for the five quantifiable aspects A , S , I , D , E for R_0 as well as the three baselines for different sizes of retrieved reviewer sets for all papers from BTW'17.	230
8.8	Average times in seconds for conduction of Step 1, Step 2 and the combination of both for all three datasets for different reviewer set sizes $ R_c $ for single manuscripts.	231
8.9	Average positions (pos) sets computed by the different configurations (conf) were ordered to in the qualitative evaluation as well as the average number of relevant reviewers ($\#rel$) and the average position of entries from the different RTs per set.	233
8.10	Average positions (pos) of sets computed by the different configurations (conf) in the qualitative evaluation as well as the average number of relevant reviewers ($\#rel$) and the average position of entries from the different RTs per set. b_1 : tf-idf + SUM , e_1 : tf-idf + MNZ , e_2 : $DBOW + Votes_{\delta=.5}$	235
8.11	Configuration (c) and RTs with corresponding scores per dataset and manually assessed average values $\in [0, 1]$ (with 1 being the best possible and 0 being the worst possible value) for aspects of sets for the twenty evaluated papers. mA : $1/3 \forall$ reviewers with h index ≥ 25 ; mS : each $1/3$ if set contains at least one senior researcher, at least one junior researcher or at least one mid-career researcher; mI : $1/3 \forall$ reviewers who published a relevant paper in the seven previous years; mD : $1/3 \forall$ reviewer pairs without overlap in their work; mE : $1/3$ for each relevant reviewer in the set. Value mSC is calculated similarly to SC , all manually evaluated aspects are multiplied.	236

9.1	Reviewer assignment results for the automatic evaluation in terms of mean workload per reviewer (mW/R) and all initial PC members ($/PC$), number of unused initial PC members (U) as well as dependency (Dep), fairness (Γ_f^S), average textual diversity (KL), and diversity (Div) of assignments per dataset and method. Methods marked with * correspond to the <u>restrictive</u> setting.	261
9.2	Reviewer suggestion results, listing average values for relevance of explanation (r), confidence (f), usefulness (u), convincingness (c), as well as suggestion ranking (NDCG) per dataset. Usefulness and convincingness are further subdivided (in parentheses) to cases with relevance below 3 ($u_<$, $c_<$) and above 3 ($u_>$, $c_>$).	263

List of Figures

1.1	Neighbourhood of publication P . Nodes symbolise publications, straight edges between papers represent citations. $X = \{x_0, \dots, x_n\}$ are papers referenced by P , $Y = \{y_0, \dots, y_m\}$ are papers citing P . Dotted edges symbolise observed relationships between publications. Group A contains distances between pairs of referenced (X) and citing papers (Y). Group B contains distances between referenced papers (X) and P . Group C contains distances from P and citing papers (Y). Group D contains distances between pairs of referenced papers (X). Group E contains distances between pairs of citing papers (Y). [29]	5
1.2	Schematic overview of our approach. The left part depicts the expert search task, the right part depicts the set of reviewers assignment task. [30]	9
5.1	SchenQL database model with relations, base concepts (underlined), specialisations (italic) and selected attributes (bold).	36
5.2	SchenQL relational data model (online in colour).	37
5.3	Overview of the SchenQL components.	39
5.4	Syntax Tree of the Query PUBLICATIONS WRITTEN BY "Ralf Schenkel".	40
5.5	SchenQL Front End for a search with suggested language components and search result.	41
5.6	Person detail view with Ego Graph depicting up to the ten most common co-authors. Nodes symbolise persons, the further an author is from the middle (person in focus), the less publications they share with the person in focus.	42

5.7	Regular (top) and detailed (bottom) BowTie view with referenced and citing papers of a person with numbers of referenced (bows left of knot) and citing (bows right of knot) papers. For the regular view year numbers limit the period of time from which a paper referenced (left) and is cited itself (right). In the detailed view, the references and citations are separated in single slices per year. Hovering over single slices depicts the year and the associated number of references from or citations acquired in the specific year. The higher the number of citations or references, the longer the bow, the longer the spanned time the higher the bow.	43
5.8	SchenQL graph-based data model (online in colour).	48
5.9	Queries with low execution time in ms.	50
5.10	Queries with medium execution time in ms.	51
5.11	Queries with high execution time in ms, overview of all formulation times.	51
5.12	Queries with high execution time in ms, zoom on formulations with low execution time.	52
5.13	Queries with very high execution time in ms, overview of all formulation times.	52
5.14	Queries with very high execution time in ms, zoom on formulations with low execution time.	53
6.1	Neighbourhood of publication P . Nodes symbolise publications, straight edges between papers represent citations. $X = \{x_0, \dots, x_n\}$ are papers referenced by P , $Y = \{y_0, \dots, y_m\}$ are papers citing P . Dotted edges symbolise observed relationships between publications. Group A contains distances between pairs of referenced (X) and citing papers (Y). Group B contains distances between referenced papers (X) and P . Group C contains distances from P and citing papers (Y). Group D contains distances between pairs of referenced papers (X). Group E contains distances between pairs of citing papers (Y).	79
6.2	Distribution of number of references and citations for seminal (blue circles), survey (orange triangles) and uninfluential (green crosses) publications.	87
6.3	Histograms of number of seminal (blue), survey (orange) and uninfluential (green) publications over the years.	88

6.4	Histograms over number of referenced (upper) and citing (lower) papers for seminal (blue), survey (orange) and uninfluential (green) publications over the years.	89
6.5	Percentages of top five topics (and all others) for publications P from classes seminal, survey and uninfluential.	91
6.6	Simplified graphical depiction of methodology.	93
6.7	Box plot of value distribution of feature $sumD$ derived from differences computed with inner products of unstemmed LDA document representations for seminal, survey and uninfluential publications in their citation networks.	98
6.8	Box plot of value distribution of feature $sumA$ derived from differences computed with cosine distance of stemmed tf-idf document representations for seminal, survey and uninfluential publications in their citation networks.	99
8.1	Schematic overview of our approach. The left part depicts the expert search task, the right part depicts the set of reviewers assignment task.	207
8.2	Numbers of reviewers which are relevant for single manuscripts per dataset.	222
8.3	Numbers of manuscripts, for which single reviewers are relevant per dataset.	223
9.1	Top: A simplified version of the flow network constructed by DiveRS. Only the depicted edges between neighbouring layers allow flow. Background nodes in the dotted ellipse are used to ensure diversity in the professional background, those in the dashed ellipse are used to enforce diversity in the continent of the assigned reviewers and those in the densely dotted ellipse guarantee diversity in seniority. Bottom: Lower (lb) and upper bounds (ub) of incoming (I) and outgoing flow (O) per edge as well as the general flow via a specific node type with ability a , demand λ , lowest load l and amount of flow depending on the node type t	252
9.2	Reviewer assignment results using manual evaluation, displaying average diversity (x-axis) against the number of suitable reviewers (y-axis). The error bars correspond to the standard deviation per method. Results are reported on the two datasets combined.	262

Preamble

1. Introduction

“A lot of the time I get obsessed by little nerdy things in my corner that no one else is interested in.”

– Bjork

Researchers are usually tackled with several tasks revolving around bibliographic metadata. For example, they need to search or browse for papers to read, authors to collaborate with, or venues to submit their papers to, identify influential publications without reading them first to narrow down their reading list or assign reviewers to submissions to a conference if they are conference chairs.

This dissertation focuses on the above-mentioned scenarios and presents the following four concrete tasks which are all associated with bibliographic metadata in order to help support researchers: *i)* general information search and exploration, *ii)* identification of influential publications, *iii)* scientific paper recommendation and *iv)* reviewer set recommendation for publications. In the following we will discuss them separately, describe their importance, how we strove to solve them and which findings we made.

1.1 Information Search and Exploration

Refers to: “*SchenQL: in-depth analysis of a query language for bibliographic metadata*”, Chapter 5.

Research usually starts with a literature review of relevant papers to read or cite, authors to consider, conferences and journals to submit to and institutions to observe. Digital libraries such as [dblp](https://dblp.uni-trier.de/)¹ [32] or Semantic Scholar² provide easy access to bibliographic metadata and offer a keyword-based search that also supports restriction of several attributes. Nevertheless, these interfaces do not support the expression of convoluted information needs such

¹<https://dblp.uni-trier.de/>

²<https://www.semanticscholar.org/>

as “Which are the five most cited articles written by person P about topic T after year Y ?”. Usage of a structured query language such as SQL could overcome these limitations but would require significant efforts from users.

To close this gap, we introduced SchenQL, a domain-specific query language and graphical user interface on bibliographic metadata. SchenQL queries are designed to resemble natural language while incorporating domain jargon. By offering domain-specific functions, we provide users with a straightforward way of satisfying possibly complex information needs. Our system supports information search via queries and exploration using the GUI for domain-experts as well as casual users of digital libraries. In SchenQL the above-mentioned query could be formulated as `MOST CITED (ARTICLES WRITTEN BY "P" ABOUT "T" AFTER Y) LIMIT 5`.

Our goals for SchenQL could be summarised as follows: make information access uncomplicated for domain experts as well as casual users of digital libraries, and support formulation of complex information needs.

The SchenQL query language uses five basic bibliographic entities as building blocks, the so-called base concepts: `publications`, `conferences`, `journals`, `persons` and `institutions`. Some of them can be refined by a specialisation (e.g. `books` instead of `publications`). Filters (e.g. `WRITTEN BY`) restrict base concepts to a subset, and functions (e.g. `MOST CITED`) can aggregate data or offer domain-specific functionalities. We implemented the SchenQL compiler with MySQL as the underlying database engine and SQL as the target language.

We evaluated the following hypotheses:

- H_1 MySQL as an underlying database engine with SQL as the target language is more suitable than the combination of Neo4j and Cypher as the target language.
- H_2 The SchenQL compiler’s constructed queries’ performance is comparable to that of manually formulated queries.
- H_3 Users operating the SchenQL’s command line interface achieve higher correctness, lower perceived difficulty to formulate queries, and lower time compared to usage of SQL.
- H_4 SchenQL is as suitable for domain-expert as it is for casual users.
- H_5 The GUI is highly suitable for users not familiar with structured query formulation.

We found that all hypotheses could be verified. We were able to achieve our aforementioned goal of providing a method for easy information access

for all users of digital libraries. SchenQL is also a suitable option to formulate possibly complex information needs.

1.2 Identification of Influential Publications

Refers to: “*Evaluating Semantometrics from Computer Science Publications*”, Chapter 6.

With the steadily increasing number of scientific publications, researchers should be supported in focusing on observing influential or seminal ones to maximise the use of their limited time and attention. Automatic approaches identifying important publications oftentimes focus on the acquired citations of papers [48] but disregard problems which might occur in this context: self-citations [42], citation practices which are area-dependent [43], different reasons for citing papers [19], uncited influences [19] and especially the non-existence of citations for new papers [53]. Another factor to consider here are the similarly high numbers of citations both for seminal works and survey papers, which might make it hard to differentiate between these two [44].

From this context we introduce the two concrete tasks of identification of influential publications: *i*) the classification of a paper with its complete citation network as seminal, survey or uninfluential, and *ii*) the prediction of the class a paper lies in if only information is observed, which is present at the time of publication. We analyse the already established method of semantometrics for these tasks and compare it to more straightforward methods. To evaluate the tasks, we present a novel and publicly available dataset, SUSdblp³. This dataset contains each 660 papers from the classes seminal, survey and uninfluential as well as their citations and references. For all papers, citations and references the dataset contains their concatenated titles and abstracts, publication years, numbers of citations, and time normalised citation scores.

Our goals for this work can be summarised as the identification of the best possibilities of using semantometrics for the identification and prediction of classes of publications and the introduction of a suitable dataset for both tasks.

Semantometrics observes features derived from distances between (groups of) publications. Distances between papers are grouped as shown in Figure 1.1: ones between references of papers (group *D*), between citations of papers (group *E*), between the paper and its references (group *B*), between the paper and its citations (group *C*) as well as references and citations of

³<https://zenodo.org/record/3693939>

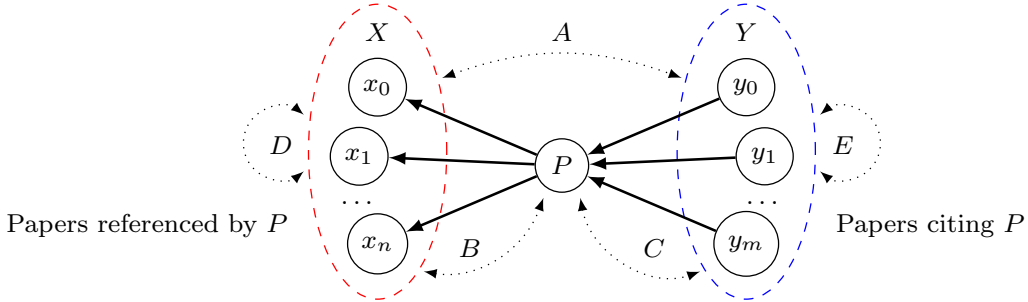


Figure 1.1: Neighbourhood of publication P . Nodes symbolise publications, straight edges between papers represent citations. $X = \{x_0, \dots, x_n\}$ are papers referenced by P , $Y = \{y_0, \dots, y_m\}$ are papers citing P . Dotted edges symbolise observed relationships between publications. Group A contains distances between pairs of referenced (X) and citing papers (Y). Group B contains distances between referenced papers (X) and P . Group C contains distances from P and citing papers (Y). Group D contains distances between pairs of referenced papers (X). Group E contains distances between pairs of citing papers (Y). [29]

a paper (group A) are observed. From the set of distances for the different groups we derive features, e.g. the sum of distances in C , average of distances in A , minimum or maximum of distances in B . We use these features as input for a classifier which predicts whether a paper (P in Figure 1.1) is from class seminal, survey or uninfluential. In our experiments, we use seven different well-established classifiers such as random forests or k-nearest neighbours. Publications are represented via stemmed or unstemmed tf-idf [41], Doc2Vec [31], stemmed or unstemmed LDA [10] or BERT [16] vectors of their textual content and publication years. We examine four different distance measures for the calculation of distances in the five groups. In case of only using information which is available at publication time, we disregard the features derived from groups where citations are relevant (see groups A , C and E in Figure 1.1).

We evaluated the following implicit hypotheses:

- H_6 Semantometrics is applicable to computer science publications.
- H_7 Usage of multiple or all semantometrics features produces higher classification accuracy than using single features only.
- H_8 Semantometrics is more useful than one-vector-representations of publications' content for their classification and prediction as seminal, survey or uninfluential.

H_9 Our newly introduced dataset SUSdblp is robust and independent of the actual years of papers.

We were able to verify all of the abovementioned hypotheses, except H_8 . The best combination of features derived from semantometrics still produced worse results compared to only using one-vector representations for the citation network. Our overall goals of identifying the best combinations for semantometrics for both introduced tasks as well as providing a suitable dataset for them could be fulfilled.

1.3 Scientific Paper Recommendation

Refers to: “*Scientific Paper Recommendation Systems: a Literature Review of recent Publications*”, Chapter 7.

With the ever-increasing number of scientific publications, the amount of possibly relevant papers [2] increases. Relevancy could e.g. be defined as papers which are related to current research [15] or ones to read to understand the current state-of-the-art [54]. To help identify which papers are relevant, so-called paper recommendation systems have been proposed. There currently is a vast number of publication recommendation approaches [40, 56, 37, 20, 51] and this amount seems to be increasing over the last years. To the best of our knowledge, the three latest⁴ surveys which summarise all the present directions were accepted for publication or published in 2019 [3, 33, 47] and therefore do not analyse more recent works.

Thus, to close this gap we present a literature review of paper recommendation approaches from January 2019 to October 2021. We categorise and briefly describe current systems, discuss datasets used in their studies, evaluation methods and provide a general overview of possible pitfalls.

Our goals for this survey can be summarised as follows: show the contrasting directions in current approaches, their datasets and diverse evaluation procedures. Additionally, we wanted to highlight challenges which are (still) encountered in or disregarded by recent publications.

We found that many problems, which have been defined in earlier work are no longer considered by current approaches, but there also were numerous novel aspects which could be improved. In general, we were able to achieve our goals: we introduced a novel classification of paper recommendation approaches, discussed their used datasets and evaluation measures, and analysed existing issues with these works from a current standpoint.

⁴As of January 2022.

1.4 Reviewer Set Recommendation for Publications

Refers to: “*RevASIDE: Evaluation of Assignments of Suitable Reviewer Sets for Publications from Fixed Candidate Pools*”, Chapter 8
and
“*Diverse Reviewer Suggestion for Extending Conference Program Committees*”, Chapter 9.

Scientific writing heavily relies on peer review as a form of insurance of quality and soundness of papers, as well as feedback to authors. The overall goal of peer review is to provide a quantifiable assessment of papers, which ensures that published works fulfil a minimum standard. Peer review should always be conducted by researchers knowledgeable in the area of the manuscript. In case of a conference, the submitted manuscripts have to be assigned to a finite predefined set of reviewer candidates, the so-called program committee (PC). The assignment between manuscripts and groups of reviewers is usually conducted by a program chair and can incorporate bidding information provided by reviewers. Problems encountered in this task are the tight time frame [14] in which submissions and PC members need to be matched and the overall complexity of the task. A reviewer is only able to review a finite number of manuscripts, they cannot have conflicts of interest with the manuscripts’ authors, they need to be knowledgeable in the area and all submissions need to be assigned a certain number of PC members.

To help with the assignment of reviewers to manuscripts there are many automatic approaches with diverse definitions of suitability of reviewer sets for manuscripts [25, 35, 39, 49]. We present two methods tackling this problem on different levels, RevASIDE and DiveRS.

RevASIDE

Suitability of reviewers in sets can be defined via properties the PC members in the set need to have. Reviewers should not have conflicts of interests [39] with authors of manuscripts to be independent from them. They should be experienced [25, 39] in the area of the manuscript to provide deep reviews. Reviewers should have authority [25, 39] in the area of the manuscript, they should be recognised in the target domain to provide credible assessments. Ideally, reviewers are interested [25] in the area of the manuscript, so they accept to review the paper and are up-to-date in the target area. The single reviewers in a set should also have diverse [35] expertise to enable broader

reviews. Current reviewer assignment systems do not support all of these properties at once.

We introduce RevASIDE, a reviewer assignment system for single manuscripts which incorporates the discussed aspects of no conflicts of interest, expertise, authority, interest, and diversity. Additionally, it ensures the diversity of reviewer sets in seniority, the sets should not be composed of senior researchers only to distribute the reviewing load more evenly and enable junior researchers to learn. Another aspect RevASIDE considers is the independence of reviewers in sets. Assigned individuals for a single manuscript cannot be affiliated with each other to ensure broader reviews [35]. Our approach does not require the step of reviewers bidding for manuscripts. To evaluate our approach we introduce three novel publicly available datasets for the task of reviewer recommendation, namely MOL’17, BTW’17 and ECIR’17⁵. These datasets contain different numbers of manuscripts accepted at the various conferences as well as the reviewer pools. Reviewers are represented by embeddings of their publications as well as their years, page length, core rank of the venue it was published in and the number of citations from the area of computer science as of 2016. For manuscripts, we provide their content as vectors. For a portion of manuscripts we also included ratings of reviewers’ fit to review them as well as ranked sets of reviewers.

The goal we pursued with RevASIDE can be summarised as the presentation of a fully automated reviewer set recommendation approach for single manuscripts which focuses on sets with expertise, authority, interest as well as diversity in expertise and seniority. We also prioritised the introduction of datasets for the task of reviewer recommendation.

Our approach consists of two parts (see Figure 1.2), an expert search part where reviewer candidates are found and a subsequent part, where those candidates are assembled to reviewer sets for single manuscripts. Authors’ publications and the manuscripts submitted to a conference can be represented as tf-idf, Doc2Vec or BERT vectors. Similarities between authors’ papers and manuscripts are aggregated with voting techniques such as sum, minimum or maximum of these similarities. In the second step, LDA and tf-idf vectors of authors’ papers sufficiently similar to a manuscript represent an authors’ profile. For combinations of authors, the set achieving the highest score for a combination of scores for expertise, authority, interest, diversity and seniority is identified.

We evaluated the following hypotheses:

H_{10} Restricting the number of possible reviewer candidates is useful for the expert search task.

⁵<https://zenodo.org/record/3826701>

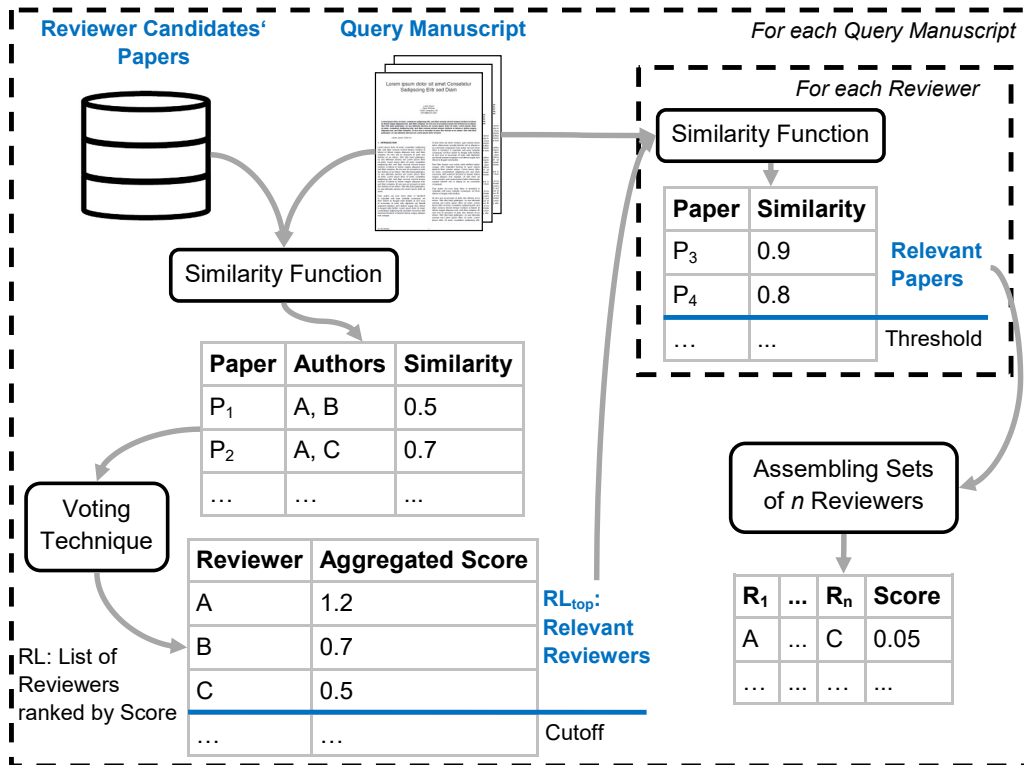


Figure 1.2: Schematic overview of our approach. The left part depicts the expert search task, the right part depicts the set of reviewers assignment task. [30]

H_{11} More advanced document vector representations produce better results in MAP, Precision@10 and nDCG in the first step of our approach compared to tf-idf.

H_{12} Different parameterisations of RevASIDE produce significantly different results.

H_{13} Using full texts of manuscripts in the second step produces worse assignments compared to only using their technical sections.

H_{14} The expert search part is helpful for the reviewer set assignment part.

H_{15} RevASIDE supports different reviewer set sizes.

H_{16} Human assessment confirms RevASIDE's usefulness.

Most of the hypotheses could be verified. Hypotheses which we were unable to verify were H_{11} and H_{13} . We found the contrary: tf-idf produced

the best results in the first step, and full texts of manuscripts represented submissions’ content better. We fulfilled our goals with the presentation of the fully automated reviewer set recommendation approach RevASIDE which considers authority, seniority, interest, diversity and expertise of reviewer sets as well as the three publicly available datasets for the task of reviewer recommendation.

DiveRS

To the best of our knowledge, all current⁶ paper recommendation systems assume the PC they use to assign reviewers to manuscripts is already perfectly composed and does not need to be modified. The program committee of a conference needs to grow and change annually to compensate for former reviewers being unavailable [21] and avoid unchanging perceptions of interesting topics [17]. As there is no way of reliably predicting the topical composition of manuscripts submitted to a conference, there could be a mismatch between the PC and incoming submissions. This mismatch could further lead to assignments of reviewer sets which do not fit the manuscripts. Reviewers not interested in the area of an assigned manuscript could review less favourably [38], reviewers inexperienced in the area of innovative and complex submissions might reject them [9] or fail to find errors [45].

To overcome these issues we propose the novel reviewer coverage problem which contains the PC extension to fit the incoming submissions as well as the reviewer assignment task. New reviewer candidates are recommended such that the actualised PC fits the incoming manuscripts. We present DiveRS, an explainable flow-based reviewer suggestion and PC extension approach tackling this task. Our approach focuses on assigning diverse reviewer sets to all submissions of a conference while complying overall load constraints of reviewers. Assigned reviewer sets are diverse in their professional background, locations and in the seniority of reviewers.

Our goal for this work can be summarised as solving the reviewer coverage task by extending the composition of the PC according to incoming submissions such that all manuscripts get diverse reviewer sets assigned.

DiveRS is a reviewer suggestion and PC extension approach which iteratively identifies submissions likely being assigned insufficient reviewers and recommends novel PC members to fit them. These new reviewers stem from an extended reviewer candidate pool and satisfy currently underrepresented diversity aspects in professional background, location and seniority. The approach solves this task as a constrained optimisation problem by constricting

⁶As of January 2022.

a multi-layer flow network. Here, reviewers' individual upper and lower reviewing bounds in terms of allocated time for the reviewing process are always considered. Reviewers in a set need to belong to industry and academia as professional background, they cannot all come from the same continents, and the sets need to contain at least one senior reviewer.

We evaluate the following implicit hypotheses:

H_{17} DiveRS is able to construct diverse and fair reviewer assignments for all submissions of conferences.

H_{18} PC chairs of former conferences confirm the suitability of reviewer sets for manuscripts recommended by DiveRS.

H_{19} The former PC chairs are satisfied with the explanations and the newly recommended reviewers.

Only the last hypothesis H_{19} could not be verified. We encountered problems with the manual evaluation of suggested reviewer candidates. PC chairs' agreements on the relevance judgements were low but when observing only relevant suggestions, convincingness and usefulness of explanations increases considerably. Nevertheless, our approach is able to estimate the confidence of suggestions. We were able to achieve our goals of presenting a solving for the novel reviewer coverage task with DiveRS. Our approach extends a current PC with respect to the submitted manuscripts and ensures the diversity of reviewers in assigned sets in their professional background, location and seniority.

1.5 Structure

The first part of this dissertation is structured as follows: first, we discuss the summarised findings of the five papers in light of the current scientific landscape in Chapter 2 before the following Chapter 3 highlights the relations between the different papers. The preamble is concluded with Chapter 4 which describes directions to further extend the works presented.

The second part 4 of the dissertation consists of the five publications representing the tasks. To conclude this work, the appendix encompasses a curriculum vitae A and a complete list of publications B.

2. Summarised Findings in Current Scientific Landscape

“Knowing comes from learning, finding from seeking.”
– Vaddey Ratner

As this dissertation encompasses five different papers from four topics, we structure the presentation of the summarised results of these works in the current scientific landscape accordingly.

2.1 Information Search and Exploration

With SchenQL we tackled the task of easy information search and exploration for casual and expert users of digital libraries. We achieved our goal of making information access uncomplicated for domain experts and non-expert users of digital libraries. SchenQL also supports the formulation of possibly complex information needs.

Our investigations revealed that users in general were satisfied with our query language and graphical user interface SchenQL. To the best of our knowledge, SchenQL currently is the only domain-specific query language directed towards bibliographic metadata. The search interface on bibliographic metadata most comparable to the functionality of SchenQL is GrapAL¹ [8]. A main difference between SchenQL and GrapAL is the skill it takes to formulate a query. For GrapAL, users need to construct queries in the declarative graph query language Cypher. In our evaluation, the users rated the handling of the SchenQL system as extremely easy learned.

SchenQL is a suitable option compared to SQL for domain-experts as well as non-experts for typical tasks encountered in digital libraries. Existing dig-

¹GrapAL is no longer supported: <https://allenai.github.io/grapal-website/>.

ital libraries such as Bibsonomy² [7], dblp, Google Scholar³, ResearchGate⁴ or Semantic Scholar do not provide the functionality to formulate queries as complex as ones supported by SchenQL. Nowadays, digital libraries often-times provide visualisations, e.g. citation influence graphs or bar charts of citations per year. We did not encounter regular participants in our user studies asking for (more) visualisations while the dblp staff from our expert interviews wished for diverse sophisticated visualisations.

While many everyday queries on bibliographic metadata can be constructed in SchenQL, certainly not all of them can be easily answered using our system. SQL is much more powerful than our domain specific query language, and thus can be used to find answers to all queries from the domain.

2.2 Identification of Influential Publications

With semantometrics we observed the two tasks of *i*) identification of influential publications while using all available information in the citation networks, and *ii*) only incorporating information present at the time of publication to determine the class of a paper as seminal, survey or uninfluential. We achieved our goals of identifying the best-performing configurations for semantometrics for these two tasks and provided a suitable dataset.

In general, semantometrics shows high potential in classifying influence classes of publications and especially predicting the class when only observing information which is available at the time of publication. However, for computer science publications, it cannot surpass simply using all information available in the citation neighbourhood of a paper for both tasks. The two tasks can be conducted by only observing publications from the citation neighbourhood which lie in the same area as the paper for which the influence class is identified. Another factor to consider is the high computational cost associated with usage of semantometrics which could be bypassed by using simple document embeddings as the input for classifiers.

To the best of our knowledge, our work was the first to define of the two tasks as a ternary classification problem that we strove to solve with semantometrics. We published SUSdblp, a dataset suitable to evaluate this task, which could also be used further to evaluate related tasks such as citation count prediction based on contents of publications.

We found that the broadness in which a paper cites other papers is highly indicative of its influence class. This property has already been observed by

²<https://www.bibsonomy.org/>

³<https://scholar.google.com/>

⁴<https://www.researchgate.net/>

prior work [48].

In computer science, analysing the papers from the citation network which stem from the same area as the main paper is sufficient to determine its influence class.

This work in total could be highly relevant for the area of literature on COVID-19 which is explosively increasing in their numbers [18], especially when it comes to pre-prints [11]. Literature on COVID-19 tends to balance over-use of positive wording and uncertainty in their abstracts [11]. This could hinder domain non-experts from finding truly relevant manuscripts. The area-specific citation networks of papers could be analysed to estimate which papers will be impactful at the time of publication only. This could help overcome the information overload for researchers working in the area, physicians searching for the best treatment for their patients, politicians deciding on health-related regulations and laws, as well as journalists who try to summarise or rephrase the current main findings for the broader public. Such a system could help all these user groups focus on fewer papers of possibly higher quality.

2.3 Scientific Paper Recommendation

In our literature review, we observed approaches from January 2019 to October 2021 which tackle the task of scientific paper recommendation. We achieved our goal of discussing the classification and methods of the paper recommendation systems, used datasets, as well as evaluation measures. We also observed possible shortcomings of these papers.

There currently is a vast number of publication recommendation approaches and this amount seems to be increasing over the last years. The three latest surveys which summarise all the present directions were accepted for publication or published in 2019 [3, 33, 47] and do not analyse more recent works. In our literature review of paper recommendation systems from this time frame, we found a discrepancy between existing classification methods for them and prevalent current ones. While approaches were usually labelled as e.g. content-based filtering, collaborative filtering, graph-based, or hybrid approaches [6], the majority of the current systems belong to the group of hybrid approaches, as they no longer utilise only one technique. As this classification is not as meaningful to differentiate between classes of systems, we propose a multi-class description of approaches targeting input, techniques, and data components used in the different works.

We found that there is a plethora of (publicly non-available) datasets which current approaches use. This makes it hard to compare their perfor-

mance. Additionally, this highlighted the need for an open, reusable dataset. Another issue which makes comparability of current approaches difficult is the diversity of reported evaluation measures.

Open issues we encountered which should be prioritised for future paper recommendation approaches are fairness and explainability of recommendations as well as conducting user studies.

With the explosive increase in papers on COVID-19 our analysis could be helpful in designing a novel paper recommendation system especially for this dataset. Here, special caution should be directed towards approaches which consider historical user interaction data due to the cold start problem [1]. The time frame might be too narrow to allow for meaningful accumulation of this data. Barolli et al. [4] suggest the combination of content-based methods with graph-based techniques for recommending literature on COVID-19. An implementation could be tested on the dataset presented by Barros et al. [5] which holds multilingual COVID-19 literature, users and their artificially inferred ratings for the recommended papers. To the best of our knowledge, there currently does not exist a paper recommendation system specifically targeting this highly relevant domain. User groups for such a system could be general researchers, physicians, politicians or journalists with their different information needs.

2.4 Reviewer Set Recommendation for Publications

We tackled the task of reviewer set recommendation for publications from two different angles. First we conducted recommendation of reviewer sets for single publications while focusing on their authority, seniority, interest, diversity as well as expertise of reviewer sets with our approach RevASIDE. The second approach, DiveRS, tackles the superordinate reviewer coverage problem, which strives to extend a current program committee to support adequate review of all submissions. Assigning suitable and diverse sets of reviewers for all manuscripts submitted to the conference is part of the observed problem.

RevASIDE

Our goals for RevASIDE were the presentation of a fully automated reviewer set recommendation approach for single manuscripts which incorporates authority, seniority, interest, diversity and expertise of reviewer sets as well as

the publication of publicly available datasets for the two tasks of reviewer recommendation and the broader expert search.

With RevASIDE we found out that using tf-idf is more suitable for the first part of the approach (the expert search task) compared to more sophisticated document vector representations such as BERT. As a general outcome, this means that simpler document representations might be able to capture specialised concepts better than complex document embeddings. Embeddings might sometimes dilute the presence of words in authors’ profiles indicating experience in specialised techniques, which then could lead to recommendation of non-optimal experts.

In general, we found that the expert search task is a suitable first step for the reviewer assignment task. RevASIDE produces the reviewer sets achieving the highest scores compared to different baselines, and a manual expert evaluation confirmed our approach’s suitability. Our approach can be applied for the construction of reviewer sets of different sizes. These results hint at the profitableness of using more aspects in the recommendation process than mere expertise as seen with several other works [12, 28, 55, 57, 26]. Additional incorporation of authority, interest, diversity and seniority of reviewer sets showed to be worthwhile in our experiments.

We found that some of our considered aspects are related (e.g. authority and expertise) or conflicting even (e.g. interest and diversity). Thus, weighing these properties manually could be difficult.

We presented three datasets, MOL’17, BTW’17 and ECIR’17. These datasets are publicly available and partially contain single reviewer and reviewer set relevancy scores for manuscripts annotated by an expert researcher. The datasets could be used for evaluation of other reviewer assignment approaches or for the assessment of the expert search task for publications.

DiveRS

With DiveRS we achieved our goal of presenting a solution to the reviewer coverage task which produces reviewer assignments for all submissions to a conference which are diverse.

Our flow-based, explainable approach DiveRS solves the novel reviewer coverage task: given a PC, a set of manuscripts and an extended reviewer candidate pool suggest members from this pool for inclusion in the PC such that *i*) reviewers’ expertise is sufficient to cover the submissions, *ii*) there are no conflicts of interests between authors of manuscripts and assigned reviewers, and *iii*) reviewer load constraints are not violated.

Veto+ [13] tackles the related problem of domain expert set expansion

by incorporating meta-paths from scholarly knowledge graphs. It can also be applied to our problem. The main difference between DiveRS and Veto+ lies in the additional constraints associated with our approach, DiveRS also considering the construction of reviewer assignments as part of the problem, and in the property that Veto+ extends the reviewer pool with novel members which are similar to the already existing ones.

We found that our approach constructs reviewer assignments to manuscripts which are comparable in fairness to the current state-of-the-art approach, which specifically optimises towards this aspect. Our assignments were also more diverse. Real assignments are much less fair and diverse than ones constructed by DiveRS. Through a manual evaluation with actual PC chairs, we found a trade-off between suitability and diversity of reviewer sets. We encountered difficulties in the manual assessment of the suitability of reviewer sets, PC chairs often disagreed on suitability.

Real assignments as well as the state-of-the-art approach did not assign all PC members with manuscripts to review. This might lead to discontent and disappointment of reviewer candidates. Our approach provides the option to enforce each PC member to be assigned at least one submission out of courtesy to value their willingness to review for a conference.

As many reviewer recommendation systems do not explicitly consider diversity of their PCs and constructed reviewer sets a priority [12, 25, 26, 28, 39, 52, 55, 57, 55], our approach provides a viable alternative. It explicitly ensures all assignments to contain at least one reviewer from industry and academia (possibly the same), locations from reviewers not all being the same and the presence of at least one senior reviewer per set.

For the reviewer suggestion part, we found that it is generally difficult to evaluate. Again, PC chairs did not agree on the relevance of novel reviewer candidates to include in the PC or the convincingness of explanations as to why they were suggested. Nevertheless, DiveRS was able to rank the suggestions according to their relevance for inclusion. If only relevant reviewer suggestions are observed, the usefulness and convincingness of the explanations for these candidates increases drastically.

3. Relations between Papers

“Everything is connected to everything else”
– Barry Commoner

The five presented works are not directly dependent on each other, as the four tasks they represent are mostly distinct in the everyday lives of researchers. However, they are related and the techniques or approaches could be combined pairwise. Combinations could increase the scientific value the components generated individually.

Information search and exploration as encountered in SchenQL can be related to semantometrics, paper recommendation and reviewer recommendation in the following ways:

- SchenQL can incorporate semantometrics either for ranking papers or to indicate importance of papers in detail views. Another option could be the reformulation of semantometrics on author networks. Using features derived from textual similarity between documents could help in estimating the importance of single researchers. Persons can be represented by a concatenation of all their authored papers, the citation network of authors they cited or were cited by provides the environment in which the features could then be computed. The prediction target for authors’ importance could e.g. base on their h-index [30], g-index [22], or actual citation counts.
- SchenQL can incorporate paper recommendation either by querying for papers related to users or input papers: **PUBLICATIONS recommended for PERSON named x**. Another application of paper recommendation in SchenQL could be the display of related papers as a feature in original papers’ or persons’ detail views.
- The SchenQL ecosystem could support reviewer recommendation and expert suggestion. With a query consisting of a manuscript, a number of required reviewers (for expert search this would be 1) and a set of researchers as possible reviewer pool (e.g. recent authors of

papers which appeared in a specific conference: PERSONS authored PUBLICATIONS after 2015 appeared in CONFERENCE x) sets of recommendations would be provided to the user. The manuscript for which experts are recommended should be a short textual description such as an abstract if the functionality tries to model reviewer recommendation (e.g. for journal submissions or grant reviewing). Initial manuscripts could also be pre-existing papers if a user searches for collaborators for future work.

Semantometrics can be related to paper recommendation and reviewer recommendation in the following ways:

- Semantometrics could help in paper recommendation to rank more influential papers higher for users. This could increase satisfaction of users who strive to read widely recognised publications. Another possibility in this scenario would be the recommendation of fitting but less influential publications for users to possibly help users discover hidden but nevertheless relevant gems.
- Semantometrics can be used to identify important papers in terms of numbers of citations. In paper recommendation the goal is also to identify relevant or important papers so this could be a connecting point. In paper recommendation importance can be defined by less quantifiable aspects, such as users' individual satisfaction. Incorporation of semantometrics in the paper recommendation process could increase overall relevancy and usefulness of recommendations.
- Semantometrics could be used for reviewer recommendation to identify the influential publications of reviewers. These papers could then have more weight in the construction of the reviewer's profile out of all their authored works. Another option could be the application of semantometrics as described before for author networks. The resulting influence labels for researchers could for example help in assigning reviewer sets for manuscripts of diverse influence.

Paper recommendation is related to reviewer recommendation in the following way: several of the open challenges which we discussed for current paper recommendation approaches can also be applied to reviewer recommendation. Evaluations in reviewer recommendation should always be conducted with users but constructed reviewer sets' suitability oftentimes does not get rated by actual experts or PC chairs [22, 23, 27]. Comparing results of approaches against each other or against artificially constructed quality factors does not necessarily capture properties of reviewer sets which PC chairs

would value. While fairness [27, 50] or diversity [22, 34, 35] of recommended reviewer sets sometimes are incorporated, explainability of sets could be considered a blind spot. Our encountered problems with the non-existence of public datasets have already been briefly mentioned in Section 9.2 so this would also be an open challenge from paper recommendation applicable to the area of reviewer recommendation.

Our two **reviewer recommendation** approaches could be combined with each other: DiveRS can be seen as a future work defined in RevASIDE. In RevASIDE we mentioned that observing reviewer coverage and PC extension could be a worthwhile task and DiveRS does exactly target this problem. However, it does not utilise the RevASIDE methodology in its core for assignment of reviewers to sets. The incorporated aspects defined in RevASIDE were authority, seniority, interest, diversity and expertise of reviewers, while DiveRS focuses on diversity in different aspects such as seniority as well as expertise represented by topical fit between reviewers' profiles and manuscripts. Current interests of reviewer candidates could be modelled into DiveRS by increasing the weight of more recent publications of reviewers and discounting importance of older papers while constructing reviewers' profiles. Authority could be another aspect to build into the flow-based approach, similar to the realisation of seniority.

More: As all different approaches, ideas or techniques can be pairwise combined to possibly increase usage, the incorporation of multiple or all of them into a single system, e.g. into the SchenQL ecosystem is also possible.

4. Future Work

“It’s not enough to be busy; so are the ants.

The question is: what are we busy about?”

– Henry David Thoreau

With the approaches presented for the different tasks, there still is lots of room for improvement or further research. Main aspects to pursue for each of the different tasks could be the following: SchenQL could be extended such that the language supports more sophisticated query types such as PageRank of persons who authored publications at a venue, the identification of hubs and authorities, centrality of authors or also allowing users to define their own functions. Even though we did not encounter a specific request for more visualisations, it could be experimented with colour-coded topics or graph visualisations as current digital libraries usually incorporate such graphics.

Semantometrics could be reevaluated on a different dataset and a different target domain to make assumptions on its overall validity and generalisation capability. Additionally, further facets such as entropy of distances in the five groups could be incorporated.

Concerning paper recommendation systems, our survey highlighted several open challenges for new approaches to consider. Construction of a paper recommender system which complies to all defined desirable goals would be worthwhile. Another direction could be the construction of a benchmarking system to automatically compare approaches and evaluation measures. With it, already existing approaches could be reevaluated on a single dataset to ensure comparability, identification of the current state-of-the-art methods as well as estimate suitability of the different evaluation measures.

RevASIDE could be extended by weighting already existing aspects, observing whole venues at once, or by considering fairness of sets. As another future direction, we defined the analysis of gaps in the program committee to try to suggest new reviewers covering them. This task was pursued with DiveRS.

To extend DiveRS, bidding information of the original PC could be incorporated to identify manuscripts which could require the inclusion of ad-

ditional reviewers. Shah et al. [46] already mentioned the combination of reviewers' bids with similarities between manuscripts and reviewer profiles as an open problem. Another direction here would be the modification of the flow network such that each reviewer set also has to contain a junior reviewer, or that newly included reviewers need to be assigned to at least two separate manuscripts.

Aside from these future directions already mentioned more in depth in the single publications, the different approaches, techniques and ideas could be pairwise combined as described in Chapter 3. The most fruitful or interesting pairwise extensions could be these following routes: Semantometrics could also be applied for authors to estimate importance, e.g. the class their h-index [30] or citation count lies in. A single author would be represented by a concatenation of their authored papers. The citation network of their cited researchers could then be observed to compute values for all features by calculating similarities between the authors.

It could be worthwhile to only combine semantometrics with paper recommendation systems or SchenQL to identify and recommend or re-rank publications based on their value. The merit of simply using semantometrics for paper recommendation where the input is either a single publication or a user profile (their combined authored papers) could e.g. be the suggestion of less influential but fitting publications to boost serendipity of a system or to recommend the most influential papers to help with users' satisfaction of the system.

DiveRS could be extended such that all different aspects considered in RevASIDE are incorporated in the flow-based network. Reviewer sets could be constructed such that they always need to contain at least one reviewer with high authority (e.g. estimated by an area dependent h-index [30]) in the fields of a manuscript. Recent publications of reviewers should influence their profile more, such that the profiles better represent current interests of researchers.

Currently, researchers usually use specialised systems for different tasks or more complex tasks consisting of subtasks, e.g. to explore scientific venues in depth and assign reviewers to manuscripts. A logical next step could be the combination of all of the approaches presented here into a single information system operating on bibliographic metadata, such as the SchenQL ecosystem. Using such a single system would prevent switches in the working sphere [36] which are problematic as they can negatively affect a user's speed and cognitive load [24].

Bibliography

- [1] Rabaa Alabdulrahman and Herna Viktor. Personalised recommendation systems and the impact of COVID-19: Perspectives, opportunities and challenges. pages 295–301, 01 2020.
- [2] Anas Alzoghbi, Victor Anthony Arrascue Ayala, Peter M. Fischer, and Georg Lausen. Pubrec: Recommending publications based on publicly available meta-data. In Ralph Bergmann, Sebastian Görg, and Gilbert Müller, editors, *LWA '15*, volume 1458 of *CEUR Workshop Proceedings*, pages 11–18. CEUR-WS.org, 2015.
- [3] Xiaomei Bai, Mengyang Wang, Ivan Lee, Zhuo Yang, Xiangjie Kong, and Feng Xia. Scientific paper recommendation: A survey. *IEEE Access*, 7:9324–9339, 2019.
- [4] Leonard Barolli, Francesco Di Cicco, and Mattia Fonisto. An investigation of Covid-19 papers for a content-based recommendation system. In Leonard Barolli, editor, *3PGCIC '22*, pages 156–164, Cham, 2022. Springer International Publishing.
- [5] Márcia Barros, Pedro Ruas, Diana Sousa, Ali Haider Bangash, and Francisco M. Couto. Covid-19 recommender system based on an annotated multilingual corpus. *Genomics & Informatics*, 19(3):e24, 2021.
- [6] Jöran Beel, Bela Gipp, Stefan Langer, and Corinna Breitingner. Research-paper recommender systems: a literature survey. *Int. J. Digit. Libr.*, 17(4):305–338, 2016.
- [7] Dominik Benz, Folke Eisterlehner, Andreas Hotho, Robert Jäschke, Beate Krause, and Gerd Stumme. Managing publications and bookmarks with bibsonomy. In Ciro Cattuto, Giancarlo Ruffo, and Filippo Menczer, editors, *HYPertext '09*, pages 323–324. ACM, 2009.
- [8] Christine Betts, Joanna Power, and Waleed Ammar. Grapal: Connecting the dots in scientific literature. In Marta R. Costa-jussà and Enrique Alfonseca, editors, *ACL '19*, pages 147–152. Association for Computational Linguistics, 2019.
- [9] Ken Birman and Fred B. Schneider. Viewpoint - program committee overload in systems. *Commun. ACM*, 52(5):34–37, 2009.
- [10] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3:993–1022, 2003.

- [11] Frederique Bordignon, Liana Ermakova, and Marianne Noel. Over-promotion and caution in abstracts of preprints during the COVID-19 crisis. Lern. Publ., 34(4):622–636, 2021.
- [12] Laurent Charlin and Richard S. Zemel. The Toronto Paper Matching System: An automated paper-reviewer assignment system. In PEER@ICML '13, 2013.
- [13] Serafeim Chatzopoulos, Thanasis Vergoulis, Theodore Dalamagas, and Christos Tryfonopoulos. Veto+: improved expert set expansion in academia. Int. J. Digit. Libr., 23(1):57–75, 2022.
- [14] Gohar Rehman Chughtai, Jia Lee, Mahnoor Shahzadi, Asif Kabir, and Muhammad Arshad Shehzad Hassan. An efficient ontology-based topic-specific article recommendation model for best-fit reviewers. Scientometrics, 122(1):249–265, 2020.
- [15] Andrew Collins and Jöran Beel. Document embeddings vs. keyphrases vs. terms for recommender systems: A large-scale online evaluation. In Maria Bonn, Dan Wu, J. Stephen Downie, and Alain Martaus, editors, JCDL '19, pages 130–133. IEEE, 2019.
- [16] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. In Jill Burstein, Christy Doran, and Thamar Solorio, editors, NAACL-HLT '19, pages 4171–4186. Association for Computational Linguistics, 2019.
- [17] Fred Douglass. Best practices for the care and feeding of a program committee, and other thoughts on conference organization. In Jeffrey C. Mogul, editor, WOWCS '08, pages 1–6. USENIX Association, 2008.
- [18] Yves Fassin. Research on covid-19: a disruptive phenomenon for bibliometrics. Scientometrics, 126(6):5305–5319, 2021.
- [19] Eugene Garfield. Can citation indexing be automated? In Statistical Association Methods for Mechanized Documentation, pages 189–192, 1965.
- [20] Guibing Guo, Bowei Chen, Xiaoyan Zhang, Zhirong Liu, Zhenhua Dong, and Xiuqiang He. Leveraging title-abstract attentive semantics for paper recommendation. In AAAI '20, pages 67–74. AAAI Press, 2020.

- [21] Shuguang Han, Jiepu Jiang, Zhen Yue, and Daqing He. Recommending program committee candidates for academic conferences. In Cornelia Caragea, C. Lee Giles, Lior Rokach, and Xiaozhong Liu, editors, CompSci@CIKM '13, pages 1–6. ACM, 2013.
- [22] Musa Ibrahim M. Ishag, Kwang-Ho Park, Jong Yun Lee, and Keun Ho Ryu. A pattern-based academic reviewer recommendation combining author-paper and diversity metrics. IEEE Access, 7:16460–16475, 2019.
- [23] Steven Jecmen, Hanrui Zhang, Ryan Liu, Nihar B. Shah, Vincent Conitzer, and Fei Fang. Mitigating manipulation in peer review via randomized reviewer assignments. In Hugo Larochelle, Marc’Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin, editors, NeurIPS '20, pages 12533–12545, 2020.
- [24] Steven Jeuris and Jakob E. Bardram. Dedicated workspaces: Faster resumption times and reduced cognitive load in sequential multitasking. Comput. Hum. Behav., 62:404–414, 2016.
- [25] Jian Jin, Baozhuang Niu, Ping Ji, and Qian Geng. An integer linear programming model of reviewer assignment with research interest considerations. Ann. Oper. Res., 291(1):409–433, 2020.
- [26] Yordan Kalmukov. An algorithm for automatic assignment of reviewers to papers. Scientometrics, 124(3):1811–1850, 2020.
- [27] Ari Kobren, Barna Saha, and Andrew McCallum. Paper matching with local fairness constraints. In Ankur Teredesai, Vipin Kumar, Ying Li, Rómer Rosales, Evimaria Terzi, and George Karypis, editors, SIGKDD '19, pages 1247–1257. ACM, 2019.
- [28] Ngai Meng Kou, Leong Hou U, Nikos Mamoulis, Yuhong Li, Ye Li, and Zhiguo Gong. A topic-based reviewer assignment system. Proc. VLDB Endow., 8(12):1852–1855, 2015.
- [29] Christin Katharina Kreutz, Prentim Sahitaj, and Ralf Schenkel. Evaluating semantometrics from computer science publications. Scientometrics, 125(3):2915–2954, 2020.
- [30] Christin Katharina Kreutz and Ralf Schenkel. RevASIDE: Assignment of suitable reviewer sets for publications from fixed candidate pools. In Eric Pardede, Maria Indrawan-Santiago, Pari Delir Haghighi, Matthias Steinbauer, Ismail Khalil, and Gabriele Kotsis, editors, iiWAS '21, pages 57–68. ACM, 2021.

- [31] Quoc V. Le and Tomas Mikolov. Distributed representations of sentences and documents. In ICML '14, volume 32 of JMLR Workshop and Conference Proceedings, pages 1188–1196. JMLR.org, 2014.
- [32] Michael Ley. DBLP - some lessons learned. Proc. VLDB Endow., 2(2):1493–1500, 2009.
- [33] Zhi Li and Xiaozhu Zou. A review on personalized academic paper recommendation. Comput. Inf. Sci., 12(1):33–43, 2019.
- [34] Xiang Liu, Torsten Suel, and Nasir D. Memon. A robust model for paper reviewer assignment. In Alfred Kobsa, Michelle X. Zhou, Martin Ester, and Yehuda Koren, editors, RecSys '14, pages 25–32. ACM, 2014.
- [35] Marcin Maleszka, Bernadetta Maleszka, Dariusz Krol, Marcin Hernes, Denis Mayr Lima Martins, Leschek Homann, and Gottfried Vossen. A modular diversity based reviewer recommendation system. In Pawel Sitek, Marcin Pietranik, Marek Krotkiewicz, and Chutimet Srinilta, editors, ACIIDS '20, volume 1178 of Communications in Computer and Information Science, pages 550–561. Springer, 2020.
- [36] Gloria Mark, Victor M. Gonzalez, and Justin Harris. No task left behind?: examining the nature of fragmented work. In Gerrit C. van der Veer and Carolyn Gale, editors, CHI '05, pages 321–330. ACM, 2005.
- [37] Chifumi Nishioka, Jorn Hauke, and Ansgar Scherp. Influence of tweets and diversification on serendipitous research paper recommender systems. PeerJ Comput. Sci., 6:e273, 2020.
- [38] Fabio Pacheco, Igor Wiese, Bruno Cartaxo, Igor Steinmacher, and Gustavo Pinto. Analyzing the evolution and diversity of SBES program committee. CoRR, abs/2002.00770, 2020.
- [39] Manos Papagelis, Dimitris Plexousakis, and Panagiotis Nikolaou. CONFIOUS: managing the electronic submission and reviewing process of scientific conferences. In Anne H. H. Ngu, Masaru Kitsuregawa, Erich J. Neuhold, Jen-Yao Chung, and Quan Z. Sheng, editors, WISE '05, volume 3806 of Lecture Notes in Computer Science, pages 711–720. Springer, 2005.
- [40] Nazmus Sakib, Rodina Binti Ahmad, Mominul Ahsan, Md. Abdul Based, Khalid Haruna, Julfikar Haider, and Saravanakumar Gurusamy. A hybrid personalized scientific paper recommendation approach integrating public contextual metadata. IEEE Access, 9:83080–83091, 2021.

- [41] Gerard Salton, Anita Wong, and Chung-Shu Yang. A vector space model for automatic indexing. Commun. ACM, 18(11):613–620, 1975.
- [42] Michael Schreiber. The influence of self-citation corrections on Egghe’s g index. Scientometrics, 76(1):187–200, 2008.
- [43] Per O. Seglen. The skewness of science. J. Am. Soc. Inf. Sci., 43(9):628–638, 1992.
- [44] Per O. Seglen. Why the impact factor of journals should not be used for evaluating research. BMJ (Clinical research ed.), 314(7079):498–502, 1997.
- [45] Anna Severin and Joanna Chataway. Overburdening of peer reviewers: A multi-stakeholder perspective on causes and effects. Learned Publishing, 34(4):537–546, 2021.
- [46] Nihar B. Shah, Behzad Tabibian, Krikamol Muandet, Isabelle Guyon, and Ulrike von Luxburg. Design and analysis of the NIPS 2016 review process. J. Mach. Learn. Res., 19:49:1–49:34, 2018.
- [47] Abdul Shahid, Muhammad Tanvir Afzal, Moloud Abdar, Mohammad Ehsan Basiri, Xujuan Zhou, Neil Y. Yen, and Jia-Wei Chang. Insights into relevant knowledge extraction techniques: a comprehensive review. J. Supercomput., 76(3):1695–1733, 2020.
- [48] Xiaolin Shi, Jure Leskovec, and Daniel A. McFarland. Citing for high impact. In Jane Hunter, Carl Lagoze, C. Lee Giles, and Yuan-Fang Li, editors, JCDL ’10, pages 49–58. ACM, 2010.
- [49] Ivan Stelmakh, Nihar B. Shah, and Aarti Singh. PeerReview4All: Fair and accurate reviewer assignment in peer review. CoRR, abs/1806.06237, 2018.
- [50] Ivan Stelmakh, Nihar B. Shah, and Aarti Singh. PeerReview4All: Fair and accurate reviewer assignment in peer review. In Aurélien Garivier and Satyen Kale, editors, ALT ’19, volume 98 of Proceedings of Machine Learning Research, pages 827–855. PMLR, 2019.
- [51] Hao Tang, Baisong Liu, and Jiangbo Qian. Content-based and knowledge graph-based paper recommendation: Exploring user preferences with the knowledge graphs for scientific paper recommendation. page e6227.

- [52] Wenbin Tang, Jie Tang, and Chenhao Tan. Expertise matching via constraint-based optimization. In Jimmy Xiangji Huang, Irwin King, Vijay V. Raghavan, and Stefan M. Rüger, editors, WI '10, pages 34–41. IEEE Computer Society, 2010.
- [53] Alex D. Wade, Kuansan Wang, Yizhou Sun, and Antonio Gulli. WSDM cup 2016: Entity ranking challenge. In Paul N. Bennett, Vanja Josifovski, Jennifer Neville, and Filip Radlinski, editors, WSDM '16, pages 593–594. ACM, 2016.
- [54] Bangchao Wang, Ziyang Weng, and Yanping Wang. A novel paper recommendation method empowered by knowledge graph: for research beginners. CoRR, abs/2103.08819, 2021.
- [55] Chen Yang, Tingting Liu, Wenjie Yi, Xiaohong Chen, and Ben Niu. Identifying expertise through semantic modeling: A modified BBPSO algorithm for the reviewer assignment problem. Applied Soft Computing, 94:106483, 06 2020.
- [56] Qiang Yang, Zhixu Li, An Liu, Guanfeng Liu, Lei Zhao, Xiangliang Zhang, Min Zhang, and Xiaofang Zhou. A novel hybrid publication recommendation system using compound information. World Wide Web, 22(6):2499–2517, 2019.
- [57] Dong Zhang, Shu Zhao, Zhen Duan, Jie Chen, Yanping Zhang, and Jie Tang. A multi-label classification method using a hierarchical and transparent representation for paper-reviewer recommendation. ACM Trans. Inf. Syst., 38(1):5:1–5:20, 2020.

Publications

SchenQL: in-depth analysis of a query language for bibliographic metadata	30
Evaluating Semantometrics from Computer Science Publications	73
Scientific Paper Recommendation Systems: a Literature Review of recent Publications	133
RevASIDE: Evaluation of Assignments of Suitable Reviewer Sets for Publications from Fixed Candidate Pools	196
Diverse Reviewer Suggestion for Extending Conference Program Committees	243

5. SchenQL: in-depth analysis of a query language for bibliographic metadata

Outline

5.1	Introduction	32
5.2	Related Work	34
5.3	SchenQL: QL and GUI	35
5.3.1	Data Model	35
5.3.2	Building Blocks	36
5.3.3	Syntax	37
5.3.4	Implementation	38
	SchenQL DB Parser	39
	SchenQL CLI and Compiler	40
	SchenQL API	41
	SchenQL Front End	41
5.4	Evaluation	44
5.4.1	Benchmarks: Database Engines and Target Language as well as SchenQL Compiler Evaluation	45
	Selection of Database Engines and Query Languages	46
	Queries	47
	Setting	48
	Analysis of H_1	49
	Discussion	52
	Setting	53
	Analysis of H_2	53
	Discussion	55
5.4.2	Qualitative Study: Interviews	55
5.4.3	Quantitative Study: SchenQL CLI vs. SQL, GUI and UEQ	56

Queries	57
Setting	59
Analysis of H_3	59
Analysis of H_4	61
Open Questions and Discussion	62
Setting	63
Partial Analysis of H_5 : users unfamiliar with query formulation	64
Partial Analysis of H_5 : UEQ	65
Open Questions and Discussion	66
5.5 Conclusion and Future Work	66
Bibliography	68

Bibliographic Information

Kreutz, C. K., Wolz, M., Knack, J., Weyers, B., Schenkel, R. (2022). SchenQL: in-depth analysis of a query language for bibliographic metadata. In *International Journal of Digital Libraries* 23(2) (pp. 113-132). Springer. <https://doi.org/10.1007/s00799-021-00317-8>.

Copyright Notice

©2022 Springer. This is an accepted but reformatted version of this article published in <https://doi.org/10.1007/s00799-021-00317-8>. Clarification of the copyright adjusted according to the guidelines of the publisher.

Keywords

Domain-Specific Query Language • Bibliographic Metadata • Digital Libraries • Graphical User Interface

Abstract

Information access to bibliographic metadata needs to be uncomplicated, as users may not benefit from complex and potentially richer data that may be difficult to obtain. Sophisticated research questions including complex aggregations could be answered with complex SQL queries. However, this comes with the cost of high complexity, which requires for a high level of expertise even for trained programmers. A domain-specific query language could provide a straightforward solution to this problem. Although less generic, it can support users not familiar with query construction in the formulation of complex information needs.

In this paper, we present and evaluate SchenQL, a simple and applicable query language that is accompanied by a prototypical GUI. SchenQL focuses on querying bibliographic metadata using the vocabulary of domain experts. The easy-to-learn domain-specific query language is suitable for domain experts as well as casual users while still providing the possibility to answer complex information demands. Query construction and information exploration is supported by a prototypical GUI. We present an evaluation of the complete system: different variants for executing SchenQL queries are benchmarked; interviews with domain-experts and a bipartite quantitative user study demonstrate SchenQL's suitability and high level of users' acceptance.

5.1 Introduction

Scientific writing almost always starts with a thorough bibliographic research on relevant papers, authors, conferences, journals and institutions. While web search is excellent for question answering and intuitively performed, not all retrieved information is correct, unbiased and categorised [3]. The arising problem is people's tendency of rather using poor information sources that are easy to query than more reliable sources which might be harder to access [4]. This introduces the need for more formal and also structured information sources such as digital libraries specialised in the underlying data, that at the same time need to be easy to query.

Currently existing interfaces of digital libraries often provide keyword search on metadata or offer to query attributes [15, 24]. However, in many cases, these interfaces do not allow to directly express more advanced queries such as "*Which are the five most cited articles written by person P about topic T after year Y?*", but require complex interaction. Popular examples

of such limited systems are dblp¹ [24] or Semantic Scholar². More complex tools, e.g. GrapAL³ [7], are capable of answering said complex queries, but come with complex and often not very intuitive query languages. Another option would be to use structured query languages such as SQL, a widespread language for querying databases, which unfortunately tends to be difficult to master [36]. This is critical as in most cases domain-experts are familiar with the schema of the data but are not experienced in using all-purpose query languages such as SQL [1, 25]. This is even worse for casual users of digital libraries who neither have knowledge of the structure of the data nor of SQL.

To close this gap, we present the SchenQL Query Language, in short SchenQL⁴, for the domain of bibliographic metadata [20, 21]. SchenQL is designed to be easily utilised by experts as well as casual users from the domain as it uses the vocabulary of digital libraries in its syntax. While domain-specific query languages (DSLs) provide a multitude of advantages [9], the most important aspect in the conception of SchenQL was that no programming skills or database schema knowledge is required to use it. For SchenQL to be widely applicable, we introduce a prototypical graphical user interface (the SchenQL GUI) which supports the construction of queries, offers visualisations of query results and an additional dimension of retrieving information by exploring data and its relations through clicking. As an example of SchenQL, the aforementioned question can be formulated as follows: `MOST CITED (ARTICLES WRITTEN BY "P" ABOUT "T" AFTER Y) LIMIT 5`.

In addition to the SchenQL query language, another major contribution of this paper is the empirical evaluation of SchenQL as domain-specific query language on bibliographic metadata including the investigation of a prototypical GUI that is designed to assist users in creating queries. SchenQL is evaluated three-fold: 1) query execution times were benchmarked to underline the suitability for interactive retrieval tasks, 2) interviews with domain-experts were conducted to identify applications as well as options for further development and 3) a quantitative user study consisting of two parts measured effectiveness, efficiency and users' satisfaction with our whole system: we first evaluated the usage of command line SchenQL against SQL, followed by a study which compared the usage of the SchenQL GUI to the previous results. Here, the User Experience Questionnaire [33] was conducted for assessing of users' experience.

The remainder of this paper is structured as follows: Section 5.2 discusses

¹<https://dblp.uni-trier.de/>

²<https://www.semanticscholar.org/>

³<https://grapal.allenai.org/>

⁴The name SchenQL is a pun on the name Ralf Schenkel and gives kudos to him as he proposed the first version of the language's grammar.

related work. Section 5.3 introduces the structure and syntax of SchenQL with a special focus on the implementation including the presentation of the SchenQL Parser, Compiler and Front End. The system is evaluated in three parts in the following Section 5.4. The last Section 5.5 describes possible future research.

This paper is an extended version of the work presented at ICADL'20 [21]. The main extensions are contained in the Sections 5.3.4 and 5.4.1.

5.2 Related Work

Areas adjacent to the one we are tackling are *search on digital libraries*, *search interfaces on bibliographic metadata*, *formalised query languages* and *domain-specific query languages*.

For *search on digital libraries*, the MARC format is a standard for information exchange [3]. While it is useful for known-item search, topical search might be problematic as contents of the corresponding fields can only be interpreted by domain-experts [3]. Most interfaces on digital libraries provide a field-based Boolean search [32] which can lead to difficulties in formulating queries that require the definition and concatenation of multiple attributes. This might cause a substantial cognitive workload on the user [6]. In contrast, withholding or restriction of faceted search on these engines fails to answer complex search tasks [5]. Thus, we focus on a search of topical information that even casual users can utilise while also offering the possibility to clearly define search terms for numerous attributes in a single query.

Several *search interfaces on bibliographic metadata* exist, the most well-known ones might be dblp [19, 24], Bibsonomy [15], Google Scholar⁵, ResearchGate⁶ or Semantic Scholar. All of those systems allow for a systematic refinement of result sets by the application of filter options via facets to varying extends. Only dblp and Semantic Scholar (on a small scale) support search on venues. The formulation of complex queries with aggregations is not targeted by any of them. In contrast, SchenQL supported by a GUI specialises on these functionalities. GrapAL⁷ [7] actually provides all functions of SchenQL but is a complex tool utilising the Cypher [13] query language (QL).

Domain-specific query languages can come in various shapes. They can be SQL-like [23], visual QLs [1, 11] or use a domain-specific vocabulary [35] but are typically specialised on a certain area. They also come in different

⁵<https://scholar.google.com/>

⁶<https://www.researchgate.net>

⁷<https://grapal.allenai.org/>

complexities: for example MathQL [14] is a query language in markup style on RDF repositories but a user needs to be mathematician to be able to operate it. The DSL proposed by Madaan [25] stems from the medical domain and is designed to be used by inexperienced patients as well as medical staff. Some DSLs are domain-unspecific such as the aforementioned Cypher [13], BiQL [12] or SnQL [26] and depend on complicated SQL-like syntax. Naturally, there are hybrid forms: some natural language to machine-readable query options are domain-specific [31] and some DSLs might be transferable to other domains [9]. With our SchenQL system, we provide a QL which uses vocabulary from the domain of bibliographic metadata while being useful for experts as well as casual users and avoiding complicated syntax.

5.3 SchenQL: QL and GUI

For simplicity, we refer to SchenQL including its GUI as the SchenQL system. SchenQL was developed to access bibliographic metadata textually, which resembles natural language for casual as well as expert users of digital libraries [20, 21]. The fundamental idea is to hide complex syntax behind plain domain-specific vocabulary. This enables usage from anyone versed in the vocabulary of the domain without experience in sophisticated query languages such as SQL. The prototypical GUI supports SchenQL: it helps in query formulation with the auto-completion and keyword suggestion. Additionally, it provides visual exploration of query results supporting two standard visualisations: Ego Graph [30] and BowTie [18].

5.3.1 Data Model

For our data model (see Figure 5.1) we assume bibliographic metadata consists of persons and the publications they authored or edited. These persons can be affiliated with certain institutions. Publications can be of multiple types and may be published in conferences or journals. Publications can reference previously published papers and might be cited themselves by more recent work building upon them.

Persons can both be authors and editors of publications and might be working for institutions. For persons we assume a unique key, their primary name, possible other names and their ORCID are given. For *institutions* we model their primary name, primary location, further names and locations as well as the location of the institution in form of city, country, latitude and longitude. *Publications* can be either of type article, book, chapter, Master's thesis or PhD thesis. For publications we assume a unique key, the

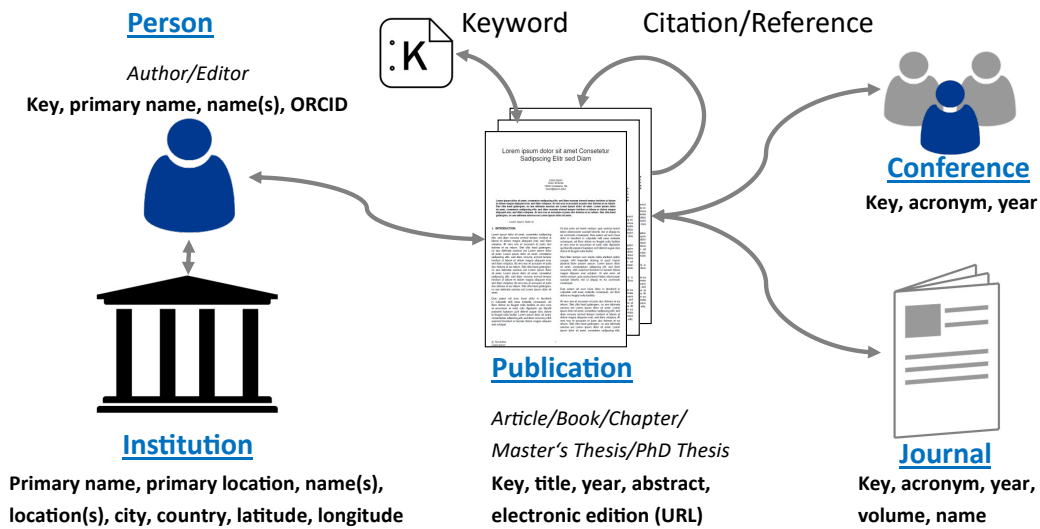


Figure 5.1: SchenQL database model with relations, base concepts (underlined), specialisations (italic) and selected attributes (bold).

title and publication year, abstract and electronic editions can be available. Publications can be associated with keywords. We additionally model links to referenced and citing papers as well as authors/editors of the publications and the publication venue. As venues for the publications we consider *conferences* and *journals*. For both of them we model a unique key, acronym and a specific year. For journals we also store volume and name information.

Figure 5.2 shows the SchenQL relational data model.

5.3.2 Building Blocks

Base concepts are the basic return objects of SchenQL. A base concept is connected to an entity of the data set and has multiple attributes. Those base concepts are **publications**, **persons**, **conferences**, **journals** and **institutions**. Upon these concepts, queries can be constructed. Base concepts can be specialised. For example **publications** can be refined by the specialisations **books**, **chapters**, **articles**, **master** or **PhD theses**. A specialisation can be used instead of a base concept in a query.

Filters can restrict base concepts by extracting a subset of the data. Literals can be used as identifiers for objects from base concepts, they can be utilised to query for specific data. Attributes of base concepts can be queried, for an overview of attributes see Figure 5.1. Table 5.1 gives an overview of literals, specialisations, filters and the standard return value for every base concept. Queries with strings as filter parameters, e.g. titles or names,

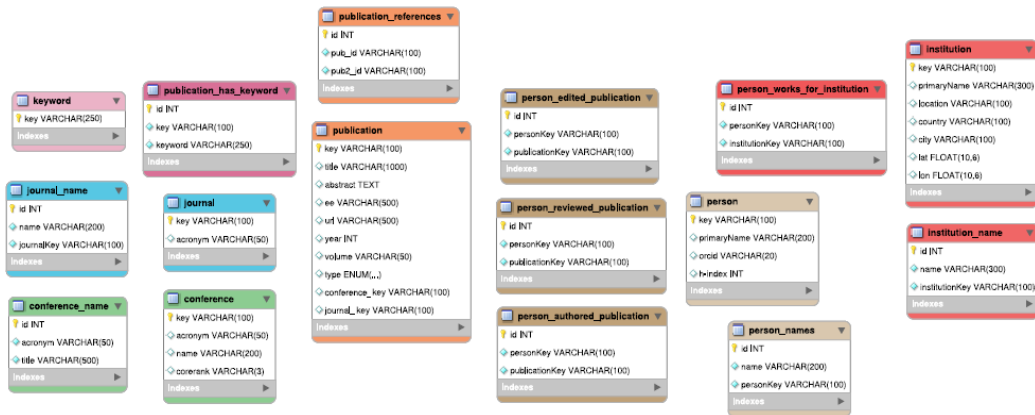


Figure 5.2: SchenQL relational data model (online in colour).

utilise exact matching in general. Prepending a \sim to such a query enables full-text search in case of titles: `PUBLICATIONS TITLED \sim "DAFFODIL"`. If \sim is used with a following person name, it provides the functionality of the original Soundex algorithm. Keywords as well as strings are case- and accent-insensitive.

Functions can be used to aggregate data or offer domain-specific operations. Right now, SchenQL provides four functions: `MOST CITED`, `COUNT`, `KEYWORD OF` and `COAUTHOR OF`. `MOST CITED (PUBLICATION)` can be applied on publications. This function returns titles as well as numbers of citations of papers in the following set. By default, the top five results are returned. `COUNT` returns the number of objects contained in the following sub-query. `KEYWORD(S) OF (PUBLICATION | CONFERENCE | JOURNAL)` returns the keywords associated with the following base concept. The next function `COAUTHOR(S) OF (PERSON)` returns the coauthors of an author. The `LIMIT x` operator with $x \in \mathbb{N}$ can be appended at the end of any query to change the number of displayed results to at most x .

5.3.3 Syntax

The syntax of SchenQL follows simple rules resulting in queries similar to natural language which are aiming at simple construction. Sub-queries have to be surrounded by parentheses. It is possible to write singular or plural when using base concepts or specialisations (e.g. `JOURNAL` or `JOURNALS`). Filters following base concepts or their specialisations can be in arbitrary order and get connected via conjunction if not specified otherwise (`OR` and `NOT` are also possible). Most filters expect a base concept as their parameter (e.g. `WRITTEN BY (PERSONS)`), however some filters anticipate a string as their

	PUBLICATION	PERSON	CONFERENCE	JOURNAL	INSTITUTION
L	key, title	key, primary name, orcid	key, acronym	key, acronym	
S	MASTERTHESIS, BOOK, CHAPTER, PHDTHESIS, ARTICLE	AUTHOR, EDITOR			
F	PUBLISHED BY (I), ABOUT (keywords), WRITTEN BY (PE), EDITED BY (PE), APPEARED IN (C J), BEFORE year, IN YEAR year, AFTER year, TITLED title, REFERENCES (PU), CITED BY (PU)	PUBLISHED IN (C J), PUBLISHED WITH (I), WORKS FOR (I), NAMED name, ORCID orcid, AUTHORED (PU), REFERENCES (PU), CITED BY (PU)	ACRONYM acronym, ABOUT (keywords), BEFORE year, IN YEAR year, AFTER year	NAMED name, ACRONYM acronym, ABOUT (keywords), BEFORE year, IN YEAR year, AFTER year, VOLUME volume	NAMED name, CITY city, COUNTRY country, MEMBERS (PE)
V	title	primary name	acronym	acronym	primary name + location

Table 5.1: SchenQL base concepts Publications (PU), persons (PE), conferences (C), journals (J) and institutions (I) with their respective literals (L), specialisations (S), filters (F) and standard return values (V, relevant for the CLI).

parameter (e.g. COUNTRY "de"). Specialisations can be used in place of base concepts. Instead of a query PERSON NAMED "Ralf Schenkel" a specialisation like AUTHOR NAMED "Ralf Schenkel" would be possible. If a filter requires a base concept, parentheses are needed except for the case of using literals for identifying objects of the base concept. For example PUBLICATIONS WRITTEN BY "Ralf Schenkel" is semantically equivalent to PUBLICATIONS WRITTEN BY (PERSONS NAMED "Ralf Schenkel"). Attributes of base concepts can be accessed by putting the queried for attribute(s) in front of a base concept and connecting both parts with an OF (e.g. "name", "acronym" OF CONFERENCES ABOUT KEYWORDS ["DL", "QLs"]).

5.3.4 Implementation

The SchenQL system contains four main components (see Figure 5.3). The *SchenQL DB Parser* parses all the different data sources and combines them in a MySQL database, the *SchenQL CLI* is the command line interface that also contains the *SchenQL Compiler* for the query language, the *SchenQL Front End* represents the web interface (introduced in Section 5.3.4), and the *SchenQL API* connects the SchenQL CLI with the SchenQL Front End. The SchenQL API also runs some direct queries on the database to execute high-level functions that SchenQL itself is not capable of. Our QL can be used in a terminal client similar to the MySQL shell or via the graphical front end.

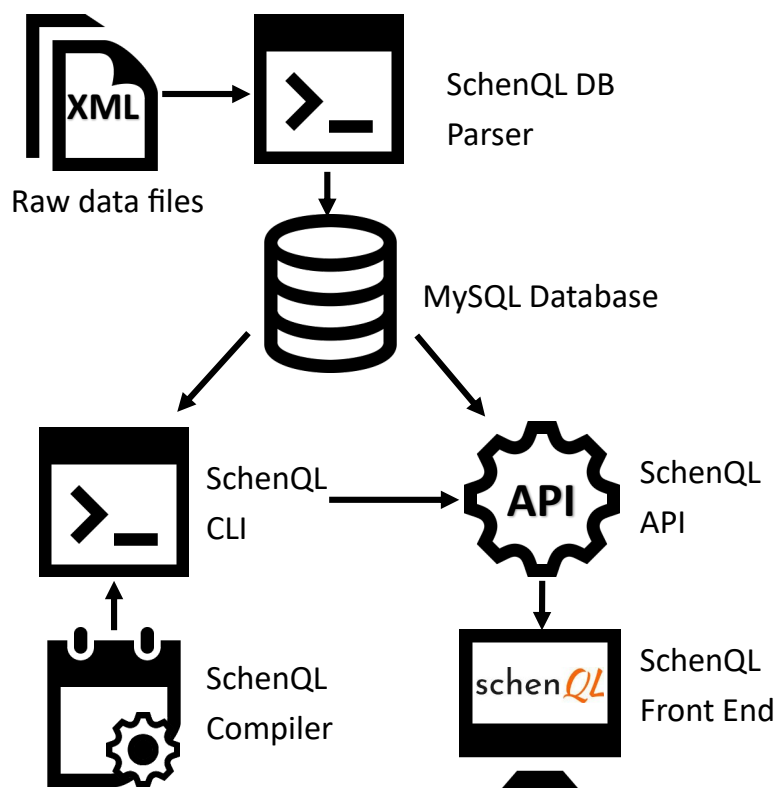


Figure 5.3: Overview of the SchenQL components.

SchenQL DB Parser

Our database model (see Figure 5.1) was specifically designed for the syntax of SchenQL so that every base concept represents an entity in the database. Data on references and citations is contained in a single table. The chosen database uses the MyISAM storage-engine instead of the MySQL 8 default InnoDB. In comparison to InnoDB, MyISAM does not support transactions, so there is no need to commit after inserting data into the database. In case of SchenQL, transactions are not required, since no data is changed after the creation of the database. On the one hand, this strongly influences the performance of the database parser and on the other hand MyISAM has a higher support for full text search, which is necessary for queries like PUBLICATIONS ABOUT "DL" or PUBLICATIONS TITLED ~ "Daffodil".

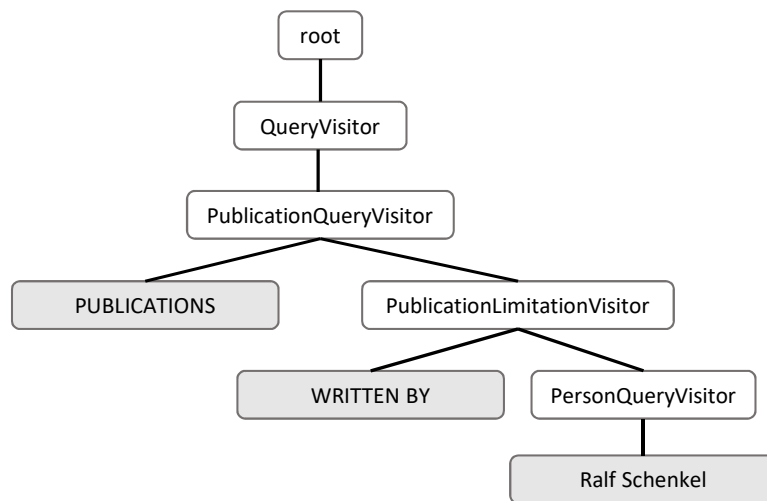


Figure 5.4: Syntax Tree of the Query PUBLICATIONS WRITTEN BY "Ralf Schenkel".

SchenQL CLI and Compiler

The SchenQL CLI defines the core of the QL. It does not only provide an interface to use the language, but it also includes the compiler. The compiler translates SchenQL queries to the target language SQL and uses Java Database Connectivity to run them against a MySQL 8.0.16 database holding the data⁸. We built the lexer and parser of our compiler using ANTLR with Java as the target language⁹. In the parser we use the *visitor* approach to iterate through the nodes in the constructed parse tree.

SQL queries are generated from SchenQL input in multiple steps: first a SchenQL expression (for example PUBLICATIONS WRITTEN BY "Ralf Schenkel") is parsed and a parse tree is constructed (see Figure 5.4). It represents the abstract syntax tree of the parsed input expression. Afterwards this syntax tree is traversed from the root onwards. The parser calls the root node where it checks whether the input query is a request for one of the basic concepts or if it is a function call (alias) to a sub-query, e.g. COUNT. Next, the child node of the root node is visited. In the case of the example from Figure 5.4, the QueryVisitor is called, the type of the query is checked and the next child node is visited (in the example: PublicationQueryVisitor). The PublicationQueryVisitor processes the child nodes by depth-first

⁸See Section 5.4.1 for an evaluation of the target language and the implementation of the SchenQL to SQL compiler.

⁹We utilise language specific functions in the lexer so that the grammar is no longer generally usable for other programming languages but would have to be adapted.

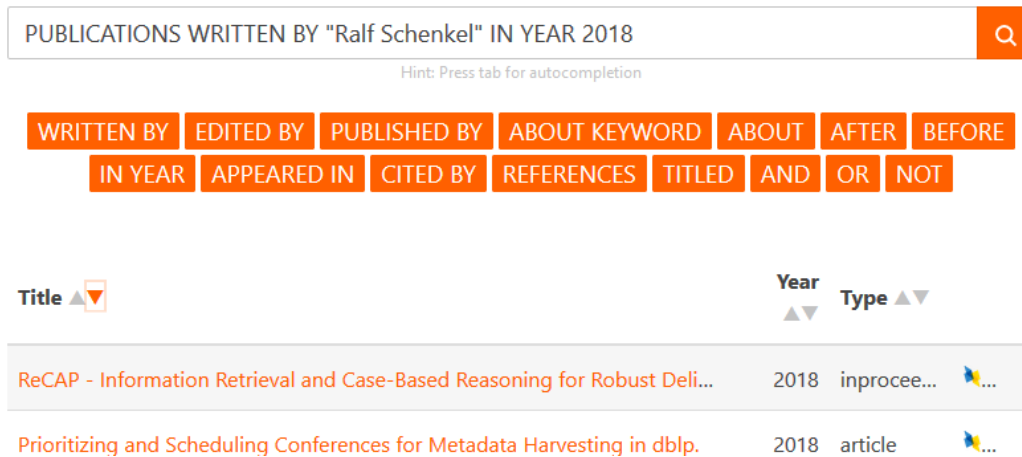


Figure 5.5: SchenQL Front End for a search with suggested language components and search result.

search and collects all filters used in the query in an array using the (Publication)LimitationVisitor. Subsequently, the PublicationQueryVisitor generates a SQL select statement and adds the filters to it. It also checks if a specialisation was used to call the query, i.e. if the user queried `ARTICLES WRITTEN BY "A"` instead of `PUBLICATIONS WRITTEN BY "A"`. This process is performed recursively until the input has been completely processed.

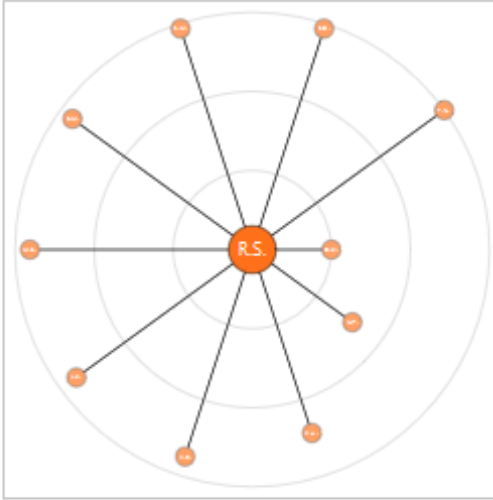
SchenQL API

The SchenQL API handles all communication between the compiler and the front end. It has two tasks: first, it implements an endpoint for handling all types of valid SchenQL queries, and second, it serves additional information based on base concepts, such as retrieving citations and references for publications or authors for performance reasons. We use the REST architecture pattern for the API. The API is implemented using Spring and Spring Boot for handling HTTP-requests.

SchenQL Front End

The SchenQL Front End (also called GUI) is inspired by results from the qualitative study described in Section 5.4.3. It provides access to information by supporting the construction of queries including the interactive navigation with the GUI. It also offers auto-completion of SchenQL query keywords and suggestions for the formulation of queries. Results of queries can be sorted for every column of the result table. In Figure 5.5 query formulation with suggested keywords and result representation in the SchenQL GUI is depicted.

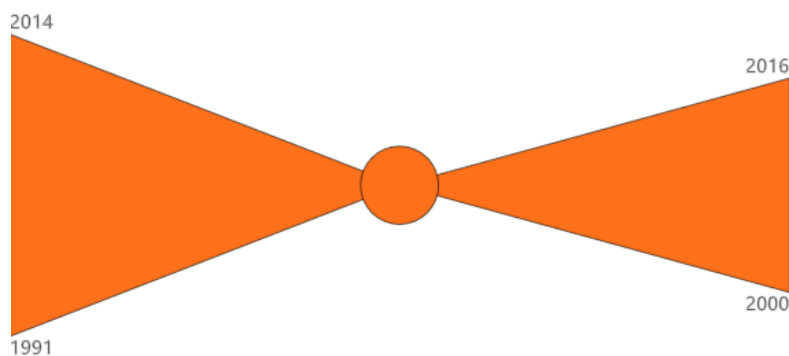
Person

Primary Name	Ralf Schenkel	Ego Graph	BowTie
ORCID	0000-0001-5379-5191	 <p>How to read this diagram?</p>	
Coauthors	Gerhard Weikum (39) Martin Theobald (30) Katja Hose (16) Show more ▾		
DBLP Key	homepages/s/RalfSchenkel		

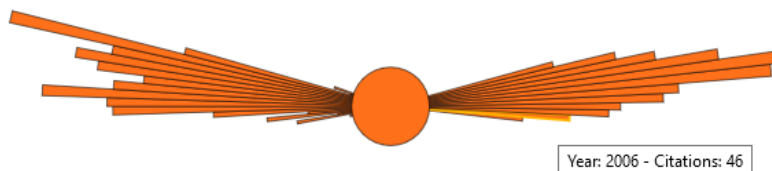
Publications

Title ▲▼	Year	Type	
SchenQL - A Domain-Specific Query Language...	2019	article	🔍...
Analyzing online schema extraction approache...	2019	inpr...	🔍...

Figure 5.6: Person detail view with Ego Graph depicting up to the ten most common co-authors. Nodes symbolise persons, the further an author is from the middle (person in focus), the less publications they share with the person in focus.



(a) Regular BowTie view.



(b) Detailed BowTie view.

Figure 5.7: Regular (top) and detailed (bottom) BowTie view with referenced and citing papers of a person with numbers of referenced (bows left of knot) and citing (bows right of knot) papers. For the regular view year numbers limit the period of time from which a paper referenced (left) and is cited itself (right). In the detailed view, the references and citations are separated in single slices per year. Hovering over single slices depicts the year and the associated number of references from or citations acquired in the specific year. The higher the number of citations or references, the longer the bow, the longer the spanned time the higher the bow.

If a search result is selected by clicking on it, detail views open (see Figure 5.6 for the detail view of a person) which offer all information available for the respective element of a base concept. Furthermore we incorporated two already established visualisations: *Ego Graph* [30] and *BowTie* [18]. The *Ego Graph* for persons (see Figure 5.6 top right part) supports the analysis of persons' most important co-authorships. At one glance the most common cooperators are visualised and compared to each other such that the overall productivity and interdependence of a person can be estimated. If a person has many equally close collaborators they might either be active in multiple fields or they could produce papers with many co-authors at once. If a person has only few very close co-authors and multiple further dependencies, this pattern could for example hint at a PhD student - supervisor relationship.

The *BowTie* visualisation can be used for the easy estimation of a person’s, publication’s or venue’s influence in terms of gained citations and its actuality (see Figure 5.7). If for example a lot of recent papers are referenced (estimated by the detailed view) by a paper in focus, one could assume that this paper is well positioned in that time’s publication landscape. The distribution of incoming citations could be very telling on whether e.g. a venue is still relevant to this day.

5.4 Evaluation

Before the actual evaluation of the SchenQL system, we conduct benchmarks for two possible database engine and target languages for the compilation of SchenQL queries: SQL (with data stored in a MySQL relational database) and Cypher (with data stored in a Neo4J graph database). Afterwards we evaluate the performance of the current implementation of the compiler that translates SchenQL into the target query language.

Our evaluation of the SchenQL system consists of a qualitative and a quantitative investigation which are followed by a performance evaluation. In a first qualitative study, we examine domain experts’ use-cases and desired functionality of a DSL such as SchenQL as well as an accompanied GUI. The major goal of this first investigation was to check SchenQL for completeness and suitability for the addressed use cases. In a subsequent step, we conducted a quantitative study in which we first compared SchenQL with SQL, both used through a command line interface (CLI) to ensure comparability. The goal was to measure the effectiveness, efficiency and users’ satisfaction with SchenQL as query language. As a follow-up, we evaluated the web-based GUI of the SchenQL system using the same queries and compared the results with those received from usage of the SchenQL CLI. We additionally investigated the SchenQL system’s user experience using the User Experience Questionnaire (UEQ) [33].

Considering the overall goals for SchenQL, we derived the following five hypotheses to be investigated:

- H_1 MySQL as database engine with SQL as a target language for the SchenQL compiler is more suitable than Neo4j as database engine with Cypher as target engine.
- H_2 The SchenQL-to-SQL compiler’s performance in translating and executing queries is comparable to that of manually formulated queries.
- H_3 Utilisation of the SchenQL CLI achieves better results in terms of

higher correctness, lower perceived difficulty of query construction as well as lower required time for query formulation than usage of SQL.

H_4 SchenQL is as suitable for domain-experts as it is for non-experts.

H_5 The SchenQL system provides high suitability and user experience (indicated by values $> .8$ for all six quality dimensions assessed with the UEQ¹⁰) for users not familiar with structured queries.

For all studies, we used a data set from the area of computer science: our structures were filled with data from dblp [24] integrated with fitting data from Semantic Scholar (for citations and abstracts) and enriched with information about institutions from Wikidata¹¹. As keys of persons, publications and venues we utilised dblp keys of the respective entities. Utilising the dblp data set as of June 2020¹² leads to 2,518,198 entries for persons, 5,095,451 entries for publications, 1849 entries for journals, 91,694 entries for conferences and 10,059 entries for institutions.

5.4.1 Benchmarks: Database Engines and Target Language as well as SchenQL Compiler Evaluation

The technical evaluation of our system consists of two parts: a comparison of execution times of queries for two different target engine candidates with their respective query languages, and a comparison of the target queries generated by the SchenQL compiler to manually optimised queries for typical query types found in digital libraries.

We first compare two specific implementations of viable target engine options: a relational database and a graph database. Both database types are reasonable options for the representation of bibliographic metadata. The actual data is clearly structured (e.g. in a publication record with clearly defined attributes) which supports usage of relational databases. The multiplicity of direct relations between bibliographic entities (e.g. persons citing papers instead of only persons writing papers and those papers citing other papers) and the graph-like structure (see Figure 5.1) hints at the utilisation of graph databases.

We then evaluate the implementation of our SchenQL compiler with regard to its suitability.

¹⁰User Experience Questionnaire Handbook: <https://www.ueq-online.org/Material/Handbook.pdf>

¹¹https://www.wikidata.org/wiki/Wikidata:Main_Page

¹²<https://dblp.org/xml/release/dblp-2020-06-01.xml.gz>

Selection of Database Engines and Query Languages

For the selection of a relational as well as a graph-based database management system (DBMS) we defined important factors which a database needed to satisfy in order to narrow down the numerous options for our application:

- Open source license. We did not want to introduce legal restrictions or license fees.
- Availability of the source code. The technical implementation should be accessible to allow research and adjustments.
- Active further development of DBMS. Guaranteed operation of the DBMS in the future was important, this was assessed by the date of latest release.
- Possibility of querying the DBMS from Java and Python programs without a further surrounding system but through a query language directly. This property ensured a low structural adaptation effort for the SchenQL ecosystem in case the underlying database is changed.

For the selection of a relational database engine, we considered the widespread options MySQL and PostgreSQL. As we did not come across clear arguments for or against one or the other¹³, MySQL was chosen. It has the advantage of providing the MyISAM storage-engine which has a higher support for full text search which we consider highly relevant. It does not support transactions, but transactions are not required in our use case. The target language for the relational database MySQL is SQL.

For the selection of a graph-based database engine¹⁴ and query language, we additionally deemed the structural and syntactical similarities to SQL important. This ensured the best possible comparability between the query languages. Consideration of the four general properties and the last one specific to the graph-based variants produced the query language Cypher as the best option. Cypher is supported by the DBMS Neo4j, Redis and AgensGraph. As a related work [7] also utilises Cypher as target language and Neo4j as DBMS, we followed their example in our decision.

Note that execution times of queries are highly dependent on the utilised execution environment. We tried to select the best possible options for the relational as well as graph-based databases and respective target languages for our specific use case. We cannot exclude that other target languages for the database types may achieve better or different results.

¹³<https://db-engines.com/en/system/MySQL%3BPostgreSQL>

¹⁴<https://db-engines.com/en/ranking/graph+dbms>

ID	Query in plain text
B_1	Titles of all publications
B_2	Titles of publications written by A
B_3	Titles of publications written by A which appeared in journal J
B_4	Primary names of persons who authored a publication with title P
B_5	Titles of publications about K
B_6	Keywords of publications with title P
B_7	Primary names of co-authors of A
B_8	Primary names of co-authors of co-authors of A
B_9	Titles of publications which reference publications which were published by institutions where A is member
B_{10}	Titles of A 's most cited publications
B_{11}	Number of A 's publications
B_{12}	Primary names of persons with a name that sounds like F
B_{13}	Titles of publications written by A or B
B_{14}	Titles of publications written by A and not by B
B_{15}	Titles of publications containing T

Table 5.2: Overview of queries evaluated in the benchmark. A , B are unique names of different authors, J is a journal acronym, P is a publication title, K is a keyword, F is a forename and T is a term.

Queries

Table 5.2 shows the different benchmarking queries we observed. They utilise a representative amount of all SchenQL language elements. The queries were inspired by typical search scenarios [8] and categorisations [28] in digital libraries. Queries B_1 to B_6 incorporate only one or two concepts and simple conditions and combinations. B_7 observes the co-authorship relation and B_8 introduces a second indirection layer to this query type. B_9 generates an especially large result set. Queries B_{10} as well as B_{11} evaluate more domain specific functions. Query B_{12} utilises Soundex. In queries B_{13} and B_{14} logical operators are combined with single concepts. B_{15} evaluates full text search. We filled the main variables of the queries randomly, the dependent variable was set with respect to the main variable. For B_3 we randomly chose a person A who has published at least one paper in a journal J , for B_{14} we randomly chose a pair of co-authors A and B where A has also published several papers without B .

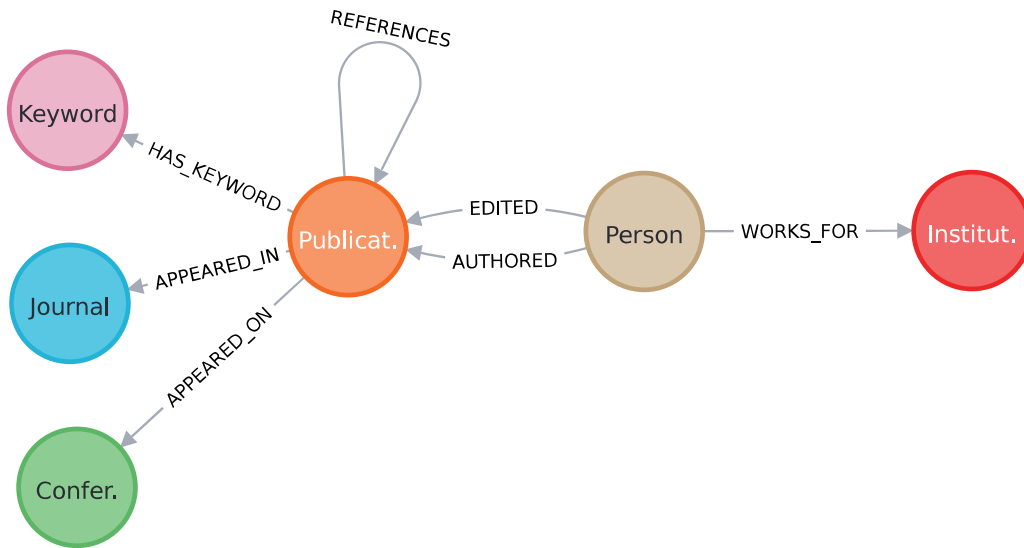


Figure 5.8: SchenQL graph-based data model (online in colour).

Setting

We run the performance benchmarks on a Ubuntu 20.04 machine with 32 GB RAM and a 2 TB SSD. A MySQL 8.0.21 database handles the SQL queries, an equivalent Neo4j 4.0.8 database handles the Cypher queries. We only set the variables for each benchmarking query once, the same variables were used throughout all experiments. Each query was run 100 times, we report average execution times on both databases. To minimise the impact of caching and prediction effects of modern hardware on the measurements, the DBMS and the surrounding docker containers were restarted after each query execution.

Figure 5.8 shows the simplified SchenQL graph-based data model without attributes. The colours of entities from this depiction correspond to the respective tables from the SchenQL relational data model in Figure 5.2. Keywords, journals and conferences are directly linked with publications. Persons are linked with publications and institutions. Person names for example are no longer stored in a separate table contrasting the relational data model but they now appear as attributes of persons.

Exec. time	Queries	Figures
Low ($s * 10^{-3}$)	$B_1, B_2, B_4, B_5, B_6, B_9, B_{11}, B_{14}$	5.9
Medium ($s * 10^{-2}$)	B_7, B_{10}, B_{15}	5.10
High (s)	B_3, B_{12}, B_{13}	5.11, 5.12
Very high ($s * 10^2$)	B_8	5.13, 5.14

Table 5.3: Evaluation query categories, queries and figures.

C	Time	User Experience
1	≤ 100	Users feel system reacts instantaneously
2	≤ 1000	User’s flow of thought is uninterrupted
3	≤ 10000	Limit for user’s attention span

Table 5.4: Categories (C) of system response times in ms and associated user experience.

Benchmark Part I: Database Engine and Target Language Performance

In this first part of the technical evaluation we assess the performance of two different database engines with specified languages for our SchenQL compiler for domain specific query types: the relation database engine MySQL with SQL and the graph-based database engine Neo4j with Cypher [13]. Here we strive to investigate the suitability of SQL as a target language for the SchenQL compiler with MySQL as database engine compared to Cypher with Neo4j as database engine and thus verify or falsify hypothesis H_1 . A target language for the SchenQL compiler has to support the formulation of typically required query types and the execution time of queries in general should not interrupt a user’s flow of thought (see Table 5.4). These two properties thus define our perception of suitability of a target engine and language. We deliberately do not include cost of learning or conciseness in our perception of suitability of a target language as users of SchenQL will not come in contact with the target language itself.

Analysis of H_1

All queries from Table 5.2 could be formulated both with SQL (run against a MySQL database) as well as Cypher (executed on a Neo4j database). Suitability in terms of both target languages being appropriate to express the information needs is therefore given, no database engine and target language surpasses the other in this aspect. So in the following we focus on the assess-

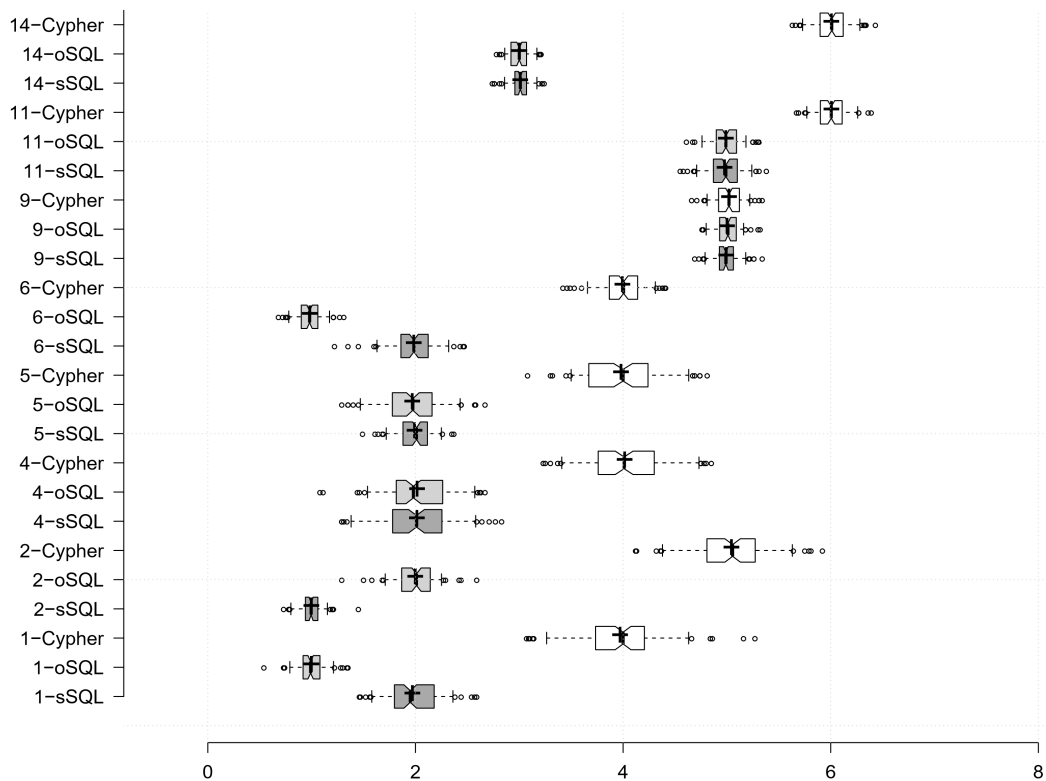


Figure 5.9: Queries with low execution time in ms.

ment of execution times for queries.

The following shows an exemplary formulation of B_{14} with Cypher:

```

MATCH (per:Person{primaryName:"A"})-[:AUTHORED]->(
  ↪ pub:Publication)
WHERE NOT EXISTS((:Person{primaryName:"B"})-[:
  ↪ AUTHORED]->(pub:Publication))
RETURN pub

```

The queries from Table 5.2 were put in four different categories depending on their execution time (see Table 5.3). The labels of the boxplots are defined as follows: *Cypher* (white boxes) describes manually written and optimised Cypher queries executed on a Neo4j database, *oSQL* (light grey boxes) marks manually constructed and optimised SQL queries and *sSQL* (dark grey boxes) are SQL queries generated by the SchenQL compiler. SQL queries were run against the MySQL database. This part of the plots is utilised in the following evaluation in Section Benchmark Part II: SchenQL Performance. Missing oSQL data points indicate that their value is identical to the respective sSQL data point. The whiskers extend to the 5th and

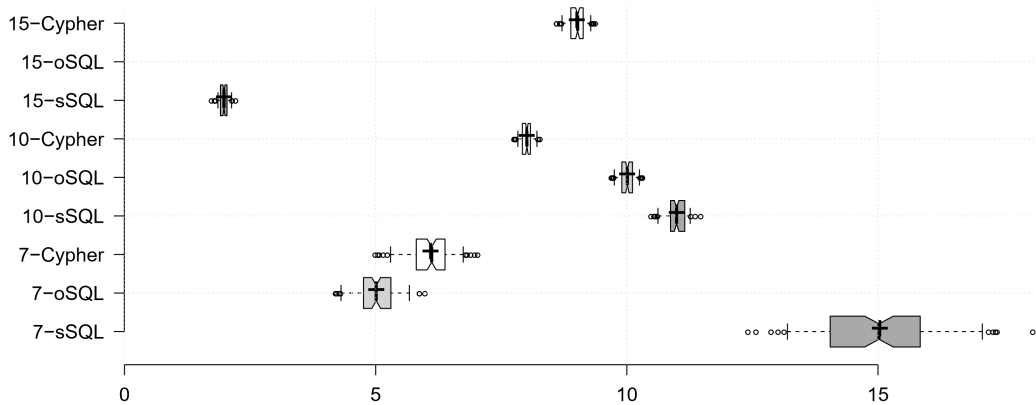


Figure 5.10: Queries with medium execution time in ms.

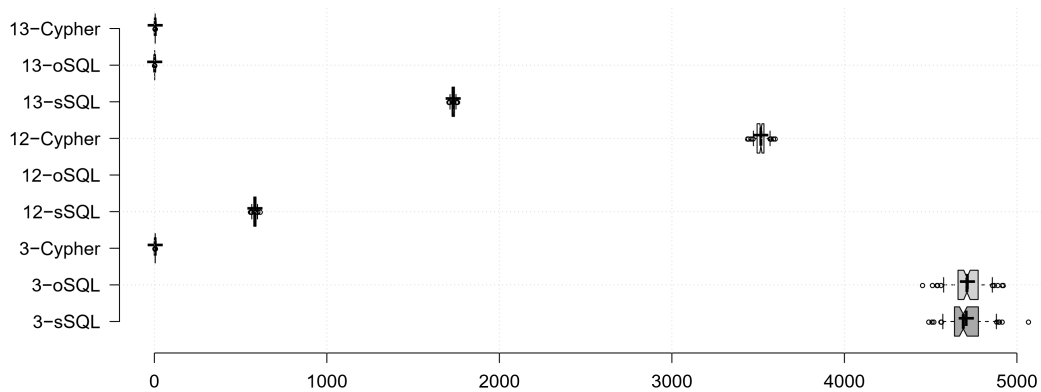


Figure 5.11: Queries with high execution time in ms, overview of all formulation times.

95th percentile [22]. The notches are defined as $\pm 1.58 \cdot \text{IQR} / \sqrt{n}$ and represent the 95% confidence interval for each median [10]. To enable conclusions about the probability that two medians differ, we can compare the notch area of two queries in the visualisation, which is only possible with linear scale. If data points of two different queries have a large separation on the time scale, we provide a separate plot with corresponding scale factor (Figures 5.13 and 5.14).

There seems to be a general execution difference of 0ms to 10ms in favour of SQL. The evaluation of the speed in relation to user experience is based on the absolute measured values according to the criteria of Nielsen [27] (see Table 5.4) constructed for response times of systems. All executions of Cypher formulations of queries except B_{12} fall into category 1. B_{12} takes 3517ms in the Cypher version and 584ms in the SQL version. Checking for a

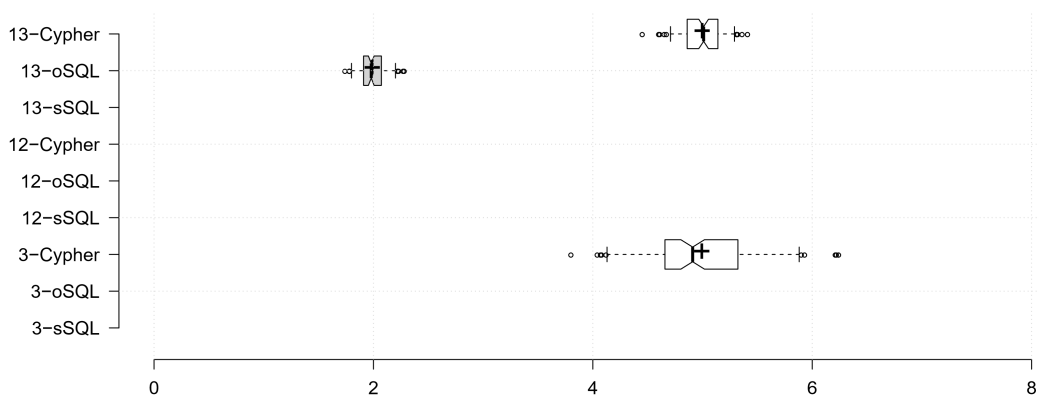


Figure 5.12: Queries with high execution time in ms, zoom on formulations with low execution time.

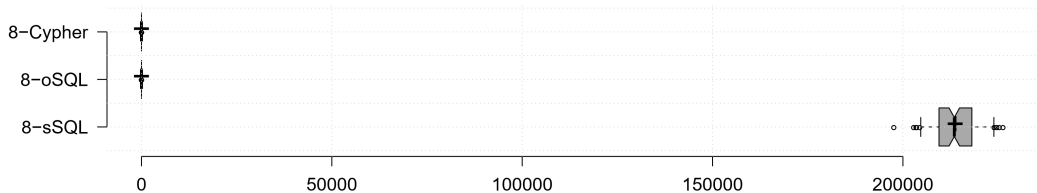


Figure 5.13: Queries with very high execution time in ms, overview of all formulation times.

substring inside a property or a field seems to cause a complete search in all relevant nodes (Cypher) and a full table scan (SQL). The relational database seems to have a more efficient way to execute this functionality compared to the graph database. The Cypher query falls in category 3, the SQL one in category 2. Execution of all SQL versions of queries except B_3 and B_{12} fall into category 1. The SQL version of B_3 , concatenation in combination with a second concept, lies in category 3 while the Cypher version falls in category 1.

Discussion

Both database engines and target languages are perfectly suitable to formulate prototypical queries with. With SQL executed on a MySQL database as well as Cypher run against a Neo4j database we found one query type which does perform badly (category 3): Soundex for Cypher (B_{12}) and combination of different full-text searches for SQL (B_3). We argue that the Soundex functionality might be more important for users of digital libraries to be computed in a fast fashion as this is an integral part of all queries containing

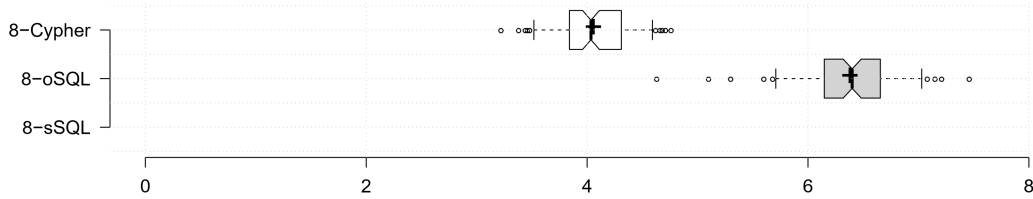


Figure 5.14: Queries with very high execution time in ms, zoom on formulations with low execution time.

author names which are oftentimes hard to spell correctly. In general the SQL queries on the relational database have lower execution time by a small margin than the Cypher ones on the graph-based DBMS. So regarding all problems we conclude that SQL run on a MySQL database is a more suitable target language for the SchenQL compiler compared to Cypher on a Neo4j database engine, thus validating hypothesis H_1 .

Benchmark Part II: SchenQL Performance

In the second part of the benchmark we want to assess the performance of our SchenQL to SQL compiler compared to queries directly formulated in SQL. Here we hope to assess the optimisation degree of compiled SchenQL queries contrasting directly formulated SQL ones. We intend to assess hypothesis H_2 .

Setting

In this part of the benchmark, we compare execution time of queries generated by our SchenQL to SQL compiler to ones in SQL we constructed already in the previous evaluation in Section Benchmark Part I: Database Engine and Target Language Performance. We again utilise the same queries (see Table 5.2) and evaluation environment described in Section 5.4.1. Table 5.5 contains SchenQL formulations of the queries.

Analysis of H_2

Here we again refer to the Figures mentioned in Table 5.3. We compare execution times for queries generated by the SchenQL compiler (labelled *sSQL*) to those of manually constructed and optimised SQL queries (labelled *oSQL*).

The query creation strategy of the SchenQL compiler relies on the DBMS optimiser to flatten subqueries and reorder joins. This approach works for

ID	Query in SchenQL
B_1	PUBLICATIONS
B_2	PUBLICATIONS WRITTEN BY "A"
B_3	PUBLICATIONS WRITTEN BY "A" AND APPEARED IN (JOURNAL NAMED "J")
B_4	PERSON AUTHORED (PUBLICATION TITLE "P")
B_5	PUBLICATIONS ABOUT KEYWORD "K"
B_6	KEYWORDS OF (PUBLICATION TITLED "P")
B_7	COAUTHORS OF "A"
B_8	COAUTHORS OF (COAUTHORS OF "A")
B_9	PUBLICATION REFERENCES (PUBLICATION PUBLISHED BY (INSTITUTION MEMBERS "A"))
B_{10}	MOST CITED (PUBLICATIONS WRITTEN BY "A")
B_{11}	COUNT (PUBLICATIONS WRITTEN BY "A")
B_{12}	PERSON NAMED \sim "F"
B_{13}	PUBLICATIONS WRITTEN BY "A" OR WRITTEN BY "B"
B_{14}	PUBLICATIONS WRITTEN BY "A" AND NOT WRITTEN BY "B"
B_{15}	PUBLICATIONS ABOUT "T"

Table 5.5: Overview of SchenQL queries derived from Table 5.2 evaluated in the benchmark. A , B are unique names of different authors, J is a journal acronym, P is a publication title, K is a keyword, F is a forename and T is a term.

most of the evaluated queries (B_1 , B_2 , B_4 , B_5 , B_6 , B_7 , B_8 , B_{10} , B_{11} , B_{14} , B_{15}) as their execution times fall in category 1 (see Table 5.3). The query execution times for B_3 and B_{12} lie in the same categories (category 3 and category 2) for compiler-generated and manually optimised SQL formulations. B_8 and B_{13} are exceptions: the manually formulated queries both belong to category 1 whereas the compiled versions lie in category 3, thus outperforming the queries generated by the SchenQL compiler by orders of magnitude. B_{13} utilises logical disjunction and joins the publication table, which contains about 5×10^6 records, with the person table (2.5×10^6 records). There could be a problem with the MySQL optimiser, the compiler-generated query seems simple to optimise (see Listing 5.1). In B_8 the two layers of indirection are causing the SchenQL-to-SQL compiler to generate a cascade of subqueries. It seems that the optimiser is unable to increase performance here so that the execution time difference between the optimised query and the generated version is about factor 10^4 .

```

SELECT DISTINCT title, year
FROM publication p join person_authored_publication
    ↪ ON publicationKey = p.key JOIN person pe ON pe
    ↪ .key = personKey
WHERE pe.key IN (
    SELECT DISTINCT key
    FROM person JOIN person_names ON personKey = key
    WHERE name = "A")
OR key IN (
    SELECT DISTINCT key
    FROM person JOIN person_names ON personKey = key
    WHERE name = "B")
LIMIT 100;

```

Listing 5.1: SchenQL compiler-generated SQL for B_{13} .

Discussion

In general it was evident that in most cases the optimiser of the MySQL RDBMS enhances the compiler-generated SQL queries to the level of the manually optimised version. Only in the case of two conditions in a query and the search for co-authors of co-authors of a person the SchenQL SQL code requires orders of magnitude more execution time than the manually written and optimised queries. From these observations we derive a general high performance of the SchenQL-to-SQL compiler which is comparable to manually formulated queries and thus validate hypothesis H_2 .

5.4.2 Qualitative Study: Interviews

To get a comprehensive picture of SchenQL's completeness and suitability, we conducted semi-structured one-on-one interviews with four employees of the dblp team to discover realistic use-cases as well as desirable functionalities and potential extensions. Leading questions were which queries they would like to answer with the data and which functions or visualisations they envisioned in a GUI. The participants do work daily on digital libraries and are thus considered highly experienced in the area. They were only aware of the domain of interest and the underlying data set but did not know anything about SchenQL.

The interviews showed that the dblp staff wished to formulate queries to compute keywords of other publications that were published in the same journal as a given publication, the determination of the most productive or cited

authors, as well as the most cited authors with few co-authors. Furthermore, a GUI should support numerous visualisations: colour coded topics of publications or co-author-groups were explicitly asked for. Another participant requested intermateable components for the visualisation of graphs to display co-publications, co-institutions or connections between different venues. Other desired functionalities were a fault-tolerant person name search and sophisticated ranking methods.

As expected, the experts' suggestions were quite specific and strongly shaped by their daily work with dblp, which may not fit classic non-expert use of digital libraries. SchenQL is able to formulate several of the desired questions, however it needs to be evaluated by non-power-users as we have done in the quantitative evaluation described below to ensure usability for casual users as well. The experts' comments on visualisation drove the design of the GUI's visual analysis components.

5.4.3 Quantitative Study: SchenQL CLI vs. SQL, GUI and UEQ

Our quantitative study consists of two parts: First, the SchenQL CLI is compared to SQL, then the usability of the GUI and thus the SchenQL system as a whole is assessed. For the first part, it is not feasible to compare a specialised system such as the SchenQL CLI to a commercial search engine, as differences between the compared systems should be minor [17]. Additionally, as stated above, search interfaces in this domain [15, 24] do not provide as many functionalities as SchenQL. We also refrained from evaluating the CLI against other DSLs such as Cypher [13] as test users would have been required to learn two new query languages. Comparing our CLI against SPARQL would have required the definition of classes, properties and labels for the data set and was therefore also disregarded in favour of the comparison against SQL.

Users participated voluntarily in the study, they were aware of being able to quit any time without negative consequences. They actively agreed on their data being collected anonymously and their screens being captured. We assume gender does not influence the measured values so it is not seen as additional factor in the evaluation. Kelly [17] advises to examine quasi-independent variables such as sex of test users if researchers believed they influenced the outcome variable. We assume domain-experts are versed in the vocabulary and connections between bibliographic objects, non-experts might have their first encounter with bibliographic metadata.

For our significance tests, we used an independent two-sample t-test in

- Q_1 What are the titles of publications written by author A ?
- Q_2 What are the names of authors which published at conference C ?
- Q_3 What are the titles of the publications referenced by author A in year Y ?
- Q_4 What are the titles of the five most cited publications written by author A ?

Table 5.6: Templates of all queries used in the qualitative evaluations. A are unique names of different authors, C is the acronym of a conference and Y is a year.

case data is normally distributed (checked with Shapiro-Wilk test) and if variances are homogeneous (checked with Levene’s test). Otherwise and if we do not specify differently we applied Mann-Whitney U tests. We consider a p -value of .05 as significance level. We use Fisher’s exact tests to check whether the frequency distributions of categorical (nominal or ordinal) variables differ from the expected distributions in cases where the expected value is less than five. If we encounter nominal and scaled variables we utilise the ETA coefficient to calculate correlations. We measure correlation between ordinal values or ordinal and scaled values with Kendall’s τ_B . Correlation between ordinal and nominal values is estimated with likelihood ratio (LR), effect size is given with Cramér’s V .

Queries

In both parts of the study, we asked the participants to find answers to the queries given in Table 5.6 using either SchenQL CLI/SQL (part I) or the GUI (part II). The used queries are inspired by everyday search tasks of users of digital libraries. Common information needs are e.g. lookup of titles of specific publications or identification of persons working in a specific area [8]. Such information needs can be classified as simple information search as well as exploratory search tasks [29]. We formulated four different types of queries targeting core concepts found in the domain. Variables were switched between query languages to prevent learning effects based on query results. Q_1 , Q_3 and Q_4 are publication searches while Q_2 targets person search. Q_1 and Q_2 can be answered by using dblp [24] alone. Except for Q_3 , Semantic Scholar could technically be used to find answers for the queries. The following formulation of Q_3 in SQL intends to show the complexity of those queries:

```
SELECT DISTINCT title
FROM publication p, publication_references r
```

Query	SchenQL	SQL
Q_1	PUBLICATIONS WRITTEN BY "A"	SELECT title FROM publication p JOIN person_authored_publication pap ON p.key=pap.publicationKey NATURAL JOIN person_names WHERE person_names.name = "A"
Q_2	AUTHORS PUBLISHED IN "C"	SELECT primaryName FROM person p JOIN person_authored_publication pap ON p.key=pap.personKey NATURAL JOIN publication WHERE conference_key = "C"
Q_4	MOST CITED (PUBLICATIONS WRITTEN BY "A")	SELECT title, COUNT (*) FROM publication p JOIN person_authored_publication pap ON p.key = pap.publicationKey NATURAL JOIN person_names JOIN publication_references pr ON p.key = pr.pub2_id WHERE name = "A" GROUP BY title ORDER BY COUNT (*) DESC LIMIT 5

Table 5.7: SchenQL and SQL formulations of queries utilised in our evaluations.

```
WHERE p.key = r.pub2_id AND r.pub_id IN (
  SELECT publicationKey
  FROM person_authored_publication pap NATURAL JOIN
    ↪ person_names JOIN publication p2 ON p2.key = pap
    ↪ .publicationKey
  WHERE person_names.name = "A" AND year = Y);
```

In SchenQL, the query could be formulated as follows (for all queries see Table 5.7):

```
PUBLICATIONS CITED BY (PUBLICATIONS WRITTEN BY "A" IN
  ↪ YEAR Y);
```

We refrained from evaluating more complex queries to keep the construction time for SQL queries feasible.

User Study Part I: SchenQL CLI vs. SQL

With this first part of the quantitative study, we assess the usability, suitability as well as user satisfaction of usage of the SchenQL CLI compared to SQL for queries typically answered with an information retrieval system operating on bibliographic metadata. Additionally, the need for a DSL in the domain of bibliographic metadata is analysed as we try to verify or falsify hypotheses H_3 and H_4 . Participants of this evaluation needed to be familiar with SQL.

Setting

We defined the evaluation process of our archetypical interactive information retrieval study [17] as follows: every user performed the evaluation alone in presence of a passive investigator on a computer with two monitors. The screens were captured in order to measure times used to formulate the queries. All participants formulated all queries in SQL and SchenQL. A query language was assigned with which a user was going to start the evaluation, it was switched between users to compensate for learning effects. Users were permitted to use the internet at any stage of the evaluation. A SchenQL cheat sheet, the ER diagram and examples for the database schema provided to test subjects can be found in Kreutz et al. [37].

At first, a video tutorial¹⁵ for the introduction and usage of SQL and the SchenQL CLI was shown, afterwards subjects were permitted to formulate queries using the system they were starting to work with. Following this optional step, users were asked to answer a first online questionnaire to assess their current and highest level of SQL knowledge (both on a scale from 1 (*no knowledge*) to 6 (*very good knowledge*)), the number of times they used SQL in the past three months (*0 times*, *1-5 times*, *more than 5 times* and *daily*) and their familiarity with the domain of bibliographic metadata. Participants were asked to submit the queries in SQL and SchenQL respectively. The queries were always formulated in the following order: Q_1 , Q_2 , Q_3 , Q_4 . We consciously ordered the queries such that more complex SQL queries followed the easier/shorter ones to help users in query formulation. This part of the first quantitative evaluation was concluded with a second online questionnaire regarding the overall impression of SchenQL, the rating of SchenQL and SQL for the formulation of queries as well as several open questions targeting possible advantages and improvements of SchenQL. We evaluated 21 participants from the area of computer science with SQL knowledge. In total, ten subjects started by using SQL, eleven participants began the evaluation using SchenQL.

Analysis of H_3

To assess the validity of hypothesis H_3 of SchenQL leading to better results than using SQL, we observe the *number of correctly formulated queries*, the *rated difficulty* and the *required time* for the formulation of queries with the SchenQL CLI and SQL. For each of these values, we first conducted significance tests on all four queries together, here the two languages SchenQL and SQL were regarded as groups, afterwards we performed significance tests

¹⁵SchenQL Evaluation CLI vs. SQL - Tutorial: <https://youtu.be/g7J64wzbE5I>

	SQL			SchenQL CLI		
	CORR	DIFF	time	CORR	DIFF	time
Q_1	90.48	2.86	4:57	90.48	1.57	2:57
Q_2	90.48	3.	4:35	100.	2.1	3:11
Q_3	23.81	4.86	8:55	47.62	2.71	3:33
Q_4	23.81	5.91	10:36	95.24	1.71	1:53

Table 5.8: Correctness (CORR) in percent, assessed average difficulty (DIFF) and average time in minutes for the four queries for SQL and the SchenQL CLI.

on each of the four queries. Table 5.8 gives an overview of correctness, average rated difficulty and average time for formulating all four queries for both languages. Difficulty was rated on a scale from 1 (*very easy*) to 7 (*very difficult*) to allow neutral ratings.

Correctness 57.14% of queries were correctly formulated using SQL whereas 83.33% of queries were correctly formulated using the SchenQL CLI. This result clearly shows the significantly (U=2604, p=0) superior effectiveness of SchenQL compared to SQL in terms of overall correctness. While Q_1 and Q_2 were answered correctly by most participants, the number of correctly formulated queries for Q_3 and Q_4 highly depends on the system. Q_4 was correctly answered by a quarter of the subjects using SQL while more than 95% of users were able to formulate the query in SchenQL, this difference is significant (U=63, p=0). These observations support the partial verification of H_3 in terms of higher number of correctly formulated queries with the SchenQL CLI compared to SQL.

Rated Difficulty The mean rating of difficulty of the formulation of queries with SQL was 4.16 ($\sigma = 1.94$), with SchenQL the mean rating was significantly lower (2.02, $\sigma = 1.11$; U=1341, p=0). On average, the query construction using SQL is rated more difficult for every query. The averaged highest rated difficulty for a query in SchenQL is still lower than the averaged lowest rated difficulty of a query in SQL. We found significantly lower ratings of difficulties of queries for all four queries (Q_1 : U=114, p=.005; Q_2 : U=143.5, p=.044; Q_3 (t-test): t=-5.539, p=0; Q_4 : U=0, p=0) when using SchenQL compared to utilisation of SQL. These observations support the partial verification of H_3 in terms lower perceived difficulty in query formulation with the SchenQL CLI compared to SQL.

Time Average construction of queries in SQL took 7:15 minutes ($\sigma = 4:47$ min.), with the SchenQL CLI the construction was significantly quicker and took 2:52 minutes ($\sigma = 1:51$ min.; U=1165.5, p=0) on average. This

documents the efficiency of SchenQL. We found significantly lower required times for query formulation for all four queries (Q_1 (t-test): $t=-3.433$, $p=.001$; Q_2 : $U=141.5$, $p=.047$; Q_3 : $U=62$, $p=0$; Q_4 : $U=7$, $p=0$) when using SchenQL compared to the utilisation of SQL. These observations support the partial verification of H_3 in terms of lower required time for query formulation with the SchenQL CLI compared to SQL.

General Results The queries Q_3 and Q_4 in SQL are assumed to be complex which is supported by the low percentage of correct formulations using SQL. They are also much longer than the respective SchenQL ones. That means the time required to write them down is higher and there is more opportunity to make mistakes which causes a query reformulation [32]. The overall rating of suitability of SchenQL for constructing the queries resulted in an average of 6.43 ($\sigma = .6$) while the rating was significantly ($U=7$, $p=0$) lower (3.14, $\sigma = 1.2$) for SQL on a scale from 1 (*very bad*) to 7 (*very good*). While SQL was rated below mediocre, SchenQL was evaluated as excellent which shows users' satisfaction with it. These results lead to the conclusion of SchenQL being highly suitable for solving the given tasks which represent everyday queries of users of digital libraries and a high user acceptance of SchenQL.

In summary, utilisation of SchenQL achieves higher correctness of queries, lower perceived difficulty and requires less time than using SQL, which together verifies hypothesis H_3 .

Analysis of H_4

To assess validity of hypothesis H_4 of SchenQL being as suitable for experts as it is for non-experts, we conduct tests of independence for correctness and rated difficulty and correlation tests for required time for query formulation. We run tests on all queries separately and on the SchenQL system as a whole. Our dependent variable is knowledge in the area of bibliographic metadata. The 21 participants from before form the two user groups: nine participants are non-experts and twelve participants are familiar with bibliographic metadata.

Correctness In general, 75% of queries were correctly formulated by domain-experts whereas the non-experts achieved only 63.89% in both Qs. Participants which were (non-)experts were able to solve 65.58% (47.22%) of queries in SQL and 85.42% (80.56%) in SchenQL. Tian et al. [35] stated that for a domain-expert, it would be easier to write queries in a DSL than in SQL. We found that the observed frequencies for correct and incorrect formulation of queries per group do not significantly deviate from the expected frequencies (separated by query and in general for all SchenQL queries; Fisher's exact

tests if there were both correct and incorrect results for queries). We did not find enough evidence to suggest that domain knowledge and correctness of formulated queries are associated.

Rated Difficulty We did not find enough evidence to suggest that domain knowledge and rated difficulty of query formulation are associated (separated by query and in general for all SchenQL queries; Fisher’s exact tests).

Time We found no strong correlation between the two groups of domain knowledge and required time for query formulation (separated by query and in general for all SchenQL queries; ETA coefficient).

Result No user group is consistently better than the other, we found no deviations from expected frequencies for correctness and rated difficulty. We also did not find strong correlations between required time for query formulation and domain knowledge. The SchenQL CLI seems to be as suitable for domain-experts as it is for non-experts, thus, H_4 is verified.

Open Questions and Discussion

In the open questions, the short, easy and intuitive SchenQL queries were complimented by many participants. Users noted the comprehensible syntax was suitable for non-computer scientists as it resembles natural language. Some noted their initial confusion due to the syntax and their incomprehension of usage of literals or limitations. Others asked for auto-completion, syntax highlighting, a documentation and more functions such as a most cited with variable return values. No participant wished for visualisations which could be caused by design fixation [16] or generally lower requirements for such a system compared to the experts from the qualitative study.

The average overall impression of the SchenQL QL was rated by the subjects as 5.05 ($\sigma = .74$) on a scale from 1 (*very bad*) to 6 (*very good*), enforcing a non-neutral rating. Assessed difficulty and required times to formulate the four queries were significantly lower when utilising SchenQL compared to SQL, the overall correctness of all queries was significantly higher for SchenQL as well. This verified hypothesis H_3 of the CLI leading to generally better results than SQL. Our hypothesis H_4 of the SchenQL system being as suitable for domain-experts as it is for casual users is also verified. No user group was found to be consistently better than the other one, we did not find significant deviations from expected frequencies. We also did not find strong correlations between required times for query formulation and knowledge of bibliographic metadata.

We performed correlation tests on the collected data of participants regarding their current and highest level of SQL knowledge as well as the number of times they used SQL in the three months preceding the eval-

uation. The participants' current skill in SQL highly correlates with their overall rating of our QL ($\tau_B = .53$). Being versed in advanced SQL could lead to a higher appreciation of complexity hidden from users in SchenQL. The quantity in which the participants were using SQL in the last three months correlates with their rating of difficulty of Q_1 ($\tau_B = .54$) and Q_2 ($\tau_B = .42$) in SQL and Q_4 with the CLI ($\tau_B = .46$). Having recently used SQL could lead to higher familiarity with it and therefore perceived easier construction of queries if they are not too complex. The number of times SQL was used in the last three months correlates with the correctness of Q_3 (LR, $V = .67$) and Q_4 (LR, $V = .87$) in SQL. Having used SQL recently seems to help persons formulate difficult queries more successfully.

This evaluation lead to the construction of the prototypical GUI with its syntax suggestion as well as auto-completion features. Additionally, although they were not mentioned by participants in this evaluation, some visualisations were included following suggestions from the qualitative evaluation.

User Study Part II: SchenQL GUI vs. CLI and User Experience Questionnaire

This second part of the quantitative study focused on evaluating the GUI and, thus, the SchenQL system as a whole. We assessed how usage of the web interface compared to users' impressions and performance when utilising the SchenQL CLI. Beside a part where test users answered queries with the GUI, we conducted the User Experience Questionnaire [33] to measure user experience with the SchenQL system. To resemble our target audience we did not pose the precondition of users being familiar with SQL or the formulation of structured queries. Here, we intend to assess the hypothesis H_5 .

Setting

This evaluation is performed analogous to the previous part: every user performed the evaluation alone but in presence of a passive investigator on a computer with two monitors. We measured times used to find answers by capturing screens. The same SchenQL cheat sheet as in the first part was provided to the test subjects. At first, a video tutorial¹⁶ introduced the usage of the SchenQL GUI. The next part was the formulation or the navigation towards solutions of the four queries introduced in Table 5.6 using the GUI. Afterwards, the subjects completed the User Experience Questionnaire [33]

¹⁶SchenQL Evaluation GUI - Tutorial: <https://youtu.be/56-23zyUDPQ>

	SchenQL GUI		
	CORR	DIFF	time
Q_1	90	1.3	1:05
Q_2	90	2.2	1:41
Q_3	40	3.6	2:56
Q_4	90	2.4	2:18

Table 5.9: Correctness (CORR) in percent, assessed average difficulty (DIFF) and average time in minutes for the four queries for the GUI.

followed by questions regarding the overall impression of the GUI as well as possible improvements.

We evaluated ten participants from the area of computer science and adjacent fields, which did not yet take part in a previous evaluation of the SchenQL system.

Partial Analysis of H_5 : users unfamiliar with query formulation

To assess partial validity of hypothesis H_5 in terms of the GUI’s suitability for users unfamiliar with query formulation, we conduct significance tests on all queries together and each separate query for correctness, rated difficulty and required time. We observe the results from usage of the SchenQL CLI from the previous evaluation and participants’ results from utilisation of the GUI as the two groups. Table 5.9 gives an overview of the correctness, average rated difficulty and average required time for all four queries when using the SchenQL system.

Correctness Except for Q_3 , participants mostly solved the queries correctly, resulting in an overall correctness of 77.5% (-10.83% compared to CLI, difference not significant). We found no significant differences between the two groups for correctness in any of the four queries.

Rated Difficulty Users rated the perceived difficulty of queries as 2.38 (+.35 compared to CLI, difference not significant) on average. We found no significant differences between the two groups for rated difficulty in any of the four queries.

Time Users took about 2:15 minutes for the retrieval of a solution (-0:37 minutes compared to CLI, difference is significant; $U=1207$, $p=.011$) on average. We found significant differences in times required to solve queries Q_1 and Q_2 . Times required for formulating the queries with the GUI were significantly (Q_1 : $U=33.5$, $p=.002$; Q_2 : $U=41$, $p=.006$) lower than those resulting from using the CLI. As these queries were relatively simple, we assume the auto-completion and suggestion-feature of the GUI is especially

dimension	description
attractiveness	overall impression, users' approval or disapproval
perspicuity	easiness to familiarise with product and to learn usage
efficiency	effort required to solve tasks, reaction times of product
dependability	security and predictability of product, level of control of users' interaction
stimulation	excitement, motivation, fun
novelty	creativity and innovation of product, sparks user's interest

Table 5.10: Description of the six dimensions measured by the UEQ to capture users' impressions of interactive products [33].

helpful in the fast construction of straightforward queries or the GUI offering other suitable ways of quickly obtaining simple bibliographic information. Usage of the GUI might be more intuitive compared to writing simple queries in the SchenQL CLI.

General Results We want to point out that participants from the first part of the quantitative study who were familiar with query formulation, but were not offered help in the construction, did not significantly differ in rating of difficulty and correctness from users of this user study. In case of the GUI, the subjects were supported in the formulation of queries but were not necessarily familiar with this kind of task. Hence, we assume the system's suggestion and auto-completion feature is useful for redemption of unequal prior knowledge in this case.

Correctness and rating of difficulty did not differ significantly between usage of CLI and GUI, but users were significantly faster in finding answers for simple queries with the GUI which underlines the suitability of the interface for everyday usage. Participants from this study resemble SchenQL's target audience, which additionally emphasises its usefulness and partly verifies hypothesis H_5 in terms of the GUI being suitable for users not versed with structured query formulation.

Partial Analysis of H_5 : UEQ

The *attractiveness*, *perspicuity*, *efficiency*, *dependability*, *stimulation* and *novelty* of interactive products can be measured with the User Experience Questionnaire (UEQ) [33] even at small sample sizes. Table 5.10 describes the aspects the UEQ measures. Here, we want to conclude the assessment of the validity of hypothesis H_5 in terms of rating of user experience.

Participants of this study answered the 26 questions of the UEQ regarding

usage of the SchenQL system. Ratings on pairs of contrasting stances (-3 to 3) such as *complicated-easy* or *boring-exciting* were then grouped to the six dimensions mentioned before. Values above .8 are generally considered as positively evaluated equalling high user experience, values above 2 are rarely encountered.

In general, users seem to enjoy using the SchenQL system (attractiveness = 2.07, $\sigma = .25$). The handling of our system is extremely easy learned (per-spicuity = 2.3, $\sigma = .19$). Tasks can be solved without unnecessary effort (efficiency = 2.03, $\sigma = .49$) and users feel in control of the system (dependability = 1.83, $\sigma = .63$). They seem exited to use the SchenQL system (stimulation = 1.73, $\sigma = .33$) and rate the system as innovative and interesting (novelty = 1.58, $\sigma = .68$).

As all six quality dimensions achieved ratings well over .8, the system is positively evaluated which equals high user experience and partially verifies H_5 .

Open Questions and Discussion

In the open questions, participants praised the intuitive usability, the auto-completion and the suggestion feature. For future development, suggestions for literals were requested and two participants wished for a voice input. Remarkably, not a single user mentioned the need for more or other visualisations, this is possibly attributed to design fixation [16] but might also stem from the advanced needs of power users from the expert interviews.

The users were significantly faster in solving simple queries when using the GUI compared to the CLI. As we found no significant impairments from utilisation of the GUI, we assume its usefulness and usability for query formulation. Participants from this study were far less familiar with the construction of structured queries compared to those of the previous study but seemed to be adequately supported by the GUI in the retrieval of information. Together with the UEQ which showed users' high ratings ($> .8$) for all six quality dimensions (which proves high user experience), hypothesis H_5 could be partially verified.

5.5 Conclusion and Future Work

We evaluated the SchenQL system, a domain-specific query language operating on bibliographic metadata from the area of computer science with accompanying GUI supporting query formulation. Our thorough evaluation against SQL showed the need for such a DSL. Test subjects' satisfaction

with the SchenQL system was assessed with application of the UEQ. The introduction of a GUI and its evaluation with users resembling our target audience did not significantly change the correctness of answers or the users' rating of difficulty of the queries compared to the CLI but instead the time needed to formulate simple queries was reduced significantly. Missing prior knowledge with structured query formulation seems to be compensated by using a GUI with a suggestions and auto-completion feature. As the CLI and the GUI proved to be viable tools for information retrieval on bibliographic metadata, users' preferences should decide which one to use.

The target language SQL run on a MySQL database engine for the SchenQL compiler was a more suitable choice than Cypher run on a Neo4j database engine (H_1) and the performance of the generated queries is as high as manually formulated ones (H_2). Using SchenQL lead to generally better results compared to the utilisation of SQL (H_3). The system seems to be as suitable for domain-experts as it is for non-experts (H_4). Our GUI has high usability for users not familiar with structured query formulation (H_5).

Future efforts could focus on the identification of query types which would better be run on a graph database and then decide which query will be translated in SQL and which one will be translated to Cypher. Enhancements of functionalities could include more visualisations such as color-coded topics or graph visualisation as the experts from the qualitative study requested. Furthermore, more specific query options such as a filter for papers with few co-authors or most cited with variable return values could be included. As visualisations were not relevant for users in our quantitative evaluation, future efforts could focus on supporting more advanced query options: algorithms for social network analysis as PageRank, computation of mutual neighbours, hubs and authorities or connected components [34] would fit. Centrality of authors, the length of a shortest path between two authors and the introduction of aliases for finding co-citations [12] would also be useful query building blocks. As user-defined functions [34] were well-received in other work [32], they are a further prospect. Incorporation of social relevance in the search and result representation process as shown in [2] could also be an extension. User profiles could store papers and keywords, which in terms influence results of search and exploration.

Bibliography

- [1] Vasco Amaral, S. Helmer, and G. Moerkotte. A visual query language for hep analysis. 2003 IEEE Nuclear Science Symposium. Conference Record (IEEE Cat. No.03CH37515), 2:829–833 Vol.2, 2003.
- [2] Sihem Amer-Yahia, Laks V. S. Lakshmanan, and Cong Yu. Socialscope: Enabling information discovery on social content sites. In Fourth Biennial Conference on Innovative Data Systems Research, CIDR 2009, Asilomar, CA, USA, January 4-7, 2009, Online Proceedings. www.cidrdb.org, 2009.
- [3] Ricardo Baeza-Yates and Berthier A. Ribeiro-Neto. Modern Information Retrieval - the concepts and technology behind search, Second edition. Pearson Education Ltd., Harlow, England, 2011.
- [4] Marcia Bates. Task force recommendation 2.3 research and design review: Improving user access to library catalog and portal information: Final report (version 3). 01 2003.
- [5] Jeffrey Beall. The weaknesses of full-text searching. The Journal of Academic Librarianship, 34(5):438–444, 09 2008.
- [6] Gerd Berget and Frode Eika Sandnes. Why textual search interfaces fail: a study of cognitive skills needed to construct successful queries. Inf. Res., 24(1), 2019.
- [7] Christine Betts, Joanna Power, and Waleed Ammar. Grapal: Connecting the dots in scientific literature. In Marta R. Costa-jussà and Enrique Alfonseca, editors, Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28 - August 2, 2019, Volume 3: System Demonstrations, pages 147–152. Association for Computational Linguistics, 2019.
- [8] Stephan Bloehdorn, Philipp Cimiano, Alistair Duke, Peter Haase, Jörg Heizmann, Ian Thurlow, and Johanna Völker. Ontology-based question answering for digital libraries. In László Kovács, Norbert Fuhr, and Carlo Meghini, editors, Research and Advanced Technology for Digital Libraries, 11th European Conference, ECDL 2007, Budapest, Hungary, September 16-21, 2007, Proceedings, volume 4675 of Lecture Notes in Computer Science, pages 14–25. Springer, 2007.

- [9] Andrey Borodin, Yuri Kiselev, Sergey Mirvoda, and Sergey Porshnev. On design of domain-specific query language for the metallurgical industry. In Stanislaw Kozielski, Dariusz Mrozek, Pawel Kasprowski, Bozena Malysiak-Mrozek, and Daniel Kostrzewa, editors, Beyond Databases, Architectures and Structures - 11th International Conference, BDAS 2015, Ustroń, Poland, May 26-29, 2015, Proceedings, volume 521 of Communications in Computer and Information Science, pages 505–515. Springer, 2015.
- [10] J. Chambers, William Cleveland, B. Kleiner, and P. Tukey. Graphical methods for data analysis (vol 17, pg 180, 1983). Journal of Sleep Research, 21:484–484, 08 2012.
- [11] Christian S. Collberg. A fuzzy visual query language for a domain-specific web search engine. In Mary Hegarty, Bernd Meyer, and N. Hari Narayanan, editors, Diagrammatic Representation and Inference, Second International Conference, Diagrams 2002, Callaway Gardens, GA, USA, April 18-20, 2002, Proceedings, volume 2317 of Lecture Notes in Computer Science, pages 176–190. Springer, 2002.
- [12] Anton Dries, Siegfried Nijssen, and Luc De Raedt. Biql: A query language for analyzing information networks. In Michael R. Berthold, editor, Bisociative Knowledge Discovery - An Introduction to Concept, Algorithms, Tools, and Applications, volume 7250 of Lecture Notes in Computer Science, pages 147–165. Springer, 2012.
- [13] Nadime Francis, Alastair Green, Paolo Guagliardo, Leonid Libkin, Tobias Lindaaker, Victor Marsault, Stefan Plantikow, Mats Rydberg, Petra Selmer, and Andrés Taylor. Cypher: An evolving query language for property graphs. In Gautam Das, Christopher M. Jermaine, and Philip A. Bernstein, editors, Proceedings of the 2018 International Conference on Management of Data, SIGMOD Conference 2018, Houston, TX, USA, June 10-15, 2018, pages 1433–1445. ACM, 2018.
- [14] Ferruccio Guidi and Irene Schena. A query language for a metadata framework about mathematical resources. In Andrea Asperti, Bruno Buchberger, and James H. Davenport, editors, Mathematical Knowledge Management, Second International Conference, MKM 2003, Bertinoro, Italy, February 16-18, 2003, Proceedings, volume 2594 of Lecture Notes in Computer Science, pages 105–118. Springer, 2003.

- [15] Andreas Hotho, Robert Jäschke, Dominik Benz, Miranda Grahl, Beate Krause, Christoph Schmitz, and Gerd Stumme. Social bookmarking am beispiel bibsonomy. In Andreas Blumauer and Tassilo Pellegrini, editors, Social Semantic Web: Web 2.0 - Was nun?, X.media.press, pages 363–391. Springer, 2009.
- [16] David Jansson and Steven Smith. Design fixation. Design Studies, 12, 01 1991.
- [17] Diane Kelly. Methods for evaluating interactive information retrieval systems with users. Found. Trends Inf. Retr., 3(1-2):1–224, 2009.
- [18] Taraneh Khazaei and Orland Hoerber. Supporting academic search tasks through citation visualization and exploration. Int. J. Digit. Libr., 18(1):59–72, 2017.
- [19] Stefan Klink, Michael Ley, Emma Rabbidge, Patrick Reuther, Bernd Walter, and Alexander Weber. Browsing and visualizing digital bibliographic data. In Oliver Deussen, Charles D. Hansen, Daniel A. Keim, and Dietmar Saupe, editors, 6th Joint Eurographics - IEEE TCVG Symposium on Visualization, VisSym 2004, Konstanz, Germany, May 19-21, 2004, pages 237–242. Eurographics Association, 2004.
- [20] Christin Katharina Kreutz, Michael Wolz, and Ralf Schenkel. Schenql: A concept of a domain-specific query language on bibliographic metadata. In Adam Jatowt, Akira Maeda, and Sue Yeon Syn, editors, Digital Libraries at the Crossroads of Digital Information for the Future - 21st International Conference on Asia-Pacific Digital Libraries, ICADL 2019, Kuala Lumpur, Malaysia, November 4-7, 2019, Proceedings, volume 11853 of Lecture Notes in Computer Science, pages 239–246. Springer, 2019.
- [21] Christin Katharina Kreutz, Michael Wolz, Benjamin Weyers, and Ralf Schenkel. Schenql: Evaluation of a query language for bibliographic metadata. In Emi Ishita, Natalie Lee-San Pang, and Lihong Zhou, editors, Digital Libraries at Times of Massive Societal Transition - 22nd International Conference on Asia-Pacific Digital Libraries, ICADL 2020, Kyoto, Japan, November 30 - December 1, 2020, Proceedings, volume 12504 of Lecture Notes in Computer Science, pages 323–339. Springer, 2020.
- [22] Martin Krzywinski and Naomi Altman. Visualizing samples with box plots. Nat Methods, 11:119–20, 02 2014.

- [23] Ulf Leser. A query language for biological networks. In ECCB/JBI'05 Proceedings, Fourth European Conference on Computational Biology/Sixth Meeting of the Spanish Bioinformatics Network (Jornadas de BioInformática), Palacio de Congresos, Madrid, Spain, September 28 - October 1, 2005, page 39, 2005.
- [24] Michael Ley. DBLP - some lessons learned. Proc. VLDB Endow., 2(2):1493–1500, 2009.
- [25] Aastha Madaan. Domain specific multi-stage query language for medical document repositories. Proc. VLDB Endow., 6(12):1410–1415, 2013.
- [26] Mauro San Martín, Claudio Gutiérrez, and Peter T. Wood. SNQL: A social networks query and transformation language. In Pablo Barceló and Val Tannen, editors, Proceedings of the 5th Alberto Mendelzon International Workshop on Foundations of Data Management, Santiago, Chile, May 9-12, 2011, volume 749 of CEUR Workshop Proceedings. CEUR-WS.org, 2011.
- [27] Jakob Nielsen. Usability engineering. Academic Press, 1993.
- [28] Piritta Numminen and Pertti Vakkari. Question types in public libraries' digital reference service in finland: Comparing 1999 and 2006. J. Assoc. Inf. Sci. Technol., 60(6):1249–1257, 2009.
- [29] Peter Pirolli. Powers of 10: Modeling complex information-seeking systems at multiple scales. Computer, 42(3):33–40, 2009.
- [30] Florian Reitz. A framework for an ego-centered and time-aware visualization of relations in arbitrary data repositories. CoRR, abs/1009.5183, 2010.
- [31] Mukesh Kumar Rohil, Rohan Kumar Rohil, Divyesakshi Rohil, and Anurag Runthala. Natural language interfaces to domain specific knowledge bases: An illustration for querying elements of the periodic table. In Yingxu Wang, Sam Kwong, Jerome Feldman, Newton Howard, Phillip C.-Y. Sheu, and Bernard Widrow, editors, 17th IEEE International Conference on Cognitive Informatics & Cognitive Computing, ICCI*CC 2018, Berkeley, CA, USA, July 16-18, 2018, pages 517–523. IEEE Computer Society, 2018.
- [32] André Schaefer, Matthias Jordan, Claus-Peter Klas, and Norbert Fuhr. Active support for query formulation in virtual digital libraries: A case study with DAFFODIL. In Andreas Rauber, Stavros Christodoulakis,

- and A Min Tjoa, editors, Research and Advanced Technology for Digital Libraries, 9th European Conference, ECDL 2005, Vienna, Austria, September 18-23, 2005, Proceedings, volume 3652 of Lecture Notes in Computer Science, pages 414–425. Springer, 2005.
- [33] Martin Schrepp, Andreas Hinderks, and Jörg Thomaschewski. Applying the user experience questionnaire (UEQ) in different evaluation scenarios. In Aaron Marcus, editor, Design, User Experience, and Usability. Theories, Methods, and Tools for Designing the User Experience - Third International Conference, DUXU 2014, Held as Part of HCI International 2014, Heraklion, Crete, Greece, June 22-27, 2014, Proceedings, Part I, volume 8517 of Lecture Notes in Computer Science, pages 383–392. Springer, 2014.
- [34] Jiwon Seo, Stephen Guo, and Monica S. Lam. Socialite: An efficient graph query language based on datalog. IEEE Trans. Knowl. Data Eng., 27(7):1824–1837, 2015.
- [35] Hao Tian, Rajshekhar Sunderraman, Robert J. Calin-Jageman, Hong Yang, Ying Zhu, and Paul S. Katz. Neuroql: A domain-specific query language for neuroscience data. In Torsten Grust, Hagen Höpfner, Arantza Illarramendi, Stefan Jablonski, Marco Mesiti, Sascha Müller, Paula-Lavinia Patranjan, Kai-Uwe Sattler, Myra Spiliopoulou, and Jef Wijsen, editors, Current Trends in Database Technology - EDBT 2006, EDBT 2006 Workshops PhD, DataX, IIDB, IIHA, ICSNW, QLQP, PIM, PaRMA, and Reactivity on the Web, Munich, Germany, March 26-31, 2006, Revised Selected Papers, volume 4254 of Lecture Notes in Computer Science, pages 613–624. Springer, 2006.
- [36] Boyan Xu, Ruichu Cai, Zhenjie Zhang, Xiaoyan Yang, Zhifeng Hao, Zijian Li, and Zhihao Liang. NADAQ: natural language database querying based on deep learning. IEEE Access, 7:35012–35017, 2019.

6. Evaluating Semantometrics from Computer Science Publications

Outline

6.1	Introduction	76
6.2	Semantometrics and Related Work	78
6.2.1	Semantometrics	78
6.2.2	Related Work	80
	Scientific Influence Assessment	80
	Identification of Important Referenced Papers	82
	Citation Behaviour Analysis	82
	Document Vector Representations	83
6.3	SUSdblp Dataset	84
6.3.1	Introduction	84
6.3.2	Contained Data	85
6.3.3	Number of References and Citations	86
6.3.4	Publication Years	87
6.3.5	Sub-fields and Topic Distributions in Publications	90
6.3.6	Description and Discussion	91
6.4	Methodology	93
6.4.1	Document Vector Representations	94
6.4.2	Distance Measures	94
6.4.3	Classification Algorithms	95
6.4.4	Implementation	95
6.5	Evaluation of the Approach	96
6.5.1	Single Features	97
6.5.2	Multiple Features	99
6.5.3	Combination	101
6.5.4	Truly Uninfluential Publications	102

6.5.5	Information Available at Publication Time	102
6.5.6	Discussion	103
6.6	Evaluation of the Dataset	105
6.6.1	Robustness of Dataset	105
6.6.2	Classification for Different Years	106
6.6.3	Discussion	107
6.7	Alternative Approaches for Classification	107
6.7.1	Classification on Number of References and Citations .	108
6.7.2	Classification on Document Vector Representations . .	109
	All Dimensions of Document Vector Representations	
	of Publications	109
	Information Available at Publication Time	110
6.7.3	Discussion	112
6.8	Conclusion and Future Work	113
6.A	Evaluation of the Approach: Single Features	115
6.B	Evaluation of the Approach: All Features	116
6.C	Evaluation of the Approach: 33 Features	116
6.D	Evaluation of the Approach: Combination	117
6.E	Evaluation of the Dataset: Classification for Different Years .	118
6.F	Evaluation of the Dataset: Abstract Length Bias	119
6.G	Other Approaches: All Dimensions of Document Vector Rep-	
	resentations of Publications	119
6.G.1	Single Dimensions of Document Vector Representations	
	of Publications	119
6.G.2	All Dimensions of Document Vector Representations	
	of Publications	121
6.G.3	All Dimensions of Document Vector Representations	
	of Referenced Papers	121
6.G.4	All Dimensions of Document Vector Representations	
	of Citing Publications	122
	Bibliography	124

Bibliographic Information

Kreutz, C. K., Sahitaj, P., Schenkel, R. (2020). Evaluating Semantometrics from Computer Science Publications. In *Scientometrics* 125(3) (pp. 2915-2954). Springer. <https://doi.org/10.1007/s11192-020-03409-5>.

Copyright Notice

©2020 Springer. This is an accepted but reformatted version of this article published in <https://doi.org/10.1007/s11192-020-03409-5>. Clarification of the copyright adjusted according to the guidelines of the publisher.

Figure 6.4 has been reformatted and the caption had to be modified accordingly to specify the location of the histograms

Keywords

Semantometrics • Classification • Citation Network • Natural Language Processing

Abstract

Identification of important works and assessment of importance of publications in vast scientific corpora are challenging yet common tasks subjected by many research projects. While the influence of citations in finding seminal papers has been analysed thoroughly, citation-based approaches come with several problems. Their impracticality when confronted with new publications which did not yet receive any citations, area-dependent citation practices and different reasons for citing are only a few drawbacks of them. Methods relying on more than citations, for example semantic features such as words or topics contained in publications of citation networks, are regarded with less vigour while providing promising preliminary results.

In this work we tackle the issue of classifying publications with their respective referenced and citing papers as either seminal, survey or uninfluential by utilising semantometrics. We use distance measures over words, semantics, topics and publication years of papers in their citation network to engineer features on which we predict the class of a publication. We present the SUSdblp dataset consisting of 1,980 labelled entries to provide a means of evaluating this approach.

A classification accuracy of up to .9247 was achieved when combining multiple types of features using semantometrics. This is +.1232 compared to the current state of the art (SOTA) which uses binary classification to identify papers from classes seminal and survey. The utilisation of one-vector representations for the ternary classification task resulted in an accuracy of .949 which is +.1475 compared to the binary SOTA. Classification based on information available at publication time derived with semantometrics resulted in an accuracy of .8152 while an accuracy of .9323 could be achieved when using one-vector representations.

6.1 Introduction

With the ever growing amount of scientific publications, automatic methods for finding influential or seminal works are indispensable. A majority of research tackles the identification of important works [25, 28, 70, 74, 59, 75]. The diffusion of scholarly knowledge in a citation network is explicitly modelled by citations and references [12], ideas from referenced papers can be utilised or amended. Common approaches are based on the observation of the number of citations which publications received. As this indicator can be highly dependent on a specific dataset, it might be problematic to utilise as a measure of impact [45, 66]. Citations need to be handled with care due

to cases of self-citations [33, 62, 63], varying citation practices in different areas [17, 33, 64, 66, 68], diverging reasons for citing [24], the non-existence of citations of new papers [74] and uncited influences [24, 42, 49].

Distinguishing between seminal publications and popular survey papers might pose a problem as both types are typically cited often [66] but reviews are over-represented amongst highly cited publications whilst not contributing any new content [3]. Seminal papers are ones which are key to a field while surveys review and compare multiple approaches and can be comprehensible summaries of a domain. For lack of space, reviews are often referenced instead of original papers [32]. Influential members of both classes can be distinguished from all other (uninfluential) publications by observing their number of citations after an initial period in which citations are accumulated. Differentiating between seminal and review papers is challenging. Therefore, methods considering more factors than the number of citations and references are required [45, 66, 74] as these observations are no sufficient proxy in the process of measuring publication impact and scientific quality [29, 66], especially at the time a paper is first published. Preferably, an approach with the potential to measure and predict the contribution of a paper and how much it advances its field should be favoured.

Herrmannova et al. [28] assume the classification of a paper as seminal or survey can be performed by observing semantometrics as a new method for research evaluation which uses differences in full texts of a citation network to determine the contribution or value of a publication [36]. Classification of a publication is conducted on features derived from distances between papers in its citation network. The distance between papers citing a seminal paper and its referenced papers is shown to be larger than this distance of a survey as the seminal publication advanced science by a considerable margin. Surveys are shown to reference papers from a broader field compared to seminal papers [28].

Experiments of Herrmannova et al. were conducted on a multi-disciplinary dataset [29], where an accuracy of up to .6897 was achieved. Kreuz et al. performed similar experiments on a dataset covering the area of computer science and achieved accuracies up to .8015 [37]. We define the identification of seminal and survey papers as our core task but also want to incorporate publications into the approach which are uninfluential. This is done to broaden the methodology and to place it in a more realistic setting. Additionally, we want to predict whether a paper is seminal, survey or uninfluential based on the information available at the time of publication. So, the usefulness of this approach is accessed on a dataset restricted to a narrower area while including a third kind of publication.

Our contribution is three-fold: First, we introduce SUSdblp, a dataset

suitable for the task of classifying a publication as seminal, survey or uninfluential by providing reference and citation information of publications from the area of computer science. Second, we analyse the approach presented by Herrmannova et al. [28] in a ternary class setting, using different document representations which encode words, semantics, topics and years of papers as well as numerous classification algorithms. Single and multiple features generated from publications of the new and homogeneous dataset are evaluated during the classification process. Third, we extend and modify the approach presented by Herrmannova et al. [28] by combining features derived from the different aspects of documents and classify solely on features which are known as soon as a paper is published. In doing so, we introduce a prediction task which differs from the former classification task.

The remainder of this paper is organised as follows: Section 6.2 gives an overview of the established conceptual background of semantometrics and related research. In Section 6.3, the SUSdblp dataset is presented. The succeeding Section 6.4 describes the used methodology in detail: utilised document vector representations, distance measures and classification algorithms to apply Herrmannova et al.’s approach [28] to the new dataset in a ternary class setting as well as the extension of the methods are introduced. A detailed evaluation of our methodology is given in Section 6.5 which is followed by the evaluation of our dataset in Section 6.6. In Section 6.7 we compare classification based on semantometrics to alternative methods.

6.2 Semantometrics and Related Work

Before the methodology is described in detail, related research is covered to integrate this work in a broader context. We first introduce the concept of semantometrics and then present adjacent areas of research.

6.2.1 Semantometrics

Feature engineering on data through mathematical descriptors is common in medical image analysis [26, 38]. For publication networks, it was initially introduced as semantometrics by Knoth and Herrmannova [36] to assess research contribution.

Herrmannova et al.’s approach [28] which uses the principles described by Knoth and Herrmannova [36] is the foundation of this work. They were the first to utilise citation networks for the classification of a publication P as seminal or survey paper. A citation network centres around P and connects it to papers referenced by P (X) and papers citing P (Y). Semantic distances

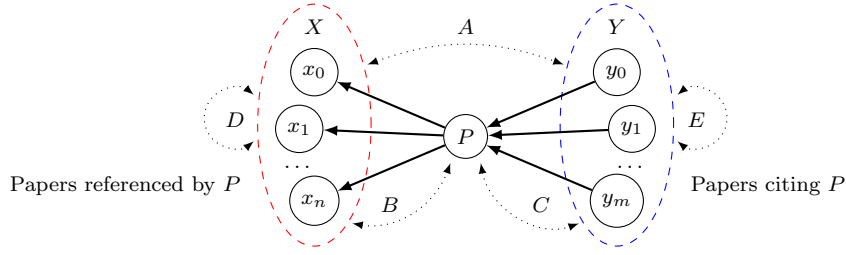


Figure 6.1: Neighbourhood of publication P . Nodes symbolise publications, straight edges between papers represent citations. $X = \{x_0, \dots, x_n\}$ are papers referenced by P , $Y = \{y_0, \dots, y_m\}$ are papers citing P . Dotted edges symbolise observed relationships between publications. Group A contains distances between pairs of referenced (X) and citing papers (Y). Group B contains distances between referenced papers (X) and P . Group C contains distances from P and citing papers (Y). Group D contains distances between pairs of referenced papers (X). Group E contains distances between pairs of citing papers (Y).

that describe the relationships amongst publications were measured: the distances between titles and abstracts of X and Y are contained in group A , distances between a publication and its referenced papers X are included in group B , and group C is composed of distances between P and its citing papers Y . The semantic distances between entries of X can be found in group D , symmetrically, distances between citing publications Y are stored in E . Figure 6.1 visualises the different groups of relations in a citation network for a given publication P . From these five groups, twelve features each were extracted such as min, max and mean distance between papers. Herrmannova et al. show that seminal papers are associated with larger semantic distances between papers from X and Y than surveys, while in turn surveys are associated with larger distances between referenced publications compared to seminal ones [28].

To enable the computation of distances between publications, P and the papers contained in X and Y need to be represented by vectors. Feature *sumB* would thus describe the summed up distance between each vector representing a reference of a paper (all papers contained in X) and the vector of paper P itself:

$$sumB = \sum_{i=0}^n dist(x_i, P), x_i \in X$$

Here, $dist(a, b)$ measures the distance between two papers a and b . Another feature *minE* would describe the smallest distance from any pairs of vectors

representing papers Y by which publication P is cited:

$$\min E = \min_{\forall y_i, y_j \in Y, i \neq j} \text{dist}(y_i, y_j)$$

The applied distance measure as well as the utilised vector representation of papers are parameters.

Herrmannova et al. [28] utilised cosine distance on tf-idf vectors of publications. A classification accuracy of .6897 was achieved in the binary classification task by using Naïve Bayes classifier on single features derived from papers in their citation network based on the TrueImpactDataset [29]. The dataset was created from a user study and contains publications from multiple distinctly labelled fields. Other previous work comes from Kreutz et al. [37]. They applied cosine distance as well as Jaccard distance on tf-idf vectors and Doc2Vec representations of publications in their citation network to derive the features for classification. The dataset used in this reevaluation of Herrmannova et al.’s approach is based in the narrower area of computer science [2]. An accuracy of up to .7432 for single features and an accuracy of .8015 using multiple features for classifying publications were achieved in the binary classification task.

Prior research tackling the task of quality estimation of papers using semantometrics solely focuses on the two classes seminal and survey but does not observe the group of papers which are neither groundbreaking nor reviews. In contrast, the task we describe includes this third group of publications (uninfluential) for the findings to be applicable in a real-world scenario where no predefined categorisation separates the different types of publications from each other.

6.2.2 Related Work

Relevant areas for our work besides semantometrics are scientific influence assessment, identification of important referenced papers, general citation behaviour analysis as well as document vector representation methods.

Scientific Influence Assessment

The fields of influence assessment of scientific papers and influence estimation of authors or author groups can be seen as alternatives to using semantometrics as an indicator for publication quality.

Several papers can be found in the field of *influence estimation of publications*. Gerrish and Blei [25] observe topical developments in corpora and topical compositions of documents to identify influential publications, this

approach is correlated with citation counts. Prediction of high-impact papers can also be done by observing similarities in full texts of citing papers. In this case, citations of topical similar papers are an indicator of wide influence [59]. The prediction of citation counts is also rooted in this domain. This influence of a paper can be accessed by using text-similarity of extracted popular terms of publications. Context-aware citation analysis on full texts and leading edge impact assessment [49] can be utilised for estimating impact of publications. Patton et al. [48] propose an audience based measure which leverages citation counts of publications and altmetrics to describe the influence of papers on the scientific community and the general public.

There are multiple approaches for the *assessment of influence of authors or author groups* and thus indirectly the influence of publications they write. The most popular one might be the h -index for estimation of author impact based on citation counts of authored publications [30]. It is defined as the highest value h such that an author has written at least h publications, each of which has been cited at least h times. There are numerous works extending, modifying, complementing or improving the h -index [8, 13, 33, 58]. The g -index is an improvement which measures global citation performance of sets of articles based on citation counts [22]. Here, publications of an author are ordered decreasingly by the received number of citations, the g -index is the largest value such that the added citation count of the top g papers is at least g^2 . An example for a complement of the h -index is the index h_α which quantifies an author's scientific leadership [31]. The papers contributing to the h -index of an author are observed, the number of times the author is the coauthor with the highest h -index of a publication in this set of papers is his h_{α} -index. Citation-based methods come with a number of drawbacks: Citations are highly influenced by the data source they are extracted from [45, 66]. Self-citations can boost scores relying solely on numbers of citations by a great margin. As their influence and meaning depends on the field, self-citations need to be handled case-by-case [33, 62, 63]. Citation practices in general are highly dependent on the different areas [17, 33, 64, 66, 68]. Another argument for caution when working with citation-based methods are diverging and unclear reasons for citing such as paying homage to pioneers, corrections of previous work or providing background reading material [24]. In cases of non-existence of citations occurring with new papers [74] or a high amount of uncited sources [24, 42, 49], citation-based methods cannot reliably perform influence estimations.

Another measure which can be utilised for influence assessment not relying on citations is the so-called research endogamy which observes fluctuation or stability of members in research teams [46]. It can be applied on different fine-grained or broad communities and venues [69] to estimate the quality

and therefore influence of papers. Research endogamy can also be combined with semantic information and citation counts [27] to classify the quality of research groups. Rocha and Moro [56] analyse research contribution of individual authors on established links between communities as a measure of influence and productivity of authors.

All of these methods inherently estimate the quality of research in retrospect, with semantometrics one could overcome this issue as evaluated in Section 6.5.5.

Identification of Important Referenced Papers

Observing content of citing papers enables identification of *important references* of publications: Hou et al. [32] classify references of papers as closely related or less related based on the number of times they are referenced in a certain paper, additionally they consider the number of references, the referenced paper and the current paper share. While Valenzuela et al. [73] label references as incidental or influential based on the section of a paper they occur in, Zhu et al. [75] use numerous features such as counts of occurrences of references, similarity of abstracts or context and position of occurrences of references to determine if a reference is influential or not. Pride and Knoth [54] state that the number of in-text occurrences of particular references as well as abstract similarity are the most descriptive features in the identification of influential references.

These approaches are directly linked to semantometrics as they all underline the varying importance of different references for publications and even conclude that the majority of referenced papers is not influential at all. Semantometrics does not consider these differing impacts but should do so as described in future work in Section 6.8.

Citation Behaviour Analysis

Citation behaviour analysis tries to explain the distributions of citations and references and properties associated with papers based on these counts. Price [19] defines publications which have at least 25 references as review-type papers and publications which received at least four citations in a single year as classics. He observed citations and references in the context of time and described a span of ten years after publication as the major period in which a paper is cited. Classics as defined by Price are roughly equivalent to seminal papers. These definitions are used in Section 6.3.6 to give quantified insight in our dataset.

The *citation half-life* describes the rate of obsolescence of a corpus of

research papers [15]. Recently, the half-life of corpora increasing has been observed [18, 43].

Citation life cycle analysis can be seen as a subtype of general citation behaviour analysis and as an alternative of the half-life referring to single publications. Differences in citedness of scientific papers can be perceived in relation to their age [64]. A citation life cycle is the period of time in which a publication is cited. Different patterns can be found in these cycles by which papers can be clustered. Avramescu [6] describes five citation frequency curves, of which a continuous steady low number of new publications as well as two curves with initially high amount of citations followed by a decline over the succeeding years appear the most in his dataset. While Aversa [5] found the two clusters **delayed rise - slow decline** and **early rise - rapid decline**, Cano and Lind [16] observed the patterns **Type A** and **Type B**. Papers of **Type A** accumulated citations fast in the first few years after the publication but the citation frequency declined after this incipient peak. Publications of **Type B** had a moderate initial amount of citations in the first six years but afterwards, the number of new citations per year was steady. Later, Aksanes [3] found several patterns for highly cited papers whereas **early rise - rapid decline** and **medium rise - slow decline** make up the highest share of his analysed papers.

As semantometrics utilises papers citing publications, awareness of citation life cycles is beneficial for constructing datasets as seen in Section 6.3.2. Incomprehensive representation of these cycles in a dataset could lead to biased results of algorithms working on it which was looked at in Section 6.6.2. Citation behaviour analysis can be used for the interpretation of evaluation results as was done in Section 6.7.1.

Document Vector Representations

In order to enable the conduction of computations on documents, these documents have to be represented as vectors. There are several approaches of transforming documents to vectors, some rely on semantic information while others regard the topical composition of texts.

Semantic information of documents can be accessed by algorithms describing input data as vectors abstracting from words to meanings behind terms. The resulting vectors are dependent on the context a word occurs in. Relations between words are learned by observing the surrounding tokens in a document. Doc2Vec [40] embeds all words that are presented in the training documents as vectors in a distributed space of fixed dimensionality. BERT [20] also learns language representations from text data but produces varying vectors for the same word in different contexts. Amongst

others, these two algorithms can be used to represent textual data as numeric features in a vector of certain length with which NLP tasks can be tackled.

Topic modelling tries to describe the topical composition of documents with a probabilistic model. A number of topics is fixed, then the topic proportions of every document in the collection are calculated as well as the word probabilities for every topic [10]. LDA is a widespread, basic topic model [10]. More complex topic models which incorporate authors' research interests [57], temporal aspects in topic developments [9] or citation information used for circulation of topics [21] tend to provide better results. Usage of these topic models requires more or other data than usage of LDA and is more computationally expensive.

Doc2Vec, BERT and LDA are going to be used as document vector representations for semantometrics as described in Section 6.4. More sophisticated topic models can be utilised as described in Section 6.8.

6.3 SUSdblp Dataset

As we strive to tackle the task of determining whether a paper is seminal, survey or uninfluential while investigating its citation network, we require a suitable dataset. Current datasets [29, 37] are only designed for binary classification tasks which leaves us with the need of constructing a new one, that is the SeminalUninfluentialSurveyDBLP (SUSdblp) dataset.

In this chapter, first we introduce the new SUSdblp dataset and describe the contained data. Afterwards we broach the issues of numbers of references and citations as well as years of publication for the three classes. Hereinafter, sub-fields contained in the dataset are observed before we describe and discuss the assumptions used in the construction of the dataset and also shed light on alternative methods for the generation of such a dataset.

6.3.1 Introduction

The SUSdblp dataset contains 1,980 publications and is an extension of the SeminalSurveyDBLP dataset [2]¹. One third of the publications are seminal (referred to as papers from class c_0), one third of the papers are surveys (referred to as papers from class c_1) and another third of the documents are uninfluential (referred to as papers from class c_2). All works are from the area of computer science and adjacent fields as they are contained in dblp [41]. For seminal publications, entries published in conferences attributed as A*

¹The SUSdblp dataset can be found at <https://zenodo.org/record/3693939/#.Xl0cF0oxlEa>

at the CORE Conference Ranking [1] CORE2018 such as *SIGIR*, *JCDL* or *SIGCOMM* were collected as publications often cited (and thus important) tend to appear in high-impact venues [3]. We assume papers published in a seminal venue as attributed by the CORE rank are seminal themselves, or they would not have been accepted for such a venue, even if they have not yet accumulated large amounts of citations. This might be a strong assumption, as not every paper from an A* conference is seminal and seminal papers can also appear in other venues, but is a simplified approximation of truly seminal papers. Surveys were extracted from *ACM Computing Surveys*, *Synthesis Digital Library of Engineering and Computer Science* and *IEEE Communications Surveys and Tutorials*. These venues are specialised in solely publishing reviews. Every paper of class seminal and survey has at least ten citations and references. Uninfluential papers are gathered from a number of venues attributed as C at the CORE Conference Ranking [1] CORE2018. They have an arbitrary number of references which in our case surpasses five but their number of citations lies between five and ten.

6.3.2 Contained Data

For each of the papers, the citing and referenced papers were collected. Citation information and abstracts from the AMiner dataset [71] were joined with dblp data to make sure they were also from computer science or adjacent domains. The join was based on matching DOIs of dblp papers with ones from AMiner or paper title and author name matches where DOIs were not present. Full texts are not included in the AMiner dataset. Citing and referenced papers not contained in dblp were omitted so the number of links for the papers might not necessarily represent the number of linkages which a paper received in the real world. For every paper, its year of release is also enclosed. The newest publications (as P and Y) contained in the dataset are from 2017 so the citation life cycle of several publications might not be completed yet. Considered publications for P , X and Y needed to have a length of at least ten terms in their combined title and abstract (in some cases the abstract was not present). The average length of the combination of title and abstract is 172.25 terms for seminal publications, 173.03 terms for surveys and 149.57 terms for uninfluential papers².

For all textual content, punctuation marks were omitted and lower case was used. A stemmed (S) and an unstemmed (U) version of the dataset are provided, the stemmed version contains 82,916 unique terms in the textual

²Influence of the different abstract lengths on classification accuracy is evaluated in the appendix 6.F.

	# papers in class	# references (X)			# citations (Y)		
		#	max	avg.	#	max	avg.
seminal	660	20,858	154	31.6	50,397	1370	76.4
survey	660	29,366	186	44.5	51,082	1365	77.4
uninfluential	660	6,629	33	10	4,263	10	6.5

Table 6.1: Numeric description of the SUSdblp dataset.

components while the unstemmed version holds 113,730 distinct words. The Porter stemmer [53] was used to create the stemmed version of the dataset.

For all papers in X and Y , the number of citations the publication received from publications from the area of computer science is also contained. Additionally, for these papers a field and time normalised citation count is included in the dataset to provide the possibility of assessing overall importance of these papers.

6.3.3 Number of References and Citations

The SUSdblp dataset is engineered to provide similar numbers of citations and references for publications of classes seminal and survey. Including this characteristic allows for methods working on this dataset to focus on hard cases. In general, seminal papers are cited numerous times while the average survey is not. Surveys typically reference a multitude of papers. Reproducing this scenario would lead to a majority of easy decisions when deciding on the class of a publication and thus divert from more challenging cases such as highly cited surveys, seminal papers referencing a multitude of publications or seminal papers which have not yet gained lots of citations. Such fringe cases would not occur often and presumably would therefore be neglected by most algorithms. Instead we decided to focus on such instances.

The total number of unique publications contained in the dataset is 129,443. This number includes all publications P as well as their referenced papers X and citing papers Y . Table 6.1 shows statistics regarding the number of citations for each type of paper. Each of the seminal and survey publications has at least ten citations and references. As the increased amount of references is assumed to be a feature of survey papers compared to seminal publications, the average and total number of references is higher and thus our dataset is unbalanced in this aspect. All papers contained in the set of uninfluential publications have between five and ten citations, the number of references was not restricted for them. Figure 6.2 shows the distribution of reference and citation cardinalities for all papers of groups seminal,

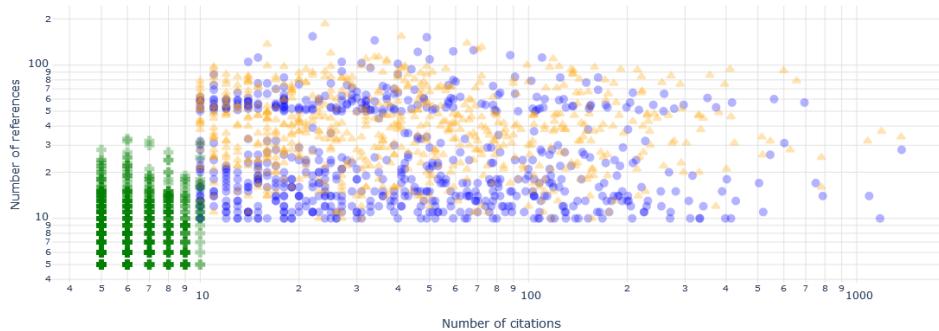


Figure 6.2: Distribution of number of references and citations for seminal (blue circles), survey (orange triangles) and uninfluential (green crosses) publications.

survey and uninfluential from the dataset. Numbers of citations are distributed rather homogeneously between the classes seminal and survey, but for references, differences in the distributions can be observed. While there are fewer publications with few references for surveys, a gap in the number of references from 40 to 50 can be seen for seminal papers. Distributions of numbers of references and citations for class uninfluential highly differ from the other two classes.

6.3.4 Publication Years

Having equal publication years for papers of the three classes seminal, survey and uninfluential as well as their references and cited publications was no priority in the construction of the dataset, so there are several differences in these features. Considering the primary focus on comparable numbers of references and citations for publications from classes seminal and survey, pursuing comparable distributions of years would have restricted the pool of publications which could be incorporated into the dataset and therefore also the number of contained papers by a considerable margin.

In Figure 6.3 the distribution of publication years of seminal, survey and uninfluential papers is depicted. While seminal papers are more common from 2002 to 2009, for the period before 1993 and from 2010 to 2014, the dataset contains more surveys. Between 1995 and 2001 as well as around 2010, the number of publications from these two classes is comparable. The number of uninfluential papers resembles the number of seminal papers be-

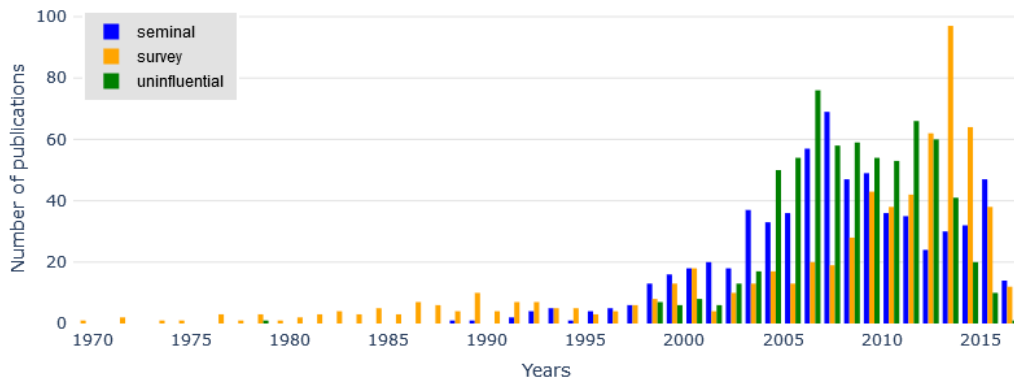


Figure 6.3: Histograms of number of seminal (blue), survey (orange) and uninfluential (green) publications over the years.

	seminal	survey	uninfluential
avg. year P	2007.13	2006.45	2008.01
avg. year X	2003.69	2002.24	2003.23
avg. year Y	2011.66	2009.55	2011.5
avg. distance years $P&X$	4.97	6.2	5.19
avg. distance years $P&Y$	4.55	4.89	3.52

Table 6.2: Average publication years for papers from P , X and Y as well as average distances in years between publications from P and X as well as P and Y for the three classes.

tween 2006 and 2008.

Table 6.2 provides the average publication years of papers from P , X and Y for the three classes as well as the average distance between the publications P and their respective referenced and citing papers. We assume that the larger distance between surveys and their referenced papers compared to the other classes might stem from the longer time a publication takes until it is published in a journal compared to the length of periods between submission and publication of papers to conferences. All papers contained in the class of surveys appeared in journals which underlines the validity of this hypothesis.

Figure 6.4 shows the number of referenced and citing papers associated with the three classes of publications over the years. As the overall number of references is considerably higher for surveys, the higher numbers of references per year for surveys were expected. The number of references and citations is notably lower for uninfluential papers. It is observable that reference distributions differ greatly. Surveys reference more publications from

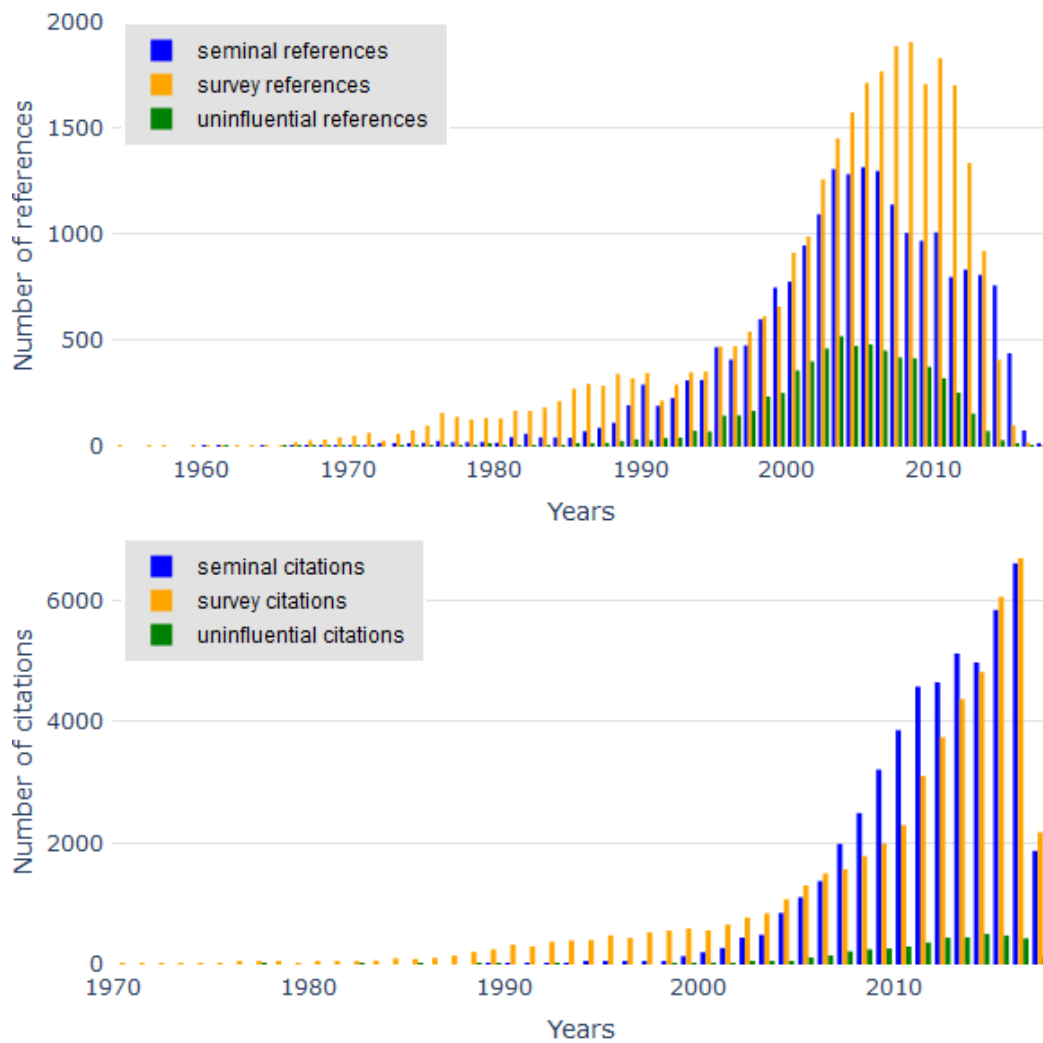


Figure 6.4: Histograms over number of referenced (upper) and citing (lower) papers for seminal (blue), survey (orange) and uninformative (green) publications over the years.

years preceding 1990 and after 2003 compared to seminal papers. Between 1990 and 2003, the numbers of references from these two classes are comparable. The number of citing papers is almost identical for papers from the two categories, but the distribution over the years differs slightly. Surveys were cited more often before 2003. From 2003 to 2008 as well as from 2013 onwards the numbers of citing papers are comparable for the two classes. Between 2009 and 2012, the number of publications citing seminal papers has increased.

To test whether publication years as well as distances in years in a citation network are equally distributed, Kruskal-Wallis H-tests for independent samples with $p = 0.05$ were conducted as requirements for standard statistical analysis were not met. The publication years of papers P , those from X and Y of works from the three classes are significantly different from each other. Publication years of P of seminal and survey publications are the only ones which are not significantly different from each other. Distances between papers in P and X , P and Y as well as X and Y are also significantly different for the three classes c_0 , c_1 and c_2 . The distances between seminal and survey papers are not significantly different for the three observed cases. This underlines the validity of the dataset for tackling the previously described hard cases based on differences in publication years even though the even distribution of years was no prerequisite in the construction of the dataset. Distinguishing between c_0 and c_1 based on distances between publication years of a paper P and its referenced papers, P and its citing papers as well as distances between publication years of referenced and citing papers is non-trivial.

6.3.5 Sub-fields and Topic Distributions in Publications

The SUSdblp dataset gathers publications from venues from several sub-fields of computer science: real time systems (conferences RTSS, ISORC), HCI (conferences CHI, ICCV, MMM, COMSWARE, ICCHP), data mining (conferences ICDM, ACII) and software engineering (conferences ICSE, ICGSE, ICCBSS) are the major fields shared between seminal and uninfluential publications. Venues of papers in class survey do not classically target a specific area but encompass all domains of computer science. This observation leads to the conclusion that the dataset at hand contains multiple areas of computer science, including their potentially differing writing and citing habits.

Figure 6.5 shows the percentages of the top five topics over all publications

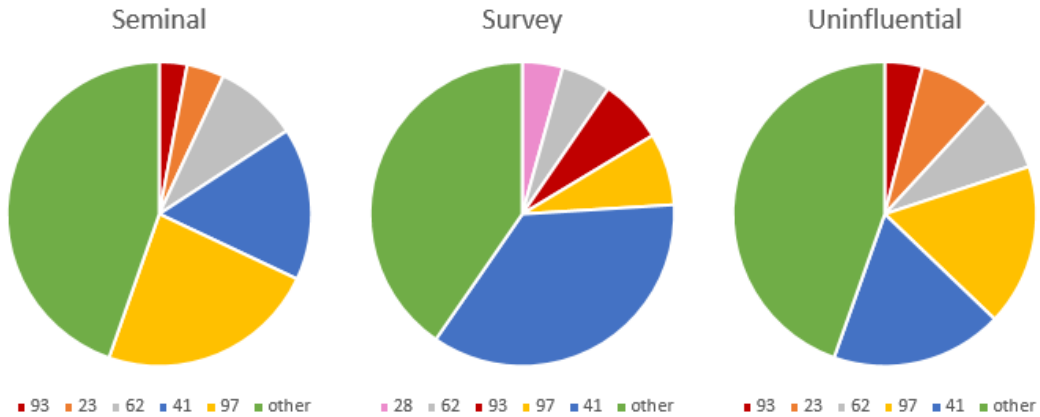


Figure 6.5: Percentages of top five topics (and all others) for publications P from classes seminal, survey and uninfluential.

P in the three different classes. All other topics are contained in share *other*. Topic distributions were calculated by usage of LDA [10] trained on dblp data [41] combined with abstracts from AMiner [71] with $k = 100$. Percentages of the different topics were added for all papers from a certain group to determine the most prevalent topics. Topics 41, 97, 62 and 93 can be found amongst all classes in the most popular topics. Although their share varies between classes, the sole existence of the topics in the top five topics hints at a topically relatively unbiased dataset.

6.3.6 Description and Discussion

The SUSdblp dataset is specifically constructed to tackle tasks by usage of semantometrics as there is no dataset suitable for its evaluation in a ternary setting. This premise prevented the inclusion of publications in class uninfluential which have no citations, i.e. really do not influence any following works. A vast percentage of scientific articles is uncited [64] but the papers contained in our dataset do not intend to represent citation distributions found in full corpora. If we included mostly uncited publications in the dataset for c_2 , methods relying on whole citation networks of publications could degenerate as much less features could be extracted from these papers. This aspect is evaluated in Section 6.5.4.

None of the publications of class survey are taken from conferences while all papers P in classes seminal and uninfluential are extracted from conferences. This might lead to longer periods between papers from groups X and P for surveys. For survey papers, it seems far more likely for them to appear in journals than in conferences. If we tried to include publications from con-

ferences in class c_1 , they would have to be collected by hand as automatic methods could only rely on keyword search and manual verification of the assigned label.

The uninfluential papers were taken from conferences which were ranked as tier C in the CORE Ranking, the assumption of being not important stems from their venue of publication and their number of citations but might be completely false for some papers as citations might not be mapped correctly or could not be mapped for these publications at all. Publications appearing in a lower tier venue might also be seminal as citation counts are unaffected by prestige of venues [3, 65]. Another aspect to consider for papers of class uninfluential might be their citation life cycle, maybe they are only on the beginning of theirs, or they are genial work which only receive citations after an initial phase of absent recognition [6]. Papers not cited in a certain year or period might be cited in a following year [19]. Influence of the different years for the classification accuracy in terms of disruption of citation life cycles is evaluated in Section 6.6.2.

Of the 660 seminal papers 24 have received a best paper award. Only incorporating publications which received an award would dramatically decrease the size of the dataset as a similar distribution of referenced and citing works of classes seminal and survey was prerequisite in its construction.

613 of the seminal papers were cited four or more times in a single year, making them classics [19]. Of the surveys, 504 reference 25 or more papers which is said to be an attribute of this type of publication [19]. Of the uninfluential papers, eleven have at least 25 references and 85 are cited at least four times in one year.

Another way to construct a dataset fit for tackling the same task would be via an email questionnaire similar to the generation of the TrueImpact-Dataset [29]. There are several challenges associated with such a procedure. Conducting a survey is entirely dependent on the participants and their willingness as well as ability to judge publication quality. In this context, problems arise in sampling of subjects for the survey leading to bias as the answers might not be representative [7]. The dataset resulting from responses would most probably be unbalanced for the three classes and its size would entirely depend on the number of responses. Answers would very likely have to be omitted as the identification of referenced papers would be impossible or they could be from completely different fields. For all answers which could be mapped to real publications, metadata as well as references and citations would have to be extracted from suitable data sources. For comparison, the TrueImpactDataset contains 166 seminal and 148 survey papers which were gathered through 184 responses. The response rate of the study was 13%. The dataset holds papers from 31 distinct scientific fields [29]. Acquiring a

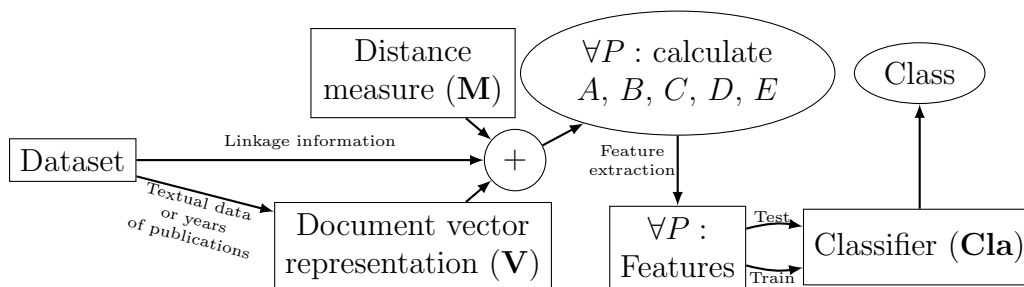


Figure 6.6: Simplified graphical depiction of methodology.

number of publications comparable to the one of the presented dataset while focusing on a single domain in this manner thus would take vast efforts for study conductors. Another factor worth considering is the caused cost of invited subjects even if they did not complete the questionnaire [7].

Furthermore, a dataset could be constructed by usage of only papers which received a test of time award for those in class seminal. The other classes could be constructed like they are now. As the number of distinguished publications per year is considerably low and these awards have not been handed out for a long time yet, it would be difficult to construct a sufficiently big dataset out of them. Another drawback using this method would pose the collection of seminal papers. Their titles would have to be scraped from web pages of conferences and then the associated metadata concerning abstracts, citations and references would have to be retrieved from external data sources. Additionally, such a dataset would not hold any recent publications in class seminal.

6.4 Methodology

Herrmannova et al. [28] proposed the usage of citation networks to extract patterns from differences between texts which can be represented by distance features for making assumptions whether publications are seminal or survey. First, document vector representations (V) of P , its referenced papers X and its citing papers Y need to be generated from a suitable dataset. In a next step, a distance measure (M) is chosen with which distances between publications for every group A to E can be calculated. From these five sets of distances, twelve features are then computed for each set: Minimum, maximum, range, mean, sum of distances in a group, standard deviation, variance, 25th percentile, 50th percentile, 75th percentile, skewness, and kurtosis. Those $12 \cdot 5 = 60$ features are named by concatenating the feature with the group it originates from, e.g. *minA* or *rangeE*. Classifiers are trained on

different sets of data which either describe a publication by one feature or multiple features. We [37] proved the classification on multiple features as useful. For test data, the classification algorithms are then able to determine the class a publication P is most likely to be associated with. Figure 6.6 displays the simplified course of action from dataset to the accuracy of a classification as described above. In this pipeline, there are several interchangeable parts where different options are available, which are indicated by rectangular boxes in the Figure.

As a considerable difference to previous work [28, 37], our dataset contains three classes, we modify the binary classification problem to become a ternary classification task. Class c_0 describes seminal publications, c_1 indicates the class containing surveys and c_2 is defined as group of comparably uninfluential papers.

In this work, we observe different aspects of citation networks of publications: words, semantics, topical compositions and publication years. Earlier work only focused on document vectors derived from words [28] or showed the helpfulness of simple semantic features [37]. We extend the approach by not only classifying on features derived from one document vector representation but observe the possibility of combining features describing multiple aspects of the dataset to improve overall classification accuracy.

6.4.1 Document Vector Representations

For document vector representation (V) methods working on words, semantics, topics and distances between publication years in a citation network were utilised. As a method working on words, tf-idf [60] is applied. Semantics of publications are depicted by usage of Doc2Vec [40] (D2V) as well as BERT [20]. Topical information of publications is constructed by using LDA [10]. We refrained from using more complex topic models since we wanted to focus on experimentation of the general usefulness of utilisation of topics in the context of semantometrics. Publication years of papers are extracted from the underlying citation network.

6.4.2 Distance Measures

As distance measure (M), cosine distance (COS) is applied as described by Herrmannova et al. [28]. Additionally, Jaccard distance (JAC) is used as a second method as seen in previous work [37]. These two measures are applied on tf-idf, Doc2Vec and BERT vectors. On LDA document representations, Earth Mover’s Distance (EMD) is applied. We apply 1 –

standard inner product (IPD or inner product distance) on all word, semantic and topical document vector representations. On publication years of papers from the three classes, differences in years in the citation network (DIST) are calculated.

6.4.3 Classification Algorithms

The set of selected classification algorithms (Cla) includes logistic regression (LR), random forests (RF), Naïve Bayes (NB), support-vector machines (SVM), gradient boosting (GB), k-nearest neighbours (KNN) and stochastic gradient descent (SGD) as seen in previous work [37]. Herrmannova et al. [28] applied SVM, LR, NB and decision trees. We wanted to include those classifiers except for decision trees which we omitted as we incorporated random forests and gradient boosting which are ensembles over decision trees and thus are able to outperform them.

6.4.4 Implementation

Python 3.6 and classifiers from scikit-learn [50] are used in this implementation. For SVM, multi-class as one-vs.-one is calculated. For LR, GB and SGD, multi-class is calculated as one-vs.-all. Implementations for kurtosis, skewness and Wasserstein distance are used from scipy [34]. Gensim [76] is used for the Doc2Vec as well as the LDA implementation. For the generation of BERT document vector embeddings, the PyTorch [47] framework is used. For statistical analysis, SPSS 26 is used.

As dataset, the SUSdblp dataset is used in a stemmed and unstemmed version. On this dataset, we constructed the document vector representations: the tf-idf values are computed on the 129,443 unique publications in the stemmed or unstemmed SUSdblp dataset, abstracts of all citing and referenced papers were included in the calculation of term frequencies. Vectors computed on the stemmed dataset consist of 82,916 dimensions while vectors representing the unstemmed dataset consist of 113,730 dimensions.

Weights for Doc2Vec are generated by usage of the English Wikipedia corpus from 20th January 2019. We refrained from using Doc2Vec on a stemmed corpus as this preprocessing is no prerequisite for achieving good results [40]. The Doc2Vec model was trained so that resulting vectors consist of 300 dimensions. This size was proposed by Lau and Baldwin [39] for general-purpose applications.

A pretrained uncased BERT base model was also used to create document embeddings. It was also only used on unstemmed publications. The BERT implementation used is only able to process input vectors of at most 512

tokens [20], documents were cut at places where punctuation marks would have been or after half of the tokens if sentences were still too long. A sliding window was used to always input two consecutive sentences to maintain as much context as possible. The model consists of overall twelve hidden layers each having 768 features. The last four layers from these twelve layers were concatenated for each token and averaged over all tokens to receive vectors of length 4 layers * 768 features = 3072 dimensions for each publication.

We ran LDA on unstemmed and stemmed titles concatenated with abstracts from all publications contained in the dblp dataset [41]. Abstracts were extracted from AMiner [71]. Following this procedure, we ensured the computed topics were from the area of computer science. The number of topics was set to 100 in both cases, resulting in the same number of dimensions for these document vector representations.

In case of years, the publication years of all papers in the citation network were extracted, resulting in one-dimensional vectors for each of the publications.

Removal of high-frequency words from titles and abstracts of publications before construction of document vector representations was out of scope for this work but we assume a possible increase in classification accuracy from conducting this pre-processing step as related tasks typically benefit from doing so [61].

The implementation of our approach including usage instructions can be found at GitHub under <https://github.com/dbis-trier-university/Semantometrics>.

6.5 Evaluation of the Approach

Our approach is evaluated by observance of different classification modalities. Classifications were conducted based on single features and all features. Additionally, classification on combinations of features derived from multiple aspects of the dataset is observed. A following experiment evaluates the performance of the approach when trying to classify truly uninfluential publications without citations. Reclining on this experiment, we predict the class of a paper based on semantometrics derived from information that is available as soon as a paper is published.

All accuracies (Acc), the 95% confidence interval (\pm) and F1 scores (F1) are rounded to four decimal places. Values have been calculated by usage of ten-fold cross-validation if not specified otherwise. In all of our experiments, there is no need for a development set as we do not perform hyperparameter tuning, data the model gets trained with is always different from the data it

is evaluated against.

If more than one classification algorithm achieved the highest accuracy, the algorithm with the highest F1 score is mentioned in a table. For all significance tests, we use a p-value of 0.05. Statistical analysis is conducted on the accuracies extracted from the ten folds of the cross-validation for each model. Normal distribution of values is evaluated by usage of Kolmogorov-Smirnov [44] and Shapiro-Wilk [67] test. Homogeneity of variances is tested with Levene’s test [14]. If an independent two-sample t-test is used, data is normally distributed in the two groups and variances are homogeneous. If a Welch t-test is conducted, data is normally distributed in the two groups but variances are not homogeneous. If a Kruskal-Wallis H-test is used, data is not normally distributed in the different groups or variances are not homogeneous. If ANOVA is used, data in the (more than two) different groups is normally distributed and variances are homogeneous.

The dataset used in this evaluation is the SUSdblp dataset.

6.5.1 Single Features

The first experiments consider the whole citation network of publications for a classification based on a single feature. Each of the 60 features derived from different aspects of a publication is used on its own as input for the classification algorithms. This evaluation delivers a baseline and enables us to relate the results to previous ones [28, 37] in a ternary setting.

At first, words, semantics, topics and publication years of seminal, survey and uninfluential papers are observed. Words of publications stem from their titles and concatenated abstracts. The tf-idf vectors of stemmed and unstemmed publications are combined with COS, JAC and IPD as distance measures. Semantics of papers were derived by applying Doc2Vec and BERT on concatenated titles and abstracts. Distances between resulting vectors of papers in the different groups were computed by using the same distance measures as before. For topics of publications, LDA was applied on stemmed and unstemmed documents. Distances between topical distributions then were calculated by usage of EDM and IPD. Lastly, distances between publication years were computed.

For each of these combinations of document vector representation and distance measure, the seven classifiers were applied which each of the 60 features as input. The best results per document vector representation can be found in Table 6.3³. No significant differences for the seven methods were

³In-depth results on all combinations of document vector representation as well as distance measure can be found in the appendix 6.A.

V	M	Cla	F	Acc	F1	Acc _{c₀}	Acc _{c₁}	Acc _{c₂}
tf-idf U	COS	LR	sumA	.6879 (\pm .0414)	.6818	.6318	.447	.9848
tf-idf S	COS	LR	sumA	.6884 (\pm .0421)	.6824	.6333	.447	.9848
D2V U	COS	LR	sumA	.6904 (\pm .0438)	.6845	.6394	.45	.9818
BERT U	COS	LR	sumA	.6889 (\pm .0472)	.6833	.6364	.453	.9773
LDA U	IPD	GB	sumD	.6939 (\pm .0723)	.6989	.2727	.5076	.9182
LDA S	IPD	LR	sumA	.6874 (\pm .0430)	.6815	.6303	.4485	.9833
years	DIST	GB	sumA	.6879 (\pm .0372)	.6856	.0848	.9121	.9742

Table 6.3: Best classifiers dependent on distance measures for all document vector representations. For the classification algorithm achieving the highest accuracy per combination, the single feature (F), the corresponding F1 score as well as accuracies for the three classes c_0 , c_1 and c_2 are displayed.

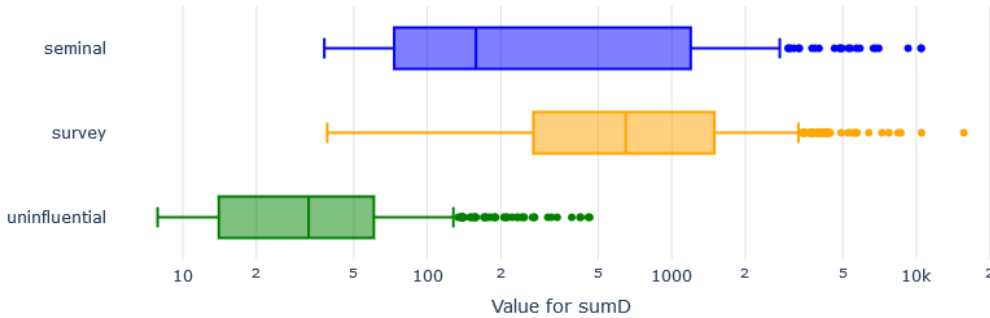


Figure 6.7: Box plot of value distribution of feature $sumD$ derived from differences computed with inner products of unstemmed LDA document representations for seminal, survey and uninfluential publications in their citation networks.

found when applying a Kruskal-Wallis H-test.

The best results can be achieved by usage of LDA on unstemmed documents combined with inner product distances and gradient boosting as classifier. The fact that feature $sumD$ is the most descriptive feature in this setting is very promising, as this feature describes the sum of distances between references. This information is already available at the time when a paper is first published. Figure 6.7 shows the distribution of values of feature $sumD$ for the three different classes.

In general, a highly descriptive feature regardless of document vector representation and distance measure is $sumA$ which describes the sum of distances between referenced and citing papers. Figure 6.8 gives an overview of the distribution of $sumA$ for the three classes at hand for tf-idf on stemmed documents and cosine distance. The relatively low values for papers from

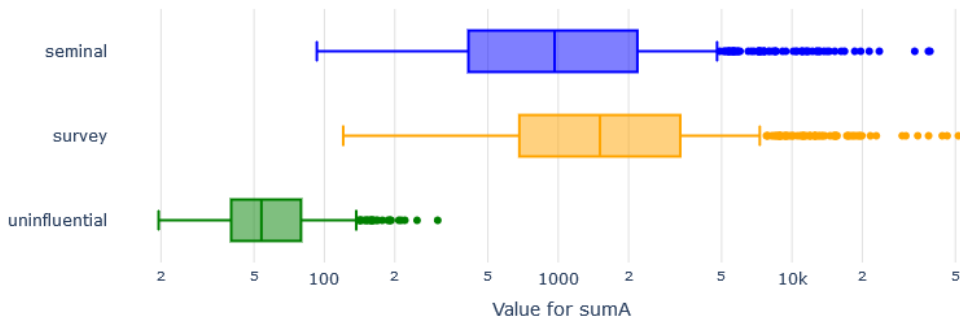


Figure 6.8: Box plot of value distribution of feature *sumA* derived from differences computed with cosine distance of stemmed tf-idf document representations for seminal, survey and uninfluential publications in their citation networks.

class uninfluential might be explained by their comparably low number of references. Surveys tend to have a higher value with this feature than seminal publications which could be explained by the inherently higher number of references per paper for reviews. The best classifier might group seminal and survey papers together, resulting in the low accuracy for c_0 . A binary classifier on features derived from distances between publications which only distinguishes seminal and survey publications might lead to better results for our core task.

Compared to Herrmannova et al. [28], we achieved an increased accuracy of .0042 which mainly only stems from the introduction of c_2 . Publications from the two other classes cannot be identified reliably using only a single feature. While features from group B , D and C were achieving good results for Herrmannova et al., they do not report on features from group A to be helpful. Kreuz et al. [37] were able to surpass our results by .0493. For them, features from group A were also not helpful in determining the class of a paper but they also found features from group D to be performing quite reliably. Contrasting earlier work, here we introduced a third class into the classification task, it is quite comprehensible for our single features to not be as descriptive as those used in a binary setting.

6.5.2 Multiple Features

For the following experiment, the complete citation network of every publication is used for the classification procedure: all 60 features are used for the classification. Features derived from all combinations of document vector representations and respective distance measures are used as input for

V	M	Cl	Acc	F1	Acc _{c₀}	Acc _{c₁}	Acc _{c₂}
tf-idf U	COS	GB	.8414 (\pm .0356)	.8415	.7894	.7515	.9833
tf-idf S	JAC	GB	.8439 (\pm .0371)	.8439	.7848	.7621	.9848
D2V U	COS	GB	.8525 (\pm .0309)	.8526	.7742	.8015	.9818
BERT U	IPD	GB	.8646 (\pm .0569)	.8647	.806	.8	.9879
LDA U	IPD	GB	.8601 (\pm .0367)	.8601	.8167	.7788	.9848
LDA S	IPD	GB	.8475 (\pm .0266)	.8477	.7803	.7773	.9848
years	DIST	RF	.8747 (\pm .0401)	.8743	.8439	.7985	.9818

Table 6.4: Best classifiers dependent on distance measures for all document vector representations. For the classification algorithm achieving the highest accuracy per combination, the corresponding F1 score as well as accuracies for the three classes c_0 , c_1 and c_2 are displayed.

the seven classifiers⁴. This evaluation enables us to relate our results to the binary classification task on seminal and survey publications performed by Kreutz et al. [37].

Table 6.4 shows the detailed accuracies for all combinations of distance measure and document vector representations⁵. Significant differences were found in the seven methods when tested with a Kruskal-Wallis H-test. The two tf-idf were significantly different from usage of years. For unstemmed tf-idf, the effect size was small ($r=.0074$), for stemmed tf-idf, effect size was medium ($r=.0103$).

When using all features derived from distances in publication years in a citation network, the best results (accuracy of .8747 with F1 score of .8743) can be achieved by usage of gradient boosting. The highest accuracy from the single feature experiments was surpassed by .1808. Accuracy values for the three classes are all relatively high, almost all unimportant papers can be reliably classified. Here, accuracy for seminal papers is higher than the accuracy for surveys. Equally distributed years of publications were no precondition in the generation of the dataset but here, instead of years, distances between years were observed. The differences between publication years already mentioned in Section 6.3.4 might be a characteristic of papers of the three classes. Nevertheless, the high descriptiveness of features derived from years might be unrepresentative of reality.

For the best performing combination, the five features with the highest

⁴An inferior second experiment using only the 33 features, which were found to be significant by Herrmannova et al. [28] can be found in the appendix 6.C.

⁵Results on all different combinations of document vector representation as well as distance measure can be found in the appendix 6.B.

influence on estimator performance are *sumA*, *sumE* with importance values of over .12 as well as *sumA*, *sumB* and *sumD* with importance values of around .006 for all ten instances of the random forest classifier in the cross-validation process.

Even though results for distances between years can be influenced by the construction premises of the dataset, applying gradient boosting on features derived from BERT embeddings and inner product distance leads to an accuracy of .8646 (F1 score .8647). Accuracies for the three classes are also considerably high. Here, no dataset topic bias should artificially boost the results based on distances between semantics of papers in their whole citation network. Papers stemming from different areas in computer science might have diverse citation practices [17, 33, 64, 66, 68] and therefore reference papers with possibly domain dependent topical compositions. This variability cannot be omitted in the automatic creation of a dataset but might hint at the approach’s suitability in terms of applicability on topical diverse citation networks.

Building upon results from Kreutz et al. [37], we were able to improve the accuracy by .0792 when using features derived from distances between publication years and achieved an increase of .0646 when utilising all features derived from BERT embeddings. Looking at the two classes c_0 and c_1 only also leads to higher overall accuracies compared to the best results from the binary classification task.

6.5.3 Combination

Experiments were conducted where all five document vector representations were utilised in concatenation. All possible combinations of stemmed and unstemmed vector representation as well as all combinations of distance measures were observed. This experiment explores the informative power of utilisation of numerous aspects of publications.

The 60 features derived from the first document vector representation with accompanying distance measure were concatenated with the following 60 features derived from document vector representations with fitting distance measure. This procedure resulted in the construction of vectors of length 300 features (5 document vector representations * 60 features) for each paper of the three classes.

An accuracy of .9247 (\pm .0446, F1 .9249, Acc_{c_0} .9015, Acc_{c_1} .8894, Acc_{c_2} .9833) was achieved for usage of features derived from inner product distances between stemmed tf-idf vectors, unstemmed LDA, Doc2Vec and BERT docu-

ment representations as well as distances between years⁶. Gradient boosting produced the highest accuracies. Results from the multi feature experiment were surpassed by .05, compared to the single feature baseline, improvements of .2308 were achieved in this experiment.

Comparison of the best performing single document representation (years) with usage of two, three, four or all five document vector representations in concatenation with ANOVA showed significant differences between the five groups. With Bonferroni correction [23] and Scheffé’s method, significant differences between utilisation of the five-vector representation approach and all other methods were found.

6.5.4 Truly Uninfluential Publications

In the SUSdblp dataset, only publications are contained that have at least five citations, as this is only an approximation of truly influential publications which did not yet accumulate any citations. In the next experiment, the capacity of the approach to recognise such papers is evaluated.

As truly influential publications, 112 publications from conferences which are not contained in the CORE Conference Ranking [1] CORE2018 were chosen at random from dblp. These publications have no citations but each of them has at least 5 references with concatenated titles and abstracts of length ≥ 10 tokens. Doc2Vec vector representations of the citation networks are constructed and cosine distance is applied to derive the 60 features. There are no values contained in groups *A*, *C* and *E* (which equals a value of 0 for all twelve features in these groups) if no citations exist for a publication.

When using all 60 features conjointly, the accuracy was 1 when using SVM. Due to this result we assume that the approach is robust to application on truly influential publications which did not yet accumulate any citations.

6.5.5 Information Available at Publication Time

When a paper is first published, only its referenced papers and the content of the publication are available. It has not yet gained any citations. In this experiment, we try to predict a class solely based on features derived from groups *B* and *D* to simulate this situation. Table 6.5 shows the best distance measure as well as accuracies and F1 scores resulting by usage of the best performing classifier for each document vector representation. Significant

⁶Results of inferior experiments using two, three or four different document vector representations in concatenation can be found in the appendix 6.D.

V	M	Cla	Acc	F1	Acc _{c₀}	Acc _{c₁}	Acc _{c₂}
tf-idf U	IPD	GB	.7131 (\pm .0499)	.7159	.6606	.7273	.7515
tf-idf S	JAC	GB	.703 (\pm .0558)	.7056	.6364	.7152	.7576
D2V U	JAC	GB	.6838 (\pm .0417)	.685	.5833	.7197	.7485
BERT U	IPD	GB	.8152 (\pm .0652)	.8155	.7364	.7288	.9803
LDA U	IPD	GB	.7192 (\pm .0625)	.7211	.6621	.7697	.7258
LDA S	IPD	GB	.7071 (\pm .0599)	.7096	.6409	.7379	.7424
years	DIST	GB	.7268 (\pm .0535)	.7281	.6773	.7318	.7712

Table 6.5: Classification accuracy and F1 scores for features of groups B and D derived from all different vector representations from the SUSdblp dataset for the best performing distance measure and classification algorithm.

differences were found for all seven methods when tested with a Kruskal-Wallis H-test. The BERT option differed significantly from the two tf-idf methods, Doc2Vec and stemmed LDA. The effect sizes were large for all of them (unstemmed tf-idf .7703, stemmed tf-idf .9673, Doc2Vec 1.2111, stemmed LDA .864).

The highest accuracy of .8152 (F1 score .8155) was achieved by using features derived from inner product distances of BERT embeddings.

This experiment underlines the usefulness of semantometrics for real information retrieval systems as a means of automatically estimating the quality of a publication. Instead of relying on the venue it appeared in or waiting for citations to accumulate in order to make assumptions on the importance of a paper, an independent prediction could directly assess the quality. This property of semantometrics is also especially helpful for preprints of papers published in arXiv where no information on a venue rank is available.

6.5.6 Discussion

The highest overall accuracy when using single features of .6939 (F1 score .6989) was achieved by usage of sumD extracted from inner product distances between unstemmed LDA document vector representations. This feature is already computable at the time a paper is first published, as it encodes the sum of differences between references of P . In general, for all other combinations of document vector embeddings and distance measures, sumA was the most descriptive feature resulting in accuracies around .69. Feature sumA contains indirect information on the number of citations and references of a publication as it describes the sum of distances between vector representations of all combinations of citing and referenced papers. The more papers

are linked to P , the more distances are computed and the higher the value of sumA becomes. Typically, uninfluential papers are cited few times, so this feature would have an overall low value. Both seminal papers as well as surveys tend to be cited often so their values should be higher than the ones for uninfluential publications. In general, surveys have more references than seminal papers, so more distances between vector representations of papers can be computed (and summed) in their citation network, leading to an even higher value for this feature.

Compared to the best single feature accuracy in the binary classification task of .7432 from our former work [37], our current accuracy is lower by .0528. Contrasting the prior work, we introduced the third class uninfluential into the classification problem so it is not surprising that we were unable to achieve such a high accuracy while only using single features.

Usage of all features resulted in the highest accuracy of .8747 (F1 score .8743) for distances between years in a citation network and application of random forests. Overall, the highest accuracies for all other combinations of document vector representations and distance measures lie between .84 and .86. Compared to the single feature variant, accuracies increased for all cases. In general, using all features resulted in +.1808 in accuracy compared to the best single feature variant.

The best accuracy from our former work [37] in a multi feature setting with binary classification was .8015, which we were able to surpass by .0732 by introducing the usage of more diverse features derived from citation networks. The newly introduced third class is not entirely responsible for the higher value, as the accuracies for c_0 and c_1 are also very high (.8439 and .7985). Due to this, one cannot argue the third class artificially boosted the accuracy as it might be too easy to distinguish uninfluential papers from seminal and survey ones.

Combining features derived from distances between all document vector representations leads to an accuracy of .9247 (F1 score .9249). The performance of the multi feature setting was increased by .05.

A Kruskal-Wallis H-test showed significant differences for the best performing model based on single features, all features and the combination of features derived all five document vector representations. The three models are all significantly different from each other.

Classification on truly uninfluential publications achieved a perfect accuracy.

When predicting classes of publications based on information which is already available as soon as they are published, an accuracy of up to .8152 was reached by using inner product distance on BERT embeddings of references and the publication. This finding is highly promising as this method could

easily be applied in real world scenarios in context of information retrieval systems.

Herrmannova et al.'s [28] best performing algorithm was Naïve Bayes, in our experiments usage of logistic regression (single feature setting), random forests (multi feature setting) as well as gradient boosting (single/multi feature setting, combination of feature sets) typically achieved the highest accuracies.

As the absence of label noise cannot be guaranteed, it is unclear if the upper bound for accuracies on this dataset truly is 1. With Herrmannova et al.'s approach [28] and our extensions to it, we were able to approximate this bound.

6.6 Evaluation of the Dataset

As we already used the SUSdblp dataset in the evaluation of our approach, we now observe the presented dataset to substantiate the reliability of our results.

The robustness of the dataset is accessed as well as differences in classification performances when using the dataset up to or from different years.

6.6.1 Robustness of Dataset

In a first experiment, the robustness of our dataset is evaluated. As the dataset is automatically constructed and based on some strong assumptions, results might be skewed by the generation process or the underlying biased dataset; citations and references are entirely dependent on the dataset they are extracted from [45, 66]. In the construction of our dataset, citing and referenced publications were omitted from inclusion into the dataset if the concatenated titles and abstract were less than ten tokens. This did happen several times when dblp entries could not be mapped to entries from Semantic Scholar and therefore no abstract was found. Other cases when citing and referenced papers were omitted are instances in which the respective publications were not contained in dblp because they were out of scope. Due to these factors, the robustness of the SUSdblp dataset with respect to slight variations in references and citations of publications is evaluated to estimate the overall reliability of our findings with regards to bias in the underlying data.

On unstemmed Doc2Vec document representations, cosine distance is applied. The document vector representation was chosen as the calculation of

it is relatively fast; the distance measure was chosen arbitrarily for the comparisons.

In a first experiment, one citation and one reference from each seminal, survey and uninfluential publication are omitted randomly in the calculation of distances for the five groups. In a second evaluation, five citations and references from each seminal and survey publication as well as two references and citations from all uninfluential publications were ignored when deriving features.

When classifying on single features, both omitting modes produce results comparable to the single feature baseline. Feature *sumA* was the most descriptive one in the best cases. No significant differences were observed when applying ANOVA.

Classification on all 60 features was significantly different for the three considered groups when analysed with ANOVA. Experiment one resulted in an accuracy of .852 (\pm .0341, F1 score .8522, Acc_{c_0} .7788, Acc_{c_1} .7955, Acc_{c_2} .9818) when using gradient boosting, which is -.0005 compared to usage of the unaltered dataset. For the second omitting mode, an accuracy of .8136 (\pm .0508, F1 score .8113, Acc_{c_0} .6985, Acc_{c_1} .7712, Acc_{c_2} .9712) was computed with GB, which is -.0389 in accuracy in comparison with the regular citation network.

Due to these slight changes in accuracies throughout the two omitting modes, robustness of the dataset is assumed. This leads to the conclusion of reliability for our findings in terms of data source bias in spite of omitting references and citations from our underlying data source.

6.6.2 Classification for Different Years

The SUSdblp dataset is biased in terms of publication years for papers of the different classes. To estimate the graveness of this bias on the used method which does not directly work on the years but instead uses distances between publication years, we restricted the SUSdblp dataset to only contain information on publications P , X and Y up to or from different years. This should simulate the effects of a possibly abrupt disruption of citation life cycle of publications on classification accuracy as has happened for papers published shortly before 2017 which was the last year from which works are included in the dataset. We observed papers up to and including 2005, 2010 as well as 2015. Additionally, we observe the performance of the algorithm when using papers P only published 2000, 2005, 2010, 2015 or later. This experiment was intended to shed light on effects of different publication years, for example surveys are included in the SUSdblp dataset from much earlier years than uninfluential publications. As document vector representation

for these experiments, Doc2Vec was utilised with cosine distance as distance measure on which features were derived from. The performed classifications used all 60 features and was evaluated against the unaltered dataset⁷.

We performed a Kruskal-Wallis H-test as the accuracies for the different models were not normally distributed. Although a significant difference can be observed when classifying on the eight datasets (≤ 2005 , ≤ 2010 , ≤ 2015 , ≥ 2000 , ≥ 2005 , ≥ 2010 , ≥ 2015 and unaltered), the datasets do not significantly differ from the unaltered option. This indicates the assumption, that the main results of the evaluation would not change if only publications from certain years an onwards or preceding a point in time are included. Disrupted citation life cycles of publications do not seem to negatively affect classification accuracy.

6.6.3 Discussion

The presented SUSdblp dataset is robust when omitting one or several references and citations. Although the classification performance decreases when doing so, accuracies comparable to the one of the unaltered dataset can be reached.

When using only publications of the dataset which are existent up until certain years, accuracy of classification drops considerably compared to the unaltered dataset. In determining the class of a publication based on all 60 features, all features from group *A*, *C* and *E* can be heavily impaired as the citing papers of a publication might not lie in the observed time frame. In the worst case, all groups are empty and thus returning 0 values for all twelve features. Usually the references of a publication are unaltered as soon as a publication is part of a certain time frame, leaving features from groups *B* and *D* unchanged when compared to the full dataset. In the single feature prediction task, sumA achieved the highest accuracy for Doc2Vec combined with cosine distance, but when restricting the years, this feature naturally seems to be less descriptive.

Overall, the dataset is appropriate for the classification task, no significant differences compared to using the unaltered and time restricted datasets were found.

6.7 Alternative Approaches for Classification

Although we were able to reach accuracies of over .92 by using semantometrics, the approach is computationally expensive as distances between all

⁷A detailed table on classification accuracies can be found in the appendix 6.E.

citing and referenced papers of publications need to be computed. Comparable or better results might be achieved by using simpler methods.

We evaluate classification based on the number of citations and references of papers as well as classification on the representation of the whole citation network of a publication in a single vector. Lastly, we evaluate how well classes of papers can be predicted based on information available at the time of publication.

6.7.1 Classification on Number of References and Citations

In a first experiment, classification solely based on numbers of references and citations of publications was conducted. This information is relatively easy to obtain and does not require vast amounts of computation and thus could provide a good alternative to usage of semantometrics.

The best accuracy of $.8126 (\pm .0462)$, F1 score $.8131$, $\text{Acc}_{c_0} .7606$, $\text{Acc}_{c_1} .697$, $\text{Acc}_{c_2} .9803$) was achieved by usage of GB. The high accuracy for class c_2 is not surprising, if the number of citations is low, publications are part of this class by definition. Even though these results are quite good, semantometrics achieved much higher accuracies. The comparably low results might be caused by the construction process of the dataset. In the dataset, not all references and citations of a publication are contained. Only those which come from the area of computer science and adjacent fields and are thus covered by dblp as well as ones for which a concatenation of titles and (if existent and assignable in the underlying dataset) abstracts has at least ten tokens. Publications not fitting these criteria are not considered in the number of references and citations of publications as they are not contained in the SUSdblp dataset.

The number of citations of publications might not be the number of citations a paper is going to accumulate until it becomes obsolete if the citation life cycle of the paper has not yet ended. As half-lives of corpora are increasing [15] and the average year of publications contained in the dataset lies between 2006 and 2008 for the three classes, this aspect is worth considering.

Another explanation for the comparably poor performance of this experiment could be the diverse reasons for citing [24] which are not covered by this method or the possibly high number of uncited influences [24, 42, 49].

6.7.2 Classification on Document Vector Representations

In the following experiments, classification based on the citation networks of publications is performed. The same information as with semantometrics is required but instead of calculating distances between P , X and Y , dimensions of these sets of papers are concatenated and used in combination for classification. This method is less complex and might also achieve reliable results.

All Dimensions of Document Vector Representations of Publications

In this experiment, whole citation networks are used to compare the results to those from semantometrics presented in Section 6.5.2⁸.

We averaged the values of all dimensions of the document vector representations for all references as well as citing papers to obtain vectors of a length which equals the number of dimensions of a certain document representation for each. Vectors generated for references are concatenated with vector representations of the publication before vectors computed for citing papers are appended for every paper. The number of dimensions of each vector equals $3 * \text{length of document vector for the publication}$. Thus, for tf-idf vectors, classifications become computationally expensive. As the dimension representing the words *survey* and *review* naturally tend to be highly descriptive in our task, we also performed classifications on stemmed and unstemmed tf-idf vectors where we omitted these two dimensions in publications P .

Tf-idf representations of unstemmed documents achieved an accuracy of .948. Using stemmed documents for the generation of tf-idf vectors results in the highest accuracy of .949. The downside to using tf-idf vector representations is the high number of dimensions which need to be considered. Omitting the dimensions for the words *survey* or *review* in the tf-idf vectors of P did decrease the accuracy only slightly, leading to the conclusion of these dimensions not being highly relevant for the ternary classification. Table 6.6 shows detailed results on accuracies of all one-vector representations, F1 scores and accuracies for the three classes. A Kruskal-Wallis H-test proved the significant differences between the observed combinations, particularly the tf-idf versions differed from usage of BERT and Doc2Vec. Between tf-idf variants, no significant differences were found.

⁸We also performed experiments on single dimensions of P , all information contained P , X and Y which can be found in the appendix 6.G.

V	Cla	Acc	F1	Acc _{c₀}	Acc _{c₁}	Acc _{c₂}
tf-idf U	GB	.948 (\pm .0252)	.948	.9424	.9273	.9742
tf-idf S	GB	.949 (\pm .0219)	.949	.9515	.9182	.9773
tf-idf wo SR U	GB	.9343 (\pm .0399)	.934	.8955	.9273	.9803
tf-idf wo SR S	GB	.9288 (\pm .0348)	.9285	.8879	.9227	.9758
D2V U	SVM	.8556 (\pm .0517)	.8556	.8348	.903	.8288
BERT U	GB	.8788 (\pm .0419)	.8788	.8758	.903	.8576
LDA U	RF	.9182 (\pm .0225)	.9177	.8773	.9015	.9758
LDA S	GB	.9096 (\pm .0223)	.9095	.8833	.8848	.9606
years	GB	.5621 (\pm .0687)	.5621	.5152	.5758	.5955

Table 6.6: Classification accuracy and F1 scores for all dimensions from the different one-vector representations of publications with their citing and referenced papers from the SUSdblp dataset for the best performing classification algorithm.

The highest values from usage of semantometrics were surpassed by .0243 proving the viability of plain document vector representations. Results from this experiment hint at the superiority of straightforward methods where no information is omitted compared to semantometrics for computer science publications.

An independent two-sample t-test is used to compare the best performing model from semantometrics and the best performing model from the one-vector representations. The two models were found to be significantly different.

The combination of multiple document vector representations might be able to produce even higher accuracies but was out of scope for this paper as we only intended to evaluate semantometrics and compare the approach to a straightforward method of classification.

Information Available at Publication Time

Here, we again want to observe prediction performance based on information which is available as soon as a paper is published. We compare our results with those from semantometrics as described in Section 6.5.5. We construct one-vector representations of the references X as described before and concatenate them with document vectors of the publications P .

Usage of tf-idf vectors results in the highest accuracy of .9323 for unstemmed publications. Similarly, for stemmed papers, an accuracy of .9318 can be achieved. Semantics of publications seem to provide good results (Doc2Vec Acc .8348, BERT Acc .8712) which is promising as these vectors

V	Cl	Acc	F1	Acc _{c₀}	Acc _{c₁}	Acc _{c₂}
tf-idf U	GB	.9323 (\pm .0193)	.9322	.9318	.8955	.9697
tf-idf S	GB	.9318 (\pm .0218)	.9317	.9333	.8864	.9758
D2V U	GB	.8348 (\pm .0513)	.8349	.8515	.8318	.8212
BERT U	GB	.8712 (\pm .0497)	.8712	.8606	.8955	.8576
LDA U	GB	.8424 (\pm .0414)	.8419	.7864	.9	.8409
LDA S	GB	.8369 (\pm .0391)	.8365	.7864	.8818	.8424
years	SVM	.5015 (\pm .0580)	.4884	.2773	.5621	.6652

Table 6.7: Classification accuracy and F1 scores for concatenated on- vector representations of references and publication from the SUSdblp dataset for the best performing classification algorithm.

can be computed quite easily as they only consist of several hundreds of dimensions.

The models are significantly different from each other by a Kruskal-Wallis H-test. The two tf-idf variants differ from BERT, LDA and years significantly (large effect size (1.3501 to .7699)). Additionally, models based on years differ from models using BERT significantly (large effect size (.8631)).

Table 6.7 provides detailed results on accuracies of all one-vector representations, F1 scores and accuracies for the three classes of vectors for references X concatenated with P . The best accuracy for this task achieved with usage of semantometrics was surpassed by .1171 in this experiment. Classification on numerous concatenations of several document representations again holds the possibility of further improvements of the results but was out of scope for this paper.

Application of a Welch t-test showed significant differences between the best performing semantometrics approach and the best performing model when utilising one-vector representations of publication data, which is available as soon as a paper is published. An independent two-sample t-test on the BERT model from this experiment and the best performing one from semantometrics on information available at publication time also showed significant differences.

Using this method allows for the prediction of the quality of a publication without having to wait for citations to accumulate. The straightforward approach produces more reliable results than semantometrics.

6.7.3 Discussion

Using the plain number of citations and references in the classification task only resulted in an accuracy worse than the ones achieved when using semantometrics. This might be owed to the creation process of the dataset and its inability to reflect the real number of references and citations of publications.

Classifying on document vector representations seemed to be more feasible. In this scenario, no features are artificially constructed but instead, all information present is used. This prevents data potentially being relevant for the classification from being omitted. Representation of the whole citation network in a single vector representation on which classification was performed achieved the highest overall results of .949 in accuracy for tf-idf vectors of stemmed publications. For all three classes, accuracies of over .91 were achieved, which indicates that the usage of semantometrics does not provide an advantage in this case.

When using only information which is available at the time a paper is published, accuracies as high as .9323 can be achieved when using unstemmed documents on which tf-idf vectors are constructed. This value is .1171 higher than the best accuracy which can be achieved while using features derived from semantometrics under the same premise, the difference is significant. As tf-idf vectors suffer from high dimensionality, usage of information derived from semantics of references and publications in form of BERT vectors could solve this issue. An accuracy of .8712 is achieved, which is +.056 compared to the highest value for utilisation of semantometrics. This method is highly relevant as it is able to predict research quality in real time compared to the approaches which are only applicable in retrospect as they rely on the accumulation of citations.

Our experiments show the significantly superior performance of one-vector representations compared to usage of features derived from semantometrics for our dataset. The SUSdblp dataset is restricted on papers from the area of computer science and adjacent fields as all observed publications are contained in dblp. Using other data sources might lead to different or contrasting results as citation behaviour in computer science is different from other domains, where typically papers from a narrow community are referenced [68]. In the SUSdblp dataset, some references or citations which exist in reality are not contained. If linked papers were not found in dblp or their combined title and abstract was less than ten tokens, they were not considered in the construction of the dataset. Although this does not represent the real world, the restriction might be a reason for the robustness of the dataset. Our dataset holds properties of a realistic use case of estimation influence of a publication. Only papers which stem from the same discipline as the one

to classify are considered. If a publication is referencing or cited by papers from other domains, they seem to be irrelevant in accessing the quality of a paper for the area of computer science.

6.8 Conclusion and Future Work

In this work, the two main tasks of classification a publication with its complete citation network as seminal, survey or uninfluential as well as quality prediction of new papers which did not yet receive citations were observed: We dissected the classification of publications in their citation network as seminal, survey or uninfluential papers based on semantometrics derived from our proposed SUSdblp dataset which is publicly available. We used words, semantics, topical composition and publication years as different aspects of publications and calculated distances in their publication networks. Extraction and usage of single features from the citation networks leads to a classification accuracy of up to .6904 for feature *sumD*, which describes the summed up distance between pairs of papers referenced by *P*, and inner product distance on LDA document vectors. Using all features resulted in a highest accuracy of .8747 if distances between publication years of papers in their citation network are used as base for the feature computation. Combining features derived from multiple aspects of a citation network increased the accuracy up to .9247 when using all five observed embeddings together. Classification based only on data which is available at the time a paper is first published before it was able to accumulate any citations lead to an accuracy of .8152 when using features derived from inner product distances from BERT document vector representations.

We presented a new dataset, the SUSdblp dataset which contains publication years, concatenated titles and abstracts as well as referencing and citing papers for each of the 660 seminal, survey and uninfluential publications. All papers come from the area of computer science and adjacent fields. Our evaluation suggested the dataset being suitable for the ternary classification task at hand.

When comparing semantometrics to established approaches like classification based on one-vector representations of the citation networks using different document embeddings, a highest accuracy of .949 was reached for tf-idf vectors. Using only information available at the time of publication of a paper, an accuracy of .9323 was achieved, labelling utilisation of semantometrics unnecessary for computer science publications.

The following three key conclusions can be derived: First, semantometrics has high potential in estimating quality of publications, especially new ones

which did not yet receive any citations. Second, usage of all information available in a citation network is significantly more potent than application of semantometrics for the two observed tasks. Feature engineering techniques such as semantometrics might perform worse than usage of all information at hand, as potentially useful information is lost. Third, assessment of the quality of a publication for the diverse discipline of computer science can apparently be performed by solely observing referenced and citing papers which are also located in the same area.

We recommend a reevaluation of our results on datasets from different or multiple areas to estimate the reliability of our findings in a broader context. Although the SUSdblp dataset spans multiple sub-fields and thus represents a somewhat diverse set of publications, observance of other domains could deliver evidence of the general inferiority of semantometrics compared to more straightforward methods. A thorough automatic evaluation of our dataset or the creation of a manually evaluated dataset with even more publications and full texts spanning multiple research areas would be desirable. A new dataset which purely holds publications which received a best paper award as seminal publications could describe another interesting bibliographic perspective to analyse.

Future work focused on semantometrics could be the incorporation of more statistical features such as entropy [26] contained in the five groups of distances. Automatic feature engineering with deep feature synthesis [35] could produce more descriptive features which in turn might lead to higher accuracies.

As BERT generated better results than Doc2Vec, more sophisticated document vector representations could produce higher accuracies. A semantic representation with fastText [11] or GloVe [51] could contribute to better results. Topic models such as DTM [9] or ATM [57] could prove to be more suitable than LDA. Other distance metrics could also be used. It could also help to assign referenced and citing publications different weights as not all referenced papers are equally important for a publication [49, 75]. For example weights based on the number of times referenced and citing papers are cited themselves in the whole dblp corpus or based on the field and time normalised citation count which is also included in the SUSdblp dataset could lead to interesting results.

Lastly, another direction for further efforts could be hyperparameter tuning via grid search or the incorporation of more advanced machine learning algorithms such as gpt-2 [55] or even a neural network as classifier. Instead of viewing the task at hand as a ternary classification problem, one could remodel it to become a binary classification task with usage of an abstaining classifier [72, 52] to describe papers which are neither seminal nor survey

publications.

6.A Evaluation of the Approach: Single Features

For the following experiment, single features from the citation network of every publication are used for the classification procedure. Table 6.8 shows the detailed accuracies for all combinations of document embeddings with all corresponding distance measures. No significant differences were found between the 17 combinations of document vector representation and distance measure when looked at with a Kruskal-Wallis H-test.

V	M	Cl	F	Acc	F1	Acc _{c₀}	Acc _{c₁}	Acc _{c₂}
tf-idf U	COS	LR	sumA	.6879 (\pm .0414)	.6818	.6318	.447	.9848
tf-idf U	JAC	LR	sumA	.6869 (\pm .0431)	.6809	.6303	.447	.9833
tf-idf U	IPD	LR	sumA	.6869 (\pm .0431)	.6809	.6303	.447	.9833
tf-idf S	COS	LR	sumA	.6884 (\pm .0421)	.6824	.6333	.447	.9848
tf-idf S	JAC	LR	sumA	.6869 (\pm .0431)	.6809	.6303	.447	.9833
tf-idf S	IPD	LR	sumA	.6869 (\pm .0431)	.6809	.6303	.447	.9833
D2V U	COS	LR	sumA	.6904 (\pm .0438)	.6845	.6394	.45	.9818
D2V U	JAC	SVM	sumA	.6768 (\pm .0359)	.6661	.6727	.3788	.9788
D2V U	IPD	GB	sumA	.6833 (\pm .0429)	.6761	.4076	.6818	.9606
BERT U	COS	LR	sumA	.6889 (\pm .0472)	.6833	.6364	.453	.9773
BERT U	JAC	LR	sumA	.6869 (\pm .0431)	.6809	.6303	.447	.9833
BERT U	IPD	GB	sumE	.6747(\pm .0687)	.6801	.6561	.7121	.6561
LDA U	EMD	GB	sumA	.6838 (\pm .0519)	.6787	.6091	.453	.9894
LDA U	IPD	GB	sumD	.6939 (\pm .0723)	.6989	.2727	.5076	.9182
LDA S	EMD	GB	sumA	.6859 (\pm .0388)	.6776	.3909	.7076	.9591
LDA S	IPD	LR	sumA	.6874 (\pm .0430)	.6815	.6303	.4485	.9833
years	DIST	GB	sumA	.6879 (\pm .0372)	.6856	.0848	.9121	.9742

Table 6.8: Observation of different distance measures for all respective document vector representations. For the classification algorithm achieving the highest accuracy, the single feature (F), the corresponding F1 score as well as accuracies for the three classes c_0 , c_1 and c_2 are displayed.

V	M	Cla	Acc	F1	Acc _{c₀}	Acc _{c₁}	Acc _{c₂}
tf-idf U	COS	GB	.8414 (\pm .0356)	.8415	.7894	.7515	.9833
tf-idf U	JAC	GB	.8323 (\pm .0299)	.8324	.7727	.7439	.9803
tf-idf U	IPD	GB	.8298 (\pm .0326)	.8299	.7712	.7364	.9818
tf-idf S	COS	GB	.8343 (\pm .0304)	.8343	.7742	.7439	.9848
tf-idf S	JAC	GB	.8439 (\pm .0371)	.8439	.7848	.7621	.9848
tf-idf S	IPD	GB	.8379 (\pm .0308)	.838	.7742	.7561	.9833
D2V U	COS	GB	.8525 (\pm .0309)	.8526	.7742	.8015	.9818
D2V U	JAC	GB	.8247 (\pm .0445)	.825	.7348	.7561	.9833
D2V U	IPD	GB	.8253 (\pm .0350)	.8249	.7076	.7879	.9803
BERT U	COS	GB	.8586 (\pm .0575)	.8584	.7788	.8152	.9818
BERT U	JAC	GB	.8561 (\pm .0546)	.8562	.7909	.7924	.9848
BERT U	IPD	GB	.8646 (\pm .0569)	.8647	.8061	.8	.9879
LDA U	EMD	GB	.8263 (\pm .0495)	.8264	.7258	.7758	.9773
LDA U	IPD	GB	.8601 (\pm .0367)	.8601	.8167	.7788	.9848
LDA S	EMD	GB	.8187 (\pm .0487)	.8183	.7212	.747	.9879
LDA S	IPD	GB	.8475 (\pm .0266)	.8477	.7803	.7773	.9848
years	DIST	RF	.8747 (\pm .0401)	.8743	.8439	.7985	.9818

Table 6.9: Best classifiers dependent on distance measures with different D2V and BERT vector representations. For the classification algorithm achieving the highest accuracy, the corresponding F1 score as well as accuracies for the three classes c_0 , c_1 and c_2 are displayed.

6.B Evaluation of the Approach: All Features

For the following experiment, all 60 features from the citation network of every publication are used for the classification procedure. Table 6.9 shows the detailed accuracies for all combinations of document embeddings with all corresponding distance measures. Significant differences between the 17 combinations of document vector representation and distance measures were observed when looked at with a Kruskal-Wallis H-test.

6.C Evaluation of the Approach: 33 Features

For the following experiment, the complete citation network of every publication is used for the classification procedure. Classification algorithms are trained on the 33 features, which were found to be significant by Herrman-

V	M	Cla	Acc	F1	Acc _{c₀}	Acc _{c₁}	Acc _{c₂}
tf-idf U	COS	GB	.8293 (\pm .0296)	.8297	.7682	.7394	.9803
tf-idf U	JAC	GB	.8293 (\pm .0428)	.8295	.7712	.7364	.9803
tf-idf U	IPD	RF	.8288 (\pm .0204)	.8294	.747	.7591	.9803
tf-idf S	COS	GB	.8293 (\pm .0377)	.8295	.7667	.7364	.9848
tf-idf S	JAC	GB	.8399 (\pm .0316)	.8401	.7909	.747	.9818
tf-idf S	IPD	GB	.8409 (\pm .0208)	.8412	.7924	.747	.9833
D2V U	COS	GB	.8404 (\pm .0268)	.8409	.7667	.7742	.9803
D2V U	JAC	GB	.8227 (\pm .0357)	.8228	.7258	.7606	.9818
D2V U	IPD	GB	.8136 (\pm .0422)	.8128	.6909	.7682	.9818
BERT U	COS	GB	.85 (\pm .0461)	.8498	.7652	.8045	.9803
BERT U	JAC	GB	.8561 (\pm .0368)	.8562	.7955	.7864	.9864
BERT U	IPD	GB	.8636 (\pm .0544)	.8637	.8015	.8061	.9833
LDA U	EMD	GB	.8066 (\pm .0383)	.8057	.6606	.7773	.9818
LDA U	IPD	GB	.8505 (\pm .0431)	.8509	.7985	.7727	.9803
LDA S	EMD	GB	.8066 (\pm .0373)	.8066	.7182	.7182	.9833
LDA S	IPD	GB	.8318 (\pm .0271)	.832	.7667	.7455	.9833
years	DIST	GB	.8712 (\pm .0283)	.8708	.8303	.8	.9833

Table 6.10: Best classifier dependent on document vector representation and distance measure. For the classification algorithm achieving the highest accuracy, the corresponding F1 score as well as accuracies for the three classes c_0 , c_1 and c_2 are displayed.

nova et al. [28]. Table 6.10 shows the detailed accuracies for all combinations of document embeddings with corresponding distance measures. Significant differences between the 17 combinations of document vector representation and distance measure were discovered when looked at with a Kruskal-Wallis H-test.

6.D Evaluation of the Approach: Combination

Experiments were conducted where two, three or four document vector representations were utilised in concatenation. Table 6.11 shows detailed results on accuracies of the best performing combinations for all numbers of combinations.

The combination of two feature sets achieving the highest accuracy of .9091 (F1 score .9093) were Jaccard distances of unstemmed BERT vectors

# combined feature sets	Acc	F1	Acc _{c₀}	Acc _{c₁}	Acc _{c₂}
2 (BERT, years)	.9096 (\pm .0342)	.9098	.8985	.8455	.9848
3 (LDA, BERT, years)	.9192 (\pm .0414)	.9192	.8879	.8833	.9864
4 (tf-idf, LDA, BERT, years)	.9237 (\pm .0345)	.9238	.8939	.8909	.9864

Table 6.11: Best accuracies dependent on the number of combined sets of features derived from distances between document vector representations.

year	Clas	Acc	F1	Acc _{c₀} (#P)	Acc _{c₁} (#P)	Acc _{c₂} (#P)
≤ 2005	GB	.7755 (\pm .0779)	.7763	.7409 (220)	.797 (197)	.7963 (162)
≤ 2010	GB	.7751 (\pm .0477)	.7732	.6904 (478)	.7044 (345)	.9156 (462)
≤ 2015	GB	.8331 (\pm .0601)	.8321	.7446 (646)	.7824 (648)	.9697 (659)
≥ 2000	GB	.8735 (\pm .0337)	.8736	.8156 (602)	.8067 (538)	.983 (646)
≥ 2005	GB	.8624 (\pm .0382)	.8626	.7857 (476)	.8025 (476)	.9801 (552)
≥ 2010	GB	.8588 (\pm .0483)	.857	.6835 (218)	.8782 (353)	.9841 (251)
≥ 2015	RF	.8525 (\pm .1814)	.8531	.8525 (61)	.84 (50)	.9091 (11)

Table 6.12: Classification accuracy and F1 scores using all features derived from Doc2Vec vector representations for the best performing classification algorithm up until or from and after different years as well as the accuracies and number of publication per class.

and distances between years. When concatenating three feature sets, the highest accuracy of .9192 (F1 score .9192) was reached when using features derived from inner product distances from stemmed LDA and unstemmed BERT document vector representations as well as distances between years. Combining four sets of features lead to an accuracy of up to .9237 (F1 score .9238). This value was reached by using all features derived from inner product distances of stemmed tf-idf, stemmed LDA and unstemmed BERT document vector embeddings as well as distances between years. Gradient boosting was the best performing classifier in all of these cases.

6.E Evaluation of the Dataset: Classification for Different Years

Table 6.12 provides classification accuracies for the dataset with publications from P , X and Y until certain years and for publications P which are older than multiple years. Doc2Vec document vector representations and cosine distance were used in these calculations.

6.F Evaluation of the Dataset: Abstract Length Bias

In the SUSdblp dataset, concatenated titles and abstracts of uninfluential papers tend to be shorter than those of seminal or survey publications. Possible classification accuracy bias caused by these different lengths is evaluated in the following experiment.

For each of the three groups, publications complete with their citation network were extracted which did not have a paper in the other two classes with the same abstract lengths. After this, 342 publications remained for each class. Doc2Vec document vector representations combined with cosine distance are utilised here to derive the 60 features on which classification was performed. The highest accuracy was reached by usage of random forests as classifier (Acc .8285 (\pm .0751), F1 .828, Acc_{c₀} .7398, Acc_{c₁} .7602, Acc_{c₂} .9854). Not significant differences were found in comparison to utilisation of the full SUSdblp dataset when using Welch-ANOVA. Bias due to different abstract lengths for the three classes could thus be suspended in terms of overall classification accuracy.

6.G Other Approaches: All Dimensions of Document Vector Representations of Publications

6.G.1 Single Dimensions of Document Vector Representations of Publications

For tf-idf vectors, LDA document representations as well as years, single dimensions are quite understandable. The meaning of Doc2Vec and BERT dimensions cannot be explained as easily. So here, we restrict the single dimension classifications on tf-idf vectors, LDA document vector representations and publication years.

The following classifications are all performed by using one dimension of the numerous dimensions of document vector representation of publications P .

When classifying on one of the dimensions of unstemmed tf-idf document vector representations, the highest accuracy of .5359 was achieved for the dimension representing the word *survey*. Unfortunately, this classifier is completely unable to identify publications of type uninfluential. For single

topic	ten most probable words
ut41	the, of, and, to, in, a, is, this, as, are
st87	the, and, of, in, to, research, thi, on, their, null

Table 6.13: Ten most probable words per topic for best performing topics in the single feature classification based on features of the publication alone, in decreasing probability.

V	Clas	DIM	Acc	F1	Acc _{c₀}	Acc _{c₁}	Acc _{c₂}
tf-idf U	KNN	u460	.5359 (\pm .0273)	.4473	1	.4394	0
tf-idf S	KNN	s380	.5616 (\pm .0316)	.4707	1	.4879	0
LDA U	GB	ut41	.5409 (\pm .0930)	.5196	.6576	.8182	.1106
LDA S	GB	st87	.5101 (\pm .0544)	.4991	.75	.676	.0424
years	GB	0	.4879 (\pm .0452)	.4837	.3773	.4591	.6273

Table 6.14: Classification accuracy and F1 scores for single dimensions DIM from different vector representations of publications P from the SUSdblp dataset for the best performing classification algorithm.

dimensions from stemmed tf-idf document vector representations, an accuracy of .5616 can be achieved for dimension 380 which refers to the word *survei* (stemmed version of survey) as the most descriptive one. Again, the classifier is not able to identify uninfluential papers. Classification on single dimensions from the LDA embedding of unstemmed publications lead to an accuracy of up to .5409 for dimension ut41. This dimension seems to represent a background topic [4] which usually is contained in all documents. In vast parts of the corpus, topic 41 can be observed as displayed in Figure 6.5. The top words of this topic can be seen in Table 6.13. When classifying on dimensions of stemmed LDA document vector representations, an accuracy of .5101 is achieved for dimension st87. From its most probable words, which are displayed in Table 6.13, this dimension again seems to describe a background topic. An accuracy of .4879 is achieved when classifying on publication years of papers P .

Table 6.14 provides detailed results on accuracies per class and F1 scores for classification based on single dimensions from the document vector representations as well as the best performing classifiers. In general, the highest accuracy decreased by .1323 when compared to the accuracy achieved with usage of the best single feature derived from citation networks of publications. The five models are significantly different from each other when looked at with Kruskal-Wallis H-test. Utilisation of years is significantly different from the two tf-idf variants. Additionally, usage of stemmed tf-idf is significantly

V	Cla	Acc	F1	Acc _{c₀}	Acc _{c₁}	Acc _{c₂}
tf-idf U	GB	.7848 (\pm .0598)	.7873	.6894	.8576	.8076
tf-idf S	GB	.7707 (\pm .0373)	.7734	.6894	.8606	.7621
tf-idf wo SR U	GB	.7722 (\pm .0308)	.7729	.6879	.8576	.7712
tf-idf wo SR S	GB	.7601 (\pm .0443)	.7605	.6879	.8591	.7333
D2V U	SVM	.7763 (\pm .0642)	.7766	.7152	.8636	.75
BERT U	LR	.7828 (\pm .0544)	.7819	.7273	.903	.7182
LDA U	GB	.704 (\pm .0756)	.7014	.6848	.8318	.5955
LDA S	GB	.7071 (\pm .0472)	.7059	.6697	.8076	.6439

Table 6.15: Classification accuracy and F1 scores for all dimensions from the different vector representations of publications P from the SUSdblp dataset for the best performing classification algorithm.

different from utilisation of stemmed LDA.

6.G.2 All Dimensions of Document Vector Representations of Publications

The next experiments observe classification on all dimensions of document vector representations of the publications P . As the dimension representing the words survey and review naturally tend to be highly descriptive in our task, we also performed classifications on stemmed and unstemmed tf-idf vectors, where we omitted these two dimensions. Table 6.15 shows detailed results on all document vector representations with their accuracies, F1 scores and accuracies for the three classes. The highest accuracy from the multi feature classification of semantometrics surpassed the best result from this experiment by .0899.

Significant differences in the eight models were found when looked at with Kuskal-Wallis H-test. The two LDA document vector representations significantly differ from all other document vector representations.

6.G.3 All Dimensions of Document Vector Representations of Referenced Papers

The following experiments required for vector representations of referenced papers of publications to be of equal length. For this, we averaged the values of all dimensions of the document vector representations for all referenced papers to obtain vectors of a length which equals the number of dimensions of a certain document representation. Using only referenced publications in

V	Cla	Acc	F1	Acc _{c₀}	Acc _{c₁}	Acc _{c₂}
tf-idf U	GB	.8778 (\pm .0349)	.8768	.8485	.8212	.9636
tf-idf S	GB	.8712 (\pm .0483)	.8699	.8318	.8121	.9697
D2V U	RF	.8152 (\pm .0368)	.8151	.8242	.7848	.8364
BERT U	GB	.8212 (\pm .0373)	.8214	.8364	.8076	.8197
LDA U	GB	.7985 (\pm .0444)	.798	.7273	.847	.8212
LDA S	RF	.797 (\pm .0501)	.7948	.6818	.8439	.8652
years	RF	.4343 (\pm .0871)	.4334	.3636	.4394	.5

Table 6.16: Classification accuracy and F1 scores for all dimensions from the different vector representations of referenced papers X from the SUSdblp dataset for the best performing classification algorithm.

the classification would equal using only features derived from group D for semantometrics. Table 6.16 shows detailed results on accuracies, F1 scores and accuracies for the three classes for classification based on one-vector representations of referenced papers X of P .

Usage of ANOVA showed significant differences between the seven observed groups. With Bonferroni correction and Scheffé’s method significant differences between classification based on the dimensions of the tf-idf methods and all other document vector representations were found. Usage of years also results in significantly differences from utilisation of all other document vector representations in the classification task.

6.G.4 All Dimensions of Document Vector Representations of Citing Publications

These experiments again required for vector representations of citing papers of publications to be of equal length. For this, we averaged the values of all dimensions of the document vector representations for all citing papers to obtain vectors of a length which equals the number of dimensions of a certain document representation. Using only citing papers in the classification would equal using only features derived from group E for semantometrics. Table 6.17 shows detailed results on accuracies, F1 scores and accuracies for the three classes for classification based on one-vector representations of citing publications Y of P .

Application of Kruskal-Wallis H-test on the seven models showed significant differences. The two models representing words and semantics of publications were significantly different from those utilising topics. Classification on BERT embeddings was also significantly different from classification on

V	Cl	Acc	F1	Acc _{c₀}	Acc _{c₁}	Acc _{c₂}
tf-idf U	GB	.8121 (\pm .0587)	.8109	.7242	.8439	.8682
tf-idf S	GB	.8035 (\pm .0410)	.8022	.7152	.8288	.8667
D2V U	SVM	.7742 (\pm .0476)	.7742	.8136	.7833	.7258
BERT U	GB	.7515 (\pm .0482)	.7513	.7803	.7621	.7121
LDA U	GB	.8707 (\pm .0429)	.8701	.8333	.8258	.9530
LDA S	RF	.8747 (\pm .0234)	.8732	.8152	.8273	.9818
years	RF	.5556 (\pm .0692)	.5476	.4288	.4788	.7591

Table 6.17: Classification accuracy and F1 scores for all dimensions from the different vector representations of citing papers Y from the SUSdblp dataset for the best performing classification algorithm.

tf-idf vectors. Utilisation of years produced significantly different accuracies than all other embeddings except Doc2Vec.

Bibliography

- [1] Core computing research and education. <http://www.core.edu.au/>. accessed: 18.12.2019.
- [2] Seminalsurveydblp dataset for classification of seminal and survey publications <https://doi.org/10.5281/zenodo.3258164>.
- [3] Dag Aksnes. Characteristics of highly cited papers. Research Evaluation, 12:159–170, 12 2003.
- [4] Loulwah AlSumait, Daniel Barbará, James Gentle, and Carlotta Domeniconi. Topic significance ranking of LDA generative models. In Wray L. Buntine, Marko Grobelnik, Dunja Mladenic, and John Shawe-Taylor, editors, Machine Learning and Knowledge Discovery in Databases, European Conference, ECML PKDD 2009, Bled, Slovenia, September 7-11, 2009, Proceedings, Part I, volume 5781 of Lecture Notes in Computer Science, pages 67–82. Springer, 2009.
- [5] Elizabeth Smith Aversa. Citation patterns of highly cited papers and their relationship to literature aging: A study of the working literature. Scientometrics, 7(3-6):383–389, 1985.
- [6] Aurel Avramescu. Actuality and obsolescence of scientific literature. J. Am. Soc. Inf. Sci., 30(5):296–303, 1979.
- [7] Sebastian Baltes and Stephan Diehl. Worse than spam: Issues in sampling software developers. In Proceedings of the 10th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement, ESEM 2016, Ciudad Real, Spain, September 8-9, 2016, pages 52:1–52:6. ACM, 2016.
- [8] Michael G. Banks. An extension of the hirsch index: Indexing scientific topics and compounds. Scientometrics, 69(1):161–168, 2006.
- [9] David M. Blei and John D. Lafferty. Dynamic topic models. In William W. Cohen and Andrew W. Moore, editors, Machine Learning, Proceedings of the Twenty-Third International Conference (ICML 2006), Pittsburgh, Pennsylvania, USA, June 25-29, 2006, volume 148 of ACM International Conference Proceeding Series, pages 113–120. ACM, 2006.
- [10] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent dirichlet allocation. J. Mach. Learn. Res., 3:993–1022, 2003.

- [11] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomáš Mikolov. Enriching word vectors with subword information. Trans. Assoc. Comput. Linguistics, 5:135–146, 2017.
- [12] Katy Börner, Shashikant Penumarthy, Mark R. Meiss, and Weimao Ke. Mapping the diffusion of scholarly knowledge among major U.S. research institutions. Scientometrics, 68(3):415–426, 2006.
- [13] Lutz Bornmann and Hans-Dieter Daniel. The citation speed index: A useful bibliometric indicator to add to the h index. J. Informetrics, 4(3):444–446, 2010.
- [14] Morton B. Brown and Alan B. Forsythe. Robust tests for the equality of variances. Journal of the American Statistical Association, 69(346):364–367, 1974.
- [15] R. E. Burton and R. W. Kebler. The “half-life” of some scientific and technical literatures. American Documentation, 11(1):18–22, 1960.
- [16] V. Cano and N. C. Lind. Citation life cycles of ten citation classics. Scientometrics, 22(2):297–312, 1991.
- [17] Blaise Cronin and Lokman I. Meho. Using the h-index to rank influential information scientists. J. Assoc. Inf. Sci. Technol., 57(9):1275–1278, 2006.
- [18] Philip M. Davis and Angela Cochran. Cited half-life of the journal literature. CoRR, abs/1504.07479, 2015.
- [19] Derek J. de Solla Price. Networks of scientific papers. Science, pages 510–515, 1965.
- [20] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. In Jill Burstein, Christy Doran, and Tamar Solorio, editors, Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers), pages 4171–4186. Association for Computational Linguistics, 2019.
- [21] Laura Dietz, Steffen Bickel, and Tobias Scheffer. Unsupervised prediction of citation influences. In Zoubin Ghahramani, editor, Machine Learning, Proceedings of the Twenty-Fourth International Conference

- (ICML 2007), Corvallis, Oregon, USA, June 20-24, 2007, volume 227 of ACM International Conference Proceeding Series, pages 233–240. ACM, 2007.
- [22] Leo Egghe. Theory and practise of the g-index. Scientometrics, 69(1):131–152, 2006.
- [23] Norbert Fuhr. Some common mistakes in IR evaluation, and how they can be avoided. SIGIR Forum, 51(3):32–41, 2017.
- [24] Eugene Garfield. Can citation indexing be automated?, 1964.
- [25] Sean Gerrish and David M. Blei. A language-based approach to measuring scholarly impact. In Johannes Fürnkranz and Thorsten Joachims, editors, Proceedings of the 27th International Conference on Machine Learning (ICML-10), June 21-24, 2010, Haifa, Israel, pages 375–382. Omnipress, 2010.
- [26] Robert J. Gillies, Paul E. Kinahan, and Hedvig Hricak. Radiomics: Images are more than pictures, they are data. Radiology, 278(2):563–577, 2016. PMID: 26579733.
- [27] Drahomira Herrmannova and Petr Knoth. Semantometrics in coauthorship networks: Fulltext-based approach for analysing patterns of research collaboration. D Lib Mag., 21(11/12), 2015.
- [28] Drahomira Herrmannova, Petr Knoth, and Robert M. Patton. Analyzing citation-distance networks for evaluating publication impact. In Nicoletta Calzolari, Khalid Choukri, Christopher Cieri, Thierry Declerck, Sara Goggi, Kôiti Hasida, Hitoshi Isahara, Bente Maegaard, Joseph Mariani, H el ene Mazo, Asunci on Moreno, Jan Odijk, Stelios Piperidis, and Takenobu Tokunaga, editors, Proceedings of the Eleventh International Conference on Language Resources and Evaluation, LREC 2018, Miyazaki, Japan, May 7-12, 2018. European Language Resources Association (ELRA), 2018.
- [29] Drahomira Herrmannova, Robert M. Patton, Petr Knoth, and Christopher G. Stahl. Citations and readership are poor indicators of research excellence: Introducing trueimpactdataset, a new dataset for validating research evaluation metrics. In Proceedings of the 1st Workshop on Scholarly Web Mining, SWM@WSDM 2017, Cambridge, United Kingdom, February 10, 2017, pages 41–48. ACM, 2017.

- [30] Jorge E. Hirsch. An index to quantify an individual’s scientific research output. Proc. Natl. Acad. Sci. USA, 102(46):16569–16572, 2005.
- [31] Jorge E. Hirsch. h_Q : An index to quantify an individual’s scientific leadership. Scientometrics, 118(2):673–686, 2019.
- [32] Wen-Ru Hou, Ming Li, and Deng-Ke Niu. Counting citations in texts rather than reference lists to improve the accuracy of assessing scientific contribution. BioEssays, 33(10):724–727, 2011.
- [33] BiHui Jin, Liming Liang, Ronald Rousseau, and Leo Egghe. The r-and ar-indices: Complementing the h-index. Chinese Science Bulletin, 52, 03 2007.
- [34] Oliphant E. Peterson P. et al. Jones, E. Scipy: Open source scientific tools for python. <http://www.scipy.org>. accessed 18 september 2019.
- [35] James Max Kanter and Kalyan Veeramachaneni. Deep feature synthesis: Towards automating data science endeavors. In 2015 IEEE International Conference on Data Science and Advanced Analytics, DSAA 2015, Campus des Cordeliers, Paris, France, October 19-21, 2015, pages 1–10. IEEE, 2015.
- [36] Petr Knoth and Drahomira Herrmannova. Towards semantometrics: A new semantic similarity based measure for assessing a research publication’s contribution. D Lib Mag., 20(11/12), 2014.
- [37] Christin Katharina Kreutz, Premtim Sahitaj, and Ralf Schenkel. Revaluating semantometrics from computer science publications. In Muthu Kumar Chandrasekaran and Philipp Mayr, editors, Proceedings of the 4th Joint Workshop on Bibliometric-enhanced Information Retrieval and Natural Language Processing for Digital Libraries (BIRNDL 2019) co-located with the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2019), Paris, France, July 25, 2019, volume 2414 of CEUR Workshop Proceedings, pages 42–55. CEUR-WS.org, 2019.
- [38] Virendra Kumar, Yuhua Gu, Satrajit Basu, Anders Berglund, Steven A. Eschrich, Matthew B. Schabath, Kenneth Forster, Hugo J.W.L. Aerts, Andre Dekker, David Fenstermacher, Dmitry B. Goldgof, Lawrence O. Hall, Philippe Lambin, Yoganand Balagurunathan, Robert A. Gatenby, and Robert J. Gillies. Radiomics: the process and the challenges. Magnetic Resonance Imaging, 30(9):1234–1248, 2012. Quantitative Imaging in Cancer.

- [39] Jey Han Lau and Timothy Baldwin. An empirical evaluation of doc2vec with practical insights into document embedding generation. In Phil Blunsom, Kyunghyun Cho, Shay B. Cohen, Edward Grefenstette, Karl Moritz Hermann, Laura Rimell, Jason Weston, and Scott Wen-tau Yih, editors, Proceedings of the 1st Workshop on Representation Learning for NLP, Rep4NLP@ACL 2016, Berlin, Germany, August 11, 2016, pages 78–86. Association for Computational Linguistics, 2016.
- [40] Quoc V. Le and Tomas Mikolov. Distributed representations of sentences and documents. In Proceedings of the 31th International Conference on Machine Learning, ICML 2014, Beijing, China, 21-26 June 2014, volume 32 of JMLR Workshop and Conference Proceedings, pages 1188–1196. JMLR.org, 2014.
- [41] Michael Ley. DBLP - some lessons learned. Proc. VLDB Endow., 2(2):1493–1500, 2009.
- [42] Michael H. MacRoberts and Barbara R. MacRoberts. Problems of citation analysis: A study of uncited and seldom-cited influences. J. Assoc. Inf. Sci. Technol., 61(1):1–12, 2010.
- [43] Alberto Martın-Martın, Enrique Orduna-Malea, Juan Manuel Ayllon, and Emilio Delgado Lopez-Cozar. Back to the past: on the shoulders of an academic search engine giant. Scientometrics, 107(3):1477–1487, 2016.
- [44] Frank J. Massey. The kolmogorov-smirnov test for goodness of fit. Journal of the American Statistical Association, 46(253):68–78, 1951.
- [45] Henk Moed. The impact-factors debate: the isi’s uses and limits. Nature, 415:731–2, 03 2002.
- [46] Sergio Lopez Montolio, David Dominguez-Sal, and Josep Lluıs Larriba-Pey. Research endogamy as an indicator of conference quality. SIGMOD Rec., 42(2):11–16, 2013.
- [47] Adam Paszke, S. Gross, Soumith Chintala, Gregory Chanan, E. Yang, Zach DeVito, Zeming Lin, Alban Desmaison, L. Antiga, and Adam Lerer. Automatic differentiation in pytorch. 2017.
- [48] Robert M. Patton, Drahomira Herrmannova, Christopher G. Stahl, Jack C. Wells, and Thomas E. Potok. Audience based view of publication impact. In 2017 ACM/IEEE Joint Conference on Digital Libraries,

- JCDL 2017, Toronto, ON, Canada, June 19-23, 2017, pages 64–68. IEEE Computer Society, 2017.
- [49] Robert M. Patton, Christopher G. Stahl, and Jack C. Wells. Measuring scientific impact beyond citation counts. D Lib Mag., 22(9/10), 2016.
- [50] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake VanderPlas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Edouard Duchesnay. Scikit-learn: Machine learning in python. J. Mach. Learn. Res., 12:2825–2830, 2011.
- [51] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. Glove: Global vectors for word representation. In Alessandro Moschitti, Bo Pang, and Walter Daelemans, editors, Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL, pages 1532–1543. ACL, 2014.
- [52] Tadeusz Pietraszek. Optimizing abstaining classifiers using ROC analysis. In Luc De Raedt and Stefan Wrobel, editors, Machine Learning, Proceedings of the Twenty-Second International Conference (ICML 2005), Bonn, Germany, August 7-11, 2005, volume 119 of ACM International Conference Proceeding Series, pages 665–672. ACM, 2005.
- [53] Martin F. Porter. An algorithm for suffix stripping. Program, 14(3):130–137, 1980.
- [54] David Pride and Petr Knoth. Incidental or influential? - challenges in automatically detecting citation importance using publication full texts. In Jaap Kamps, Giannis Tsakonas, Yannis Manolopoulos, Lazaros S. Iliadis, and Ioannis Karydis, editors, Research and Advanced Technology for Digital Libraries - 21st International Conference on Theory and Practice of Digital Libraries, TPD L 2017, Thessaloniki, Greece, September 18-21, 2017, Proceedings, volume 10450 of Lecture Notes in Computer Science, pages 572–578. Springer, 2017.
- [55] Alec Radford, Jeff Wu, R. Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. 2019.
- [56] Lais M. A. Rocha and Mirella M. Moro. Research contribution as a measure of influence. In Fatma Özcan, Georgia Koutrika, and Sam

- Madden, editors, Proceedings of the 2016 International Conference on Management of Data, SIGMOD Conference 2016, San Francisco, CA, USA, June 26 - July 01, 2016, pages 2259–2260. ACM, 2016.
- [57] Michal Rosen-Zvi, Thomas L. Griffiths, Mark Steyvers, and Padhraic Smyth. The author-topic model for authors and documents. In David Maxwell Chickering and Joseph Y. Halpern, editors, UAI '04, Proceedings of the 20th Conference in Uncertainty in Artificial Intelligence, Banff, Canada, July 7-11, 2004, pages 487–494. AUAI Press, 2004.
- [58] Ronald Rousseau and Fred Y. Ye. A proposal for a dynamic h-type index. J. Assoc. Inf. Sci. Technol., 59(11):1853–1855, 2008.
- [59] Anup Sawant Brian Uzzi & Noshir Contractor Ryan Whalen, Yun Huang. Natural language processing, article content & bibliometrics: Predicting high impact science. In ASCW'15 Workshop at Web Science 2015, pages 6–8, 2015.
- [60] Gerard Salton, Anita Wong, and Chung-Shu Yang. A vector space model for automatic indexing. Commun. ACM, 18(11):613–620, 1975.
- [61] Alexandra Schofield, Måns Magnusson, and David M. Mimno. Pulling out the stops: Rethinking stopword removal for topic models. In Mirella Lapata, Phil Blunsom, and Alexander Koller, editors, Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2017, Valencia, Spain, April 3-7, 2017, Volume 2: Short Papers, pages 432–436. Association for Computational Linguistics, 2017.
- [62] Michael Schreiber. Self-citation corrections for the hirsch index. EPL (Europhysics Letters), 78:30002, 04 2007.
- [63] Michael Schreiber. The influence of self-citation corrections on egghe's g index. Scientometrics, 76(1):187–200, 2008.
- [64] Per O. Seglen. The skewness of science. J. Am. Soc. Inf. Sci., 43(9):628–638, 1992.
- [65] Per O. Seglen. Casual relationship between article citedness and journal impact. J. Am. Soc. Inf. Sci., 45(1):1–11, 1994.
- [66] Per O. Seglen. Why the impact factor of journals should not be used for evaluating research. BMJ, 314(7079):497, 1997.

- [67] S. S. Shapiro and M. B. Wilk. An analysis of variance test for normality (complete samples). *Biometrika*, 52(3/4):591–611, 1965.
- [68] Xiaolin Shi, Jure Leskovec, and Daniel A. McFarland. Citing for high impact. In Jane Hunter, Carl Lagoze, C. Lee Giles, and Yuan-Fang Li, editors, Proceedings of the 2010 Joint International Conference on Digital Libraries, JCDL 2010, Gold Coast, Queensland, Australia, June 21-25, 2010, pages 49–58. ACM, 2010.
- [69] Thiago H. P. Silva, Mirella M. Moro, Ana Paula Couto da Silva, Wagner Meira Jr., and Alberto H. F. Laender. Community-based endogamy as an influence indicator. In IEEE/ACM Joint Conference on Digital Libraries, JCDL 2014, London, United Kingdom, September 8-12, 2014, pages 67–76. IEEE Computer Society, 2014.
- [70] Mikhail V. Simkin and Vwani P. Roychowdhury. Copied citations create renowned papers. *Annals of Improbable Research*, 11(1):24–27, 2005.
- [71] Jie Tang, Jing Zhang, Limin Yao, Juanzi Li, Li Zhang, and Zhong Su. Arnetminer: extraction and mining of academic social networks. In Ying Li, Bing Liu, and Sunita Sarawagi, editors, Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Las Vegas, Nevada, USA, August 24-27, 2008, pages 990–998. ACM, 2008.
- [72] Francesco Tortorella. An optimal reject rule for binary classifiers. In Francesc J. Ferri, José Manuel Iñesta Quereda, Adnan Amin, and Pavel Pudil, editors, Advances in Pattern Recognition, Joint IAPR International Workshops SSPR 2000 and SPR 2000, [8th International Workshop on Structural and Syntactic Pattern Recognition, 3rd International Workshop on Statistical Techniques in Pattern Recognition], Alicante, Spain, August 30 - September 1, 2000, Proceedings, volume 1876 of Lecture Notes in Computer Science, pages 611–620. Springer, 2000.
- [73] Marco Valenzuela, Vu Ha, and Oren Etzioni. Identifying meaningful citations. In Cornelia Caragea, C. Lee Giles, Narayan L. Bhamidipati, Doina Caragea, Sujatha Das Gollapalli, Saurabh Kataria, Huan Liu, and Feng Xia, editors, Scholarly Big Data: AI Perspectives, Challenges, and Ideas, Papers from the 2015 AAAI Workshop, Austin, Texas, USA, January, 2015, volume WS-15-13 of AAAI Workshops. AAAI Press, 2015.

- [74] Alex D. Wade, Kuansan Wang, Yizhou Sun, and Antonio Gulli. WSDM cup 2016: Entity ranking challenge. In Paul N. Bennett, Vanja Josifovski, Jennifer Neville, and Filip Radlinski, editors, Proceedings of the Ninth ACM International Conference on Web Search and Data Mining, San Francisco, CA, USA, February 22-25, 2016, pages 593–594. ACM, 2016.
- [75] Xiaodan Zhu, Peter D. Turney, Daniel Lemire, and André Vellino. Measuring academic influence: Not all citations are equal. J. Assoc. Inf. Sci. Technol., 66(2):408–427, 2015.
- [76] Radim Řehůřek and Petr Sojka. Software framework for topic modelling with large corpora. In Proceedings of LREC 2010 workshop New Challenges for NLP Frameworks, pages 46–50, Valletta, Malta, 2010. University of Malta.

7. Scientific Paper Recommendation Systems: a Literature Review of recent Publications

Outline

7.1	Introduction	136
7.2	Problem Statement	137
7.3	Literature Review	138
7.3.1	Scope	138
7.3.2	Meta analysis	140
7.3.3	Categorisation	140
	Former Categorisation	140
	Novel Categorisation	143
7.3.4	Paper Recommendation Systems	144
7.3.5	Other relevant Work	153
	Surrounding Paper Recommendation	153
	(R)Evaluations	153
	Living Labs	154
	Multilingual/Cross-lingual Recommendation	154
	Related Recommendation Systems	155
7.4	Datasets	155
7.4.1	dblp based datasets	158
7.4.2	SPRD based datasets	158
7.4.3	CiteULike based datasets	158
7.4.4	ACM based datasets	159
7.4.5	Scopus based datasets	159
7.4.6	AMiner based datasets	159
7.4.7	AAN based datasets	159

7.4.8	Sowiport based datasets	160
7.4.9	CiteSeerX based datasets	160
7.4.10	Patents based datasets	160
7.4.11	Hep-TH based datasets	160
7.4.12	MAG based datasets	161
7.4.13	Others	161
7.5	Evaluation	161
7.5.1	Relevance and Assessment	161
7.5.2	Evaluation Measures	164
	Commonly Used Evaluation Measures	164
	Rarely used Evaluation Measures	165
7.5.3	Evaluation Types	166
7.6	Open Challenges and Objectives	167
7.6.1	Challenges Highlighted in Previous Works	167
	Neglect of User Modelling	167
	Focus on Accuracy	168
	Translating Research into Practice	168
	Persistence and Authority	168
	Cooperation	169
	Information Scarcity	169
	Cold Start	170
	Sparsity or Reduce Coverage	170
	Scalability	170
	Privacy	171
	Serendipity	171
	Unified Scholarly Data Standards	172
	Synonymy	172
	Gray Sheep	172
	Black Sheep	172
	Shilling attack	172
7.6.2	Emerging Challenges	173
	User Evaluation	173
	Target audience	173
	Recommendation Scenario	174
	Fairness/Diversity	175
	Complexity	175
	Explainability	176
	Public Dataset	176
	Comparability	177
7.6.3	Discussion	177
7.7	Conclusion	178

Bibliographic Information

Kreutz, C. K., Schenkel, R. (2022). Scientific Paper Recommendation Systems: a Literature Review of recent Publications. <https://arxiv.org/abs/2201.00682>.

Copyright Notice

©This is a reformatted preprint of an article submitted to a journal. It is published in arXiv with a non-exclusive license to distribute the article.

Keywords

Paper Recommendation System • Publication Suggestion • Literature Review

Abstract

Scientific writing builds upon already published papers. Manual identification of publications to read, cite or consider as related papers relies on a researcher’s ability to identify fitting keywords or initial papers from which a literature search can be started. The rapidly increasing amount of papers has called for automatic measures to find the desired *relevant* publications, so-called paper recommendation systems.

As the number of publications increases so does the amount of paper recommendation systems. Former literature reviews focused on discussing the general landscape of approaches throughout the years and highlight the main directions. We refrain from this perspective, instead we only consider a comparatively small time frame but analyse it fully.

In this literature review we discuss used methods, datasets, evaluations and open challenges encountered in all works first released between January 2019 and October 2021. The goal of this survey is to provide a comprehensive and complete overview of current paper recommendation systems.

7.1 Introduction

The rapidly increasing number of publications leads to a large quantity of possibly relevant papers [6] for more specific tasks such as finding related papers [26], finding ones to read [104] or literature search in general to inspire new directions and understand the state-of-the-art approaches [43]. Overall researchers typically spend a large amount of time on searching for relevant related work [7]. Keyword based search options are insufficient to find relevant papers [9, 49, 104], they require some form of initial knowledge about a field. Oftentimes, users’ information needs are not explicitly specified [53] which impedes this task further.

To close this gap, a plethora of paper recommendation systems have been proposed recently [34, 36, 83, 99, 112]. These systems should fulfil different functions: for junior researchers systems should recommend a broad variety of papers, for senior ones the recommendations should align more with their already established interests [9] or help them discover relevant interdisciplinary research [95]. In general paper recommendation approaches positively affect researchers’ professional lives as they enable finding relevant literature more likely and faster [47].

As there are many different approaches, their objectives and assumptions are also diverse. A simple problem definition of a paper recommendation system could be the following: given one paper recommend a list of papers

fitting the source paper [64]. This definition would not fit all approaches as some specifically do not require any initial paper to be specified but instead observe a user as input [34]. Some systems recommend sets of publications fitting the queried terms only if they are observed together [57, 58], most of the approaches suggest a number of single publications as their result [34, 36, 83, 112]. Most approaches assume that all required data to run a system to be present already [34, 112] but some works [36, 83] explicitly crawl general publication information or even abstracts and keywords from the web.

In this literature review we observe papers recently published in the area of scientific paper recommendation between and including January 2019 and October 2021¹. We strive to give comprehensive overviews on their utilised methods as well as their datasets, evaluation measures and open challenges of current approaches. Our contribution is 4-fold:

- We propose a novel multidimensional characterisation of current paper recommendation approaches.
- We compile a list of recently used datasets in evaluations of paper recommendation approaches.
- We compile a list of recently used evaluation measures for paper recommendation.
- We analyse existing open challenges and identify current novel problems in paper recommendation which could be specifically helpful for future approaches to address.

In the following Section 7.2 we describe the general problem statement for paper recommendation systems before we dive into the literature review in Section 7.3. Section 7.4 gives insight into datasets used in current work. In the following Section 7.5 different definitions of relevance, relevance assessment as well as evaluation measures are analysed. Open challenges and objectives are discussed in detail in Section 7.6. Lastly Section 7.7 concludes this literature review.

7.2 Problem Statement

Over the years different formulations for a problem statement of a paper recommendation system have emerged. In general they should specify the

¹The most recent surveys [9, 55, 87] focusing on scientific paper recommendation appeared in 2019 such that this time frame is not yet covered.

input for the recommendation system, the type of recommendation results, the point in time when the recommendation will be made and which specific goal an approach tries to achieve. Additionally, the target audience should be specified.

As *input* we can either specify an initial paper [26], keywords [112], a user [34], a user and a paper [5] or more complex information such as user-constructed knowledge graphs [104]. Users can be modelled as a combination of features of papers they interacted with [17, 19], e.g. their clicked [24] or authored publications [20]. Papers can for example be represented by their textual content [83].

As *types of recommendation* we could either specify single (independent) papers [34] or a set of paper which is to be observed in its full form [58]. A study by Beierle et al. [16] found that existing digital libraries recommend between three and ten single papers, in their case the optimal number of suggestions to display users was five to six.

As for the *point in time*, most work focuses on immediate recommendation of papers. Only few approaches also consider delayed suggestion via newsletter for example [53].

In general, recommended papers should be relevant in one way or another to achieve certain *goals*. They could e.g. be related to an initial paper [26] or publications which should be read [104].

Different *target audiences* for example junior or senior researcher have different demands from paper recommendation systems [9]. Usually paper recommendation approaches target single users but there are also works which strive to recommend papers for sets of users [105, 106].

7.3 Literature Review

In this chapter we first clearly define the scope of our literature review (see Sect. 7.3.1) before we conduct a meta analysis on the observed papers (see Sect. 7.3.2). Afterwards our categorisation or lack thereof is discussed in depth (see Sect. 7.3.3), before we give short overviews of all paper recommendation systems we found (see Sect. 7.3.4) and some other relevant related work (see Sect. 7.3.5).

7.3.1 Scope

To the best of our knowledge the literature reviews by Bai et al. [9], Li and Zou [55] and Shahid et al. [87] are the most recent ones targeting the domain of scientific paper recommendation systems. They were accepted for

publication or published in 2019 so they only consider paper recommendation systems up until 2019 at most. We want to bridge the gap between papers published after their surveys were finalised and current work so we only focus on the discussion of publications which appeared between January 2019 and October 2021 when this literature search was conducted.

We conducted our literature search on the following digital libraries: ACM², dblp³, GoogleScholar⁴ and Springer⁵. Titles of considered publications had to contain either *paper*, *article* or *publication* as well as some form of *recommend*. Papers had to be written in English to be observed. We judged relevance of retrieved publications by observing titles and abstracts if the title alone did not suffice to assess their topical relevance. In addition to these papers found by systematically searching digital libraries, we also considered their referenced publications if they were from the specified time period and of topical fit. For all papers their date of first publication determines their publication year. E.g. for journal articles we consider the point in time when they were first published online instead of the data on which they were published in an issue, for conference articles we consider the date of the conference instead a later date when they were published online.

We refrain from including works in our study which do not identify as scientific paper recommendation systems such as Wikipedia article recommendation [66, 74, 80] or general news article recommendation [29, 40, 98]. Citation recommendation systems [69, 85, 118] are also out of scope of this literature review. Even though citation and paper recommendation can be regarded analogously [42] we argue the differing functions of citations [31] and tasks of these recommendation systems [63] should not be mixed with the problem of paper recommendation. We also consciously refrain from discussing the plethora of more area-independent recommender systems which could be adopted to the domain of scientific paper recommendation.

Our literature research resulted in 76 relevant papers. We found 12 manuscripts which do not present paper recommendation systems but are relevant works for the area nonetheless, they are discussed in Section 7.3.5. This left 64 publications describing paper recommendation systems for us to analyse in the following.

²<https://dl.acm.org/>

³<https://dblp.uni-trier.de/>

⁴<https://scholar.google.com/>

⁵<https://link.springer.com/>

Type	Venue	#p
Journal	IEEE Access	5
Journal	Scientometrics	2
Journal	PeerJ CS	2
Conference	WWW	2
Conference	ChineseCSCW	2
Conference	CSCWD	2

Table 7.1: Top most common venues where relevant papers were published together with their type and number of papers (#p).

7.3.2 Meta analysis

For papers within our scope we consider their publication year as stated in the citation information. Of the 64 relevant system papers, 21 were published in 2019, 23 were published in 2020 and 20 were published in 2021. On average each paper has 4.0625 authors (std. dev.=1.7036) and 12.5781 pages (std. dev.=9.2192). 34 of the papers appeared as conference papers, 27 papers were published in journals and there were two preprints which have not yet been published otherwise. There has been one master’s thesis within scope. The most common venues for publications were the ones depicted in Table 7.1. Some papers [70, 71, 72, 88, 89] described the same approach without modification or extension of the actual paper recommendation methodology e.g. by providing evaluations. This left us with 61 different paper recommendation systems to discuss.

7.3.3 Categorisation

Former Categorisation

The already mentioned three most recent [9, 55, 87] and one older but highly influential [14] literature reviews in scientific paper recommendation utilise different categorisations to group approaches. Beel et al. [14] categorise observed papers by their underlying recommendation principle in stereotyping, content-based filtering, collaborative filtering, co-occurrence, graph based, global relevance and hybrid models. Bai et al. [9] only utilise the classes content-based filtering, collaborative filtering, graph-based methods, hybrid methods and other models. Li and Zou [55] use the categories content-based recommendation, hybrid recommendation, graph-based recommendation and recommendation based on deep learning. Shahid et al. [87] label approaches by the criterion they identify relevant papers with: content, metadata, col-

laborative filtering and citations.

The four predominant categories thus are content-based filtering, collaborative filtering, graph-based and hybrid systems. Most of these categories are defined sharply but graph-based approaches are not always characterised concisely: *Content-based filtering* (CBF) methods are said to be ones where user interest is inferred by observing their historic interactions with papers [9, 14, 55]. Recommendations are composed by observing features of papers and users [5]. In *collaborative filtering* (CF) systems the preferences of users similar to a current one are observed to identify likely relevant publications [9, 14, 55]. Current users' past interactions need to be similar to similar users' past interactions [9, 14]. *Hybrid* approaches are ones which combine multiple types of recommendations [9, 14, 55].

Graph-based methods can be characterised in multiple ways. A very narrow definition only encompasses ones which observe the recommendation task as a link prediction problem or utilise random walk [5]. Another less strict definition identifies these systems as ones which construct networks of papers and authors and then apply some graph algorithm to estimate relevance [9]. Another definition specifies this class as one using graph metrics such as random walk with restart, bibliographic coupling or co-citation inverse document frequency [101]. Li and Zhou [55] abstain from clearly characterising this type of systems directly but give examples which hint that in their understanding of graph-based methods somewhere in the recommendation process, some type of graph information e.g. bibliographic coupling or co-citation strength, should be used. Beel et al. [14] as well as Bai et al. [9] follow a similar line, they characterise graph-based methods broadly as ones which build upon the existing connections in a scientific context to construct a graph network.

When trying to classify approaches by their recommendation type, we encountered some problems:

1. We have to refrain from only utilising the labels the works give themselves (see Table 7.2 for an overview of self-labels of works which do classify themselves). Works do not necessarily (clearly) state, which category they belong to [26, 46, 57]. Another problem with self-labelling is authors' individual definitions of categories while disregarding all possible ones (as e.g. seen with Afsar et al. [1] or Ali et al. [5]). Mis-definition or omitting of categories could lead to an incorrect classification.
2. When considering the broadest definition of graph-based methods many recent paper recommendation systems tend to belong to the class of hybrid methods. Most of the approaches [5, 43, 45, 46, 54, 83, 100, 112] utilise some type of graph structure information as part of the

Work	Label	c
[1]	knowledge-based	×
[3]	hybrid	✓
[4]	deep learning-based	✓
[5]	unified model	×
[17]	graph-based	✓
[19]	user-specific	×
[22]	hybrid	✓
[27]	graph-based	✓
[28]	active one-shot learning	×
[34]	collaborative filtering	✓
[36]	hybrid	✓
[38]	hybrid	✓
[41]	hybrid	✓
[42]	hybrid	✓
[43]	hybrid	✓
[52]	hybrid	✓
[54]	network-based	×
[56]	content-based	✓
[58]	graph-based	✓
[59]	neuro-collaborative filtering	×
[60]	meta-path based	×
[61]	heterogeneous graph representation based	×
[30]	social network-based	×
[65]	hybrid	✓
[67]	content-based	✓
[70, 71, 72]	content-based	✓
[79]	hybrid	✓
[81]	content-based	✓
[84]	collaborative filtering	✓
[83]	hybrid	✓
[88, 89]	in-text citation frequencies-based	×
[91]	hybrid	✓
[93]	content-based	✓
[99]	hybrid	✓
[101]	graph-based	✓
[103]	hybrid	✓
[108]	knowledge-aware path recurrent network	×
[104]	graph-based	✓
[105]	hybrid	✓
[106]	hybrid	✓
[112]	hybrid	✓
[113]	network	×
[117]	hybrid	✓

Table 7.2: Indications as what type of paper recommendation system works describe themselves with indication if the description is a common used label (c).

approach which would classify them as graph-based but as they also utilise historic user-interaction data or descriptions of paper features (see e.g. Li et al. [54] who describe their approach as network-based while using a graph structure, textual components and user profiles) which would render them as either CF or CBF also.

Thus we argue the former categories do not suffice to classify the particu-

larities of current approaches in a meaningful way. So instead, we introduce more dimensions by which systems could be grouped.

Novel Categorisation

Recent paper recommendation systems can be categorised in 20 different dimensions by data and methods, which are part of the approach:

- Personalisation (Personal.): The approach produces personalised recommendations.
- Input: The approach requires some form of input, either a paper (p), keywords (k), user (u) or something else, e.g. an advanced type of input (o). Hybrid forms are also possible. In some cases the input is not clearly specified throughout the paper so it is unknown (?).
- Title: The approach utilises titles of papers.
- Abstract: The approach utilises abstracts of papers.
- Keyword: The approach utilises keywords of papers.
- Text: The approach utilises some type of text of papers which is not clearly specified as titles, abstracts or keywords. In the evaluation this approach might utilise specified text fragments of publications.
- Citation: The approach utilises citation information, e.g. numbers of citations or co-references.
- Historic interaction (interaction): The approach uses some sort of historic user-interaction data, e.g. previously authored, cited or liked publications. An approach can only include historic user-interaction data if it also somehow contains user profiles.
- User profile (user): The approach constructs some sort of user profile or utilises profile information.
- Popularity: The approach utilises some sort of popularity indication, e.g. CORE rank, numbers of citations or number of likes.
- Key phrase: The approach utilises key phrases.
- Embedding: The approach utilises some sort of text or graph embedding technique, e.g. BERT or Doc2Vec.

- Topic model (TM): The approach utilises some sort of topic model, e.g. LDA.
- Knowledge graph (KG): The approach utilises or builds some sort of knowledge graph.
- Graph: The approach actively builds or directly uses a graph structure, e.g. a knowledge graph or scientific heterogeneous network. Utilisation of a neural network is not considered in this dimension.
- Meta-path (Path): The approach utilises meta-paths.
- Random Walk (with Restart) (RW(WR)): The approach utilises Random Walk or Random Walk with Restart.
- Advanced machine learning (AML): The approach utilises some sort of advanced machine learning component in its core such as a neural network. Utilisation of established embedding methods which themselves use neural networks (e.g. BERT) are not considered in this dimension.
- Crawling: The approach conducts some sort of web crawling step.
- Cosine similarity (cosine): The approach utilises cosine similarity at some point.

Of the observed paper recommendation systems, six were general systems or methods which were only applied on the domain of paper recommendation [3, 4, 22, 57, 113, 115]. Two were targeting explicit set-based recommendation of publications where only all papers in the set together satisfy users' information needs [57, 58], two recommend multiple papers [39, 67] (e.g. on a path [39]), all the other approaches focused on recommendation of k single papers. Only two approaches focus on recommendation of papers to user groups instead of single users [105, 106]. Only one paper [53] supports subscription-based recommendation of papers, all other approaches solely regarded a scenario in which papers were suggested straight away.

Table 7.3 classifies the observed approaches according to the afore discussed dimensions.

7.3.4 Paper Recommendation Systems

The 64 relevant works identified in our literature search are described in this section. We deliberately refrain from trying to structure the section by classifying papers by an arbitrary dimension and instead point to Table 7.3

Work	General		Data						Methods											
	Personal.	Input	Title	Abstract	Keyword	Text	Citation	Interaction	User	Popularity	Key phrase	Embedding	TM	KG	Graph	Path	RW(WF)	AML	Crawling	Cosine
[1]	•	u	•					•	•					•	•					
[2]		p	•	•						•									•	•
[3]	•	u				•		•	•											
[4]	•	u				•		•	•											
[5]	•	pu				•		•	•											
[17]	•	u				•		•	•						•					•
[20]	•	u	•		•			•	•						•					
[19]	•	u	•	•	•			•	•											•
[23]	•	k				•		•	•											
[22]	•	k			•			•	•											
[24]	•	ku	•					•	•									•		
[25]		u				•		•	•											
[26]		p	•	•				•	•	•										•
[27]		p				•		•	•	•					•					•
[28]	•	pu				•		•	•	•								•	•	•
[34]	•	u	•	•				•	•	•										•
[35]		p				•		•	•	•									•	•
[36]		p	•	•				•	•	•									•	•
[38]	•	u				•		•	•	•										•
[39]		p				•		•	•	•					•					•
[41]		p				•		•	•	•										•
[42]		p	•	•	•			•	•	•										•
[43]		p				•		•	•	•									•	•
[45]		p				•		•	•	•										•
[46]	•	u						•	•	•										•
[52]	•	u	•	•	•			•	•	•										•
[53]	•	k	•	•	•			•	•	•				•						•
[54]	•	u				•		•	•	•										•
[56]		k			•			•	•	•					•					•
[57]		k			•			•	•	•										•
[58]		k			•			•	•	•										•
[59]	•	u	•	•				•	•	•										•
[60]	•	u						•	•	•										•
[61]	•	u	•	•	•			•	•	•										•
[30]	•	pu	•	•	•			•	•	•										•
[65]	•	u	•	•	•			•	•	•										•
[67]	•	p	•	•	•			•	•	•										•
[70, 71, 72]	•	u				•		•	•	•										•
[79]	•	u	•	•	•			•	•	•										•
[81]		p			•			•	•	•										•
[84]		p						•	•	•										•
[83]		p	•	•	•			•	•	•					•					•
[88, 89]		p				•		•	•	•									•	•
[90]		p				•		•	•	•										•
[91]		p	•	•				•	•	•										•
[93]		k				•		•	•	•										•
[99]	•	u	•	•	•			•	•	•										•
[101]		p				•		•	•	•										•
[102]		p				•		•	•	•										•
[103]		p				•		•	•	•										•
[108]		u			•			•	•	•										•
[104]	•	ko	•	•	•			•	•	•										•
[105]	•	u				•		•	•	•										•
[106]	•	u			•			•	•	•										•
[110]	•	ku			•			•	•	•										•
[111]		pk	•	•	•			•	•	•										•
[112]		k	•	•				•	•	•										•
[113]	•	u						•	•	•										•
[115]		?	•	•				•	•	•										•
[116]		?	•	•				•	•	•										•
[117]	•	u	•	•				•	•	•										•

Table 7.3: Indications whether works utilise the specific data or methods. Papers describing the same approach without extension of the methodology (e.g. only describing more details or an evaluation) are regarded in combination with each other.

to identify those dimensions in which a reader is interested to navigate the following short descriptions. The works are ordered by the surname of the first author and ascending publication year. An exception to this rule are papers presenting extensions of previous approaches with different first authors. These papers are ordered to their preceding approaches.

Afsar et al. [1] propose KERS, a multi armed bandit approach for patients to help with medical treatment decision making. It consists of two phases: first an exploration phase identifies categories users are implicitly interested in. This is supported by an expert-built knowledge base. Afterwards an exploitation phase takes place where articles from these categories are recommended until a user's focus changes and another exploitation phase is initiated. The authors strive to minimise the exploration efforts while maximising users' satisfaction.

Ahmedi et al. [3] propose a personalised approach which can also be applied to more general recommendation scenarios which include user profiles. They utilise Collaborative Topic Regression to mine association rules from historic user interaction data.

Alfarhood and Cheng [4] introduce Collaborative Attentive Autoencoder, a deep learning-based model for general recommendation targeting the data sparsity problem. They apply probabilistic matrix factorisation while also utilising textual information to train a model which identifies latent factors in users and papers.

Ali et al. [5] construct PR-HNE, a personalised probabilistic paper recommendation model based on a joint representation of authors and publications. They utilise graph information such as citations as well as co-authorships, venue information and topical relevance to suggest papers. They apply SBERT and LDA to represent author embeddings and topic embeddings respectively.

Bereczki [17] models users and papers in a bipartite graph. Papers are represented by their contents' Word2Vec or BERT embeddings, users' vectors consist of representations of papers they interacted with. These vectors are then aggregated with simple graph convolution.

Bulut et al. [20] focus on current user interest in their approach which utilises k-Means and KNN. Users' profiles are constructed from their authored papers. Recommended papers are the highest cited ones from the cluster most similar to a user. In a subsequent work they extended their research group to again work in the same domain. Bulut et al. [19] again focus on users' features. They represent users as the sum of features of their papers. These representations are then compared with all papers' vector representations to find the most similar ones. Papers can be represented by TF-IDF, Word2Vec or Doc2Vec vectors.

Chaudhuri et al. [23] use indirect features derived from direct features of papers in addition to direct ones in their paper recommendation approach: keyword diversification, text complexity and citation analysis. In an extended group Chaudhuri et al. [24] later propose usage of more indirect features such as quality in paper recommendation. Users' profiles are composed of their clicked papers. Subsequently they again worked on an approach in the same area but in a slightly smaller group. Chaudhuri et al. [22] propose the general Hybrid Topic Model and apply it on paper recommendation. It learns users' preferences and intentions by combining LDA and Word2Vec. They compute user's interest from probability distributions of words of clicked papers and dominant topics in publications.

Chen and Ban [25] introduce CPM, a recommendation model based on topically clustered user interests mined from their published papers. They derive user need models from these clusters by using LDA and pattern equivalence class mining. Candidate papers are then ranked against the user need models to identify the best-fitting suggestions.

Collins and Beel [26] propose the usage of their paper recommendation system Mr. DLib as a recommender as-a-service. They compare representing papers via Doc2Vec with a key phrase-based recommender and TF-IDF vectors.

Du et al. [27] introduce HNPR, a heterogeneous network method using two different graphs. The approach incorporates citation information, co-author relations and research areas of publications. They apply random walk on the networks to generate vector representations of papers.

Du et al. [28] propose Polar++, a personalised active one-shot learning-based paper recommendation system where new users are presented articles to vote on before they obtain recommendations. The model trains a neural network by incorporating a matching score between a query article and the recommended articles as well as a personalisation score dependant on the user.

Guo et al. [34] recommend publications based on papers initially liked by a user. They learn semantics between titles and abstracts of papers on word- and sentence-level, e.g. with Word2Vec and LSTMs to represent user preferences.

Habib and Afzal [35] crawl full texts of papers from CiteSeer. They then apply bibliographic coupling between input papers and a clusters of candidate papers to identify the most relevant recommendations. In a subsequent work Afzal again used a similar technique. Ahmad and Afzal [2] crawled papers from CiteSeerX. Cosine similarity of TF-IDF representations of key terms from titles and abstracts is combined with co-citation strength of paper pairs. This combined score then ranks the most relevant papers the highest.

Haruna et al. [36] incorporate paper-citation relations combined with contents of titles and abstracts of papers to recommend the most fitting publications for an input query corresponding to a paper.

Hu et al. [38] present ADRCR, a paper recommendation approach incorporating author-author and author-paper citation relationships as well as authors' and papers' authoritativeness. A network is built which uses citation information as weights. Matrix decomposition helps learning the model.

Hua et al. [39] propose PAPR which recommends relevant paper sets as an ordered path. They strive to overcome recommendation merely based on similarity by observing topics in papers changing over time. They combine similarities of TF-IDF paper representations with random-walk on different scientific networks.

Jing and Yu [41] build a three-layer graph model which they traverse with random-walk with restart in an algorithm named PAFRWR. The graph model consists of one layer with citations between papers' textual content represented via Word2Vec vectors, another layer modelling co-authorships between authors and the third layer encodes relationships between papers and topics contained in them.

Kanakia et al. [42] build their approach upon the MAG dataset and strive to overcome the common problems of scalability and cold-start. They combine TF-IDF and Word2Vec representations of the content with co-citations of papers to compute recommendations. Speedup is achieved by comparing papers to clusters of papers instead of all other single papers.

Kang et al. [43] crawl full texts of papers from CiteSeer and construct citation graphs to determine candidate papers. Then they compute a combination of section-based citation and key phrase similarity to rank recommendations.

Kong et al. [45] present VOPRec, a model combining textual components in form of Doc2vec and Paper2Vec paper representations with citation network information in form of Struc2vec. Those networks of papers connect the most similar publications based on text and structure. Random walk on these graphs contributes to the goal of learning vector representations.

L et al. [46] base their recommendation on lately accessed papers of users as they assume future accessed papers are similar to recently seen ones. They utilise a sliding window to generate sequences of papers, on those they construct a GNN to aggregate neighbouring papers to identify users' interests.

Li et al. [53] introduce a subscription-based approach which learns a mapping between users' browsing history and their clicks in the recommendation mails. They learn a re-ranking of paper recommendations by using its meta-data, recency, word representations and entity representations by knowledge graphs as input for a neural network. Their defined target audience are new

users.

Li et al. [52] present HNTA a paper recommendation method utilising heterogeneous networks and changing user interests. Paper similarities are calculated with Word2Vec representations of words recommended for each paper. Changing user interest is modelled with help of an exponential time decay function on word vectors.

Li et al. [54] utilise user profiles with a history of preferences to construct heterogeneous networks where they apply random walks on meta-paths to learn personalised weights. They strive to discover user preference patterns and model preferences of users as their recently cited papers.

Lin et al. [56] utilise authors' citations and years they have been publishing papers in their recommendation approach. All candidate publications are matched against user-entered keywords, the two factors of authors of these candidate publications are combined to identify the overall top recommendations.

Liu et al. [57] explicitly do not require all recommended publications to fit the query of a user perfectly. Instead they state the set of recommended papers fulfils the information need only in the complete form. Here they treat paper recommendation as a link prediction problem incorporating publishing time, keywords and author influence. In a subsequent work, part of the previous research group again observes the same problem. In this work Liu et al. [58] propose an approach utilising numbers of citations (author popularity) and relationships between publications in an undirected citation graph. They compute Steiner trees to identify the sets of papers to recommend.

Lu et al. [59] propose TGMF-FMLP, a paper recommendation approach focusing on the changing preferences of users and novelty of papers. They combine category attributes (such as paper type, publisher or journal), a time-decay function, Doc2Vec representations of the papers' content and a specialised matrix factorisation to compute recommendations.

Ma et al. [61] introduce HIPRec, a paper recommendation approach on heterogeneous networks of authors, papers, venues and topics specialised on new publications. They use the most interesting meta-paths to construct significant meta-paths. With these paths and features from these paths they train a model to identify new papers fitting users. Together with another researcher Ma further pursued this research direction. Ma and Wang [60] propose HGRec, a heterogeneous graph representation learning-based model working on the same network. They use meta-path-based features and Doc2Vec paper embeddings to learn the node embeddings in the network.

Manju et al. [30] attempt to solve the cold-start problem with their paper recommendation approach coding social interactions as well as topical

relevance into a heterogeneous graph. They incorporate believe propagation into the network and compute recommendations by applying random walk.

Mohamed Hassan et al. [65] adopt an existing tag prediction model which relies on a hierarchical attention network to capture semantics of papers. Matrix factorisation then identifies the publications to recommend.

Nair et al. [67] propose C-SAR, a paper recommendation approach using a neural network. They input GloVe embeddings of paper titles into their Gated Recurrent Union model to compute probabilities of similarities of papers. The resulting adjacency matrix is input to an association rule mining a priori algorithm which generates the set of recommendations.

Nishioka et al. [70, 71] state serendipity of recommendations as their main objective. They incorporate users' tweets to construct profiles in hopes to model recent interests and developments which did not yet manifest in users' papers. They strive to diversity the list of recommended papers. In more recent work Nishioka et al. [72] explained their evaluation more in depth.

Rahdari and Brusilovsky [79] observe paper recommendation for participants of scientific conferences. Users' profiles are composed of their past publications. Users control the impact of features such as publication similarity, popularity of papers and its authors to influence the ordering of their suggestions.

Renuka et al. [81] propose a paper recommendation approach utilising TF-IDF representations of automatically extracted keywords and key phrases. They then either use cosine similarity between vectors or a clustering method to identify the most similar papers for an input paper.

Sakib et al. [84] present a paper recommendation approach utilising second-level citation information and citation context. They strive to not rely on user profiles in the paper recommendation process. Instead they measure similarity of candidate papers to an input paper based on co-occurred or co-occurring papers. In a follow-up work with a bigger research group Sakib et al. [83] combine contents of titles, keywords and abstracts with their previously mentioned collaborative filtering approach. They again utilise second-level citation relationships between papers to find correlated publications.

Shahid et al. [89] utilise in-text citation frequencies and assume a reference is more important to a referencing paper the more often it occurs in the text. They crawl papers from CiteSeerX to retrieve the top 500 citing papers. In a follow-up work with a partially different research group Shahid et al. [88] evaluate the previously presented approach with a user study.

Sharma et al. [90] propose IBM PARSe, a paper recommendation system for the medical domain to reduce the number of papers to review for keeping an existing knowledge graph up-to-date. Classifiers identify new papers from target domains, named entity recognition finds relevant medical concepts

before papers' TF-IDF vectors are compared to ones in the knowledge graph. New publications most similar to already relevant ones with matching entities are recommended to be included in the knowledge base.

Subathra and Kumar [93] constructed an paper recommendation system which applies LDA on Wikipedia articles twice. Top related words are computed using pointwise mutual information before papers are recommended for these top words.

Tang et al. [99] introduce CGPrec, a content-based and knowledge graph-based paper recommendation system. They focus on users' sparse interaction history with papers and strive to predict papers on which users are likely to click. They utilise Word2Vec and a Double Convolutional Neural Network to emulate users' preferences directly from paper content as well as indirectly by using knowledge graphs.

Tanner et al. [101] consider relevance and strength of citation relations to weigh the citation network. They fetch citation information from the parsed full texts of papers. On the weighted citation networks they run either weighted co-citation inverse document frequency, weighted bibliographic coupling or random walk with restart to identify the highest scoring papers.

Tao et al. [102] use embeddings and topic modelling to compute paper recommendations. They combine LDA and Word2Vec to obtain topic embeddings. Then they calculate most similar topics for all papers using Doc2Vec vector representations and afterwards identify the most similar papers. With PageRank on the citation network they re-rank these candidate papers.

Waheed et al. [103] propose CNRN, a recommendation approach using a multilevel citation and authorship network to identify recommendation candidates. From these candidate papers ones to recommend are chosen by combining centrality measures and authors' popularity. Highly correlated but unrelated Shi et al. [91] present AMHG, an approach utilising a multilayer perceptron. They also construct a multilevel citation network as described before with added author relations. Here they additionally utilise vector representations of publications and recency.

Wang et al. [108] introduce a knowledge-aware path recurrent network model. An LSTM mines path information from the knowledge graphs incorporating papers and users. Users are represented by their downloaded, collected and browsed papers, papers are represented by TF-IDF representations of their keywords.

Wang et al. [104] require users to construct knowledge graphs to specify the domain(s) and enter keywords for which recommended papers are suggested. From the keywords they compute initially selected papers. They apply Doc2Vec and emotion-weighted similarity between papers to identify recommendations.

Wang et al. [105] regard paper recommendation targeting a group of people instead of single users and introduce GPRAH_ER. They employ a two-step process which first individually predicts papers for users in the group before recommended papers are aggregated. Here users in the group are not considered equal, different importance and reliability weights are assigned such that important persons' preferences are more decisive of the recommended papers. Together with a different research group two authors again pursued this definition of the paper recommendation problem. Wang et al. [106] recommend papers for groups of users in an approach called GPMF_ER. As with the previous approach they compute TF-IDF vectors of keywords of papers to calculate most similar publications for each user. Probabilistic matrix factorisation is used to integrate these similarities in a model such that predictive ratings of all users and papers can be obtained. In the aggregation phase the number of papers read by a user is determined to replace the importance component.

Xie et al. [111] propose JTIE, an approach incorporating contents, authors and venues of papers to learn paper embeddings. Further, directed citation relations are included into the model. Based on users' authored and referenced papers personalised recommendations are computed. They consider explainability of recommendations. In a subsequent work part of the researchers again work on this topic. Xie et al. [110] specify on recommendation of papers from different areas for user-provided keywords or papers. They use hierarchical LDA to model evolving concepts of papers and citations as evidence of correlation in their approach.

Yang et al. [112] incorporate the age of papers and impact factors of venues as weights in their citation network-based approach named PubTeller. Papers are clustered by topic, the most popular ones from the clusters most similar to the query terms are recommendation candidates. In this approach, LDA and TF-IDF are used to represent publications.

Yu et al. [113] propose ICMN, a general collaborative memory network approach. User and item embeddings are composed by incorporating papers' neighbourhoods and users' implicit preferences.

Zhang et al. [115] propose W-Rank, a general approach weighting edges in a heterogeneous author, paper and venue graph by incorporating citation relevance and author contribution. They apply their method on paper recommendation. Network- (via citations) and semantic-based (via AWD) similarity between papers is combined for weighting edges between papers, harmonic counting defines weights of edges between authors and papers. A HITS-inspired algorithm computes the final authority scores. In a subsequent work in a slightly smaller group they focus on a specialised approach for paper recommendation. Here Zhang et al. [116] strive to emulate a hu-

man expert recommending papers. They construct a heterogeneous network with authors, papers, venues and citations. Citation weights are determined by semantic- and network-level similarity of papers. Lastly, recommendation candidates are re-ranked while combining the weighted heterogeneous network and recency of papers.

Zhao et al. [117] present a personalised approach focusing on diversity of results which consists of three parts. First LFM extracts latent factor vectors of papers and users from the users' interactions history with papers. Then BERT vectors are constructed for each word of the papers, with those vectors as input and the latent factor vectors as label a BiGRU model is trained. Lastly, diversity and a user's rating weights determine the ranking of recommended publications for the specific user.

7.3.5 Other relevant Work

We now briefly discuss some papers which did not present novel paper recommendation approaches but are relevant in the scope of this literature review nonetheless.

Surrounding Paper Recommendation

Here we present two works which could be classified as ones to use on top of or in combination with existing paper recommendation systems: Lee et al. [48] introduce LIMEADE, a general approach for opaque recommendation systems which can for example be applied on any paper recommendation system. They produce explanations for recommendations as a list of weighted interpretable features such as influential paper terms.

Beierle et al. [16] use the recommendation-as-a-service provider Mr. DLib to analyse choice overload in user evaluations. They report several click-based measures and discuss effects of different study parameters on engagement of users.

(R)Evaluations

The following three works can be grouped as ones which provide (r)evaluations of already existing approaches. Their results could be useful for the construction of novel systems: Ostendorff [73] suggests considering the context of paper similarity in background, methodology and findings sections instead of undifferentiated textual similarity for scientific paper recommendation.

Mohamed Hassan et al. [64] compare different text embedding methods such as BERT, ELMo, USE and InferSent to express semantics of papers. They perform paper recommendation and re-ranking of recommendation candidates based on cosine similarity of titles.

Le et al. [47] evaluate the already existing paper recommendation system Mendeley Suggest, which provides recommendations with different collaborative or content-based approaches. They observe different usage behaviours and state utilisation of paper recommendation systems does positively effect users' professional lives.

Living Labs

Living labs help researchers conduct meaningful evaluations by providing an environment, in which recommendations produced by experimental systems are shown to real users in realistic scenarios [12]. We found three relevant works for the area of scientific paper recommendation: Beel et al. [12] proposed a living lab for scholarly recommendation built on top of Mr. DLib, their recommender-as-a-service system. They log users' actions such as clicks, downloads and purchases for related recommended papers. Additionally, they plan to extend their living lab to also incorporate research grant or research collaborator recommendation.

Gingstad et al. [33] propose ArXivDigest, an online living lab for explainable and personalised paper recommendations from arXiv. Users can either be suggested papers while browsing their website or via email as a subscription-type service. Different approaches can be hooked into ArXivDigest, the recommendations generated by them can then be evaluated by users. A simple text-based baseline compares user-input topics with articles. Target values of evaluations are users' clicked and saved papers.

Schaer et al. [86] held the Living Labs for Academic Search (LiLAS) where they hosted two shared tasks: dataset recommendation for scientific papers and ad-hoc multi-lingual retrieval of most relevant publications regarding specific queries. To overcome the gap between real-world and lab-based evaluations they allowed integrating participants' systems into real-world academic search systems, namely LIVIO and GESIS Search.

Multilingual/Cross-lingual Recommendation

The previous survey by Li and Zhou [55] identifies cross-language paper recommendation as a future research direction. The following two works could be useful for this aspect: Keller and Munz [44] present their results of participating on the CLEF LiLAS challenge where they tackled recommendation of

multilingual papers based on queries. They utilised a pre-computed ranking approach, Solr and pseudo-relevance feedback to extend queries and identify fitting papers.

Safaryan et al. [82] compare different already existing techniques for cross-language recommendation of publications. They compare word by word translation, linear projection from a Russian to an English vector representation, VecMap alignment and MUSE word embeddings.

Related Recommendation Systems

Some recommendation approaches are slightly out of scope of pure paper recommendation systems but could still provide inspiration or relevant results: Ng [68] proposes CRec, a children’s book recommendation system utilising matrix factorisation. His goal is to encourage good reading habits of children. The approach combines readability levels of users and books with TF-IDF representations of books to find ones which are similar to ones which a child may have already liked.

Patra et al. [75] recommend publications relevant for datasets to increase reusability. Those papers could describe the dataset, use it or be related literature. The authors represent datasets and articles as vectors and use cosine similarity to identify the best fitting papers. Re-ranking them with usage of Word2Vec embeddings results in the final recommendation.

7.4 Datasets

As the discussed paper recommendation systems utilise different inputs or components of scientific publications and pursue slightly different objectives, datasets to experiment on are also of diverse nature. We do not consider datasets of approaches which do not evaluate [57] or do not evaluate the actual paper recommendation [2, 23, 35, 79, 81]. We also do not discuss datasets where only the data sources are mentioned but no remarks are made regarding the size or composition of the dataset [19, 99] or ones where we were not able to identify actual numbers [30]. Table 7.4 gives an overview of datasets used in the evaluation of the considered discussed methods. Many of the datasets are unavailable only few years after publication of the approach. Most approaches utilise their own modified version of a public dataset which makes exact replication of experiments hard. In the following the main underlying data sources and publicly available datasets are discussed. Non-publicly available datasets are briefly described in Table 7.5.

Name	A?	Used by
DBLP + Citations v1 [100]	✓	Yang et al. [112]
DBLP + Citations v8 [100]	×	Ma and Wang [60], Ma et al. [61]
DBLP + Citations v11	✓	Ali et al. [5]
dblp + IEEE + ACM + Pubmed	×	Bulut et al. [20]
DBLP paths	×	Hua et al. [39]
DBLP-Citation-network f. AMiner	×	Jing and Yu [41]
dblp	×	Li et al. [54]
DBLP-REC	×	Shi et al. [91]
dblp + AMiner KG	×	Wang et al. [108]
dblp + AMiner + venue	×	Xie et al. [111]
SPRD.Senior	✓	Chen and Ban [25]
SPRD [96]	✓	Haruna et al. [36], Sakib et al. [84], Sakib et al. [83]
Citeulike-a [107]	✓	Ahmedi et al. [3], Alfarhood and Cheng [4], Guo et al. [34], L et al. [46], Mohammed Hassan et al. [65], Tang et al. [99], Yu et al. [113], Zhao et al. [117]
Citeulike-t [107]	✓	Alfarhood and Cheng [4]
Citeulike.huge	×	Lu et al. [59]
Citeulike.medium	×	Wang et al. [105]
Citeulike.tiny	×	Wang et al. [106]
ACM paths	×	Hua et al. [39]
ACM citation network V8	×	Nishioka et al. [70], Nishioka et al. [71], Nishioka et al. [72]
Scopus.tiny	×	Chaudhuri et al. [22, 24]
ScienceDirect+Scopus	×	Li et al. [53]
Scopus	×	Xie et al. [110]
AMiner	×	Li et al. [54]
AMiner + Wanfang	×	Du et al. [27]
AMiner.tiny	×	Du et al. [28]
AMiner.huge	×	Waheed et al. [103]
ACM C-D	×	Xie et al. [110]
AAN.original [78]	✓	Nair et al. [67]
AAN.modified	×	Ali et al. [5], L et al. [46]
AAN.tiny	×	Tanner et al. [101]
Sowiprot	×	Collins and Beel [26]
RARD.tiny	×	Du et al. [28]
CiteSeer	×	Kang et al. [43]
CiteSeer.tiny	×	Shahid et al. [89]
CiteSeer.medium	×	Shahid et al. [87]
Patents.tiny	×	Du et al. [28]
Patents	×	Xie et al. [111]
ACM H-I	×	Xie et al. [110]
Hep-TH graph	×	Liu et al. [58]
arXiv Hep-TH	×	Zhang et al. [115]
MSA	×	Yang et al. [112]
MAG 2017	×	Zhang et al. [115]
MAG 2018	×	Kanakia et al. [42]
BBC	✓	Afsar et al. [1]
PRSDataset	✓	Guo et al. [34], L et al. [46]
Physical Review A	×	Kong et al. [45]
ACL selection network	×	Tao et al. [102]
prostate cancer	×	Afsar et al. [1]
Peltarion	×	Bereczki [17]
Jabref	×	Collins and Beel [26]
DM	×	Hu et al. [38]
Graphs	×	L et al. [46]
SCHOLAT	×	Li et al. [52]
IEEE Xplore	×	Lin et al. [56]
KGs	×	Wang et al. [104]
Wanfang	×	Kang et al. [43]
Watson TM for Genomics	×	Sharma et al. [90]
Wikipedia	×	Subathra and Kumar [93]
LibraryThing	×	Zhao et al. [117]

Table 7.4: Overview of datasets utilised in most recent related work with (unofficial) names, public availability of the possibly modified dataset which was used (A?), and a list of papers it was used in. Datasets are grouped by their underlying data source if possible.

DBLP + Citations v8 [100]	[60, 61]	2,133 <i>p</i> from 20 <i>v</i> from 2000 to 2016, 39,530 <i>a</i> , 15,708 <i>p</i> topics
dblp + IEEE + ACM + Pubmed	[20]	sources: dblp, IEEE, ACM, Pubmed. 3,394,616 <i>p</i> (titles), <i>a</i> , publication years, keywords, <i>r</i>
DBLP paths	[39]	1,782,700 <i>p</i> (titles, abstracts, keywords), 2,052,414 <i>a</i> , 18,936 <i>v</i> , 100,000 <i>t</i> , 9,590,600 <i>i</i>
DBLP-Citation-network f. AMiner	[41]	63,469 <i>p</i> from 2013 to 2019, 152,586 <i>a</i>
dblp	[54]	2,126,267 <i>p</i> , 8686 <i>v</i> , 1,221,259 <i>a</i> , 256,214 <i>t</i> , 3765 <i>u</i> relations
DBLP-REC	[91]	DBLP-Citation-network v11 + ScienceDirect + IEEE, 3,590,853 <i>p</i> , 3,276,803 <i>a</i> , 35,254,530 <i>c</i>
dblp + AMiner KG	[108]	KG with 223,431 <i>a</i> , 337,561 <i>p</i> , 5578 <i>v</i> , 1179 keyword nodes, 16,328,642 <i>c</i>
dblp + AMiner + venue	[111]	3,056,388 <i>p</i> (titles, abstracts, keywords), 1,752,401 <i>a</i> , 354,693 keywords, 11,397 <i>v</i> , <i>c</i> , discipline labels
Name	Used by	Description
Citeulike_huge	[59]	210,137 <i>p</i> , 3,039 <i>u</i> , 284,960 <i>u-p</i> <i>i</i> from Nov 2004 to Dec 2007
Citeulike_medium	[105]	2,065 users, 718 groups, 85,542 <i>p</i>
Citeulike_tiny	[106]	1,659 users, 718 groups, 82,376 <i>p</i> , 198,744 <i>i</i>
ACM paths	[39]	2,385,057 <i>p</i> (titles, abstracts, keywords), 2,004,398 <i>a</i> , 269,467 <i>v</i> , 61,618 <i>t</i> , 12,048,682 <i>i</i>
ACM citation network V8	[70, 71, 72]	1,669,237 <i>p</i> (titles, abstracts), <i>v</i> , <i>a</i>
Scopus_tiny	[22, 24]	2,000 <i>p</i>
ScienceDirect + Scopus	[53]	<i>u</i> 's browsed <i>p</i> prior to first email from ScienceDirect, <i>p</i> metadata from Scopus, 4,392 recommendation sessions (emails with clicks on <i>p</i> , <i>u</i> ' browsing history)
Scopus	[110]	528,224 <i>p</i> , <i>a</i> , <i>r</i> , discipline tags
Scopus + venue	[111]	1,304,907 <i>p</i> (titles, abstracts, keywords), 482,602 <i>a</i> , 127,630 keywords, 7653 <i>v</i> , <i>c</i> , discipline labels
AMiner	[54]	2,070,699 <i>p</i> , 263,250 <i>v</i> , 1,557,147 <i>a</i> , 735,059 <i>t</i> , 9398 <i>u</i> relations
AMiner + Wanfang	[27]	4 mio <i>p</i> . 3 sets: data from 2018 and 2019 (221,076 <i>p</i> , 503,945 <i>a</i>), mathematical analysis (98,702 <i>p</i> , 117,183 <i>a</i>), image processing (49,098 <i>p</i> , 107,290 <i>a</i>)
AMiner_tiny	[28]	188 input <i>p</i> , 10 candidate <i>p</i> for each input
AMiner_huge	[103]	2,092,356 <i>p</i> , 1,712,433 <i>a</i> , 8,024,869 <i>c</i> , 4,258,615 co-autorships
ACM C-D	[110]	43,380 <i>p</i> from AMiner, <i>a</i> , ACM CSS tags
AAN_modified	[5, 46]	21,455 <i>p</i> from 312 <i>v</i> from NLP, 17,342 <i>a</i> , 113,367 <i>c</i>
AAN_tiny	[101]	2082 <i>p</i> (ids, titles, publication year), 8194 <i>c</i> , avg. 7.87 <i>c</i> per <i>p</i> , <i>a</i> , <i>v</i>
Sowiport	[26]	<i>u</i> <i>i</i> data from Mar 2017 to Oct 2018, 0.1% click-through rate
RARD_tiny	[28]	800 input <i>p</i> from Related-Article Recommendation Dataset from Sowiport [11]
CiteSeer	[43]	1,100 <i>p</i> , 10 sets of relevant <i>p</i>
CiteSeer_tiny	[89]	400 <i>c</i> -pairs, 1,230 <i>c</i> contexts
CiteSeer_medium	[87]	10 <i>p</i> , 226 <i>c</i> -pairs
Patents.tiny	[28]	67 input patents, 20 candidate patents for each input
Patents	[111]	182,260 patents, 73,974 <i>a</i>
ACM H-I	[110]	70,090 patents with ownership from 2017, <i>r</i> , ACM CSS tags
Hep-TH graph	[58]	graph with 8,721 <i>p</i> (keywords)
arXiv Hep-TH	[115]	~29,000 <i>p</i> , 350,000 <i>c</i> , 14,909 <i>a</i> , 428 journals
MSA	[112]	101,205 <i>p</i> , 190,146 <i>c</i> in 300 conferences
MAG 2017	[115]	based on data until 2017, area: intrusion detection in cyber security, 6428 <i>p</i> , 94,887 <i>c</i> , 18,890 <i>a</i> , 6428 journals
MAG 2018	[42]	based on MAG Azure database from Oct 2018, 206,676,892 <i>p</i>
Physical Review A	[45]	393 <i>p</i> from 2007 to 2009 with 2,664 <i>c</i> from American Physical Society
ACL selection network	[102]	18,718 <i>p</i> (titles, summaries) from ACL proceedings
prostate cancer	[1]	500 <i>p</i> tagged with 5 categories
Peltarion	[17]	290 <i>p</i> , <i>u</i> <i>i</i> from Dec 2018 to May 2021 of <i>u</i> of Peltarion Knowledge Center who have read ≥ 5 <i>p</i>
Jabref	[26]	<i>u</i> <i>i</i> data from Mar 2017 to Oct 2018, 0.22% click-through rate
DM	[38]	8,301 <i>p</i> from journals: DMKD, TKDE + conferences: KDD, ICDM, SDM
Graphs	[46]	Cora (1 graph, 2.7k nodes), TU-IMDB (1.5k graphs, 13 nodes each), TU-MUTAG (188 molecules, 18 nodes)
SCHOLAT	[52]	34,518 <i>p</i> (titles, abstracts, keywords), <i>a</i>
IEEE Xplore	[56]	3 <i>p</i> (keywords), <i>r</i> , <i>a</i> appeared in IEEE between 2010 and 2017
KGs	[104]	knowledge graphs, 600 <i>p</i> from information retrieval + machine learning
Wanfang	[43]	500 <i>p</i> , 5 sets of relevant <i>p</i>
Watson TM for Genomics	[90]	15,320 <i>p</i> from top 10 percentile genomics journals from Jun 2016
Wikipedia	[93]	1000 <i>p</i> from Wikipedia, 20 topics
LibraryThing	[117]	120,150 books (titles, abstracts), <i>u</i> , 185,210 favourites records, 150,216 ratings, 139,530 reviews of 12,350 <i>u</i>

Table 7.5: Description of private datasets utilised in most recent related work with (unofficial) names. Datasets are grouped by their underlying data source if possible. We used the following abbreviations: user(s) *u*, paper(s) *p*, interaction(s) *i*, author(s) *a*, venue(s) *v*, reference(s) *r*, citation(s) *c*, term(s) *t*.

7.4.1 dblp based datasets

The dblp computer science bibliography (dblp) is a digital library offering metadata on authors, papers and venues from the area of computer science and adjacent fields [51]. They provide publicly available short-time stored daily and longer-time stored monthly data dumps⁶.

The *dblp + Citations v1* dataset [100] builds upon a dblp version from 2010 mapped on AMiner. It contains 1,632,442 publications with 2,327,450 citations.

The *dblp + Citations v11* dataset⁷ builds upon dblp. It contains 4,107,340 papers, 245,204 authors, 16,209 venues and 36,624,464 citations

Descriptions of non-public datasets based on dblp (*dblp + IEEE + ACM + Pubmed*, *DBLP paths*, *DBLP-Citation-network f. AMiner*, *dblp*, *DBLP + Citations v8*, *DBLP-REC*, *dblp + AMiner KG*, *dblp + AMiner + venue*) can be found in Table 7.5.

7.4.2 SPRD based datasets

The Scholarly Paper Recommendation Dataset (SPRD)⁸ was constructed by collecting publications written by 50 researchers of different seniority from the area of computer science which are contained in dblp from 2000 to 2006 [96, 97, 55]. The dataset contains 100,351 candidate papers extracted from the ACM Digital Library as well as citations and references for papers. Relevance assessments of papers relevant to their current interests of the 50 researchers are also included.

A subset of SPRD, *SPRD_Senior*, which contains only the data of senior researchers can also be constructed [94].

7.4.3 CiteULike based datasets

CiteULike [18] was a social bookmarking site for scientific papers. It contained papers and their metadata. Users were able to include priorities, tags or comments for papers on their reading list. There were daily data dumps available from which datasets could be constructed.

Citeulike-a [107]⁹ contains 5,551 users, 16,980 papers with titles and abstracts from 2004 to 2006 and their 204,986 interactions between users and papers. Papers are represented by their title and abstract.

⁶<https://dblp.uni-trier.de/xml/>

⁷<https://www.aminer.org/citation>

⁸<https://www.db.soc.i.kyoto-u.ac.jp/~sugiyama/SchPaperRecData.html>

⁹<https://github.com/js05212/citeulike-a>

Citeulike-t [107]¹⁰ contains 7,947 users, 25,975 papers and 134,860 user-paper interactions. Papers are represented by their pre-processed title and abstract.

The description of a non-public dataset based on CiteULike (*Citeulike_huge*, *Citeulike_medium*, *Citeulike_tiny*) can be found in Table 7.5.

7.4.4 ACM based datasets

The ACM Digital Library (ACM) is a semi-open digital library offering information on scientific authors, papers, citations and venues from the area of computer science¹¹. They offer an API to query for information.

Descriptions of non-public datasets based on ACM (*ACM paths*, *ACM citation network V8*) can be found in Table 7.5.

7.4.5 Scopus based datasets

Scopus is a semi-open digital library containing metadata on authors, papers and affiliations in different scientific areas¹². They offer an API to query for data.

Descriptions of non-public datasets based on Scopus (*Scopus_tiny*, *ScienceDirect + Scopus*, *Scopus*, *Scopus + venue*) can be found in Table 7.5.

7.4.6 AMiner based datasets

ArnetMiner (AMiner) [100] is an open academic search system modelling the academic network consisting of authors, papers and venues from all areas¹³. They provide an API to query for information.

Descriptions of non-public datasets based on AMiner (*AMiner*, *AMiner + Wanfang*, *AMiner_tiny*, *AMiner_huge*, *ACM C-D*) can be found in Table 7.5.

7.4.7 AAN based datasets

The ACL Anthology Network (AAN) [76, 77, 78] is a networked database containing papers, authors and citations from the area of computational linguistics¹⁴. It consists of three networks representing paper-citation relations,

¹⁰<https://github.com/js05212/citeulike-t>

¹¹<https://dl.acm.org/>

¹²<https://www.scopus.com/home.uri>

¹³<https://www.aminer.org/>

¹⁴<https://aan.how/download/>

author-collaboration relations and the author-citation relations. The original dataset contains 24,766 papers and 124,857 citations [67].

Descriptions of non-public datasets based on AAN (*AAN_modified*, *AAN_tiny*) can be found in Table 7.5.

7.4.8 Sowiport based datasets

Sowiport was an open digital library containing information on publications from the social sciences and adjacent fields [13, 37]. It contained author names, keywords and venue titles by which the constructed social network could be traversed by users. Sowiport co-operated with the recommendation-as-a-service system Mr. DLib [26].

Descriptions of non-public datasets based on Sowiport (*Sowiport*, *RARD_tiny*) can be found in Table 7.5.

7.4.9 CiteSeerX based datasets

CiteSeerX [32, 109] is a digital library focused on metadata and full-texts of open access literature¹⁵. It is the overhauled form of the former digital library CiteSeer.

Descriptions of non-public datasets based on CiteSeerX (*CiteSeer*, *CiteSeer_tiny*, *CiteSeer_medium*) can be found in Table 7.5.

7.4.10 Patents based datasets

The Patents dataset provides information on patents and trademarks granted by the United States Patent and Trademark Office¹⁶.

Descriptions of non-public datasets based on Patents (*Patents_tiny*, *Patents*, *ACM H-I*) can be found in Table 7.5.

7.4.11 Hep-TH based datasets

The original unaltered *Hep-TH* [50] dataset¹⁷ stems from the area of high energy physics theory. It contains papers in a graph which were published between 1993 and 2003. It was released as part of KDD Cup 2003.

Descriptions of non-public datasets based on Hep-TH (*Hep-TH graph*, *arXiv Hep-TH*) can be found in Table 7.5.

¹⁵<https://citeseerx.ist.psu.edu/index>

¹⁶<https://bulkdata.uspto.gov/>

¹⁷<https://snap.stanford.edu/data/cit-HepTh.html>

7.4.12 MAG based datasets

The Microsoft Academic Graph (MAG) [92] was an open scientific network containing metadata on academic communication activities¹⁸. Their heterogeneous graph consists of nodes representing fields of study, authors, affiliations, papers and venues.

Descriptions of non-public datasets based on MAG (*MSA*, *MAG 2017*, *MAG 2018*) can be found in Table 7.5.

7.4.13 Others

The following datasets have no common underlying data source: The *BBC*¹⁹ dataset contains 2,225 BBC news articles which stem from 5 topics.

*PRSDataset*²⁰ contains 2,453 users, 21,940 items and 35,969 pairs of users and items.

Descriptions of all other non-public datasets can be found in Table 7.5.

7.5 Evaluation

Due to the vast differences in approaches and datasets used to apply the methods, there is also a spectrum of used evaluation measures and objectives. In this section first we observe different notions of relevance of recommended papers and individual assessment strategies for relevance. Afterwards we analyse commonly used evaluation measures and list ones which are only rarely encountered in evaluation of paper recommendation systems. Lastly we shed light on the different types of evaluation which authors conducted.

In this discussion we again only consider paper recommendation systems which also evaluate their actual approach. We disregard approaches which do evaluate other properties [2, 23, 35, 79, 81, 116] or contain no evaluation [57]. Thus we observe 54 different approaches in this analysis.

7.5.1 Relevance and Assessment

Relevance of recommended publications can be evaluated against multiple target values: clicked papers [22, 53, 99], references [41, 110], references of recently authored papers [54], papers an author interacted with in the past [46], degree-of-relevancy which is determined by citation strength [89], a

¹⁸<https://docs.microsoft.com/en-us/academic-services/graph/>

¹⁹<http://mlg.ucd.ie/datasets/bbc.html>

²⁰<https://sites.google.com/site/tinhuynhuit/dataset>

Work	Relevancy				Target value					Measures						
	Human rating	Dataset	Papers	Clicked	Read	Cited	Liked	Relevancy	Other	Precision	Recall	F1	nDCG	MMR	MAP	Other
[1]			•	•												•
[3]		•					•									
[4]		•					•									•
[5]		•						•								
[17]		•			•											
[20]	•							•								
[19]	•					•										
[22]	•							•								•
[24]	•							•								•
[25]	•	•						•								•
[26]	•							•					•			•
[27]	•		•						•							•
[28]	•	•						•								•
[34]		•						•								•
[36]		•						•								•
[38]		•	•			•										•
[39]	•							•								•
[41]		•	•			•										•
[42]	•		•					•								•
[43]	•		•					•								•
[45]		•	•			•							•			•
[46]		•						•								•
[52]		•		•											•	
[53]		•	•					•								•
[54]		•	•			•										•
[56]		•	•			•									•	
[58]		•	•					•								•
[59]		•	•					•								•
[60]		•	•			•										•
[61]		•	•			•										•
[30]	•		•			•										•
[65]		•	•					•								•
[67]		•	•			•										•
[70, 71, 72]	•		•			•										•
[84]		•	•					•								•
[83]		•	•					•								•
[88, 89]	•	•	•					•					•			•
[90]		•	•					•								•
[91]	•		•					•								•
[93]		•	•					•								•
[99]		•	•		•											•
[101]		•	•					•								•
[102]		•	•			•										•
[103]	•		•					•								•
[108]		•	•			•										•
[104]	•	•	•					•								•
[105]		•	•					•								•
[106]		•	•					•								•
[110]		•	•			•							•			•
[111]		•	•			•							•			•
[112]		•	•			•							•			•
[113]		•	•					•					•			•
[115]		•	•					•					•			•
[117]		•	•					•					•			•

Table 7.6: Indications whether approaches utilise the specified relevancy definitions, target values of evaluations and evaluation measures.

ranking based on future citation numbers [115] as well as papers accepted [24] or deemed relevant by authors [36, 83].

Assessing the relevance of recommendations can also be conducted in different ways: the top n papers recommended by a system can be judged by either a referee team [104] or single persons [24, 70, 71]. Other options for relevance assessment are the usage of a dataset with user ratings [36, 83] or emulation of users and their interests [1, 54].

Table 7.6 holds information on utilised relevance indicators and target values which indicate relevance for the 54 discussed approaches. *Relevancy* describes the method that defines which of the recommended papers are relevant:

- Human rating: The approach is evaluated using assessments of real users of results specific to the approach.
- Dataset: The approach is evaluated using some type of assessment of a target value which is not specific to the approach but from a dataset. The assessment was either conducted for another approach and re-used or it was collected independent of an approach.
- Papers: The approach is evaluated by some type of assessment of a target value which is directly generated from the papers contained in the dataset such as citations or their keywords.

The *target values* in Table 7.6 describe the entities which the approach tried to approximate:

- Clicked: The approximated target value is derived from users' clicks on papers.
- Read: The approximated target value is derived from users' read papers.
- Cited: The approximated target value is derived from cited papers.
- Liked: The approximated target value is derived from users' liked papers.
- Relevancy: The approximated target value is derived from users' relevance assessment of papers.
- Other: The approximated target value is derived from other entities, e.g. papers with identical references or interest.

	P	R	F1	nDCG	MMR	MAP
%	48.15	24.07	50	25.92	27.78	22.22

Table 7.7: Common evaluation measures and percentage of observed evaluations of paper recommendation systems in which they were applied. Percentages are rounded to two decimal places.

Only three approaches evaluate against multiple target values [19, 28, 99]. Six approaches (11.11%) utilise clicks of users, only one approach (1.85%) uses read papers as target value. Even though cited papers are not the main objective of paper recommendation systems but rather citation recommendation systems, this target was approximated by 13 (24.07%) of the observed systems. Ten approaches (18.52%) evaluated against liked papers, 15 (27.78%) against relevant papers and 13 (24.07%) against some other target value.

7.5.2 Evaluation Measures

We differentiate between commonly used and rarely used evaluation measures for the task of scientific paper recommendation. They are described in the following sections. Table 7.6 holds indications of utilised evaluation measures for the 54 discussed approaches. *Measures* are the methods used to evaluate the approach’s ability to approximate the target value which can be of type precision, recall, f1 measure, nDCG, MMR, MAP or another one.

Out of the observed systems, twelve approaches [1, 26, 28, 46, 56, 61, 65, 67, 70, 71, 72, 102, 111, 110] (22.22%) only report one single measure, all others report at least two different ones.

Commonly Used Evaluation Measures

Bai et al [9] identify *precision* (P), *recall* (R), *F1*, *nDCG*, *MMR* and *MAP* as evaluation features which have been used regularly in the area of paper recommendation systems. Table 7.7 gives usage percentages of each of these measures in observed related work.

Alfarhood and Cheng [4] argue against the use of precision when utilising implicit feedback. If a user gives no feedback for a paper it could either mean disinterest or that a user does not know of the existence of the specific publication.

Measure	Used by	Description
Average precision	[103]	area under precision-recall curve
Receiver operating characteristic	[115]	plot of true positives against false positives
AUC	[34, 99]	area under receiver operating characteristic curve
Computation time	[24, 58]	time to compute recommendation list
DCG	[4]	summed up relevancy divided by logarithm of rank + 1
Click-through-rates	[22, 26]	percentage of Clicks on recommendations
Reward	[1, 33]	weighted sum of interactions of users with recommendations, e.g. clicked and saved papers
Spearman correlation coefficient	[42, 115]	correlation between ranks of paper lists
Hit ratio	[59, 108, 113]	percentage of relevant items in top k recommendations
Accuracy	[19, 61, 87]	percentage of relevant papers which the approach identified
Specificity	[19]	true negative rate
Mean absolute error	[38]	average difference between real and predicted values
Root mean square error	[38]	expected squared difference between real and predicted values
Fallout	[30]	percentage of irrelevant recommendations out of all irrelevant papers
Support	[67]	frequency of occurrences of set
TopN	[104]	probability that target keywords are encountered in first n recommended papers
FindN	[104]	number of target keywords which are encountered in first n recommended papers
Coverage	[117]	method's ability to discover the long tail of papers
Popularity	[117]	average logarithm of the number of ratings of papers in recommendation, indicates novelty of results
Average paper popularity	[58]	paper popularity divided by number of recommendations
Intra-list similarity	[117]	dissimilarity between recommended papers, smaller value indicates more diverse recommendation
Serendipity score	[70, 71, 72]	summed up usefulness divided by unexpectedness of recommended papers
Success rate	[58]	number of recommendations ; $2 \times$ number of keywords
Number of recommended papers	[58]	size of set of recommended papers

Table 7.8: Overview of rare existing measures used in evaluations of observed approaches.

Rarely used Evaluation Measures

We found a plethora of rarer used evaluation measures which have either been utilised only by the work they were introduced in or to evaluate few approaches. Our analysis in this aspect might be highly influenced by the narrow time frame we observe. Novel measures might require more time to be adopted by a broader audience. Thus we differentiate between novel rarely used evaluation measures and ones where authors do not explicitly claim they are novel. A list of rare but already defined evaluation measures can be found in Table 7.8. In total 25 approaches (46.3%) did use an evaluation measure not considered common.

Novel rarely used Evaluation Measures. In our considered approaches we only encountered three novel evaluation measures: *Recommendation quality* as defined by Chaudhuri et al. [24] is the acceptance of recommendations by users rated on a Likert scale from 1 to 10.

TotNP_EU is a measure defined by Manju et al. [30] specifically introduced for measuring performance of approaches regarding the cold start problem. It indicates the number of new publications suggested to users with a prediction value above a certain threshold.

TotNP_AVG is another measure defined by Manju et al. [30] for measuring performance of approaches regarding the cold start problem. It indicates the

average number of new publications suggested to users with a prediction value above a certain threshold.

7.5.3 Evaluation Types

Evaluations can be classified into different categories. We follow the notion of Beel and Langer [15] who differentiate between user studies, online evaluations and offline evaluations. They define *user studies* as ones where users' satisfaction with recommendation results is measured by collecting explicit ratings. *Online evaluations* are ones where users do not explicitly rate the recommendation results; relevancy is derived from e.g. clicks. In *offline evaluations* a ground truth is used to evaluate the approach.

From the 54 observed approaches we found four using multiple evaluation types [27, 43, 87, 89, 104]. Twelve (22.22%) were conducting user studies which describe the size and composition of the participant group.²¹ Only two approaches [26, 30] (3.7%) in the observed papers were evaluated with an online evaluation. We found 44 approaches (81.48%) providing an offline evaluation. Offline evaluations being the most common form of evaluation is unsurprising as this tendency has also been observed in an evaluation of general scientific recommender systems [21]. Offline evaluations are fast and do not require users [21]. Nevertheless the margin by which this form of evaluation is conducted could be rather surprising.

A distinction in *lab-based* vs. *real world* user studies can be conducted [14, 15]. User studies where participants rate recommendations according to some criteria and are aware of the study are lab-based, all others are considered real world studies. Living labs [33, 12, 86] for example enable real world user studies. On average the lab-based user studies were conducted with 17.83 users. Table 7.9 holds information on the number of participants for all studies as well as the composition of groups in terms of seniority.

For offline evaluation, they can either be ones with an *explicit* ground truth given by a dataset containing user rankings, *implicit* ones by deriving user interactions such as liked or cited papers or *expert* ones with manually collected expert ratings [15]. We found 22 explicit offline evaluations (40.74%) corresponding to ones using datasets to estimate relevance (see Table 7.6) and 21 implicit offline evaluations (38.89%) corresponding to ones using paper information to identify relevant recommendations (see Table 7.6). We did not find any expert offline evaluations.

²¹Shi et al. [91] also conduct a user study but do not describe their participants.

work	#p	composition
Bulut et al. [20]	50	PhD students studying in Turkey in 2019
Bulut et al. [19]	10 + 30	researchers
Chaudhuri et al. [22]	50	NA
Chaudhuri et al. [24]	45	from 9 different areas, different seniority levels: 12 faculty members, 20 postgraduate students, 13 undergraduate students
Du et al. [28]	NA	college students or patent analysis experts
Hua et al. [39]	10	experts
Kanakia et al. [42]	40	full-time computer science researchers at Microsoft Research
Kang et al. [43]	12	postgraduates
Nishioka et al. [70, 71, 72]	22	seniority based on highest degree: 2 Master's, 13 PhD, 7 lecturers/professors; 2 female, 20 male; 17 working in academia, 3 working in industry
Shahid et al. [88]	20	post-graduate students
Waheed et al. [103]	20	researchers
Wang et al. [104]	5	1 doctoral supervisor, 2 master supervisors, 2 graduate students

Table 7.9: For all observed works with user studies we list their number of participants (#p) and their composition. NA indicates that #p or compositions were not described in a specific user study.

7.6 Open Challenges and Objectives

All paper recommendation approaches which were considered in this survey could have been improved in some way or another. Some papers did not conduct evaluations which would satisfy a critical reader, others could be more convincing if they compared their methods to appropriate competitors. The possible problems we encountered within the papers can be summarised in different open challenges, which papers should strive to overcome. We separate our analysis and discussion of open challenges in those which have already been described by previous literature reviews (see Section 7.6.1) and ones we identify as new or emerging problems (see Section 7.6.2). Lastly we briefly discuss the presented challenges (see Section 7.6.3).

7.6.1 Challenges Highlighted in Previous Works

In the following we will explain possible shortcomings which were already explicitly discussed in previous literature reviews [9, 14, 87]. We regard these challenges in light of current paper recommendation systems to identify problems which are nowadays still encountered.

Neglect of User Modelling

Neglect of user modelling has been described by Beel et al. [14] as identification of target audiences' information needs. They describe the trade-off between specifying keywords which brings recommendation systems closer to search engines and utilising user profiles as input.

Currently only some approaches consider users of systems to influence the recommendation outcome, as seen with Table 7.3 users are not always

Group	Papers
Capital University of Science and Technology	[2, 35]
Firat University	[20, 19]
IIT Kharagpur	[23, 22, 24]
Qufu Normal University	[57, 58]
Kyoto-Kiel-Essex	[70, 71, 72]
University of Malaya-Bayero University	[84, 83]
Pakistan	[88, 89]
Hefei University of Technology	[105, 106]
Shandong University	[110, 111]
Australia	[115, 116]

Table 7.10: Overview of research groups with multiple papers.

part of the input to systems. Instead many paper recommendation systems assume that users do not state their information needs explicitly but only enter keywords or a paper. With paper recommendation systems where users are not considered, the problem of neglecting user modelling still holds.

Focus on Accuracy

Focus on accuracy as a problem is described by Beel et al. [14]. They state putting users' satisfaction with recommendations on a level with accuracy of approaches does not depict reality. More factors should be considered.

Only over one fourth of current approaches do not only report precision or accuracy but also observe more diversity focused measures such as MMR. We also found usage of less widespread measures to capture different aspects such as popularity, serendipity or click-through-rate.

Translating Research into Practice

The missing translation of research into practice is described by Beel et al. [14]. They mention the small percentage of approaches which are available as prototype as well as the discrepancy between real world systems and methods described in scientific papers.

Only four of our observed approaches definitively must have been available online at any point in time [26, 42, 30, 79]. We did not encounter any of the more complex approaches being used in widespread paper recommendation systems.

Persistence and Authority

Beel et al. [14] describe the lack of persistence and authority in the field of paper recommendation systems as one of the main reasons why research is not adapted in practice.

#	2	3	4	5	6	7	8
%	14.06	31.25	14.06	23.44	7.81	3.13	3.13

Table 7.11: Percentage of the 64 considered papers with different numbers of authors (#). Publications with 1 and 10 authors were encountered only once (1.56% each).

The analysis of this possible shortcoming of current work could be highly affected by the short time period from which we observed works. We found several groups publishing multiple papers as seen in Table 7.10 which corresponds to 29.69% of approaches. The most papers a group published was three so this amount still cannot fully mark a research group as authority in the area.

Cooperation

Problems with cooperation are described by Beel et al. [14]. They state even though approaches have been proposed by multiple authors building upon prior work is rare. Corporations between different research groups are also only encountered sporadically.

Here again we want to point to the fact that our observed time frame of less than three years might be too short to make substantive claims regarding this aspect. Table 7.11 holds information on the different numbers of authors for papers and the percentage of papers out of the 64 observed ones which are authored by groups of this size. We only encountered little cooperation between different co-author groups (see Haruna et al. [36] and Sakib et al. [83] for an exception). There were several groups not extending their previous work [115, 116]. We refrain from analysing citations of related previous approaches as our considered period of less than three years is too short for all publications to have been able to be recognised by the wider scientific community.

Information Scarcity

Information scarcity is described by Beel et al. [14] as researchers' tendency to only provide insufficient detail to re-implement their approaches. This leads to problems with reproducibility.

Many of the approaches we encountered did not provide sufficient information to make a re-implementation possible: with Afsar et al. [1] it is unclear how the knowledge graph and categories were formed, Collins and Beel [26] do not describe their Doc2Vec enough, Liu et al. [58] do not specify the extraction of keywords for papers in the graph and Tang et al. [99] do not

clearly describe their utilisation of Word2Vec. In general oftentimes details are missing [3, 4, 57, 112]. Exceptions to these observations are e.g. found with Berezki [17], Nishioka et al. [70, 71, 72] and Sakib et al. [83].

We did not find a single papers' code e.g. provided as a link to GitHub.

Cold Start

Pure collaborative filtering systems encounter the cold start problem as described by Bai et al. [9] and Shahid et al. [87]. If new users are considered, no historical data is available, they cannot be compared to other users to find relevant recommendations.

While this problem still persists, most current approaches are no pure collaborative filtering based recommendation systems (see Section 7.3.3). Systems using deep learning could overcome this issue [55]. There are approaches specifically targeting this problem [56, 91], some [56] also introduced specific evaluation measures (totNP_EU and avgNP_EU) to quantify systems' ability to overcome the cold start problem.

Sparsity or Reduce Coverage

Bai et al. [9] state the user-paper-matrix being sparse for collaborative filtering based approaches. Shahid et al. [87] also mention this problem as the *reduce coverage problem*. This trait makes it hard for approaches to learn relevancy of infrequently rated papers.

Again, while this problem is still encountered, current approaches mostly are no longer pure collaborative filtering based systems but instead utilise more information (see Section 7.3.3). Using deep learning in the recommendation process might reduce the impact of this problem [55].

Scalability

The problem of scalability was described by Bai et al. [9]. They state paper recommendation systems should be able to work in huge, ever expanding environments where new users and papers are added regularly.

A few approaches [35, 43, 83, 104] contain a web crawling step which directly tackles challenges related to outdated or missing data. Some approaches [24, 58] evaluate the time it takes to compute paper recommendations which also indicates their focus on this general problem. But most times scalability is not explicitly mentioned by current paper recommendation systems. There are several works [39, 42, 91, 103, 111] evaluating on bigger datasets with over 1 million papers and which thus are able to handle big amounts of data. Sizes of current relevant real-world data collections

exceed this threshold many times over (see e.g. PubMed with over 33 million papers²² or SemanticScholar with over 203 million papers²³). Kanakia et al. [42] explicitly state scalability as a problem their approach is able to overcome. Instead of comparing each paper to all other papers they utilise clustering to reduce the number of required computations. They present the only approach running on several hundred million publications. Nair et al. [67] mention scalability issues they encountered even when only considering around 25,000 publications and their citation relations.

Privacy

The problem of privacy in personalised paper recommendation is described by Bai et al. [9]. Shahid et al. [87] also mention this as a problem occurring in collaborative filtering approaches. An issue is encountered when sensitive information such as habits or weaknesses that users might not want to disclose is used by a system. This leads to users' having negative impressions of systems. Keeping sensitive information private should therefore be a main goal.

In the current approaches, we did not find a discussion of privacy concerns. Some approach even explicitly utilise likes [79] or association rules [3] of other users while failing to mention privacy altogether. In approaches not incorporating any user data, this issue does not arise at all.

Serendipity

Serendipity is described by Bai et al. [9] as an attribute often encountered in collaborative filtering [14]. Usually paper recommender systems focus on identification of relevant papers even though also including not obviously relevant ones might enhance the overall recommendation. Junior researchers could profit from stray recommendations to broaden their horizon, senior researchers might be able to gain knowledge to enhance their research. The ratio between clearly relevant and serendipitous papers is crucial to prevent users from losing trust in the recommender system.

A main objective of the works of Nishioka et al. [70, 71, 72] is serendipity. Other approaches do not mention this aspect.

²²<https://pubmed.ncbi.nlm.nih.gov/>

²³<https://www.semanticscholar.org/product/api>

Unified Scholarly Data Standards

Different data formats of data collections is mentioned as a problem by Bai et al. [9]. They mention digital libraries containing relevant information which needs to be unified in order to use the data in a paper recommendation system. Additionally the combination of datasets could also lead to problems.

Many of the approaches we observe do not consider data collection or preparation as part of the approach, they often only mention the combination of different datasets as part of the evaluation (see e.g. Du et al. [27], Li et al. [53] or Xie et al. [110]). An exception to this general rule are systems which contain a web crawling step for data (see e.g. Ahmad and Afzal [2] or Sakib et al. [83]). Even with this type of approaches the combination of datasets and their diverse data formats is not identified as a problem.

Synonymy

Shahid et al. [87] describe the problem of synonymy encountered in collaborative filtering approaches. They define this problem as different words having the same meaning.

Even though there are still approaches (not necessarily CF ones) utilising basic TF-IDF representations of papers [2, 39, 81, 90], nowadays this problem can be bypassed by using a text embedding method such as Doc2Vec or BERT.

Gray Sheep

Gray sheep is a problem described by Shahid et al. [87] as an issue encountered in collaborative filtering approaches. They describe it as some users not consistently (dis)agreeing with any reference group.

We did not find any current approach mentioning this problem.

Black Sheep

Black sheep is a problem described by Shahid et al. [87] as an issue encountered in collaborative filtering approaches. They describe it as some users not (dis)agreeing with any reference group.

We did not find any current approach mentioning this problem.

Shilling attack

Shilling attacks are described by Shahid et al. [87] as a problem encountered in collaborative filtering approaches. They define this problem as users being

able to manually enhance visibility of their own research by rating authored papers as relevant while negatively rating any other recommendations.

Although we did not find any current approach mentioning this problem we assume maybe it is no longer highly relevant as most approaches are no longer pure collaborative filtering ones. Additionally from the considered collaborative filtering approaches no one explicitly stated to feed relevance ratings back into the system.

7.6.2 Emerging Challenges

In addition to the open challenges discussed in former literature reviews by Bai et al. [9], Beel et al. [14] and Shahid et al. [87] we identified the following problems and derive desirable goals for future approaches from them.

User Evaluation

Paper recommendation is always targeted at human users. But oftentimes an evaluation with real users to quantify users' satisfaction with recommended publications is simply not conducted [79]. Conducting huge user studies is not feasible [35]. So sometimes user data to evaluate with is fetched from the presented datasets [36, 83] or user behaviour is artificially emulated [1, 17, 54]. Noteworthy counter-examples²⁴ are the studies by Bulut et al. [20] who emailed 50 researchers to rate relevancy of recommended articles or Chaudhuri et al. [24] who asked 45 participants to rate their acceptance of recommended publications. Another option to overcome this issue is utilisation of living labs as seen with ArXivDigest [33], Mr. DLib's living lab [12] or LiLAS for the related tasks of dataset recommendation for scientific publications and multi-lingual document retrieval [86].

Desirable goal. Paper recommendation systems targeted at users should always contain a user evaluation with a description of the composition of participants.

Target audience

Current works mostly fail to clearly characterise the intended users of a system altogether and the varying interests of different types of users are not examined in their evaluations. There are some noteworthy counter-examples: Afsar et al. [1] mention cancer patients and their close relatives as intended target audience. Bereczki [17] identifies new users as a special group they want to recommend papers to. Hua et al. [39] consider users which start

²⁴For a full list of approaches conducting user studies see Table 7.9.

diving into a topic which they have not yet researched before. Sharma et al. [90] name subject matter experts incorporating articles into a medical knowledge base as their target audience. Shi et al. [91] clearly state use cases for their approach which always target users which are unaware of a topic but already have one interesting paper from the area. They strive to recommend more papers similar to the first one.

User characteristics such as registration status of users are already mentioned by Beel et al. [14] as a factor which is disregarded in evaluations. We want to extend on this point and highlight the oftentimes missing or inadequate descriptions of intended users of paper recommendation systems. Traits of users and their information needs are not only important for experiments but should also be regarded in the construction of an approach. The targeted audience of a paper recommendation system should influence its suggestions. Bai et al. [9] highlight different needs of junior researchers which should be recommended a broad variety of papers as they still have to figure out their direction. They state recommendations for senior researchers should be more in line with their already established interests. Sugiyama and Kan [95] describe the need to help discover interdisciplinary research for this experienced user group. Most works do not recognise possible different functions of paper recommendation systems for users depending on their level of seniority. If papers include an evaluation with real persons, they e.g. mix Master's students with professors but do not address their different goals or expectations from paper recommendation [70]. Chaudhuri et al. [24] have junior, experienced and expert users as participants of their study and give individual ratings but do not calculate evaluation scores per user group. In some studies the exact composition of test users is not even mentioned (see Table 7.9).

Desirable goal. Definition and consideration of a specific target audience for an approach and evaluation with members of this audience. If there is no specific person group a system should suit best, this should be discussed, executed and evaluated accordingly.

Recommendation Scenario

Suggested papers from an approach should either be ones to read [104, 41], to cite or fulfil another specified information need such as help patients in cancer treatment decision making [1]. Most work does not clearly state which is the case. Instead recommended papers are only said to be related [4, 26], relevant [4, 5, 24, 25, 35, 39, 42, 45, 53, 54, 100, 110, 112], satisfactory [39, 58], suitable [19], appropriate and useful [20, 83] or a description which scenario is tackled is skipped altogether [3, 34, 36, 79].

In rare cases if the recommendation scenario is mentioned there is the possibility of it not perfectly fitting the evaluated scenario. This can e.g. be seen in the work of Jing and Yu [41] where they propose paper recommendation for papers to read but evaluate papers which were cited. Cited papers should always be ones which have been read beforehand but the decision to cite papers can be influenced by multiple aspects [31].

Desirable goal. The clear description of the recommendation scenario is important for comparability of approaches as well as the validity of the evaluation.

Fairness/Diversity

Anand et al [8] define fairness as the balance between relevance and diversity of recommendation results. Only focusing on fit between the user or input paper and suggestions would lead to highly similar results which might not be vastly different from each other. Having diverse recommendation results can help cover multiple aspects of a user query instead of only satisfying the most prominent feature of the query [8]. In general more diverse recommendations provide greater utility for users [72].

Most of the current paper recommendation systems do not consider fairness but some approaches specifically mention diversity [24, 70, 71, 72] while striving to recommend relevant publications. Thus these systems consider fairness.

Over one fourth of considered approaches with an evaluation report MMR as a measure of their system's quality. This at least seems to show researchers' awareness of the general problem of diverse recommendation results.

Desirable Goal. Diversification of suggested papers to ensure fairness of the approach.

Complexity

Paper recommendation systems tend to become more complex, convoluted or composed of multiple parts. We observed this trend by regarding the classification of current systems compared to previous literature reviews (see Section 7.3.3). While systems' complexity increases, users' interaction with the systems should not become more complex. If an approach requires user interaction at all, it should be as simple as possible. Users should not be required to construct sophisticated knowledge graphs [104] or enter multiple rounds of keywords for an approach to learn their user profile [22].

Desirable Goal. Maintain simplicity of usage even if approaches become more complex.

Explainability

Confidence in the recommendation system has already been mentioned by Beel et al. [14] as an example of what could enhance users' satisfaction but what is overlooked in approaches in favour of accuracy. This aspect should be considered with more vigour as the general research area of explainable recommendation has gained immense traction [114]. Gingstad et al. [33] regard explainability as a core component of paper recommendation systems. Xie et al. [111] mention explainability as a key feature of their approach but do not state how they achieve it or if their explanations satisfy users. Suggestions of recommendation systems should be explainable to enhance their trustworthiness and make them more engaging [62]. Here, different explanation goals such as effectiveness, efficiency, transparency or trust and their influence on each other should be considered [10]. If an approach uses neural networks [22, 34, 46, 53] it is oftentimes impossible to explain why the system learned, that a specific suggested paper might be relevant.

Lee et al. [48] introduce a general approach which could be applied to any paper recommendation system to generate explanations for recommendations. Even though this option seems to help solve the described problem it is not clear how valuable post-hoc explanations are compared to systems which construct them directly.

Desirable Goal. The conceptualisation of recommendation systems which comprehensibly explain their users why a specific paper is suggested.

Public Dataset

Current approaches utilise many different datasets (see Table 7.4). A large portion of them are built by the authors such that they are not publicly available for others to use as well [1, 28, 106]. Part of the approaches already use open datasets in their evaluation but a large portion still does not seem to regard this as a priority (see Table 7.5). Utilisation of already public data sources or construction of datasets which are also published and remain available thus should be a priority in order to support reproducibility of approaches.

Desirable Goal Utilisation of publicly available datasets in the evaluation of paper recommendation systems.

Comparability

From the approaches we observed many identified themselves as paper recommendation ones but only evaluated against systems, which are more general recommendation systems or ones utilising some same methodologies but not from the sub-domain of paper recommendation (seen with e.g. Guo et al [34], Tanner et al. [101] or Yang et al. [112]). While some of the works might claim to only be applied on paper recommendation and be of more general applicability (see e.g. the works by Ahmedi et al. [3] or Alfarhood and Cheng [4]) we state that they should still be compared to ones, which mainly identify as paper recommendation systems as seen in the work of Chaudhuri et al. [22]. Only if a more general approach is compared to a paper recommendation approach, its usefulness for the area of paper recommendation can be fully assessed.

Several times, the baselines to evaluate against are not even other works but artificially constructed ones [2, 35] or no other approach at all [20].

Desirable Goal. Evaluation of paper recommendation approaches, even those which are applicable in a wider context, should always be against at least one paper recommendation system to clearly report relevance of the proposed method in the claimed context.

7.6.3 Discussion

From the already existing problems, several of them are still encountered in current paper recommendation approaches. Users are not always part of the approaches so users are not always modelled but this also prevents privacy issues. Accuracy seems to still be the main focus of recommendation systems. Novel techniques proposed in papers are not available online or applied by existing paper recommendation systems. Approaches do not provide enough details to enable re-implementation.

Other problems mainly encountered in pure collaborative filtering systems such as the cold start problem, sparsity, synonymy, gray sheep, black sheep and shilling attacks do not seem to be as relevant anymore. We observed a trend towards hybrid models, this recommendation system type can overcome these issues. These hybrid models should also be able to produce serendipitous recommendations.

Unifying data sources is conducted often but nowadays it does not seem to be regarded as a problem. With scalability we encountered the same. Approaches are oftentimes able to handle millions of papers, here they do not specifically mention scalability as a problem they overcome but they also mostly do not consider huge datasets with several hundreds of millions of

publications.

Due to the limited scope of our survey we are not able to derive substantive claims regarding cooperation and persistence. We found around 30% of approaches published by groups which authored multiple papers and very few collaborations between different author groups.

As for the newly introduced problems part of the observed approaches conducted evaluations with users, on publicly available datasets and against other paper recommendation systems. Many works considered a low complexity for users.

Target audiences in general were rarely defined, the recommendation scenario was mostly not described. Diversity was considered by few. Overall the explainability of recommendations was dismissed.

To conclude, there are many challenges which are not constantly considered by current approaches. They define the requirements for future works in the area of paper recommendation systems.

7.7 Conclusion

This literature review of publications targeting paper recommendation between January 2019 and October 2021 provided comprehensive overviews of their methods, datasets and evaluation measures. We showed the need for a richer multi-dimensional characterisation of paper recommendation as former ones no longer seem sufficient in classifying the increasingly complex approaches. We also revisited known open challenges in the current time frame and highlighted possibly under-observed problems which future works could focus on.

Efforts should be made to standardise or better differentiate between the varying notions of relevancy and recommendation scenarios when it comes to paper recommendation. Future work could try reevaluate already existing methods with real humans and against other paper recommendation systems. This could for example be realised in an extendable paper recommendation benchmarking system similar to the in a living lab environments ArXivDigest [33], Mr. DLib's living lab [12] or LiLAS [86] but with the additional property that it also provides build-in offline evaluations. As fairness and explainability of current paper recommendation systems have not been tackled widely, those aspects should be further explored. Another direction could be the comparison of multiple rare evaluation measures on the same system to help identify those which should be focused on in the future. As we observed a vast variety in datasets utilised for evaluation of the approaches (see Table 7.4), construction of publicly available and widely reusable ones would

be worthwhile.

Bibliography

- [1] Mohammad Mehdi Afsar, Trafford Crump, and Behrouz H. Far. An exploration on-demand article recommender system for cancer patients information provisioning. In Eric Bell and Fazel Keshtkar, editors, Proceedings of the Thirty-Fourth International Florida Artificial Intelligence Research Society Conference, North Miami Beach, Florida, USA, May 17-19, 2021, 2021.
- [2] Shahbaz Ahmad and Muhammad Tanvir Afzal. Combining metadata and co-citations for recommending related papers. Turkish J. Electr. Eng. Comput. Sci., 28(3):1519–1534, 2020.
- [3] Lule Ahmedi, Edonit Rexhepi, and Eliot Bytyçi. Using association rule mining to enrich user profiles with research paper recommendation. Int. J. Com. Dig. Sys., 2021.
- [4] Meshal Alfarhood and Jianlin Cheng. Collaborative attentive autoencoder for scientific article recommendation. In M. Arif Wani, Taghi M. Khoshgoftaar, Dingding Wang, Huanjing Wang, and Naeem Seliya, editors, 18th IEEE International Conference On Machine Learning And Applications, ICMLA 2019, Boca Raton, FL, USA, December 16-19, 2019, pages 168–174. IEEE, 2019.
- [5] Zafar Ali, Guilin Qi, Khan Muhammad, Bahadar Ali, and Waheed Ahmed Abro. Paper recommendation based on heterogeneous network embedding. Knowl. Based Syst., 210:106438, 2020.
- [6] Anas Alzoghbi, Victor Anthony Arrascue Ayala, Peter M. Fischer, and Georg Lausen. Pubrec: Recommending publications based on publicly available meta-data. In Ralph Bergmann, Sebastian Görg, and Gilbert Müller, editors, Proceedings of the LWA 2015 Workshops: KDML, FGWM, IR, and FGDB, Trier, Germany, October 7-9, 2015, volume 1458 of CEUR Workshop Proceedings, pages 11–18. CEUR-WS.org, 2015.
- [7] Maha Amami, Rim Faiz, Fabio Stella, and Gabriella Pasi. A graph based approach to scientific paper recommendation. In Amit P. Sheth, Axel Ngonga, Yin Wang, Elizabeth Chang, Dominik Slezak, Bogdan Franczyk, Rainer Alt, Xiaohui Tao, and Rainer Unland, editors, Proceedings of the International Conference on Web Intelligence, Leipzig, Germany, August 23-26, 2017, pages 777–782. ACM, 2017.

- [8] Ankesh Anand, Tanmoy Chakraborty, and Amitava Das. Fairscholar: Balancing relevance and diversity for scientific paper recommendation. In Joemon M. Jose, Claudia Hauff, Ismail Sengör Altingövde, Dawei Song, Dyaa Albakour, Stuart N. K. Watt, and John Tait, editors, Advances in Information Retrieval - 39th European Conference on IR Research, ECIR 2017, Aberdeen, UK, April 8-13, 2017, Proceedings, volume 10193 of Lecture Notes in Computer Science, pages 753–757, 2017.
- [9] Xiaomei Bai, Mengyang Wang, Ivan Lee, Zhuo Yang, Xiangjie Kong, and Feng Xia. Scientific paper recommendation: A survey. IEEE Access, 7:9324–9339, 2019.
- [10] Krisztian Balog and Filip Radlinski. Measuring recommendation explanation quality: The conflicting goals of explanations. In Jimmy Huang, Yi Chang, Xueqi Cheng, Jaap Kamps, Vanessa Murdock, Ji-Rong Wen, and Yiqun Liu, editors, Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval, SIGIR 2020, Virtual Event, China, July 25-30, 2020, pages 329–338. ACM, 2020.
- [11] Jöran Beel, Zeljko Carevic, Johann Schaible, and Gábor Neusch. RARD: the related-article recommendation dataset. D Lib Mag., 23(7/8), 2017.
- [12] Jöran Beel, Andrew Collins, Oliver Kopp, Linus W. Dietz, and Petr Knöth. Online evaluations for everyone: Mr. dlib’s living lab for scholarly recommendations. In Leif Azzopardi, Benno Stein, Norbert Fuhr, Philipp Mayr, Claudia Hauff, and Djoerd Hiemstra, editors, Advances in Information Retrieval - 41st European Conference on IR Research, ECIR 2019, Cologne, Germany, April 14-18, 2019, Proceedings, Part II, volume 11438 of Lecture Notes in Computer Science, pages 213–219. Springer, 2019.
- [13] Jöran Beel, Siddharth Dinesh, Philipp Mayr, Zeljko Carevic, and Raghvendra Jain. Stereotype and most-popular recommendations in the digital library sowiport. In Maria Gäde, Violeta Trkulja, and Vivien Petras, editors, Everything Changes, Everything Stays the Same? Understanding Information Spaces. Proceedings of the 15th International Symposium of Information Science (ISI 2017), Berlin, Germany, March 13-15, 2017, volume 70 of Schriften zur Informationswissenschaft, pages 96–108. Verlag Werner Hülsbusch, 2017.

- [14] Jöran Beel, Bela Gipp, Stefan Langer, and Corinna Breiting. Research-paper recommender systems: a literature survey. Int. J. Digit. Libr., 17(4):305–338, 2016.
- [15] Jöran Beel and Stefan Langer. A comparison of offline evaluations, online evaluations, and user studies in the context of research-paper recommender systems. In Sarantos Kapidakis, Cezary Mazurek, and Marcin Werla, editors, Research and Advanced Technology for Digital Libraries - 19th International Conference on Theory and Practice of Digital Libraries, TPDL 2015, Poznań, Poland, September 14-18, 2015. Proceedings, volume 9316 of Lecture Notes in Computer Science, pages 153–168. Springer, 2015.
- [16] Felix Beierle, Akiko Aizawa, Andrew Collins, and Joeran Beel. Choice overload and recommendation effectiveness in related-article recommendations. Int. J. Digit. Libr., 21(3):231–246, 2020.
- [17] Márk Bereczki. Graph neural networks for article recommendation based on implicit user feedback and content. Master’s thesis, KTH, School of Electrical Engineering and Computer Science (EECS), 2021.
- [18] Toine Bogers and Antal van den Bosch. Recommending scientific articles using citeulike. In Pearl Pu, Derek G. Bridge, Bamshad Mobasher, and Francesco Ricci, editors, Proceedings of the 2008 ACM Conference on Recommender Systems, RecSys 2008, Lausanne, Switzerland, October 23-25, 2008, pages 287–290. ACM, 2008.
- [19] Betül Bulut, Esra Gündoğan, Buket Kaya, Reda Alhajj, and Mehmet Kaya. User’s Research Interests Based Paper Recommendation System: A Deep Learning Approach, pages 117–130. Springer International Publishing, Cham, 2020.
- [20] Betül Bulut, Buket Kaya, and Mehmet Kaya. A paper recommendation system based on user interest and citations. In 2019 1st International Informatics and Software Engineering Conference (UBMYK), pages 1–5, 2019.
- [21] Zohreh Dehghani Champiri, Adeleh Asemi, and Siti Salwah Binti Salim. Meta-analysis of evaluation methods and metrics used in context-aware scholarly recommender systems. Knowl. Inf. Syst., 61(2):1147–1178, 2019.

- [22] Arpita Chaudhuri, Debasis Samanta, and Monalisa Sarma. Modeling user behaviour in research paper recommendation system. CoRR, abs/2107.07831, 2021.
- [23] Arpita Chaudhuri, Monalisa Sarma, and Debasis Samanta. Advanced feature identification towards research article recommendation: A machine learning based approach. In TENCON 2019 - 2019 IEEE Region 10 Conference (TENCON), Kochi, India, October 17-20, 2019, pages 7–12. IEEE, 2019.
- [24] Arpita Chaudhuri, Nilanjan Sinhababu, Monalisa Sarma, and Debasis Samanta. Hidden features identification for designing an efficient research article recommendation system. Int. J. Digit. Libr., 22(2):233–249, 2021.
- [25] Jinpeng Chen and Zhijie Ban. Academic Paper Recommendation Based on Clustering and Pattern Matching, pages 171–182. Springer, 07 2019.
- [26] Andrew Collins and Jöran Beel. Document embeddings vs. keyphrases vs. terms for recommender systems: A large-scale online evaluation. In Maria Bonn, Dan Wu, J. Stephen Downie, and Alain Martaus, editors, 19th ACM/IEEE Joint Conference on Digital Libraries, JCDL 2019, Champaign, IL, USA, June 2-6, 2019, pages 130–133. IEEE, 2019.
- [27] Nana Du, Jun Guo, Chase Q. Wu, Aiqin Hou, Zimin Zhao, and Daguang Gan. Recommendation of academic papers based on heterogeneous information networks. In 17th IEEE/ACS International Conference on Computer Systems and Applications, AICCSA 2020, Antalya, Turkey, November 2-5, 2020, pages 1–6. IEEE, 2020.
- [28] Zhengxiao Du, Jie Tang, and Yuhui Ding. POLAR++: active one-shot personalized article recommendation. IEEE Trans. Knowl. Data Eng., 33(6):2709–2722, 2021.
- [29] Shuaishuai Feng, Junyan Meng, and Jiaying Zhang. News recommendation systems in the era of information overload. J. Web Eng., 20(2):459–470, 2021.
- [30] Manju G., Abhinaya P., Hemalatha M. R., Manju Ganesh G., and Manju G. G. Cold start problem alleviation in a research paper recommendation system using the random walk approach on a heterogeneous user-paper graph. Int. J. Intell. Inf. Technol., 16(2):24–48, 2020.

- [31] Eugene Garfield. Can citation indexing be automated?, 1964.
- [32] C. Lee Giles, Kurt D. Bollacker, and Steve Lawrence. Citeseer: An automatic citation indexing system. In Proceedings of the 3rd ACM International Conference on Digital Libraries, June 23-26, 1998, Pittsburgh, PA, USA, pages 89–98. ACM, 1998.
- [33] Kristian Gingstad, Øyvind Jekteberg, and Krisztian Balog. Arxivdigest: A living lab for personalized scientific literature recommendation. In Mathieu d’Aquin, Stefan Dietze, Claudia Hauff, Edward Curry, and Philippe Cudré-Mauroux, editors, CIKM ’20: The 29th ACM International Conference on Information and Knowledge Management, Virtual Event, Ireland, October 19-23, 2020, pages 3393–3396. ACM, 2020.
- [34] Guibing Guo, Bowei Chen, Xiaoyan Zhang, Zhirong Liu, Zhenhua Dong, and Xiuqiang He. Leveraging title-abstract attentive semantics for paper recommendation. In The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020, pages 67–74. AAAI Press, 2020.
- [35] Raja Habib and Muhammad Tanvir Afzal. Sections-based bibliographic coupling for research paper recommendation. Scientometrics, 119(2):643–656, 2019.
- [36] Khalid Haruna, Maizatul Akmar Ismail, Atika Qazi, Habeebah Adamu Kakudi, Mohammed Hassan, Sanah Abdullahi Muaz, and Haruna Chiroma. Research paper recommender system based on public contextual metadata. Scientometrics, 125(1):101–114, 2020.
- [37] Daniel Hienert, Frank Sawitzki, and Philipp Mayr. Digital library research in action: Supporting information retrieval in sowiport. D Lib Mag., 21(3/4), 2015.
- [38] Donglin Hu, Huifang Ma, Yuhang Liu, and Xiangchun He. Scientific paper recommendation using author’s dual role citation relationship. In Zhongzhi Shi, Sunil Vadera, and Elizabeth Chang, editors, Intelligent Information Processing X - 11th IFIP TC 12 International Conference, IIP 2020, Hangzhou, China, July 3-6, 2020, Proceedings, volume 581 of

- IFIP Advances in Information and Communication Technology, pages 121–132. Springer, 2020.
- [39] Shengjun Hua, Wei Chen, Zhixu Li, Pengpeng Zhao, and Lei Zhao. Path-based academic paper recommendation. In Zhisheng Huang, Wouter Beek, Hua Wang, Rui Zhou, and Yanchun Zhang, editors, Web Information Systems Engineering - WISE 2020 - 21st International Conference, Amsterdam, The Netherlands, October 20-24, 2020, Proceedings, Part II, volume 12343 of Lecture Notes in Computer Science, pages 343–356. Springer, 2020.
- [40] Zhenyan Ji, Mengdan Wu, Hong Yang, and José Enrique Armendáriz-Iñigo. Temporal sensitive heterogeneous graph neural network for news recommendation. Future Gener. Comput. Syst., 125:324–333, 2021.
- [41] Sun Jing and Sun Yu. Research of paper recommendation system based on citation network model. In Xiaofeng Chen, Hongyang Yan, Qiben Yan, and Xiangliang Zhang, editors, Machine Learning for Cyber Security - Third International Conference, ML4CS 2020, Guangzhou, China, October 8-10, 2020, Proceedings, Part III, volume 12488 of Lecture Notes in Computer Science, pages 237–247. Springer, 2020.
- [42] Anshul Kanakia, Zhihong Shen, Darrin Eide, and Kuansan Wang. A scalable hybrid research paper recommender system for microsoft academic. CoRR, abs/1905.08880, 2019.
- [43] Ying Kang, Aiqin Hou, Zimin Zhao, and Daguang Gan. A hybrid approach for paper recommendation. IEICE Transactions on Information and Systems, E104.D(8):1222–1231, 2021.
- [44] Jüri Keller and Leon Paul Mondrian Munz. TEKMA at CLEF-2021: BM-25 based rankings for scientific publication retrieval and data set recommendation. In Guglielmo Faggioli, Nicola Ferro, Alexis Joly, Maria Maistro, and Florina Piroi, editors, Proceedings of the Working Notes of CLEF 2021 - Conference and Labs of the Evaluation Forum, Bucharest, Romania, September 21st - to - 24th, 2021, volume 2936 of CEUR Workshop Proceedings, pages 1700–1711. CEUR-WS.org, 2021.
- [45] Xiangjie Kong, Mengyi Mao, Wei Wang, Jiaying Liu, and Bo Xu. Voprec: Vector representation learning of papers with text information and structural identity for recommendation. IEEE Trans. Emerg. Top. Comput., 9(1):226–237, 2021.

- [46] Hao L, Shijun Liu, and Li Pan. Paper recommendation based on author-paper interest and graph structure. In Weiming Shen, Jean-Paul A. Barthès, Junzhou Luo, Yanjun Shi, and Jinghui Zhang, editors, 24th IEEE International Conference on Computer Supported Cooperative Work in Design, CSCWD 2021, Dalian, China, May 5-7, 2021, pages 256–261. IEEE, 2021.
- [47] Minh Le, Subhradeep Kayal, and Andrew Douglas. The impact of recommenders on scientific article discovery: The case of mendeley suggest. In Oren Sar Shalom, Dietmar Jannach, and Ido Guy, editors, Proceedings of the 1st Workshop on the Impact of Recommender Systems co-located with 13th ACM Conference on Recommender Systems, ImpactRS@RecSys 2019, Copenhagen, Denmark, September 19, 2019, volume 2462 of CEUR Workshop Proceedings. CEUR-WS.org, 2019.
- [48] Benjamin Charles Germain Lee, Kyle Lo, Doug Downey, and Daniel S. Weld. Explanation-based tuning of opaque machine learners with application to paper recommendation. CoRR, abs/2003.04315, 2020.
- [49] Joonseok Lee, Kisung Lee, and Jennifer G. Kim. Personalized academic research paper recommendation system. CoRR, abs/1304.5457, 2013.
- [50] Jure Leskovec, Jon M. Kleinberg, and Christos Faloutsos. Graphs over time: densification laws, shrinking diameters and possible explanations. In Robert Grossman, Roberto J. Bayardo, and Kristin P. Bennett, editors, Proceedings of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Chicago, Illinois, USA, August 21-24, 2005, pages 177–187. ACM, 2005.
- [51] Michael Ley. DBLP - some lessons learned. Proc. VLDB Endow., 2(2):1493–1500, 2009.
- [52] Weisheng Li, Chao Chang, Chaobo He, Zhengyang Wu, Jiongsheng Guo, and Bo Peng. Academic paper recommendation method combining heterogeneous network and temporal attributes. In Yuqing Sun, Dongning Liu, Hao Liao, Hongfei Fan, and Liping Gao, editors, Computer Supported Cooperative Work and Social Computing, pages 456–468, Singapore, 2021. Springer Singapore.
- [53] Xinyi Li, Yifan Chen, Benjamin Pettit, and Maarten de Rijke. Personalised reranking of paper recommendations using paper content and user behavior. ACM Trans. Inf. Syst., 37(3):31:1–31:23, 2019.

- [54] Yi Li, Ronghui Wang, Guofang Nan, Dahui Li, and Minqiang Li. A personalized paper recommendation method considering diverse user preferences. Decis. Support Syst., 146:113546, 2021.
- [55] Zhi Li and Xiaozhu Zou. A review on personalized academic paper recommendation. Comput. Inf. Sci., 12(1):33–43, 2019.
- [56] Shi-jie Lin, Guanling Lee, and Sheng-Lung Peng. Academic Article Recommendation by Considering the Research Field Trajectory, pages 447–454. Springer, 04 2021.
- [57] Hanwen Liu, Huaizhen Kou, Xiaoxiao Chi, and Lianyong Qi. Combining time, keywords and authors information to construct papers correlation graph (S). In Angelo Perkusich, editor, The 31st International Conference on Software Engineering and Knowledge Engineering, SEKE 2019, Hotel Tivoli, Lisbon, Portugal, July 10-12, 2019, pages 11–19. KSI Research Inc. and Knowledge Systems Institute Graduate School, 2019.
- [58] Hanwen Liu, Huaizhen Kou, Chao Yan, and Lianyong Qi. Keywords-driven and popularity-aware paper recommendation based on undirected paper citation graph. Complex., 2020:2085638:1–2085638:15, 2020.
- [59] Yibo Lu, Yi He, Yixiang Cai, Zelin Peng, and Yong Tang. Time-aware neural collaborative filtering with multi-dimensional features on academic paper recommendation. In Weiming Shen, Jean-Paul A. Barthès, Junzhou Luo, Yanjun Shi, and Jinghui Zhang, editors, 24th IEEE International Conference on Computer Supported Cooperative Work in Design, CSCWD 2021, Dalian, China, May 5-7, 2021, pages 1052–1057. IEEE, 2021.
- [60] Xiao Ma and Ranran Wang. Personalized scientific paper recommendation based on heterogeneous graph representation. IEEE Access, 7:79887–79894, 2019.
- [61] Xiao Ma, Yin Zhang, and Jiangfeng Zeng. Newly published scientific papers recommendation in heterogeneous information networks. Mob. Networks Appl., 24(1):69–79, 2019.
- [62] James McInerney, Benjamin Lacker, Samantha Hansen, Karl Higley, Hugues Bouchard, Alois Gruson, and Rishabh Mehrotra. Explore, exploit, and explain: personalizing explainable recommendations with

- bandits. In Sole Pera, Michael D. Ekstrand, Xavier Amatriain, and John O’Donovan, editors, Proceedings of the 12th ACM Conference on Recommender Systems, RecSys 2018, Vancouver, BC, Canada, October 2-7, 2018, pages 31–39. ACM, 2018.
- [63] Zoran Medic and Jan Snajder. A survey of citation recommendation tasks and methods. J. Comput. Inf. Technol., 28(3):183–205, 2020.
- [64] Hebatallah A. Mohamed Hassan, Giuseppe Sansonetti, Fabio Gasparetti, Alessandro Micarelli, and Jöran Beel. Bert, elmo, USE and inferent sentence encoders: The panacea for research-paper recommendation? In Marko Tkalčić and Sole Pera, editors, Proceedings of ACM RecSys 2019 Late-Breaking Results co-located with the 13th ACM Conference on Recommender Systems, RecSys 2019 Late-Breaking Results, Copenhagen, Denmark, September 16-20, 2019, volume 2431 of CEUR Workshop Proceedings, pages 6–10. CEUR-WS.org, 2019.
- [65] Hebatallah A. Mohamed Hassan, Giuseppe Sansonetti, and Alessandro Micarelli. Tag-aware document representation for research paper recommendation, 07 2020.
- [66] Oleksii Moskalenko, Diego Sáez-Trumper, and Denis Parra. Scalable recommendation of wikipedia articles to editors using representation learning. In Toine Bogers, Marijn Koolen, Casper Petersen, Bamshad Mobasher, Alexander Tuzhilin, Oren Sar Shalom, Dietmar Jannach, and Joseph A. Konstan, editors, Proceedings of the Workshops on Recommendation in Complex Scenarios and the Impact of Recommender Systems co-located with 14th ACM Conference on Recommender Systems (RecSys 2020), Online, September 25, 2020, volume 2697 of CEUR Workshop Proceedings. CEUR-WS.org, 2020.
- [67] Akhil M. Nair, Oshin Benny, and Jossy George. Content based scientific article recommendation system using deep learning technique. In V. Suma, Joy Iong-Zong Chen, Zubair Baig, and Haoxiang Wang, editors, Inventive Systems and Control, pages 965–977, Singapore, 2021. Springer Singapore.
- [68] Yiu-Kai Ng. CBRec: a book recommendation system for children using the matrix factorisation and content-based filtering approaches. International Journal of Business Intelligence and Data Mining, 16(2):129–149, January 2020.

- [69] Yiu-Kai Ng. Research paper recommendation based on content similarity, peer reviews, authority, and popularity. In 32nd IEEE International Conference on Tools with Artificial Intelligence, ICTAI 2020, Baltimore, MD, USA, November 9-11, 2020, pages 47–52. IEEE, 2020.
- [70] Chifumi Nishioka, Jörn Hauke, and Ansgar Scherp. Research paper recommender system with serendipity using tweets vs. diversification. In Adam Jatowt, Akira Maeda, and Sue Yeon Syn, editors, Digital Libraries at the Crossroads of Digital Information for the Future - 21st International Conference on Asia-Pacific Digital Libraries, ICADL 2019, Kuala Lumpur, Malaysia, November 4-7, 2019, Proceedings, volume 11853 of Lecture Notes in Computer Science, pages 63–70. Springer, 2019.
- [71] Chifumi Nishioka, Jörn Hauke, and Ansgar Scherp. Towards serendipitous research paper recommender using tweets and diversification. In Antoine Doucet, Antoine Isaac, Koraljka Golub, Trond Aalberg, and Adam Jatowt, editors, Digital Libraries for Open Knowledge - 23rd International Conference on Theory and Practice of Digital Libraries, TPDL 2019, Oslo, Norway, September 9-12, 2019, Proceedings, volume 11799 of Lecture Notes in Computer Science, pages 339–343. Springer, 2019.
- [72] Chifumi Nishioka, Jörn Hauke, and Ansgar Scherp. Influence of tweets and diversification on serendipitous research paper recommender systems. PeerJ Comput. Sci., 6:e273, 2020.
- [73] Malte Ostendorff. Contextual document similarity for content-based literature recommender systems. CoRR, abs/2008.00202, 2020.
- [74] Malte Ostendorff, Corinna Breitingner, and Bela Gipp. A qualitative evaluation of user preference for link-based vs. text-based recommendations of wikipedia articles. CoRR, abs/2109.07791, 2021.
- [75] Braja Gopal Patra, Vahed Maroufy, Babak Soltanalizadeh, Nan Deng, W. Jim Zheng, Kirk Roberts, and Hulin Wu. A content-based literature recommendation system for datasets to improve data reusability – a case study on gene expression omnibus (geo) datasets. Journal of Biomedical Informatics, 104:103399, 2020.
- [76] Dragomir R. Radev, Mark Thomas Joseph, Bryan Gibson, and Pradeep Muthukrishnan. A Bibliometric and Network Analysis of the field

of Computational Linguistics. Journal of the American Society for Information Science and Technology, 2009.

- [77] Dragomir R. Radev, Pradeep Muthukrishnan, and Vahed Qazvinian. The ACL anthology network corpus. In Proceedings, ACL Workshop on Natural Language Processing and Information Retrieval for Digital Libraries, Singapore, 2009.
- [78] Dragomir R. Radev, Pradeep Muthukrishnan, Vahed Qazvinian, and Amjad Abu-Jbara. The acl anthology network corpus. Language Resources and Evaluation, pages 1–26, 2013.
- [79] Behnam Rahdari and Peter Brusilovsky. User-controlled hybrid recommendation for academic papers. In Proceedings of the 24th International Conference on Intelligent User Interfaces: Companion, Marina del Ray, CA, USA, March 16-20, 2019, pages 99–100. ACM, 2019.
- [80] Behnam Rahdari, Peter Brusilovsky, Khushboo Thaker, and Jordan Barria-Pineda. Knowledge-driven wikipedia article recommendation for electronic textbooks. In Carlos Alario-Hoyos, María Jesús Rodríguez-Triana, Maren Scheffel, Inmaculada Arnedillo Sánchez, and Sebastian Dennerlein, editors, Addressing Global Challenges and Quality Education - 15th European Conference on Technology Enhanced Learning, EC-TEL 2020, Heidelberg, Germany, September 14-18, 2020, Proceedings, volume 12315 of Lecture Notes in Computer Science, pages 363–368. Springer, 2020.
- [81] S. Renuka, G. S. S. Raj Kiran, and Palakodeti Rohit. An unsupervised content-based article recommendation system using natural language processing. In I. Jeena Jacob, Selvanayaki Kolandapalayam Shanmugam, Selwyn Piramuthu, and Przemyslaw Falkowski-Gilski, editors, Data Intelligence and Cognitive Informatics, pages 165–180, Singapore, 2021. Springer Singapore.
- [82] Anna Safaryan, Petr Filchenkov, Weijia Yan, Andrey Kutuzov, and Irina Nikishina. Semantic recommendation system for bilingual corpus of academic papers. In Wil M. P. van der Aalst, Vladimir Batagelj, Alexey Buzmakov, Dmitry I. Ignatov, Anna A. Kalenkova, Michael Yu. Khachay, Olessia Koltsova, Andrey Kutuzov, Sergei O. Kuznetsov, Irina A. Lomazova, Natalia V. Loukachevitch,

- Ilya Makarov, Amedeo Napoli, Alexander Panchenko, Panos M. Pardalos, Marcello Pelillo, Andrey V. Savchenko, and Elena Tutubalina, editors, Recent Trends in Analysis of Images, Social Networks and Texts - 9th International Conference, AIST 2020, Skolkovo, Moscow, Russia, October 15-16, 2020 Revised Supplementary Proceedings, volume 1357 of Communications in Computer and Information Science, pages 22–36. Springer, 2020.
- [83] Nazmus Sakib, Rodina Binti Ahmad, Mominul Ahsan, Md. Abdul Based, Khalid Haruna, Julfikar Haider, and Saravanakumar Gurusamy. A hybrid personalized scientific paper recommendation approach integrating public contextual metadata. IEEE Access, 9:83080–83091, 2021.
- [84] Nazmus Sakib, Rodina Binti Ahmad, and Khalid Haruna. A collaborative approach toward scientific paper recommendation using citation context. IEEE Access, 8:51246–51255, 2020.
- [85] Abdul Samad, Muhammad Arshad Islam, Muhammad Azhar Iqbal, and Muhammad Aleem. Centrality-based paper citation recommender system. EAI Endorsed Trans. Ind. Networks Intell. Syst., 6(19):e2, 2019.
- [86] Philipp Schaer, Timo Breuer, Leyla Jael Castro, Benjamin Wolff, Johann Schaible, and Narges Tavakolpoursaleh. Overview of lilas 2021 - living labs for academic search (extended overview). In Guglielmo Faggioli, Nicola Ferro, Alexis Joly, Maria Maistro, and Florina Piroi, editors, Proceedings of the Working Notes of CLEF 2021 - Conference and Labs of the Evaluation Forum, Bucharest, Romania, September 21st - to - 24th, 2021, volume 2936 of CEUR Workshop Proceedings, pages 1668–1699. CEUR-WS.org, 2021.
- [87] Abdul Shahid, Muhammad Tanvir Afzal, Moloud Abdar, Mohammad Ehsan Basiri, Xujuan Zhou, Neil Y. Yen, and Jia-Wei Chang. Insights into relevant knowledge extraction techniques: a comprehensive review. J. Supercomput., 76(3):1695–1733, 2020.
- [88] Abdul Shahid, Muhammad Tanvir Afzal, Abdullah Alharbi, Hanan Aljuaid, and Shaha Al-Otaibi. In-text citation’s frequencies-based recommendations of relevant research papers. PeerJ Comput. Sci., 7:e524, 2021.

- [89] Abdul Shahid, Muhammad Tanvir Afzal, Muhammad Qaiser Saleem, M. S. Elsayed Idrees, and Majzoob K. Omer. Extension of direct citation model using in-text citations. Computers, Materials & Continua, 66(3):3121–3138, 2021.
- [90] Bhuvan Sharma, Van C Willis, Claudia S Huettner, Kirk Beaty, Jane L Snowdon, Shang Xue, Brett R South, Gretchen P Jackson, Dilhan Weeraratne, and Vanessa Michelini. Predictive article recommendation using natural language processing and machine learning to support evidence updates in domain-specific knowledge graphs. JAMIA Open, 3(3):332–337, 09 2020.
- [91] Hui Shi, Wei Ma, Xiaoliang Zhang, Junyan Jiang, Yanbing Liu, and Shujuan Chen. A hybrid paper recommendation method by using heterogeneous graph and metadata. In 2020 International Joint Conference on Neural Networks, IJCNN 2020, Glasgow, United Kingdom, July 19-24, 2020, pages 1–8. IEEE, 2020.
- [92] Arnab Sinha, Zhihong Shen, Yang Song, Hao Ma, Darrin Eide, Bo-June Paul Hsu, and Kuansan Wang. An overview of microsoft academic service (MAS) and applications. In Aldo Gangemi, Stefano Leonardi, and Alessandro Panconesi, editors, Proceedings of the 24th International Conference on World Wide Web Companion, WWW 2015, Florence, Italy, May 18-22, 2015 - Companion Volume, pages 243–246. ACM, 2015.
- [93] P. Subathra and P. N. Kumar. Recommending research article based on user queries using latent dirichlet allocation. In V. Sivakumar Reddy, V. Kamakshi Prasad, Jiacun Wang, and K. T. V. Reddy, editors, Soft Computing and Signal Processing, pages 163–175, Singapore, 2020. Springer Singapore.
- [94] Kazunari Sugiyama and Min-Yen Kan. Scholarly paper recommendation via user’s recent research interests. In Jane Hunter, Carl Lagoze, C. Lee Giles, and Yuan-Fang Li, editors, Proceedings of the 2010 Joint International Conference on Digital Libraries, JCDL 2010, Gold Coast, Queensland, Australia, June 21-25, 2010, pages 29–38. ACM, 2010.
- [95] Kazunari Sugiyama and Min-Yen Kan. Serendipitous recommendation for scholarly papers considering relations among researchers. In Glen Newton, Michael J. Wright, and Lillian N. Cassel, editors, Proceedings of the 2011 Joint International Conference on Digital Libraries, JCDL

- 2011, Ottawa, ON, Canada, June 13-17, 2011, pages 307–310. ACM, 2011.
- [96] Kazunari Sugiyama and Min-Yen Kan. Exploiting potential citation papers in scholarly paper recommendation. In J. Stephen Downie, Robert H. McDonald, Timothy W. Cole, Robert Sanderson, and Frank Shipman, editors, 13th ACM/IEEE-CS Joint Conference on Digital Libraries, JCDL '13, Indianapolis, IN, USA, July 22 - 26, 2013, pages 153–162. ACM, 2013.
- [97] Kazunari Sugiyama and Min-Yen Kan. A comprehensive evaluation of scholarly paper recommendation using potential citation papers. Int. J. Digit. Libr., 16(2):91–109, 2015.
- [98] Panagiotis Symeonidis, Lidija Kirjackaja, and Markus Zanker. Session-based news recommendations using simrank on multi-modal graphs. Expert Syst. Appl., 180:115028, 2021.
- [99] Hao Tang, Baisong Liu, and Jiangbo Qian. Content-based and knowledge graph-based paper recommendation: Exploring user preferences with the knowledge graphs for scientific paper recommendation. Concurr. Comput. Pract. Exp., 33(13), 2021.
- [100] Jie Tang, Jing Zhang, Limin Yao, Juanzi Li, Li Zhang, and Zhong Su. Arnetminer: extraction and mining of academic social networks. In Ying Li, Bing Liu, and Sunita Sarawagi, editors, Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Las Vegas, Nevada, USA, August 24-27, 2008, pages 990–998. ACM, 2008.
- [101] William Tanner, Esra Akbas, and Mir Hasan. Paper recommendation based on citation relation. In Chaitanya Baru, Jun Huan, Lati-fur Khan, Xiaohua Hu, Ronay Ak, Yuanyuan Tian, Roger S. Barga, Carlo Zaniolo, Kisung Lee, and Yanfang Fanny Ye, editors, 2019 IEEE International Conference on Big Data (Big Data), Los Angeles, CA, USA, December 9-12, 2019, pages 3053–3059. IEEE, 2019.
- [102] Min Tao, Xinmin Yang, Gao Gu, and Bohan Li. Paper Recommend Based on LDA and PageRank, pages 571–584. Springer, 09 2020.
- [103] Waleed Waheed, Muhammad Imran, Basit Raza, Ahmad Kamran Malik, and Hasan Ali Khattak. A hybrid approach toward research paper recommendation using centrality measures and author ranking. IEEE Access, 7:33145–33158, 2019.

- [104] Bangchao Wang, Ziyang Weng, and Yanping Wang. A novel paper recommendation method empowered by knowledge graph: for research beginners. CoRR, abs/2103.08819, 2021.
- [105] Gang Wang, Hanru Wang, Ying Yang, Dong-Ling Xu, Jian-Bo Yang, and Feng Yue. Group article recommendation based on ER rule in scientific social networks. Appl. Soft Comput., 110:107631, 2021.
- [106] Gang Wang, Xinyue Zhang, Hanru Wang, Yan Chu, and Zhen Shao. Group-oriented paper recommendation with probabilistic matrix factorization and evidential reasoning in scientific social network. IEEE Transactions on Systems, Man, and Cybernetics: Systems, pages 1–15, 2021.
- [107] Hao Wang, Binyi Chen, and Wu-Jun Li. Collaborative topic regression with social regularization for tag recommendation. In Francesca Rossi, editor, IJCAI 2013, Proceedings of the 23rd International Joint Conference on Artificial Intelligence, Beijing, China, August 3-9, 2013, pages 2719–2725. IJCAI/AAAI, 2013.
- [108] Xiao Wang, Hanchuan Xu, Wenjie Tan, Zhongjie Wang, and Xiaofei Xu. Scholarly paper recommendation via related path analysis in knowledge graph. In 2020 International Conference on Service Science, ICSS 2020, Xining, China, August 24-26, 2020, pages 36–43. IEEE, 2020.
- [109] Jian Wu, Kunho Kim, and C. Lee Giles. Citeseerx: 20 years of service to scholarly big data. In Huajin Wang and Keith Webster, editors, Proceedings of the Conference on Artificial Intelligence for Data Discovery and Reuse, AIDR 2019, Pittsburgh, PA, USA, May 13-15, 2019, pages 1:1–1:4. ACM, 2019.
- [110] Yi Xie, Yuqing Sun, and Elisa Bertino. Learning domain semantics and cross-domain correlations for paper recommendation. In Fernando Diaz, Chirag Shah, Torsten Suel, Pablo Castells, Rosie Jones, and Tetsuya Sakai, editors, SIGIR '21: The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, Canada, July 11-15, 2021, pages 706–715. ACM, 2021.
- [111] Yi Xie, Shaoqing Wang, Wei Pan, Huaibin Tang, and Yuqing Sun. Embedding based personalized new paper recommendation. In Yuqing Sun, Dongning Liu, Hao Liao, Hongfei Fan, and Liping Gao, editors,

Computer Supported Cooperative Work and Social Computing, pages 558–570, Singapore, 2021. Springer Singapore.

- [112] Qiang Yang, Zhixu Li, An Liu, Guanfeng Liu, Lei Zhao, Xiangliang Zhang, Min Zhang, and Xiaofang Zhou. A novel hybrid publication recommendation system using compound information. World Wide Web, 22(6):2499–2517, 2019.
- [113] Mei Yu, Yue Hu, Xuwei Li, Mankun Zhao, Tianyi Xu, Hongwei Liu, Linying Xu, and Ruiguo Yu. Paper recommendation with item-level collaborative memory network. In Christos Douligeris, Dimitris Karagiannis, and Dimitris Apostolou, editors, Knowledge Science, Engineering and Management - 12th International Conference, KSEM 2019, Athens, Greece, August 28-30, 2019, Proceedings, Part I, volume 11775 of Lecture Notes in Computer Science, pages 141–152. Springer, 2019.
- [114] Yongfeng Zhang and Xu Chen. Explainable recommendation: A survey and new perspectives. Found. Trends Inf. Retr., 14(1):1–101, 2020.
- [115] Yu Zhang, Min Wang, Florian Gottwalt, Morteza Saberi, and Elizabeth Chang. Ranking scientific articles based on bibliometric networks with a weighting scheme. Journal of Informetrics, 13(2):616–634, 2019.
- [116] Yu Zhang, Min Wang, Morteza Saberi, and Elizabeth Chang. Towards expert preference on academic article recommendation using bibliometric networks. In Wei Lu and Kenny Q. Zhu, editors, Trends and Applications in Knowledge Discovery and Data Mining, pages 11–19, Cham, 2020. Springer International Publishing.
- [117] Xu Zhao, Hui Kang, Tie Feng, Chenkun Meng, and Ziqing Nie. A hybrid model based on LFM and bigru toward research paper recommendation. IEEE Access, 8:188628–188640, 2020.
- [118] Yifan Zhu, Qika Lin, Hao Lu, Kaize Shi, Ping Qiu, and Zhendong Niu. Recommending scientific paper via heterogeneous knowledge embedding based attentive recurrent neural networks. Knowl. Based Syst., 215:106744, 2021.

8. RevASIDE: Evaluation of Assignments of Suitable Reviewer Sets for Publications from Fixed Candidate Pools

Outline

8.1	Introduction	199
8.2	Related Work	201
8.2.1	Team Formation	201
8.2.2	Expert Search	202
8.2.3	Reviewer Set Recommendation	203
8.3	Aspects	205
8.3.1	Aspect 1: No Conflicts of Interests	205
8.3.2	Aspect 2: Disjoint Publications	205
8.3.3	Aspect 3: Expertise	205
8.3.4	Aspect 4: Authority	206
8.3.5	Aspect 5: Diverse Expertise (Diversity)	206
8.3.6	Aspect 6: Current Interest	206
8.3.7	Aspect 7: Diverse Experience (Seniority)	206
8.4	Approach	207
8.4.1	Step 1: Expert Search	207
8.4.2	Step 2: Reviewer Set Assignment	209
	Expertise E	210
	Authority A	211
	Diverse Expertise (Diversity) D	211
	Current Interest I	211
	Diverse Experience (Seniority) S	212
	Final Equation	212
8.4.3	Run Time Analysis	213

8.5	Datasets	214
8.5.1	Data Acquisition	215
8.5.2	Document Representations	216
8.5.3	MOL'17, BTW'17 and ECIR'17	217
8.5.4	Discussion and Challenges	217
8.6	Evaluation: Preface	218
8.6.1	Hypotheses	219
8.7	Evaluation: Step 1 - Expert Search Task	219
8.7.1	Setting	221
8.7.2	Results (Analysis of H_1 and H_2)	222
8.8	Evaluation Step 2 - Reviewer Set Assignment Task	224
8.8.1	Setting	224
8.8.2	Quantitative Evaluation	225
	Significance of Factors (Analysis of H_3)	225
	Configurations achieving highest Scores (Analysis of H_4 and H_5)	226
	Highest Values for Aspects	228
	Number of Reviewers R_c (Analysis of H_6)	229
	Run Time (Analysis of H_6)	231
8.8.3	Qualitative Evaluation (Analysis of H_7)	232
	Best Configurations from Automatic Evaluation	232
	Best Configurations From Step 1	234
8.9	Conclusion and Future Work	237
	Bibliography	239

Bibliographic Information

Kreutz, C. K., Schenkel, R. (to appear). RevASIDE: Evaluation of Assignments of Suitable Reviewer Sets for Publications from Fixed Candidate Pools. In *Journal of Data Intelligence* (to appear). Rinton Press.

Copyright Notice

©2022 Rinton Press. This is an accepted but reformatted version of this article. Clarification of the copyright adjusted according to the guidelines of the publisher.

Keywords

Reviewer Assignment • Recommendation System • Expertise Modelling

Abstract

Scientific publishing heavily relies on the assessment of quality of submitted manuscripts by peer reviewers. Assigning a set of matching reviewers to a submission is a highly complex task which can be performed only by domain experts.

We introduce and deeply evaluate RevASIDE, a reviewer recommendation system that assigns suitable sets of complementing reviewers from a predefined candidate pool without requiring manually defined reviewer profiles. Here, suitability includes not only reviewers' expertise, but also their authority in the target domain, their diversity in their areas of expertise and experience, and their interest in the topics of the manuscript.

We present three new data sets for the expert search and reviewer set assignment tasks and compare the usefulness of simple text similarity methods to document embeddings for expert search. We analyse the appropriateness of the approach for different sizes of reviewer sets. Furthermore, a quantitative evaluation demonstrates significantly better results in reviewer set assignment compared to baselines. A qualitative evaluation also shows their superior perceived quality.

8.1 Introduction

Peer review is a popular method of ensuring scientific standards for conferences and journals. It requires the assignment of suitable experts for each submission, which is often done manually [7]. These reviewers then provide objective assessment of the manuscript and recommend accepting or rejecting the submission [10]. All of this has to be performed in a tight time frame [3].

The continuously increasing number of submissions as well as the high complexity of the task even for experienced chairs of program committees (PC) or journal editors calls for fully automatic methods of expert assignment. Furthermore, it is not sufficient to focus on the quality of single reviewers, but a good set of complementing reviewers should be recommended for each manuscript. The reviewer assignment problem tackles the task of retrieving sets of suitable reviewers for manuscripts submitted to a venue.

Even though the construction of sets of reviewers fitting submitted manuscripts has been studied frequently, most work focuses on construction of sets with the highest possible expertise but does not consider (m)any other aspects. Such other aspects could help reduce reviewers' work load and increase comprehensiveness of reviews. Additionally, actual human evaluation of the sets and thus a reliable confirmation of results is generally not conducted.

Numerous works [6, 10, 12, 13, 14, 19, 27, 29] tackle the reviewer assignment problem in different ways, with slightly different definitions for the suitability of reviewers. While expertise of a reviewer with the topic of the manuscript [6, 10, 12, 13, 14, 27, 29] has been dominating in existing work, other features like authority [12, 13], research interest [12] and diversity [10, 13, 19] were considered in some existing work, but not in a holistic way. Additionally, these aspects were defined heterogeneously in present works. We incorporate the following five aspects into our definition of suitability of reviewer sets: *expertise* of reviewers in general topics and methods of a submission, *authority* of reviewers in the domain of the manuscript, *diversity* in terms of reviewers differing in their areas of expertise, *interest* of reviewers in the topics of the submission and diversity in terms of *seniority* aspects of the reviewer set.

In this work, we embark on finding the best reviewer sets for a submitted scientific paper from a predefined candidate pool in terms of these five aspects. Unlike some existing work [1, 21], we explicitly do not require the manual definition of keywords or bids on manuscripts from reviewer candidates¹. To achieve this, we make two important contributions: 1) We propose and thoroughly evaluate RevASIDE, a new and completely automated technique for recommending sets of reviewers from a fixed set of candidates for single manuscripts. For this, we introduce seniority as a completely new aspect and its combination with already established but redefined features. 2) We publish three different datasets suitable for expert search as well as reviewer set assignment.

While we build on established expert retrieval methods to find reviewers with high expertise, our method is the first to incorporate all of the complementary factors authority, diversity, interest of candidates and seniority to solve the reviewer set assignment problem. Our approach consists of two steps. Step 1 identifies topically relevant reviewers based on the similarity of their research direction to the manuscript, utilising expert search methods. Step 2 then assembles sets from these experts and determines the reviewer set that performs best in the five aspects. To the best of our knowledge, this is the first work that utilises the expert search task as a preparatory step for the reviewer set assignment task.

This paper is an extended version of the work presented at iiWAS'21 [15]. The main extensions are contained in the Sections 8.2.1, 8.4.3, 8.5.4, 8.8.2, 8.8.2 and 8.8.2.

¹Assignments containing bidding information could be problematic as they require e.g. the randomisation of the order of presented manuscripts to reviewers or the explicit promotion of bid-less manuscripts [5].

8.2 Related Work

The reviewer assignment problem is closely related to the *team or group formation problem*, as reviewer assignment can be seen as a specialised form of group formation.

Retrieval-based approaches for scientific reviewer assignment treat the manuscript for which reviewers are searched as a query. They determine fitting reviewers based on different aspects, often under additional constraints. Such methods can be divided into ones recommending single reviewers for manuscripts, so-called expert search, and those tackling the assignment of whole reviewer sets.

8.2.1 Team Formation

The team or group formation problem has the goal of retrieving sets of experts with different skills best suited for a specific task or use case [20]. It is relevant in a multitude of disciplines, for example in social sciences, product design, product marketing campaigns, customer services [24], participatory sensing or software product development [8].

Group formation majorly differs from the reviewer assignment problem as in the former, the task for the group oftentimes is of collaborative nature [8]. Collaboration in groups needs to be effective to achieve the highest possible productivity [8] but it also comes with coordination costs. Minimising them would result in groups of experts being close or similar to each other, which in general is rather undesirable for recommended sets of reviewers [1, 2]. Reviewer assignment thus can be defined as a variation of the group formation problem with a non-collaborative task and therefore without coordination costs.

Seleznova et al. [24] propose a group exploration framework utilising reinforcement learning. They recommend exploration actions suitable for different user datasets and tasks. Retrieved candidate groups are relevant targets and increase the overall exploration quality, e.g. in diversity or coverage. Nikolakai et al. [20] propose team formation where these teams do not need to meet all of the requirements of the tasks, but instead only cover tasks partially. They assume the lower the load of experts, the higher is their performance.

A specialised form of group formation utilises social networks and surpasses the mere resource allocation problem: Datta et al. [8] work on forming effective teams which meet the requirements of tasks by utilising social connections of members in social networks. Their approach minimises different social collaboration cost measures to identify the optimal teams for multi-

ple tasks while not overburdening the respective experts. Anagnostopoulos et al. [2] study automated online team formation for a stream of different tasks where the skills and compatibility of experts are modelled in a social network. They also minimise coordination costs in teams and propose a fair allocation of experts to tasks.

8.2.2 Expert Search

Several papers target the recommendation of single reviewers for manuscripts, which contrasts our goal of recommending reviewer sets. We identify and assemble the best fitting experts to a suitable set, while the following works only handle the expert search task, which disregards set effects.

Numerous works pursue the expert search task as a matching problem between the query manuscript and expert profiles formed by their past publications. Some of them also consider more aspects than textual similarity: MINARET [23] is a recommendation framework based on publications and affiliations of experts as well as expanded keywords for manuscripts. After an initial filtering step, it returns a ranked list of reviewers. Candidates receive a score based on topical coverage, impact, recency, experience in reviewing and their familiarity with the target venue. Chughtai et al. [7] suggest ontology-based and topic-specific recommendation of single experts fitting a submission. Macdonald and Ounis [17] propose twelve voting techniques to find suitable experts for query manuscripts. These techniques base on similarity of the reviewer candidates and the manuscripts. We use and extend their methods in Step 1 of our approach.

Other works transform single expert finding into a classification problem: Yang et al. [28] base their approach on word-semantic relatedness via Wikipedia. Reviewers are ranked with respect to a manuscript by experience in the domain of the submission and their number of papers. Zhao et al. [31] utilise word embeddings of keywords from author profiles and manuscripts to suggest fitting reviewers. Similar to this approach, we use embedding methods to abstract from words while searching for reviewer candidates.

Tran et al. [26] pursue another direction by defining expert search as a task between a single expert and a group of researchers instead of an expert and a query. They incorporate only non-textual features such as citation information or co-authorships depending on venues in their recommendation approach for experts given an existing program committee.

Table 8.1: Observed properties expertise (E), authority (A), diversity (D), interest (I) and seniority (S) in related work (● indicates a paper covers this aspect but might define it differently from us) as well as indication if the approach is targeting the whole venue (wv?) or can be fully automated (fa?).

Approach	E	A	D	I	S	wv?	fa?
Charlin and Zemel [3]	●					yes	yes
Ishag et al. [10]	●		●			no	yes
Jin et al. [12]	●	●		●		yes	yes
Kalmukov [13]	●					yes	yes
Kou et al. [14]	●					yes	yes
Liu et al. [13]	●	●	●			yes	yes
Maleszka et al. [19]	●		●			no	no
Papagelis et al. [21]	●			●		yes	no
Tang et al. [22]	●	●				yes	yes
Yang et al. [27]	●					yes	yes
Zhang et al. [29]	●					no	yes
RevASIDE	●	●	●	●	●	no	yes

8.2.3 Reviewer Set Recommendation

Reviewer set recommendation can be observed for *single papers* or *multiple/all papers of a venue*. The following approaches tackle reviewer set recommendation but consider different or fewer aspects compared to RevASIDE for estimating the quality of reviewer sets. Table 8.1 compares the presented approaches in a coherent form.

Ishag et al. [10] incorporate the h index of reviewers, citation counts and paper diversity into their approach based on itemset mining. They return reviewer sets fitting a query manuscript and estimate the sets' impact. Contrasting their definition of diversity which uses the number of different affiliations of authors of a single paper, we define diversity as a measure between authors to estimate the actual topical differences in reviewer sets. Maleszka et al. [19] tackle the reviewer set assignment problem for one manuscript at a time by focusing on diversity aspects in expertise, the co-authorship graph and style of reviewers. They begin the set recommendation process with a single reviewer determined by another method. Zhang et al. [29] utilise a multi-label classifier for the construction of reviewer sets. The approach bases on predicted research labels for manuscripts and predicts reviewers with similar labels. Set-based effects are ignored, which contrasts our approach.

Works tackling the reviewer set recommendation for *multiple papers* can

be divided in ones relying on *manual inputs* such as bidding by reviewers and *fully automated* ones. Some of the papers incorporating *manual inputs*, contrasting our fully automated method, are the following: The Toronto Paper Matching System (TPMS) [3] conducts automatic reviewer assignment for all manuscripts submitted to a conference by using either word count representation or LDA topics, but can also incorporate reviewers' bids on submissions. TPMS supports some constraints: papers must be reviewed by three reviewers, and reviewers are assigned not more than a given limit of papers. Reviewers for manuscripts are determined based on expertise extracted from their publications. TPMS is applied, for example, by the SIGMOD research track [1], where reviewers upload a representative set of their publications. Papagelis et al. [21] present a system which incorporates reviewers' interests in terms of paper topics, their bids on papers, conflicts of interests and overall workload balance for the reviewer assignment task. It can either assign reviewer sets automatically if the bidding is completed or the PC chair can manually adjust the sets.

The following works are *fully automated* recommendation approaches intended to work with multiple manuscripts²: Liu et al. [13] recommend n reviewers for each manuscript, which are dependent on each other. They model reviewers' expertise, authority and diversity as a graph which they traverse with random walk with restart. The number of co-authorships is modelled as authority, which contrasts our definition of authority. Kou et al. [14] introduce an assignment system for sets of n reviewers which bases on the topic distributions of reviewers and the manuscripts computed with the Author-Topic Model. They define expertise of reviewer sets in certain topics as the maximum expertise for the topic found in the set; our definition of expertise deviates. Jin et al. [12] assume reviewers have a certain relevance in a topic, which is determined by their publications and usage of the Author-Topic Model. Additionally, authority in form of citations and research interest of researchers are important factors. Here, the number of reviewers per paper and the maximum number of papers a reviewer is assigned to can be predefined. Amongst others, we also observe these factors but define them differently. Yang et al. [27] utilise LDA to represent manuscripts as well as past publications of reviewer candidates. They then use a discrete optimisation model which focuses on expertise to assign reviewers to all manuscripts. Likewise, we also incorporate LDA in our approach, but we additionally consider more aspects beyond expertise. Kalmukov [13]

²Note that we currently refrain from this task as it would require an evaluation dataset which includes all submissions to the venue, even the rejected ones and their authors. Such a dataset does not exist currently, to the best of our knowledge.

uses a weighted bipartite graph to compute sets of reviewers for multiple manuscripts and solely focuses on maximising the expertise for each one. Additionally, he incorporates the worthiness of a manuscript to be assigned to a reviewer and equally distributes reviewing load. He prioritises assignments for papers which have a low number of possible reviewers. Contrasting this approach, we incorporate more factors and do not solely strive to maximise experience of reviewer sets. Tang et al. [22] propose a constraint-based optimisation framework that proposes sets of reviewers for query manuscripts and user feedback if available. They incorporate expertise matching, authority aspects, load balance and want to maximise the topic coverage between reviewer sets and the manuscripts. For this, they utilise LDA, which we also use. A major difference is their definition of authority, they define different expertise levels similar to our concept of seniority.

8.3 Aspects

In our work, we assess the appropriateness of a reviewer set with respect to a submission based on the following seven aspects:

8.3.1 Aspect 1: No Conflicts of Interests

Reviewers in a reviewer set cannot have *conflicts of interests*: they can be neither authors of the submission nor prior co-authors of its authors [21]. This aspect aims at ensuring unbiased and objective candidates. While we (as well as others [21]) regard this aspect quite vigorously, less restrictive variants (e.g. disallowing co-authorships in the three years prior to the submission) are also feasible.

8.3.2 Aspect 2: Disjoint Publications

Reviewers cannot be co-authors of any other reviewer in the set. Reviewers having *disjoint publications* enforces a broader spectrum of different backgrounds. This could produce broader reviews [19] which is a desirable property in peer review [1].

8.3.3 Aspect 3: Expertise

Reviewers need to be *experienced* in the area of the manuscript [14]. The topic of the paper should be relevant for them and fit their research profile.

Not only the content, but also the number of papers in the area of a submission contributes to our understanding of experience. This aspect ensures deep reviews, another desirable feature of assessments [1].

8.3.4 Aspect 4: Authority

Reviewers need to hold *authority* in the research area of the submission. Reviews of the papers have to be credible, reviewers should be well recognised in the target domain [13]. Authority can be assessed, for example, by an area-dependent *h* index and citation counts of candidates.

8.3.5 Aspect 5: Diverse Expertise (Diversity)

Reviewers need to be *diverse* in their area of expertise. Typically, as many topics as possible of a submission should be assessed to create a comprehensive review [1]. Reviewers that are proficient in different topics from each other support this goal, as the candidates in a set have unique perspectives formed by their different experiences and backgrounds [19].

8.3.6 Aspect 6: Current Interest

Reviewers need to be *currently interested* in the topics of the manuscripts so they accept the reviewing request [12] and are not asked to review topics they no longer work in. Scientific progress makes it impossible to be up-to-date in all areas they were formerly interested in. Thus, time-aware suggestion should weigh recent works of reviewers much higher than older publications. If reviewers are interested in the area of the manuscript (e.g. signalled by bidding on a paper), they should be able to provide sharp and confident reviews [5].

8.3.7 Aspect 7: Diverse Experience (Seniority)

Reviewers of a manuscript should not solely consist of *senior* researchers, but they need to be diverse with respect to the amount of their experience. Senior researchers provide vast reviewing experience and a global vision, but they should be handled as a sparse resource as they are asked to review many submissions. Junior researchers are ambitious and resilient while not having that much experience. Usually, they are less frequently asked to review and more of an unexhausted resource. Reviewing load needs to be distributed between senior and junior researchers, such that the lower load for senior researchers and incorporation of newer researchers benefits the overall

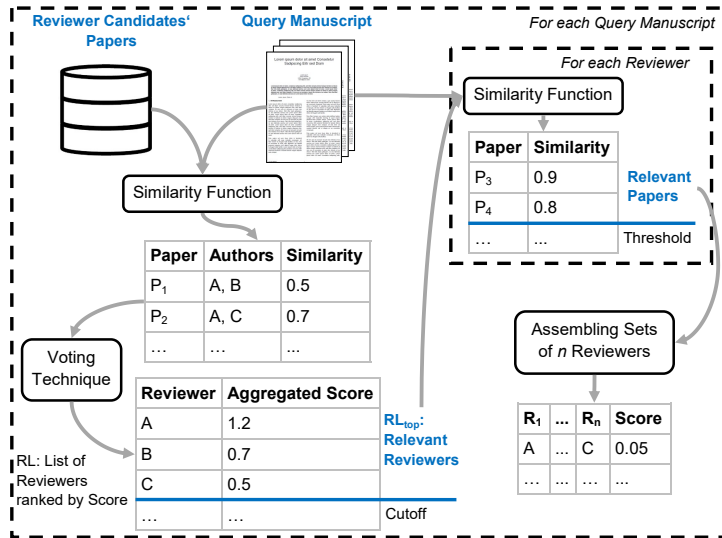


Figure 8.1: Schematic overview of our approach. The left part depicts the expert search task, the right part depicts the set of reviewers assignment task.

quality of reviews. Additionally, junior researchers could provide new and refreshing perspectives, while the reviewing activity might also benefit their own development. Breaking up well-established reviewer constellations with new candidates could also avoid research cliques [3].

8.4 Approach

RevASIDE is a system for assigning sets of **Reviewers** utilising **Authority**, **Seniority**, **Interest**, **Diversity** and **Expertise** of reviewers to find the most suitable reviewer set out of a fixed set of candidates, the reviewer candidate pool RCP , for a given manuscript M . Our approach is composed of two steps: in Step 1, suitable reviewers are identified from the pool of reviewer candidates; in Step 2, they are assembled to the most suitable set for the manuscript. Figure 8.1 depicts the schematic overview of our approach.

8.4.1 Step 1: Expert Search

Step 1 handles the left part of Figure 8.1. We represent publications as tf-idf vectors or ones constructed with BERT [9] or Doc2Vec [16], which allows depicting semantics of documents instead of single tokens. This enables capturing similarity of concepts of papers.

Table 8.2: Voting techniques VT and accompanying formulas for reviewer R and manuscript M .

VT	Formula
$Votes_\delta$	$\sum_{P \in P(R) \wedge sim(P, M) \geq \delta} 1$
SUM	$\sum_{P \in P(R)} sim(P, M)$
AVG	$\frac{\sum_{P \in P(R)} sim(P, M)}{ P(R) }$
MNZ	$ P(R) * \sum_{P \in P(R)} sim(P, M)$
SUM_n	$\sum_{P \in P(R) \wedge rank(P, R, M) \leq n} sim(P, M)$
MIN	$min(\{sim(P, M) P \in P(R)\})$
MAX	$max(\{sim(P, M) P \in P(R)\})$
RR	$\sum_{P \in P(R)} \frac{1}{rank(P, M)}$
mRR	$\frac{1}{ P(R) } \sum_{P \in P(R)} \frac{1}{rank(P, M)}$
$BordaFuse$	$\sum_{P \in P(R)} (\bigcup_{R_i \in RCP} P(R_i) - rank(P, M))$
exp^{SUM}	$\sum_{P \in P(R)} e^{sim(P, M)}$
exp^{AVG}	$\frac{\sum_{P \in P(R)} e^{sim(P, M)}}{ P(R) }$
exp^{MNZ}	$ P(R) * \sum_{P \in P(R)} e^{sim(P, M)}$

Let M be the manuscript for which a reviewer set should be computed. We ignore any reviewers for which a conflict of interest with the authors of M exists (Aspect 1). For the remaining reviewers from the reviewer candidate pool RCP , let $P(R)$ be the set of publications written by reviewer R . The similarity between a publication P and a manuscript M is given by $sim(P, M)$; the utilised similarity measure can be changed between the two steps. In our experiments, we will use the cosine similarity of the corresponding vectors. We then sort R 's papers in descending order by their similarity to the manuscript M and denote by $rank(P, R, M)$ the rank of a certain publication P of reviewer R in this order. Similarly, we sort all publications in the collection in descending order by their similarity to manuscript M and denote by $rank(P, M)$ the rank of a publication P in this order.

To obtain a ranked list RL of reviewers, we apply a number of voting techniques (VTs) that score reviewer candidates with respect to a manuscript. These voting techniques base on the ones applied by Macdonald and Ounis [17] for expert search. Table 8.2 shows the exact formulas for the 13 voting techniques considered in our approach. Higher scores signal better fit of a reviewer to the given manuscript. $Votes_\delta$ computes the number of papers of a reviewer with a similarity to the query manuscript not smaller than a threshold δ ; note that the method was introduced without such a threshold in [17], which corresponds to $\delta = 0$ in our definition. SUM sums

up the similarities of the papers of a reviewer with the query manuscript, *AVG* uses this score and normalizes it by the total number of papers of the reviewer. *MNZ* multiplies the *SUM* score by the number of papers of the reviewer. SUM_n sums the similarities of the n papers of the reviewer most similar to the manuscript. *MIN* returns the smallest similarity of the reviewer’s paper with the manuscript, *MAX* is defined analogously. *RR* sums up the reciprocal ranks of the reviewer’s papers in the ordered list of all papers. We additionally introduce *mRR* which normalizes this score by the number of papers written by the reviewer. *BordaFuse* utilises Borda-fuse as score. The three voting techniques exp^{SUM} , exp^{ANZ} and exp^{MNZ} are defined as their non-exponential forms but instead of using similarities, they apply the exponential function on similarities.

For a fixed voting technique, this step generates a ranked list *RL* of reviewers, i.e. experts, fitting the manuscript in question.

8.4.2 Step 2: Reviewer Set Assignment

Step 2 handles the right part of Figure 8.1, i.e. the actual formation of reviewer sets for manuscript *M* based on the ranked list *RL* of reviewers generated in Step 1. We denote the top k reviewers from *RL* by RL_{top} ; if $k = |RL|$, the first step becomes irrelevant. A smaller k restricts the observed candidates in the second step drastically and is especially useful to improve run time.³

We now represent documents by term-based vectors weighted with tf-idf and by topic-based vectors computed with LDA [3]; this allows us to capture concrete terms as well as general topics of publications of reviewer candidates and the submission.⁴ Additionally, these document vector representations allow us to easily weight and combine vectors of publications without destroying their expressiveness as each vector dimension represents a single token or topic which can be present in a document to a certain extent. This starkly contrasts BERT or Doc2Vec embeddings, where single dimensions do not have a comprehensible semantics, but instead the combination of all dimensions represents a document entirely. These tf-idf and LDA vectors can be constructed either on all parts of manuscripts or only on the technical sections, which consist of the methodology as well as the evaluation.

For each reviewer *R* this step considers the set $r_t(R, M)$ of their publications whose similarity to manuscript *M* is not lower than a threshold t ; i.e.

³The influence of cut-off k on the overall performance is evaluated in Section 8.8.2.

⁴Both for tf-idf and LDA vector representations of documents, values in all dimensions are non-negative.

$r_t(R, M) = \{P | P \in P(R) \wedge sim(P, M) \geq t\}$, with $t \in [0, 1]$. The threshold is utilised to define the selectivity of the research area relevant for the submission. If $t = 0$ all papers of a reviewer are included, a value closer to 1 restricts the number of papers taken into account in the second step. We assume similarities lie in $[0, 1]$.

Let $rep(P, V)$ be the representation of publication P as a vector of type $V \in \{L, T\}$ with L representing LDA vectors and T representing tf-idf vectors. Both document vector representations (DVs) can be used to compute $r_t(R, M)$, e.g. using the cosine of the corresponding vectors as a similarity function.

Lastly, let $P_{V,R,M,t} = \frac{\sum_{P \in r_t(R,M)} rep(P,V)}{\|\sum_{P \in r_t(R,M)} rep(P,V)\|_2}$ be the length normalized aggregation vector of type V that combines all information on relevant publications of a reviewer R with respect to M .

We now consider all possible candidate reviewer sets of a predefined size (for example 3) and assess, for each candidate set R_c , its suitability with respect to the aspects defined in Section 8.3. We prohibit reviewers in a set R_c to be co-authors of each other (Aspect 2); sets that include such reviewers are not considered further, they are assigned a final *score* of 0. In addition, we observe five different quantifiable aspects for suitability for each such set R_c of reviewer candidates. These reviewers are taken from RL_{top} produced in Step 1. Scores for all aspects are normalised to $[0, 1]$ with 1 being the best and 0 being the worst possible value.

Expertise E

Expertise describes the relevance of the reviewers in a set to the manuscript (Aspect 3). Reviewers should have solid knowledge with terms and topics of the manuscript, substantiated by numerous publications. Particularly, the submission should be similar to publications written by the reviewers [13] and their number of such papers should be high. Contrasting Liu et al.’s work [13] we use the number of co-authorships of reviewer candidates not as an indicator of authority but rather as an indicator of expertise. In E_2 we utilise an adapted definition of Cabanac [4] who states the topical similarity between researchers can be measured by the cosine similarity between their tf-idf vectors. These conditions are measured by the following scores:

$$E_1(R_c, M, t) = \frac{\sum_{R_i \in R_c} sim(P_{L,R_i,M,t}, M)}{|R_c|}$$

$$E_2(R_c, M, t) = \frac{\sum_{R_i \in R_c} sim(P_{T,R_i,M,t}, M)}{|R_c|}$$

$$E_3(R_c, M, t) = \frac{\sum_{R_i \in R_c} |r_t(R_i, M)|}{|R_c| \cdot \max_{R \in RL_{top}} |r_t(R, M)|}$$

These scores are then linearly combined to the final expertise score, with $\epsilon_i \in [0, 1]$ weighting parameters and $\epsilon_1 + \epsilon_2 + \epsilon_3 = 1$:

$$E(R_c, M, t) = \epsilon_1 E_1(R_c, M, t) + \epsilon_2 E_2(R_c, M, t) + \epsilon_3 E_3(R_c, M, t)$$

Authority A

Reviewers should hold authority in the area the manuscript belongs to (Aspect 4). We propose two scores to measure authority: the average h index of reviewers [13] $h(R, M, t)$ calculated on papers relevant to the manuscript $r_t(R, M)$ (measured by A_1), and the average number of their obtained citations on these papers (measured by A_2):

$$A_1(R_c, M, t) = \frac{\sum_{R_i \in R_c} h(R_i, M, t)}{|R_c| \cdot \max_{R \in RL_{top}} h(R, M, t)} \quad (8.1)$$

$$A_2(R_c, M, t) = \frac{\sum_{R_i \in R_c} \sum_{P_j \in r_t(R_i, M)} c(P_j)}{|R_c| \cdot \max_{R \in RL_{top}} \sum_{P \in r_t(R, M)} c(P)}$$

with $c(P)$ being the number of citations a paper P has obtained. These scores are then linearly combined to the final authority score, with $\alpha \in [0, 1]$ a weighting parameter:

$$A(R_c, M, t) = \alpha A_1(R_c, M, t) + (1 - \alpha) A_2(R_c, M, t)$$

Diverse Expertise (Diversity) D

We define diversity as a measure to ensure that the expertise of reviewers is distributed to areas as disjunct as possible (Aspect 5). This allows for reviews to cover multiple aspects of the manuscript. The corresponding score rewards if topics in which reviewers are proficient overlap as little as possible [13]:

$$D(R_c, M, t) = 1 - \frac{\sum_{R_i, R_j \in R_c, i < j} \text{sim}(P_{L, R_i, M, t}, P_{L, R_j, M})}{|R_c| \cdot (|R_c| - 1)/2}$$

Current Interest I

As research objectives of scientists change over time, interest measures the fit of reviewers and the manuscript with respect to their temporal development (Aspect 6). Interest of reviewers denotes their willingness to review

submissions from certain areas [12]. These interests change over time. If a reviewer was involved in a topic several years ago but then changed their focus, they probably no longer follow the rapid developments in the former research area. Thus, they might not be willing or even able to review current submissions from this area. To represent the time-aware profiles of reviewers, we combine the publications of reviewers with regard to their age to a length-normalized vector where recent papers are weighted stronger than older ones. This measure works on topical representations of documents:

$$I(R_c, M, t) = |R_c|^{-1} \cdot \sum_{R_i \in R_c} \text{sim} \left(\frac{\sum_{P_j \in r_t(R_i, M)} \frac{\text{rep}(P_j, L)}{a(P_j)}}{\|\sum_{P_j \in r_t(R_i, M)} \frac{\text{rep}(P_j, L)}{a(P_j)}\|_2}, M \right) \quad (8.2)$$

with $a(P)$ describing the age of a publication P in years.

Diverse Experience (Seniority) S

In terms of seniority, reviewer sets are desirable which do not solely consist of senior researchers (Aspect 7). In the recommended group of candidates, at least one senior researcher should be contained who is familiar with the methodology of the paper [1] (measured by S_2). Further it is desirable to have a diverse group in terms of seniority, the set should include at least one junior researcher (measured by S_1). These requisitions are modelled in the following equations:

$$S_1(R_c, M, t) = 1 - \frac{\min_{R_i \in R_c} \text{range}(R_i, M, t)}{\max_{R \in RL_{top}} \text{range}(R, M, t)}$$

$$S_2(R_c, M, t) = \min \left(\frac{\max_{R_i \in R_c} \text{range}(R_i, M, t)}{\text{quantile}_{.75, R \in RL_{top}} \text{range}(R, M, t)}, 1 \right)$$

with $\text{range}(R, M, t) = 1 + \max_{P \in r_t(R, M)} a(P) - \min_{P \in r_t(R, M)} a(P)$ denoting the temporal range in which reviewer R has published on topics relevant to M . These scores are then linearly combined to the final seniority score, with $\sigma \in [0, 1]$ a weighting parameter:

$$S(R_c, M, t) = \sigma S_1(R_c, M, t) + (1 - \sigma) S_2(R_c, M, t)$$

Final Equation

We combine all of these five quantifiable aspects to obtain a single score SC for each reviewer sets. Good reviewer sets will have high values in all aspects;

we thus multiply the per-aspect scores:

$$SC(R_c, M, t) = A(R_c, M, t) \cdot S(R_c, M, t) \cdot I(R_c, M, t) \cdot D(R_c, M, t) \cdot E(R_c, M, t) \quad (8.3)$$

The candidate reviewer set R_c achieving the highest SC is the most suitable one and recommended for the manuscript as result of Step 2. We will denote this result as R_0 in the experimental evaluation.

8.4.3 Run Time Analysis

We separate this run time analysis into parts which are independent on the single specific manuscripts and ones which have to be conducted for each manuscript. Manuscript non-specific parts only need to be conducted once for the assignment per PC. Manuscript specific parts need to be conducted for each manuscript for which reviewers are assigned.

1. Data preparation: Let $n \in \mathbf{N}$ be the number of publications of all possible reviewers from RPC . We load the set of information ($\mathcal{O}(n)$). Additionally, we load the set of co-authors for each reviewer ($\mathcal{O}(n)$). This part ($\mathcal{O}(n)$) is manuscript non-specific.

2. Step 1: In Step 1 of the algorithm, the similarity of all publications of reviewers to the manuscript M needs to be calculated ($\mathcal{O}(n)$). Then for all reviewers, the similarity of their publications needs to be summed ($\mathcal{O}(n)$). Furthermore, all publications need to be ordered according to their similarity with M ($\mathcal{O}(n \log n)$). These three parts are summed up such that the final run time for Step 1 of the algorithm is $\mathcal{O}(n \log n)$.

3. Restrict $|RL|$: Before Step 2 of the algorithm can be tackled, we resolve conflicts of interests in RL_{top} with the observation of co-authors of reviewers. This potentially excludes persons from the set of possible reviewers and results in $RL_{top'}$; $|RL_{top'}| \leq |RL_{top}|$. The run time of this restriction is negligible ($\mathcal{O}(1)$).

4. Pre-computations for Step 2: For the second part of the algorithm, we separate some pre-calculations which need to be performed for all R_c and the calculations, which are dependent on the current R_c . Let $m \in \mathbf{N}$, $m \leq n$ be the number of publications of the k reviewers in $RL_{top'}$. Per definition $|R_c| \geq 2$. The pre-calculations include the calculation of the ranges of all reviewers ($\mathcal{O}(k)$), the citation count of all publications ($\mathcal{O}(m)$), the h index of all reviewers ($\mathcal{O}(k)$), the max citation count ($\mathcal{O}(m)$), the max h index ($\mathcal{O}(k)$), the similarities of topics and words of publications with the manuscript M ($\mathcal{O}(m)$) and the length normalized one vector representations of all reviewers ($\mathcal{O}(k)$).

Similarities of the papers of reviewers from $RL_{top'}$ do not need to be computed, as they were already computed in the first step, here we need to assess if these similarities surpass the similarity threshold t ($\mathcal{O}(m)$). Similarities of all restricted profiles of reviewer candidates with all other restricted profiles of reviewer candidates need to be computed as these values are required in the calculation of diversity in step 2. The cost of compiling the k restricted profiles is $\mathcal{O}(m)$ such that the similarity calculation takes $\mathcal{O}(m + k!)$.

All scores for reviewer candidates for the partial aspects $E_1, E_2, E_3, A_1, A_2, I, S_1$ and S_2 (with $\mathcal{O}(k)$ for each aspect) can be pre-computed per manuscript. The algorithm then combines the reviewer dependent values once the respective reviewers are part of a reviewer set. The only R_c dependent part is D but as combinations of reviewer pairs can occur in multiple R_c , this aspect can also be pre-computed ($\mathcal{O}(k!)$).

The pre-calculations can thus be summed up to a run time of $\mathcal{O}(m + k!)$.

5. ASIDE part of Step 2: The combination of values for the ASIDE part needs to be performed for each candidate reviewer set $|R_c|$, i.e., $\binom{k}{|R_c|}$ times; $|R_c| \in \{1, \dots, k\}$. The check for disjoint publications of possible reviewers is negligible ($\mathcal{O}(1)$). The run times for the look-ups for the combined scores for authority ($\mathcal{O}(1)$), seniority ($\mathcal{O}(1)$), interest ($\mathcal{O}(1)$), diversity ($\mathcal{O}(\binom{|R_c|}{2})$) and expertise ($\mathcal{O}(1)$) are summed up. The complete run time of the second step is $\mathcal{O}(m + k! + \binom{k}{|R_c|} * \binom{|R_c|}{2})$.

Overall Run Time: Except for the first part, all others are dependent on the specific manuscript. We thus report the overall run times if we observe j different manuscripts in a single venue. The full run time in the average and best case with $m < n$, $k < |RL|$ and $|R_c| < k$ is $\mathcal{O}(n + j * (n \log n + m + k! + \binom{k}{|R_c|} * \binom{|R_c|}{2})) = \mathcal{O}(n \log n + m + k! + \binom{k}{|R_c|} * \binom{|R_c|}{2})$.

In the worst case where $m = n$ and $k = |RL| = |RPC|$, so the list of possible reviewers $RL_{top'}$ is not restricted, the first step of the approach can be omitted resulting in a run time of $\mathcal{O}(n + j * (n + |RPC|! + \binom{|RPC|}{|R_c|} * \binom{|R_c|}{2})) = \mathcal{O}(n + |RPC|! + \binom{|RPC|}{|R_c|} * \binom{|R_c|}{2})$.

8.5 Datasets

To evaluate our proposed reviewer set recommendation approach, we develop three novel evaluation datasets. We consider manuscripts from three different workshops and conferences of different size and thematic focus that took place in 2017, namely MOL, BTW, and ECIR. As it is practically impossible to obtain all papers submitted to a conference, we use all accepted papers as an approximation instead. Note that this might lead to

non-representative topic distributions of manuscripts and unrealistically low number of manuscripts to be reviewed. Additional fuzziness is introduced since we do not distinguish between long, short and demo papers as program committees are oftentimes published in a merged form.

8.5.1 Data Acquisition

We built three different datasets⁵ based on data from dblp [12]⁶ which was merged with abstracts, citations and references from the AMiner part of the Open Academic Graph [25, 71]⁷ where available as well as full texts of accepted manuscripts. Information from AMiner was joined with dblp data (based on matching DOIs where available, or on matching paper titles, author names and publication years otherwise); this allowed to focus on publications from computer science or adjacent domains and to build rather precise reviewer profiles due to dblp’s author disambiguation efforts, compared to using reviewer names only. Full texts of accepted manuscripts are not included in the AMiner dataset but stem from pdfs collected by hand which were converted to text files using Science Parse⁸.

Information on program committees was either taken from conference websites or conference proceedings. Reviewer names were manually mapped to dblp authors.⁹ For each reviewer, we set up a list of their publications identified by their dblp keys. Here, only papers up to 2016 were taken into consideration, corresponding to a reviewer selection process in early 2017. For each of the papers the dataset contains its publication year, the paper length, the CORE rank¹⁰ of the venue it was published in, the number of citations it accumulated and the average h index of its authors. The concatenated title and abstract (where available) of papers needed to consist of at least three terms to be considered for the dataset. Citing papers which are not contained in dblp were omitted. Thus, the number of incoming links might not necessarily represent the number of citations which publications received in the real world. This influences the number of citations and the average h index.

⁵Available under <https://doi.org/10.5281/zenodo.4071874>. Data acquired from the manual evaluations in Section 8.8.3 as well as templates showcasing the structure of the files are also included in the datasets.

⁶As of January 1, 2020; <https://dblp.org/xml/release/dblp-2020-01-01.xml.gz>

⁷V1 from mid 2017; <https://www.aminer.org/open-academic-graph>

⁸V2.03; <https://github.com/allenai/science-parse>

⁹Note that the dblp dataset is being revised continuously, reviewers’ profiles might be imperfect due to disambiguation problems.

¹⁰<http://www.core.edu.au/>

For each manuscript of our three test conferences the datasets contain a pool of possible reviewers. It consists of all members of the program committee, but excludes those with obvious conflicts of interest accessible by (former) co-authorships of authors of the manuscripts and reviewers. For each of the papers published by possible reviewers, our datasets also contain tf-idf, Doc2Vec [16], LDA [3] and BERT [9] vector representations of its title and abstract where available. For submitted manuscripts, these four kinds of document representation are contained for the full text as well as only the research sections of the paper (which consist of all sections excluding the abstract, introduction, related work, conclusion, references and acknowledgements). The textual content of the papers is not contained. We consider only English documents for the construction of our datasets.

8.5.2 Document Representations

We calculated the document frequencies of words for tf-idf on unstemmed titles of all publications contained in dblp up to 2016 concatenated with abstracts from AMiner where available which were written in English. In total we used 2,940,996 documents. The final tf-idf vectors are calculated for unstemmed textual data available in the respective datasets including all papers of reviewers and submitted manuscripts.

For the construction of BERT [9] vectors, we used the base pretrained uncased model.¹¹ Since the BERT implementation used is only able to process input vectors of at most 512 tokens, documents were cut at punctuation marks or after half of the tokens if sentences were still too long. A sliding window was used to always input two consecutive sentences to maintain as much context as possible. The model consists of overall twelve hidden layers each having 768 features. The last four layers from these twelve layers were concatenated for each token and averaged over all tokens to receive vectors of length $4 \text{ layers} \times 768 \text{ features} = 3072 \text{ dimensions}$ for each publication. [37]

Weights for Doc2Vec [16] are trained on the English Wikipedia corpus from 1st February 2020¹². We refrained from using Doc2Vec on a stemmed corpus as this preprocessing is no prerequisite for achieving good results [16]. We trained two Doc2Vec models, one distributed bag of words (DBOW) and one distributed memory (DM) model, so that resulting vectors consist of 300 dimensions each. This size was proposed by Lau and Baldwin [39] for general-purpose applications.¹³ [37]

¹¹We utilised the BERT implementation and model by https://huggingface.co/transformers/model_doc/bert.html.

¹²<https://dumps.wikimedia.org/enwiki/20200201/>

¹³We utilised the Doc2Vec implementation by <https://radimrehurek.com/gensim/>

For LDA [3] we again used the 2,940,996 documents which we already utilised for the computation of the document frequency in tf-idf. This procedure ensured the computed topics were from the area of computer science. The number of topics was set to 100 resulting in the same number of dimensions for vector representations of manuscripts and publications.¹⁴ [37]

8.5.3 MOL'17, BTW'17 and ECIR'17

MOL'17 The dataset contains 12 manuscripts in English language which were accepted at *Meeting on the Mathematics of Language '17*, 22 program committee members and their papers in dblp. We excluded extended abstracts. No distinction between different paper types and program committees was made. On average, each manuscript has 21 possible reviewers, which do not have conflicts of interests. This dataset represents a small biannual international conference with a different focus than the other two datasets.

BTW'17 The dataset contains 36 manuscripts in English language which were accepted at *Datenbanksysteme für Business, Technologie und Web '17* (the German database conference), 56 program committee members and their papers in dblp. We again excluded extended abstracts. No distinction between different paper types was made but the program committees members are split in scientific, industry and demo paper committee. On average, each manuscript has 47.78 possible reviewers, which do not have conflicts of interests. This dataset represents a medium-sized biannual national conference with several lesser-known reviewers.

ECIR'17 The dataset contains 80 manuscripts in English language which were accepted at *European Conference on Information Retrieval '17*, 151 program committee members and their papers in dblp. A distinction between full-paper meta-reviewers, full-paper program committee, short paper program committee and demonstration reviewers was made. On average, each manuscript has 141.35 possible reviewers, which do not have conflicts of interests. This dataset represents a medium to large annual European conference attributed with CORE rank A and mostly well-known reviewers.

8.5.4 Discussion and Challenges

Utilising scientific citation data always comes with challenges: citations need to be handled with care due to self-citations which could vastly improve the perceived authority of researchers [62], varying citation practices in different

models/doc2vec.html.

¹⁴We utilised the LDA implementation by <https://radimrehurek.com/gensim/models/ldamodel.html>.

areas [17, 66, 68], ambiguous reasons for citing works [24], the non-existence of citations of newly published papers [74] and the generally uncited influences [24, 42, 48]. In our case, all observed publications come from the area of computer science and closely related fields in general. Further restriction of our focus on the three conferences hopefully also helps in attenuating this effect. New papers hopefully also do not influence our problem vastly as they should be present for almost all reviewer candidates and thus cancelling each other out. Unfortunately, the other challenges associated with usage of citations cannot be tackled in the context of this work.

As we base our datasets on the dblp data, we are dependent on their disambiguation efforts. Their author profiles are revised continuously, but the disambiguation might not always be perfect [22]. So, this also influences our reviewers' and manuscripts' data. Additionally, names of reviewers were partially extracted from conference websites where multiple clerical errors were made and nicknames or abbreviations of names were included. We corrected obvious mistakes but cannot ensure total correctness of the manual mapping of names to reviewers' dblp profiles.

It would be desirable to observe all publications which were submitted to a conference, especially the rejected ones. As such a dataset does not exist currently to the best of our knowledge, the suitability of our dataset for reviewer recommendation for whole venues is possibly restricted.

We want to point to the fact that even though it would be possible to consider multiple versions of the same venue as datasets, they would still need to be considered separately as submitted manuscripts, reviewer committee sets as well as papers of reviewers are year-dependent and liable to changes. The need for bigger datasets could only be fulfilled in observing bigger conferences.

8.6 Evaluation: Preface

In our experiments, if not stated otherwise¹⁵, we solely focus on sets consisting of three reviewers, even though our approach is applicable for different numbers of reviewers per manuscript as well. This number was chosen as a widespread norm [3] to reduce the dimensionality of further evaluation steps. We evaluate our approach on the three introduced datasets MOL'17, BTW'17 and ECIR'17 where we disregard the different manuscript and committee types. By observing the performance of our approach in venues of different sizes, we strive to make assumptions on its general applicability.

¹⁵See the evaluation of hypothesis H_6 in Section 8.8.2 and Section 8.8.2.

We use Cosine similarity as similarity measure. This ensures similarity values in $[0, 1]$ for Step 2 as tf-idf and LDA document vector representations hold non-negative values for all dimensions. For the voting techniques of the algorithm we run tests with $n \in \{5, 10\}$ and $\delta \in \{0, .25, .5, .9\}$.

For all significance tests, we use a p -value of .05. We evaluate the normal distribution of values using Kolmogorov-Smirnov tests and test the homogeneity of variances with Levene's tests. All depicted values are rounded on four decimal places.

8.6.1 Hypotheses

Considering the overall challenges and goals of RevASIDE, we investigate the following seven hypotheses:

- H_1 Step 1 is useful for the expert search task.
- H_2 Usage of more advanced document vector representations leads to significantly better overall results for Step 1 compared to more basic ones.
- H_3 Utilisation of different document vector representations, voting techniques, cut-off values k of the result list RL , content types and thresholds t leads to significantly different overall RevASIDE scores and values for the five quantifiable aspects in Step 2.
- H_4 Utilisation of the full texts of manuscripts leads to worse overall results than restriction of the manuscripts' content to the technical sections in Step 2.
- H_5 The conduction of Step 1 is profitable for Step 2.
- H_6 RevASIDE is suitable for different sizes of reviewer sets.
- H_7 Results of Step 2 are confirmed by human assessment, thus RevASIDE is useful for the reviewer set assignment task.

8.7 Evaluation: Step 1 - Expert Search Task

In this part of the evaluation, we intend to assess hypotheses H_1 of Step 1 being useful for the expert search task and H_2 of utilisation of more advanced DVs producing better results.

Table 8.3: Mean average precision@10 (MAP), precision@10 (P@10) and nDCG@10 (nDCG) for all combinations of voting techniques (VT) and document vector representations of manuscripts from BTW’17 (upper half) and ECIR’17 (lower half). Best combination in BTW’17: tf-idf + *SUM* (short b_1). Best combinations in ECIR’17: tf-idf + *MNZ* (short e_1), *DBOW* + *Votes $_{\delta=.5}$* (short e_2). Column SD gives information on whether or not MAP (m), P@10 and nDCG (n) significantly differ between the different DVs. If \checkmark , all three measures are significantly different.

dataset		BTW’17											SD	
DV	VT\measure	tf-idf			DM			DBOW			BERT			
		MAP	P@10	nDCG	MAP	P@10	nDCG	MAP	P@10	nDCG	MAP	P@10	nDCG	
	<i>Votes$_{\delta=0}$</i>	.1705	.34	.3677	.1732	.345	.3706	.1705	.34	.3677	.1705	.34	.3677	
	<i>Votes$_{\delta=.25}$</i>	.0712	.2	.1881	.2056	.36	.4017	.1714	.335	.3685	.1705	.34	.3677	\checkmark
	<i>Votes$_{\delta=.5}$</i>	.0712	.2	.1881	.1584	.25	.3088	.1966	.325	.373	.1705	.34	.3677	\checkmark
	<i>Votes$_{\delta=.9}$</i>	.0712	.2	.1881	.0712	.2	.1881	.0712	.2	.1881	.1766	.35	.3701	\checkmark
	<i>SUM</i>	.2947	.42	.4923	.1816	.345	.3721	.1749	.34	.3722	.168	.34	.3635	
	<i>AVG</i>	.2612	.385	.4246	.1222	.265	.2754	.1682	.345	.345	.0604	.16	.1593	\checkmark
	<i>MNZ</i>	.273	.41	.4761	.1787	.345	.3777	.1755	.345	.3725	.1704	.34	.367	
	<i>SUM$_n=5$</i>	.0303	.1	.1007	.043	.15	.1398	.0697	.18	.1827	.0385	.115	.1113	mn
	<i>SUM$_n=10$</i>	.0329	.08	.0891	.0421	.135	.1321	.0659	.175	.1813	.0231	.095	.0872	\checkmark
	<i>MIN</i>	.0364	.155	.1194	.0301	.135	.1023	.0391	.145	.1316	.0168	.08	.0622	
	<i>MAX</i>	.2779	.39	.4589	.2872	.405	.4812	.256	.395	.4517	.2162	.34	.3758	
	<i>RR</i>	.0949	.235	.242	.1027	.265	.2632	.1005	.25	.2386	.2326	.365	.4311	\checkmark
	<i>mRR</i>	.0519	.165	.1505	.0613	.2	.1882	.0683	.185	.1857	.1757	.33	.3614	\checkmark
	<i>BordaFuse</i>	.1545	.325	.3405	.1385	.305	.3192	.12	.275	.2768	.1633	.345	.3459	
	<i>expSUM</i>	.1705	.34	.3677	.1764	.345	.3756	.1725	.34	.3695	.168	.34	.3635	
	<i>expAVG</i>	.2612	.385	.4246	.1248	.265	.2759	.171	.34	.3458	.0589	.16	.1542	\checkmark
	<i>expMNZ</i>	.1705	.34	.3677	.1761	.345	.3752	.171	.34	.3681	.1708	.34	.3679	
dataset		ECIR’17											SD	
DV	VT\measure	tf-idf			DM			DBOW			BERT			
		MAP	P@10	nDCG	MAP	P@10	nDCG	MAP	P@10	nDCG	MAP	P@10	nDCG	
	<i>Votes$_{\delta=0}$</i>	.1116	.45	.4748	.1132	.455	.4784	.1116	.45	.4748	.1116	.45	.4748	
	<i>Votes$_{\delta=.25}$</i>	.031	.21	.2095	.1308	.485	.5245	.1195	.48	.4937	.1116	.45	.4748	\checkmark
	<i>Votes$_{\delta=.5}$</i>	.0317	.21	.2147	.1239	.43	.4733	.164	.555	.5992	.1116	.45	.4748	\checkmark
	<i>Votes$_{\delta=.9}$</i>	.0317	.21	.2147	.0317	.21	.2147	.0317	.21	.2147	.1252	.48	.5081	\checkmark
	<i>SUM</i>	.1217	.475	.4789	.1283	.49	.5173	.124	.475	.5007	.115	.46	.482	
	<i>AVG</i>	.0664	.315	.3129	.047	.285	.2639	.0567	.33	.3167	.0349	.19	.1843	\checkmark
	<i>MNZ</i>	.1647	.545	.5908	.124	.475	.5012	.1163	.46	.4826	.1132	.455	.4788	
	<i>SUM$_n=5$</i>	.0309	.19	.1792	.0202	.135	.1353	.0286	.185	.1679	.0431	.175	.1736	
	<i>SUM$_n=10$</i>	.0284	.16	.1691	.0243	.12	.1311	.0297	.15	.1615	.0473	.165	.1731	
	<i>MIN</i>	.031	.23	.2004	.0181	.165	.149	.0206	.14	.1392	.0274	.17	.1588	
	<i>MAX</i>	.1205	.41	.4429	.1496	.535	.5449	.1525	.535	.565	.0717	.38	.362	\checkmark
	<i>RR</i>	.0535	.275	.2858	.084	.375	.3926	.0651	.37	.3451	.0765	.39	.3693	
	<i>mRR</i>	.0141	.135	.1186	.0311	.23	.2197	.0318	.24	.2299	.0264	.185	.1913	\checkmark
	<i>BordaFuse</i>	.099	.445	.4393	.0957	.425	.427	.0921	.415	.4137	.1014	.43	.4404	
	<i>expSUM</i>	.1116	.45	.4748	.117	.465	.485	.116	.46	.4828	.115	.46	.4818	
	<i>expAVG</i>	.0658	.315	.3164	.049	.295	.2725	.0574	.33	.3171	.038	.19	.1871	\checkmark
	<i>expMNZ</i>	.1116	.45	.4748	.115	.46	.482	.115	.46	.482	.1132	.455	.4788	

8.7.1 Setting

We randomly selected 20 manuscripts from each of the BTW'17 and ECIR'17 datasets. The manuscripts are represented by their full texts, the profiles of reviewers are represented by their papers' titles and abstracts where available. To create a ground-truth of relevant reviewers, the top 10 reviewer candidates are computed with all 13 (17 with variants) voting techniques and combined. The resulting pools of reviewers for each manuscript from the BTW'17 dataset contained 48.35 entries on average and 101.5 entries on average for manuscripts from ECIR'17. In the former case, about all possible reviewers were contained in the respective lists, contrasting the ECIR'17 lists which contain a lower percentage of possible reviewers. Unfortunately, a more extensive manual evaluation with more manuscripts would not be feasible.

The manuscripts' title and abstract as well as the potential reviewers and a link to their dblp profile were presented to an independent senior researcher in the field who evaluated the reviewers in terms of appropriateness for the given manuscript. For the manual evaluation of relevance, only papers up to 2016 of reviewers were considered. The expert was not aware which method retrieved which reviewers. If the expert observed missing relevant reviewers, they were also included in the ground-truth. In BTW'17, each paper has 10.05 relevant reviewers on average (min=5, max=14, median=10, standard deviation=2.762). In ECIR'17, each paper has 27.2 relevant reviewers on average (min=3, max=55, median=25, standard deviation=13.5671). On average, a reviewer from the program committee is relevant for 3.5893 manuscripts for BTW'17 and 3.1813 manuscripts for ECIR'17.

We report result quality with three established metrics, examining the first 10 retrieved reviewers of each method. Precision@10 measures the fraction of the top-10 recommended reviewers that were actually relevant. Non-interpolated mean average precision@10 (MAP) averages the precision at ranks where a relevant reviewer appears, using a precision of 0 for each relevant reviewer not appearing in the result list. Normalized cumulative discounted gain (nDCG) [11] aggregates relevance of all reviewers appearing in the result, but with a logarithmic discount for later ranks; this follows the intuition that later ranks are less important to a user than earlier ranks. In addition, it normalizes this aggregation by the cumulative discounted gain achieved by an ideal ranking where all relevant reviewers appear in front, thus showing how close the result is to an optimal result and allowing to compare across different queries with different numbers of relevant results.

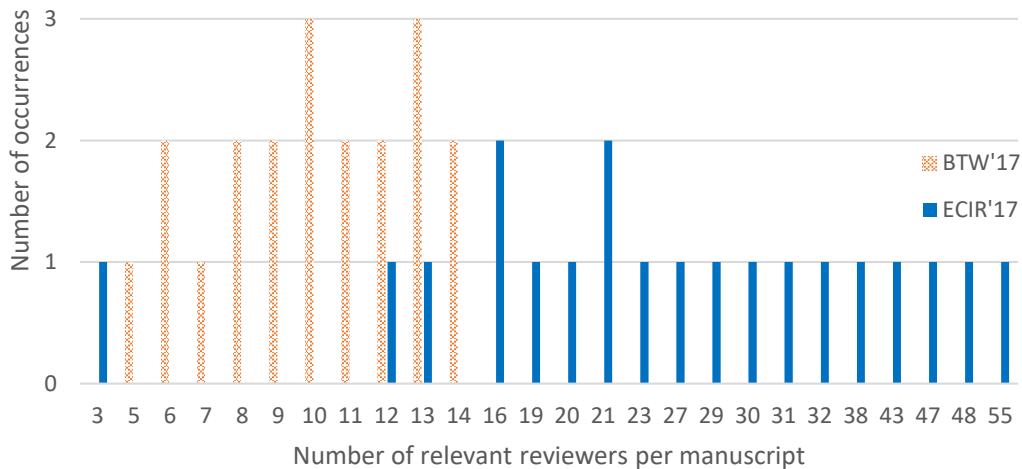


Figure 8.2: Numbers of reviewers which are relevant for single manuscripts per dataset.

8.7.2 Results (Analysis of H_1 and H_2)

In BTW'17, each of the 20 papers has 10.05 relevant reviewers on average (min=5, max=14, median=10, standard deviation=2.762). In ECIR'17, each of them has 27.2 relevant reviewers on average (min=3, max=55, median=25, standard deviation=13.5671). Figure 8.2 shows the number of reviewers which are relevant for the 20 observed manuscripts per dataset. All manuscripts have at least three relevant reviewers. We observe major differences between the different sized datasets. In ECIR'17, for most manuscripts many relevant reviewers could be found, for BTW'17 these numbers are much lower. This might partially be influenced by the explicit topical breadth of BTW'17 but also by the smaller size of the reviewer candidate pool for this dataset.

On average, a reviewer from the program committee is relevant for 3.5893 manuscripts for BTW'17 and 3.1813 manuscripts for ECIR'17 out of the 20 observed ones. Figure 8.3 shows the amounts of manuscripts from the 20 observed ones, for which single reviewers are relevant per dataset. Most reviewers are relevant for only few of the observed manuscripts, few reviewers are relevant for many of them. A non-negligible share of reviewers is not relevant for any of the 20 observed papers in both datasets.

The upper part of Table 8.3 shows result quality for all combinations of document vector representation and voting technique for the twenty manuscripts from BTW'17. $Votes_{\delta=0}$ is exactly the same for each document vector representation, as this voting technique solely considers the number of papers of reviewer candidates and not their similarity with query manuscripts. The

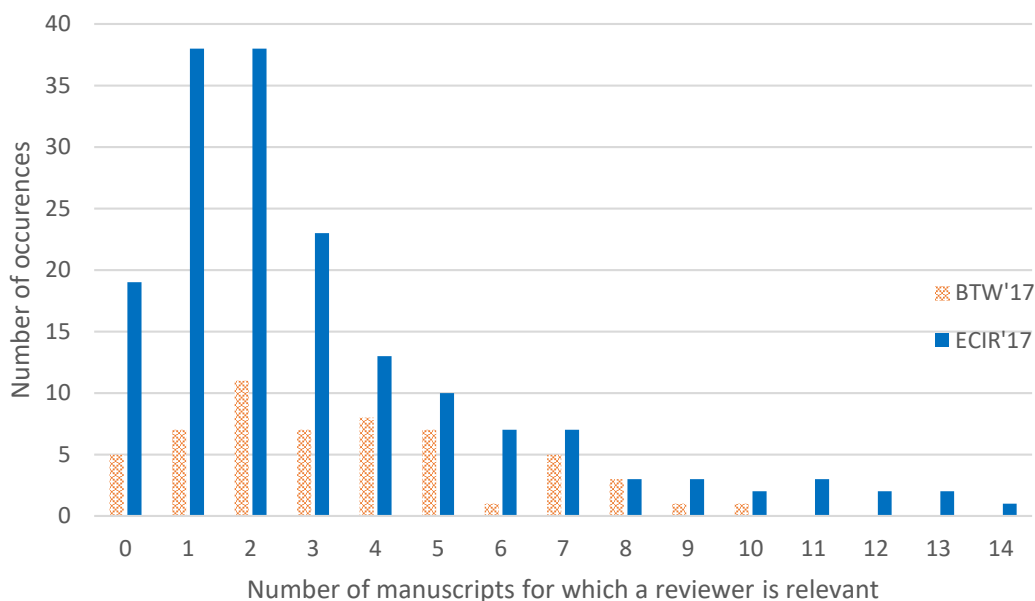


Figure 8.3: Numbers of manuscripts, for which single reviewers are relevant per dataset.

lower part of Table 8.3 shows the same for the twenty manuscripts from ECIR'17.

In BTW'17, each paper has 2.7801 relevant reviewers per combination of VT and DV on average, in ECIR'17 this value is significantly (Mann-Whitney U test) higher (3.4838). These assessments lead to the assumption of the VTs and DVs presented here being useful for the expert search task and therefore verifying H_1 .

We found significant (Kruskal-Wallis H tests) differences between the four DVs for several voting techniques, but not for all of them (see rightmost column of Table 8.3). The more advanced document vector representations Doc2Vec and especially BERT did not achieve better results than tf-idf.

The best voting techniques seem to depend on the dataset and the utilised document vector representation. BERT performs worse than both tf-idf and the Doc2Vec models. Usage of tf-idf and DM achieves comparable results for the best performing VTs for BTW'17; for ECIR'17, tf-idf and DBOW with their respective best VTs result in similar values. BERT seems to generalise the concepts of papers too much, such that the VTs cannot clearly distinguish between relevant and non-relevant reviewers. This is underlined by the fact that three versions of $Votes_\delta$ generate the same values for MAP, P@10 as well as nDCG. Tf-idf has high selectivity and is able to identify experts versed in the exact same techniques described in a manuscript. Hence, hypothesis H_2

of more sophisticated VRs being more suitable than basic VRs is rejected.

For the ECIR'17 dataset, P@10 and nDCG are higher than for BTW'17. This might be caused by ECIR'17 having higher overall numbers of reviewers as well as more relevant reviewers per manuscript. This disadvantages the smaller BTW'17 dataset.

8.8 Evaluation Step 2 - Reviewer Set Assignment Task

The evaluation of Step 2 of our algorithm consists of a quantitative and a qualitative evaluation. These parts each encompass numerous experiments.

8.8.1 Setting

In Equation 8.3 we set $\epsilon_1 = \epsilon_2 = \epsilon_3 = \frac{1}{3}$ and $\alpha = \sigma = .5$. Furthermore, except for experiments in Section 8.8.2 and Section 8.8.2 we only observe $|R_c|$ as three as a widespread lower bound for sizes of reviewer sets [13] and to reduce the complexity of our following calculation

As a first baseline B_{t3} , the three highest ranked reviewers in the ranked list RL for each VT and DV are considered as a reviewer set for a manuscript. Such an approach is common in reviewer set recommendation [13, 29]. Our second baseline B_{tr} chooses three random reviewers from RL_{top} . Our third baseline B_r chooses three random reviewers from the whole program committee, excluding only those with a conflict of interest. For the latter, we cap values of E_3 , A_1 and A_2 at 1.¹⁶

We experiment with cut-offs k of reviewers in RL to generate RL_{top} at position 10 and 20 after Step 1 and without cut-off, i.e. all reviewers without conflicts of interests for the manuscripts were utilised as a comparison to evaluate the usefulness of Step 1. If we do not restrict the number of candidate reviewers, i.e. $|RL| = k$, the voting technique used in Step 1 (which determines the reviewer candidates considered in Step 2) becomes irrelevant for Step 2 but still influences the creation of the baselines. We also experiment with different thresholds $t \in \{0, .25, .5, .9\}$.

We divide the manuscripts in non-technical and research sections to better estimate their true content. Non-technical sections include abstract, introduction, related work, conclusion, acknowledgements and references. Research sections are all other parts. We compare the effect of using the full

¹⁶Reviewers from B_r are possibly not contained in RL_{top} and thus could theoretically produce values > 1 for the three partial aspects. The score for this baseline is still calculated based on maxima of papers of relevant candidate reviewers.

Table 8.4: Significant differences between the groups in SC as well as the five quantifiable aspects by datasets MOL’17 (m), BTW’17 (b) and ECIR’17 (e).

grouped by	SC	A	S	I	D	E
DV	mb	mb	b	e	mbe	mbe
VT	mb	mbe	mbe	mbe	mbe	mbe
k	mbe	mbe	mbe	mbe	mbe	mbe
CT	mbe			mbe		mbe
t in Step 2	mbe	mbe	mbe	mbe	mbe	mbe
RT	mbe	mbe	mbe	mbe	mbe	mbe

text in Step 2 to using only the content of research sections. Profiles of reviewers are represented by their papers’ titles and abstracts where available, which are similar enough (threshold t) to the query manuscript.

8.8.2 Quantitative Evaluation

In this part of the evaluation, we focus on understanding the influence of the different factors of our approach and prepare the qualitative evaluation by identifying the combinations achieving the highest scores. In this context, we intend to assess hypotheses H_3 and H_4 as well as H_5 which observes the usefulness of Step 1.

In these experiments, for each combination of document vector representation in Step 1, voting technique, cut-off of relevant reviewers utilised in Step 2, similarity threshold t in Step 2 as well as used content type (CT) in Step 2 we observe the following result types (RT): the three baselines (B_{t3} , B_{tr} , B_r) and the best result returned by RevASIDE (R_0).

Significance of Factors (Analysis of H_3)

In this section, we want to evaluate hypothesis H_3 which claims the utilisation of different DVs, VTs, cut-off values, content types and thresholds results in significantly different scores SC as well as values for the five quantifiable aspects.

We test for significant differences between groups of experiments to determine which factors really influence the overall score SC (as computed by Equation 8.3) and the five quantifiable aspects introduced in Sections 8.4.2 to 8.4.2. Kruskal-Wallis H tests are used for the following experiments since in most of our observed cases, data is not normally distributed in the different groups or variances are not homogeneous. Table 8.4 indicates between

Table 8.5: Configuration (conf), DV, VT, RL_{top} cut-off value k , utilised content type and threshold t resulting in the highest average scores and corresponding values for A , S , I , D as well as E per dataset and result type.

conf	dataset	RT	DV	VT	CT	k	t	SC	A	S	I	D	E
c_1	MOL'17	R_0	BERT	MIN	full	20	.5	.053	.6675	.8696	.5277	.3783	.4614
c_2	MOL'17	B_{t3}	BERT	$Votes_{\delta=0}$	full	20	.25	.0439	.5972	.8696	.5283	.3109	.5056
c_3	MOL'17	B_{tr}	DBOW	exp^{AVG}	full	10	.25	.0348	.69	.8158	.5158	.3014	.3738
c_4	MOL'17	B_r	DM	$BordaFuse$	full	10	.5	.0251	.4642	.8199	.5408	.2654	.4467
c_5	BTW'17	R_0	BERT	MIN	full	20	.5	.0528	.5827	.9935	.5243	.555	.3152
c_6	BTW'17	B_{t3}	BERT	$SUM_{n=5}$	full	10	.5	.0218	.4106	.887	.6078	.3332	.2891
c_7	BTW'17	B_{tr}	tf-idf	$SUM_{n=10}$	full	10	.5	.0303	.826	.7274	.5559	.2298	.3884
c_8	BTW'17	B_r	BERT	MIN	full	10	.25	.0193	.3797	.8963	.5192	.3743	.2693
c_9	ECIR'17	R_0	BERT	mRR	full	20	.9	.0438	.6348	.8077	.6614	.3162	.4148
c_{10}	ECIR'17	B_{t3}	BERT	mRR	full	10	.9	.0192	.5312	.7315	.6704	.2038	.3471
c_{11}	ECIR'17	B_{tr}	DM	$SUM_{n=5}$	full	10	.5	.0319	.9517	.6077	.6675	.1977	.4309
c_{12}	ECIR'17	B_r	DBOW	$BordaFuse$	full	10	.5	.0171	.5228	.7271	.6715	.1451	.4581

which groups of experiments we found significant differences in the scores or the five quantifiable aspects. We observe 1,632 (4 DVs \times 17 VTs \times 3 cut-offs k \times 2 CTs \times 4 t in Step 2) experimental setups per dataset. Experiments were grouped by document vector type such that there were four groups of experiments, ones using tf-idf in the first step, ones using Doc2Vec DM, ones using Doc2Vec DBOW and ones using BERT document vector representations. Grouping by VT in Step 1 results in 17 different groups of experiments. When experiments are grouped by the number of observed candidates k three different groups result. When grouping by content type, two groups of experiments result, ones which utilise the full text in Step 2 and ones utilising only the research sections of the query manuscript. Grouping by the threshold value t in Step 2 results in four different groups. Lastly, grouping by RT produces four groups containing experiments of types B_{t3} , B_r , B_{tr} and R_0 .

DV does influence some aspects significantly, but overall, the scores of the ECIR'17 dataset are not significantly influenced by it. VT significantly influences the five aspects for all datasets as well as the score for the two smaller ones. The content type which is utilised in Step 2 is significantly influential for values for all datasets except for authority, diversity, and seniority. These values are not calculated by directly utilising the query manuscript and therefore are not influenced by the content type. The cut-off value k which is chosen for RL_{top} , the threshold value t as well the result types significantly influence the results in all three datasets. From these observations, we derive the overall validity of hypothesis H_3 .

Configurations achieving highest Scores (Analysis of H_4 and H_5)

Table 8.5 shows the best combinations of DV, VT, cut-off values, content type and threshold, measured in terms of the highest overall average scores

for R_0 and the three baselines B_{t3} , B_r and B_{tr} for each of the three datasets. SC is calculated with Equation 8.3 and can take values between 0 and 1 with 1 being the best. As it is multiplicative, a score of .05 can be reached if e.g. values of all quantifiable aspects A , S , I , D , and E are around .55.

R_0 achieves the highest SC results for each dataset. This, together with the significant differences between result types observed in the previous experiment (see Table 8.4), leads to the conclusion that RevASIDE produces significantly higher average SC scores than the baselines. This applies to all three different sized datasets, which highlights the general applicability of our approach.

Utilising full texts of query manuscripts yields better results than only taking the research sections into account. This leads to the rejection of hypothesis H_4 .

The restriction of RL_{top} to $k = 10$ leads to the best average scores for MOL'17 and ECIR'17; for BTW'17, no restriction of RL_{top} leads to the highest average scores (not depicted in the table). This indicates that the reduction of the number of considered reviewers for Step 2 (and therefore the entirety of Step 1) is a major factor in small and large datasets. It also decreases the overall computation time, which in general verifies H_5 . MOL'17 as well as ECIR'17 represent relatively focused areas, while BTW'17 is more diverse. For focused datasets it suffices to regard the few most relevant reviewers to compose a suitable set, but for a diverse conference, it seems more reviewers need to be considered. When grouping all 1,632 experiments by voting technique and threshold, the highest average scores for MOL'17 are achieved by exp^{SUM} and .5; for BTW'17 $SUM_{n=5}$ and .25; and for ECIR'17 $SUM_{n=10}$ and .9. BERT is the DV which on average performs best for each dataset. They outperform the other VTs and thresholds on average but do not appear as a combination in Table 8.5 under the overall best configurations. Remarkably, the best results for R_0 in MOL'17 as well as BTW'17 were achieved by the same combination of DV, VT, CT, k as well as t . The combination of BERT with MIN or mRR did not achieve any good results in our manual evaluation of Step 1 but did prove to be useful in Step 2.

The highest scores for MOL'17 (.0516), BTW'17 (.0462) as well as ECIR'17 (.0399) for the best performing combinations from Step 1 of our approach (b_1 : tf-idf + SUM, e_1 : tf-idf + MNZ , e_2 : DBOW + $Votes_{\delta=.5}$) are independent of DV and VT as they are achieved by $|RL_{top}| = |RL|$. The threshold t is set to .5. These results cannot surpass the best configurations from Table 8.5 for the same data, but also do not significantly differ from them.

For BTW'17 as well as ECIR'17, we found no significant correlation between the scores produced by the twelve (eleven as $c_1 = c_5$) best config-

Table 8.6: DV, VT, content type CT (rc symbolises content restricted to research sections), cut-off k , threshold t , RT and scores SC for the highest values for each of the different aspects A , S , I , D and E for all three datasets.

MOL'17											
DV	VT	CT	k	t	RT	SC	A	S	I	D	E
DM	$SUM_{n=5}$	rc	10	.9	R_0	.0455	.7359	.7919	.6073	.3045	.4248
BERT	MIN	full	10	0	R_0	.0003	.3616	.9972	.0041	.7670	.1786
DBOW	$SUM_{n=10}$	full	10	.9	BL_{top3}	.0110	.2632	.8544	.6423	.2523	.2869
DM	$SUM_{n=5}$	full	10	0	R_0	.0011	.4871	.9762	.0166	.8698	.1498
BERT	$Votes_{\delta=0}$	full	10	$\geq .25$	BL_{top3}	.0434	.5972	.8598	.5283	.3109	.5056
BTW'17											
DV	VT	CT	k	t	RT	SC	A	S	I	D	E
tf-idf	$SUM_{n=10}$	rc	10	.25	BL_{t3r}	.0211	.8967	.6433	.5252	.1831	.3807
BERT	MIN	full	20	.25	R_0	.0505	.5831	.9943	.5251	.5531	.2997
DBOW	AVG	full	10	.5	R_0	.0277	.5931	.8849	.6441	.2099	.3986
BERT	$SUM_{n=5}$	rc	10	0	BL_{top3}	~ 0	.2847	.7776	~ 0	.7532	.0577
DM	SUM	full	$ RL $	$\geq .25$	BL_{top3}	.0155	.6297	.7540	.5726	.1243	.4548
ECIR'17											
DV	VT	CT	k	t	RT	SC	A	S	I	D	E
BERT	MIN	full	10	.25	BL_{t3r}	.0254	.9743	.5849	.6480	.1673	.4201
BERT	AVG	rc	10	0	R_0	.0011	.4591	.9920	.0199	.5817	.1127
DBOW	$Votes_{\delta=.5}$	full	10	.5	BL_{top3}	.0105	.5297	.7591	.7357	.0811	.4492
DM	$SUM_{n=5}$	rc	20	0	R_0	.0009	.5271	.9432	.0133	.7315	.1264
DM	exp^{SUM}	full	$ RL $	$\geq .5$	BL_{top3}	.0150	.4389	.7762	.6961	.1183	.5334

urations from Table 8.5 for R_0 and the number of relevant reviewers per manuscript for the forty manuscripts observed in the evaluation of Step 1 with Kendall's τ_B .

We want to point to the fact that some of the DVs and VTs present in Table 8.5 achieve low results in the evaluation of Step 1 (for BTW'17 .0168 to .171 in MAP, .08 to .34 in P@10 and .0622 to .3677 in nDCG; for ECIR'17 .0264 to .1116 in MAP, .135 to .45 in P@10 and .1353 to .4748 in nDCG). This hints at possible problems with aspects with opposing objectives, which will be regarded in depth in the following Section as well as the qualitative evaluation in Section 8.8.3.

Highest Values for Aspects

In the following evaluation, we observe the highest values that the five quantifiable aspects were able to reach on average per dataset. The goal of this part is to better understand the factors influencing the aspects.

Table 8.6 depicts the highest values for the five quantifiable aspects per dataset, together with the corresponding configuration and result types.

For high authority cut-off $k = 10$, so a low number of observed reviewer candidates in RL_{top} , seems to be helpful. If few reviewer candidates are observed, the probability of one having a high h index might be lower. This in terms highly influences the calculation of the aspect, as the denominator in the respective Equation 8.1 is smaller, thus increasing the overall value for

A_1 .

Seniority seems to be maximised for R_0 , a low threshold t and usage of BERT as document vector representation in Step 1. This means the true content of the papers of reviewers is completely disregarded here. Only the span of years the reviewer candidates published in is relevant.

High interest of reviewer sets was achieved with utilisation of full texts of publications, their representation in Step 1 with DBOW, a threshold t of at least .5 and $k = 10$. A high influence of papers of reviewer candidates relatively similar to the full content of the manuscript on this aspect is not surprising, whereas the cut-off value k is not even used in the calculation of this aspect (see Equation 8.2).

For a high diversity, all papers of reviewers needed to be considered, not only those which were similar to the manuscript ($t = 0$). Additionally, $SUM_{n=5}$ seems to be helpful to maximise this aspect. If the five most similar papers of reviewer candidates are observed to construct the retrieved sets, reviewers from completely disjunct fields - maybe even ones irrelevant to the manuscript - achieve high diversity values. This assumption is strengthened by the low expertise and overall score SC of these sets.

High expertise of reviewer sets can be achieved by observing full content of manuscript, setting the similarity threshold t to at least .25 and usage of result type BL_{top3} . Utilisation of publications which are at least somewhat similar to all the information of the manuscript in question seems like a reasonable approach. Unsurprisingly, the baseline set resulting in the highest expertise is the one consisting of the three reviewers most similar to the manuscript.

All depicted scores are lower than the highest computed scores per dataset (see Table 8.5). In general, we found that one cannot only optimise after one of the quantifiable aspects to increase the whole score SC . The single different aspects profit from varying, sometimes opposing configurations.

Number of Reviewers R_c (Analysis of H_6)

In this part of the automatic evaluation, we observe the influence of different sizes for the set of recommended reviewers for manuscripts R_c and thus want to commence the evaluation of H_6 . This hypothesis tackles the usefulness of RevASIDE for different reviewer set sizes. In the previous experiments, $|R_c|$ was set to three. This specification might not depict reality, as different venues determine different numbers of reviewers per manuscript.

To restrict dimensionality of this observation, we exemplarily utilise the configuration which achieved the best results for Step 1 e_2 (DBOW + $Votes_\delta = .5$) and only use the BTW'17 dataset. Baselines are adjusted to the

Table 8.7: Average cores SC as well as values for the five quantifiable aspects A , S , I , D , E for R_0 as well as the three baselines for different sizes of retrieved reviewer sets for all papers from BTW'17.

$ R_c $	RT	SC	A	S	I	D	E
2	R_0	.0401	.621	.7948	.5738	.3693	.3956
2	BL_{topX}	.0063	.4858	.69415	.6221	.1092	.362
2	BL_{tXr}	.0055	.3552	.6971	.5919	.1471	.302
2	BL_{rand}	.0062	.2516	.7357	.5784	.216	.2741
4	R_0	.0245	.5969	.8353	.5958	.2293	.3642
4	BL_{topX}	.0026	.4467	.7917	.6113	.1142	.3436
4	BL_{tXr}	.0038	.3448	.8014	.5958	.1488	.307
4	BL_{rand}	.0051	.2686	.8201	.5441	.2184	.2565
5	R_0	.0214	.5638	.8399	.5911	.217	.3536
5	BL_{topX}	.0008	.4333	.8045	.6082	.1141	.3365
5	BL_{tXr}	.004	.3299	.8088	.5942	.1624	.2994
5	BL_{rand}	.0041	.2819	.8628	.5526	.2227	.2675
6	R_0	.0191	.5284	.8399	.5956	.2058	.3513
6	BL_{topX}	.0006	.4123	.8145	.6067	.1198	.3336
6	BL_{tXr}	.0019	.3407	.8182	.5899	.1511	.3037
6	BL_{rand}	.0042	.2693	.8799	.5577	.204	.2761

respective reviewer set sizes. BL_{top3} is named BL_{topX} , BL_{t3r} will be called BL_{tXr} for these experiments.

Table 8.7 shows the average values for the score SC as well as the five quantifiable aspects for different sizes of the retrieved reviewer set for manuscripts. We experiment with sets of size 2 to 6 as higher numbers seem highly unusual for the presented problem. Other works define 3 to 5 [22], at least 2 [5] and 3 or more [26] reviewers as suitable.

R_0 achieves the best results for all reviewer set sizes. BL_{topX} achieving the worst results for set sizes bigger than 3 stands out sharply. This fact is attributed to the highly increased probability of reviewers which are on the very top of RL having joint publications. With Aspect 2 we prohibited this, so these sets are attributed with scores of 0.

Expertise is the aspect which most work focuses on [3, 13, 14, 27, 29]. For this aspect R_0 also achieves the highest average values in these experiments for all sizes of the constructed reviewer set.

From these observations as well as the ones before where we experimented with $|R_c| = 3$ we derive partial validity of hypothesis H_6 . RevASIDE seems highly suitable for different numbers of reviewer set sizes in terms of numeric

Table 8.8: Average times in seconds for conduction of Step 1, Step 2 and the combination of both for all three datasets for different reviewer set sizes $|R_c|$ for single manuscripts.

dataset	MOL'17				
task\ R_c \	2	3	4	5	6
avg. Step 1	.1988	.1986	.1975	.1976	.1979
avg. Step 2	1.0341	1.0588	1.1833	1.6478	3.0069
avg. Step 1 + Step 2	1.2329	1.2574	1.3808	1.8454	3.2048
dataset	BTW'17				
task\ R_c \	2	3	4	5	6
avg. Step 1	0.9839	.9857	.985	.9782	.9814
avg. Step 2	2.3582	2.3914	2.6006	3.481	6.264
avg. Step 1 + Step 2	3.3421	3.3771	3.5856	4.4592	7.2454
dataset	ECIR'17				
task\ R_c \	2	3	4	5	6
avg. Step 1	4.1918	4.189	4.1513	4.1712	4.1652
avg. Step 2	2.7968	2.8432	3.1639	4.3623	8.2469
avg. Step 1 + Step 2	6.9886	7.0322	7.3152	8.5335	12.4121

quality.

Run Time (Analysis of H_6)

To fully evaluate H_6 of RevASIDE's suitability for varying reviewer set sizes, the run time for the construction of recommended sets is analysed in this section. We report the average times of 10 runs of the program.

There are two pre-computations which are required for the conduction of Step 1: *first*, the collecting and conversion of papers of reviewers in the different document vector representations, in all cases the construction of tf-idf and LDA vectors which are needed for Step 2 also. In theory, this part can be conducted before the manuscripts are submitted to save some time. *Second*, the computation of the document vector representation for the manuscript(s) for which reviewer sets need to be recommended. The actual execution time of both steps of our approach is observed here, pre-computations which are independent of the actual manuscripts and only performed once were disregarded in the following calculations. For MOL'17 the pre-computation took 1.2837 seconds on average, for BTW'17 it took 6.9949 seconds on average and 19.4046 seconds on average for ECIR'17.

We performed the experiments on a server with 251 GB ram.

Again, to restrict dimensionality of this observation, we used the configuration which achieved the best results for Step 1 e_2 (DBOW + $Votes_{\delta=.5}$) which we already utilised in the previous evaluation in Section 8.8.2. Here in Step 1, papers of reviewer candidates are summed up, if they have a similarity of at least .5 with a manuscript. The complexity of this calculation is comparable to the other presented voting techniques, so we assume the generalisation of our experiments is possible.

Table 8.8 shows the average run times for both steps of our algorithm for varying reviewer set sizes in seconds. We observe differences in run times for the three observed datasets as well as $|R_c|$ but in general, both steps can be performed in few seconds. The observed time frame of 12 seconds per manuscript at most is hard to beat in a manual conduction of the reviewer set assignments. For fewer reviewers in a reviewer set, this period only decreases.

In general, the size of the reviewer candidate pool of a venue is influential. However, the bigger the recommended reviewer set size, the higher the probability of two or more authors from the observed set having a joint publication. This opposes Criterion 2, where we defined the need for disjoint publications of authors in a reviewer set. In case of co-authorships, the current reviewer set is disregarded and a score of 0 is returned. This also strongly affects run times.

In this experiment, we showed the usability of RevASIDE in terms of run times for varying recommended reviewer set sizes. Together with our previous evaluation in Section 8.8.2 we conclude the validity of H_6 of RevASIDE being suitable for different sizes of reviewer sets.

8.8.3 Qualitative Evaluation (Analysis of H_7)

In this part of the evaluation we assess hypothesis H_7 which covers the manual assessment of the sets resulting from Step 2 and RevASIDE's overall usefulness.

Best Configurations from Automatic Evaluation

In our first qualitative evaluation of Step 2, we examine the eleven (as $c_1 = c_5$) configurations which performed best for the different result types from the three datasets (see Table 8.5) in the quantitative evaluation. For the forty (twenty from ECIR'17 and twenty from BTW'17) documents which were used in the first manual evaluation, we compute lists of four reviewer sets for all configurations, consisting of one reviewer set produced by each of the three baselines as well as R_0 . We present the lists to an expert who then ranks the four entries according to suitability for the query manuscript from 1 (best) to

Table 8.9: Average positions (pos) sets computed by the different configurations (conf) were ordered to in the qualitative evaluation as well as the average number of relevant reviewers (#rel) and the average position of entries from the different RTs per set.

dataset result type	BTW'17							
	R_0		B_{t3}		B_{tr}		B_r	
pos \forall conf	2.3292		2.9083		2.4083		2.3373	
conf\measure	#rel	pos	#rel	pos	#rel	pos	#rel	pos
$c_1 = c_5$.3	2.05	0	3.65	.25	2.35	.65	1.95
c_2	.75	2.8	1.1	2.1	.85	2.2	.55	2.9
c_3	1.1	1.95	.85	3	1.1	2.25	.65	2.8
c_4	.6	2.55	.85	2.25	.55	2.65	.9	2.55
c_6	.4	2.5	.25	2.65	.5	2.5	.55	2.35
c_7	.4	2.6	.25	2.95	.25	2.65	.65	1.75
c_8	.2	2.1	0	4	.3	2.25	.5	1.65
c_9	.8	2.35	1.05	2.3	.85	2.4	.6	2.95
c_{10}	1.05	2.25	1.05	2.25	.95	2.65	.7	2.8
c_{11}	.65	2.3	.3	3.4	.55	2.4	1	1.9
c_{12}	.65	2.45	.7	2.7	.8	2.25	.65	2.5
dataset result type	ECIR'17							
	R_0		B_{t3}		B_{tr}		B_r	
pos \forall conf	2.1454		2.7272		2.4818		2.5818	
conf\measure	#rel	pos	#rel	pos	#rel	pos	#rel	pos
$c_1 = c_5$.6	2.4	.2	3.2	.6	2.3	.4	2.1
c_2	1	2.2	1.6	1.8	1.6	2.0	.7	3.7
c_3	.7	2.3	.7	2.2	.7	2.6	.7	2.6
c_4	1	1.6	1.5	1.9	.8	2.8	.5	3.7
c_6	.2	2.8	.4	2.6	.4	2.7	.3	1.9
c_7	.7	2	.5	3.3	.6	2.5	.4	2.2
c_8	.6	1.8	.2	3.7	.3	2.7	.8	1.8
c_9	.5	1.5	.5	3.2	.5	2.5	.2	2.8
c_{10}	.5	2.4	.5	2.9	.5	2.7	.7	2.0
c_{11}	.5	2.1	.3	3.4	.3	2.6	.7	1.9
c_{12}	1	2.5	1.4	1.8	1.4	1.9	.6	3.7

4 (worst), with the option of ties if two or more entries are equally suitable. Table 8.9 shows the average ranks of the result types in the evaluated lists for the two datasets, their average number of relevant reviewers per configuration and the average positions that entries from a specific RT achieved.

For BTW'17, the combination achieving the best results is c_8 (BERT, *MIN*, $k = 10$, $t = .25$) and (surprisingly) B_r . For ECIR'17, the combination achieving the best results came from configuration c_9 (BERT, *mRR*, $k = 20$, $t = .9$) and R_0 .

Overall, R_0 achieves the best results out of all combinations and datasets. B_r generates the best results for BTW'17, but highly depends on the configuration as it also achieves considerably bad results, especially for the ECIR'17 dataset. Although the results are greatly influenced by the configuration, R_0 performs consistently well in general. The combination of configuration and result type achieving the highest number of mean relevant reviewers per dataset is not the one achieving the best results in terms of positions, e.g. ECIR'17 + c_c + B_{tr} . This leads to the conclusion that it is not sufficient to consider only topical relevance in determining the most suitable combination. In both datasets, the RT achieving the best average positions is R_0 .

As data was not normally distributed in the different groups for both datasets, we used Kruskal-Wallis H tests on positions of the four RT for the two datasets, which resulted in significant differences. We conducted Mann-Whitney U tests on the positions of R_0 and each of the three baselines resulting from all configurations together on the respective datasets. In the BTW'17 dataset, R_0 performed significantly better than B_{t3} but no significant differences were found when compared to the two other baselines. In the ECIR'17 dataset, R_0 performed significantly better than all three baselines.

Best Configurations From Step 1

In a second manual evaluation of Step 2, we examined the best combinations from Step 1 (b_1 , e_1 and e_2) with CT = full, $k = 20$, $t = .5$ as the best performing combinations from Step 2 performed bad in Step 1. Configuration e_2 has already been utilised in the previous evaluation of H_6 (see Section 8.8.2 and Section 8.8.2). Table 8.10 was constructed exactly as described previously for Table 8.9. For both datasets, the best performing RT is R_0 . It achieves the best average position for all configurations together, and the combination resulting in the best position is e_2 with R_0 . For ECIR'17, R_0 also produces the best overall position for e_1 .

To better understand the impact of the five aspects, a human assessor also evaluated the quality of the results with the best combinations from Step 1 with respect to each aspect, assigning a value between 0 and 1 for each aspect.

Table 8.10: Average positions (pos) of sets computed by the different configurations (conf) in the qualitative evaluation as well as the average number of relevant reviewers (#rel) and the average position of entries from the different RTs per set. b_1 : tf-idf + *SUM*, e_1 : tf-idf + *MNZ*, e_2 : *DBOW* + *Votes $_{\delta=.5}$* .

dataset		BTW'17							
result type	R_0		B_{t3}		B_{tr}		B_r		
pos \forall conf	1.5833		1.95		2.933		3.5167		
conf \ measure	#rel	pos	#rel	pos	#rel	pos	#rel	pos	
b_1	.9	1.7	1.45	1.95	.95	3.15	.8	3.2	
e_1	.85	1.55	1.4	1.85	1.05	3.0	.6	3.6	
e_2	.8	1.5	1.25	2.05	.75	2.65	.6	3.75	
dataset		ECIR'17							
result type	R_0		B_{t3}		B_{tr}		B_r		
pos \forall conf	1.2667		1.3333		1.6167		3.2167		
conf \ measure	#rel	pos	#rel	pos	#rel	pos	#rel	pos	
b_1	1.6	1.6	1.55	1.45	1.25	1.55	.4	3.1	
e_1	1.55	1.1	2.1	1.15	1.3	1.4	.5	3.15	
e_2	1.6	1.1	2.05	1.4	1.3	1.9	.9	3.4	

Table 8.11 depicts average scores according to Equation 8.3 for combinations of the three best methods from Step 1 with all result types, manually assessed average values for the five aspects and “manual” scores computed by multiplying the per-aspect values. In this evaluation, we wanted to compare the manually constructed scores to the automatic ones and evaluate possible effects of opposing aspects. We observe vast differences in the manual scores mSC and the computed $scores$, in almost all cases B_{t3} achieves the highest mSC . As we have already seen in Tables 8.9 and 8.10, R_0 generally achieves the best average positions for sets of reviewers. This discrepancy further underlines the suitability of our approach. RevASIDE produces reviewer sets based on calculated aspects which are preferable in a manual evaluation to the sets from B_{t3} which achieved the highest mSC in the manual assessment of aspects.

We found a positive correlation of aspects mA and mI (.597 for BTW'17, .802 for ECIR'17) which is significant with Pearson's correlation coefficient for both datasets. A higher authority might be equivalent to a higher number of papers, especially in the last seven years, which might increase the probability of one of these papers being from the area of the manuscript and thus signals reviewers' interest. Also for both datasets, the negative correlation

Table 8.11: Configuration (c) and RTs with corresponding scores per dataset and manually assessed average values $\in [0, 1]$ (with 1 being the best possible and 0 being the worst possible value) for aspects of sets for the twenty evaluated papers. mA : $1/3 \forall$ reviewers with h index ≥ 25 ; mS : each $1/3$ if set contains at least one senior researcher, at least one junior researcher or at least one mid-career researcher; mI : $1/3 \forall$ reviewers who published a relevant paper in the seven previous years; mD : $1/3 \forall$ reviewer pairs without overlap in their work; mE : $1/3$ for each relevant reviewer in the set. Value mSC is calculated similarly to SC , all manually evaluated aspects are multiplied.

dataset $c \times RT$	BTW'17						
	SC	mSC	mA	mS	mI	mD	mE
$b_1 \times R_0$.0316	.0383	.8667	.6667	.25	.8833	.3
$b_1 \times B_{t3}$.0089	.0767	.8333	.6333	.3167	.95	.4833
$b_1 \times B_{tr}$.0079	.0412	.7333	.7333	.25	.9667	.3167
$b_1 \times B_r$.0056	.024	.65	.6833	.2167	.9333	.2667
$e_1 \times R_0$.0308	.0397	.9833	.6667	.2333	.9167	.2833
$e_1 \times B_{t3}$.0107	.0667	.9833	.55	.2833	.9333	.4667
$e_1 \times B_{tr}$.0082	.0446	.8333	.6667	.2333	.9833	.35
$e_1 \times B_r$.0085	.0142	.6667	.7333	.15	.9667	.2
$e_2 \times R_0$.0296	.0338	1	.6333	.2	1	.2667
$e_2 \times B_{t3}$.0078	.0704	.9	.65	.3333	.8667	.4167
$e_2 \times B_{tr}$.0082	.0223	.8167	.6667	.1667	.9833	.25
$e_2 \times B_r$.0069	.0076	.45	.75	.1167	.9667	.2
dataset $c \times RT$	ECIR'17						
$c \times RT$	SC	mSC	mA	mS	mI	mD	mE
$b_1 \times R_0$.0337	.1539	.8667	.6667	.5167	.9667	.5333
$b_1 \times B_{t3}$.0087	.1182	.8667	.55	.5333	.9	.5167
$b_1 \times B_{tr}$.009	.0753	.85	.6	.3667	.9667	.4167
$b_1 \times B_r$.0072	.006	.55	.6167	.1333	1	.1333
$e_1 \times R_0$.03	.122	1	.5667	.4166	1	.5167
$e_1 \times B_{t3}$.0082	.2432	1	.6167	.65	.8667	.7
$e_1 \times B_{tr}$.0087	.1097	.9667	.65	.4167	.9667	.4333
$e_1 \times B_r$.0043	.0089	.6667	.6833	.1167	1	.1667
$e_2 \times R_0$.0271	.1493	1	.6	.4667	1	.5333
$e_2 \times B_{t3}$.0096	.1686	.9167	.5	.5667	.95	.6833
$e_2 \times B_{tr}$.007	.084	.8667	.65	.35	.9833	.4333
$e_2 \times B_r$.0044	.0297	.5667	.6667	.2667	.9833	.3

between mI and mD (-.589 for BTW'17, -.72 for ECIR'17) is significant with Pearson's correlation coefficient. If reviewers in a set are very interested in a manuscript, it seems likely that the set is not as diverse. In BTW'17, mA is significantly correlated with mS (-.789), in ECIR'17 this negative correlation is not significant with Pearson's correlation coefficient. This observation can be explained as sets having high authority normally consist solely of researchers with high seniority. We found opposing objectives coded into the aspects which might have led to methods from Table 8.5 achieving low results in Step 1 but being useful in Step 2.

In general, average positions of sets from the different RTs are highly dependent on the configuration in BTW'17 and ECIR'17 for the best performing configurations in Step 2, but the overall best results are achieved independent of configuration by R_0 . From these observations, we conclude that R_0 and thereby RevASIDE is a well-performing solution of the reviewer set assignment problem which is generally applicable. Thus, hypothesis H_7 is verified.

8.9 Conclusion and Future Work

In this paper, we proposed and evaluated RevASIDE, a method for assigning complementing reviewer sets for submissions from fixed candidate pools. Our approach incorporates authority, seniority, interests of researchers, diversity of the reviewer set as well as candidates' expertise. Additionally, we presented three new datasets suitable for reviewer set recommendation.

In this context, we examine the expert search as well as the reviewer set assignment tasks and show RevASIDE's general applicability: for the first task, we reevaluated expert voting techniques utilising different document representations. We verified the general usefulness of Step 1 for the expert search (addressed with hypothesis H_1) and reviewer set recommendation task (addressed with hypothesis H_5). Additionally, we have shown the suitability of simple textual similarity methods utilising tf-idf compared to more advanced techniques using BERT, which in terms rejected hypothesis H_2 . For the second task, RevASIDE produces significantly higher overall scores for reviewer set assignment compared to three baselines in a quantitative evaluation, which shows the approach's usefulness. Our approach is useful for different recommended reviewer set sizes (see hypothesis H_6). In a qualitative evaluation, we observed that sets assembled by our system are generally significantly more suitable recommendations compared to our three baselines. We were able to confirm the results from the quantitative evaluation and thus verified H_7 .

Possible extensions might include weighting the different quantifiable aspects defined in Step 2 of the approach and incorporating the venue which reviewers are recommended for. The number of assigned reviewers could be varied for each submission to take into account papers with broad content.

Future work will focus on recommending suitable reviewer sets for whole venues. Here, the optimisation problem of single manuscripts is extended to include all manuscripts and several constraints such as individually differing maximal numbers of papers per reviewer come into consideration. Such an approach should also consider fairness [18] of the recommended reviewer sets. It would be interesting to observe gaps in the expertise displayed by the program committee in terms of fit with submitted manuscripts, together with suggesting new reviewers matching the missing criteria. Another feasible extension might be the recommendation of a program committee based on former and recent conferences and anticipated submissions. Here, topical development between years is important. Furthermore, explainability [30] of the recommended reviewer sets should be a priority. In our case, radar charts could be used for example to visualise the values which the sets achieved in the different quantifiable aspects.

Bibliography

- [1] Anastasia Ailamaki, Periklis Chrysogelos, Amol Deshpande, and Tim Kraska. The SIGMOD 2019 research track reviewing system. SIGMOD Record, 48(2):47–54, 2019.
- [2] Aris Anagnostopoulos, Luca Becchetti, Carlos Castillo, Aristides Giannis, and Stefano Leonardi. Online team formation in social networks. In Alain Mille, Fabien Gandon, Jacques Misselis, Michael Rabinovich, and Steffen Staab, editors, Proceedings of the 21st World Wide Web Conference 2012, WWW 2012, Lyon, France, April 16-20, 2012, pages 839–848. ACM, 2012.
- [3] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent dirichlet allocation. J. Mach. Learn. Res., 3:993–1022, 2003.
- [4] Guillaume Cabanac. Accuracy of inter-researcher similarity measures based on topical and social clues. Scientometrics, 87(3):597–620, 2011.
- [5] Guillaume Cabanac and Thomas Preuss. Capitalizing on order effects in the bids of peer-reviewed conferences to secure reviews by expert referees. J. Assoc. Inf. Sci. Technol., 64(2):405–415, 2013.
- [6] Laurent Charlin and Richard S. Zemel. The toronto paper matching system: An automated paper-reviewer assignment system. 2013.
- [7] Gohar Rehman Chughtai, Jia Lee, Mahnoor Shahzadi, Asif Kabir, and Muhammad Arshad Shehzad Hassan. An efficient ontology-based topic-specific article recommendation model for best-fit reviewers. Scientometrics, 122(1):249–265, 2020.
- [8] Samik Datta, Anirban Majumder, and K. V. M. Naidu. Capacitated team formation problem on social networks. CoRR, abs/1205.3643, 2012.
- [9] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. In NAACL-HLT 2019, pages 4171–4186. ACL, 2019.
- [10] Musa Ibrahim M. Ishag, Kwang-Ho Park, Jong Yun Lee, and Keun Ho Ryu. A pattern-based academic reviewer recommendation combining author-paper and diversity metrics. IEEE Access, 7:16460–16475, 2019.
- [11] Kalervo Järvelin and Jaana Kekäläinen. Cumulated gain-based evaluation of IR techniques. ACM Trans. Inf. Syst., 20(4):422–446, 2002.

- [12] Jian Jin, Baozhuang Niu, Ping Ji, and Qian Geng. An integer linear programming model of reviewer assignment with research interest considerations. Ann. Oper. Res., 291(1):409–433, 2020.
- [13] Yordan Kalmukov. An algorithm for automatic assignment of reviewers to papers. Scientometrics, 124(3):1811–1850, 2020.
- [14] Ngai Meng Kou, Leong Hou U, Nikos Mamoulis, Yuhong Li, Ye Li, and Zhiguo Gong. A topic-based reviewer assignment system. Proc. VLDB Endow., 8(12):1852–1855, 2015.
- [15] Christin Katharina Kreutz and Ralf Schenkel. Revaside: Assignment of suitable reviewer sets for publications from fixed candidate pools. In Eric Pardede, Maria Indrawan-Santiago, Pari Delir Haghighi, Matthias Steinbauer, Ismail Khalil, and Gabriele Kotsis, editors, iiWAS2021: The 23rd International Conference on Information Integration and Web Intelligence, Linz, Austria, 29 November 2021 - 1 December 2021, pages 57–68. ACM, 2021.
- [16] Quoc V. Le and Tomas Mikolov. Distributed representations of sentences and documents. In ICML 2014, volume 32 of JMLR Workshop and Conference Proceedings, pages 1188–1196. JMLR.org, 2014.
- [17] Craig Macdonald and Iadh Ounis. Voting for candidates: adapting data fusion techniques for an expert search task. In CIKM 2006, pages 387–396. ACM, 2006.
- [18] Lucas Machado and Kostas Stefanidis. Fair team recommendations for multidisciplinary projects. In WI 2019, pages 293–297. ACM, 2019.
- [19] Marcin Maleszka, Bernadetta Maleszka, Dariusz Król, Marcin Hernes, Denis Mayr Lima Martins, Leschek Homann, and Gottfried Vossen. A modular diversity based reviewer recommendation system. In ACIIDS 2020, volume 1178 of Communications in Computer and Information Science, pages 550–561. Springer, 2020.
- [20] Sofia Maria Nikolakaki, Mingxiang Cai, and Evimaria Terzi. Finding teams that balance expert load and task coverage. CoRR, abs/2011.04428, 2020.
- [21] Manos Papagelis, Dimitris Plexousakis, and Panagiotis Nikolaou. CONFIOUS: managing the electronic submission and reviewing process of scientific conferences. In WISE 2005, volume 3806 of LNCS, pages 711–720. Springer, 2005.

- [22] Florian Reitz. Harnessing historical corrections to build test collections for named entity disambiguation. In Eva Méndez, Fabio Crestani, Cristina Ribeiro, Gabriel David, and João Correia Lopes, editors, Digital Libraries for Open Knowledge, 22nd International Conference on Theory and Practice of Digital Libraries, TPDL 2018, Porto, Portugal, September 10-13, 2018, Proceedings, volume 11057 of Lecture Notes in Computer Science, pages 47–58. Springer, 2018.
- [23] Sherif Sakr, Mohamed Ragab, Mohamed Maher, and Ahmed Awad. MINARET: A recommendation framework for scientific reviewers. In EDBT 2019, pages 538–541. OpenProceedings.org, 2019.
- [24] Mariia Seleznova, Behrooz Omidvar-Tehrani, Sihem Amer-Yahia, and Eric Simon. Guided exploration of user groups. Proc. VLDB Endow., 13(9):1469–1482, 2020.
- [25] Arnab Sinha, Zhihong Shen, Yang Song, Hao Ma, Darrin Eide, Bo-June Paul Hsu, and Kuansan Wang. An overview of microsoft academic service (MAS) and applications. In WWW (Companion Volume), pages 243–246. ACM, 2015.
- [26] Hong Diep Tran, Guillaume Cabanac, and Gilles Hubert. Expert suggestion for conference program committees. In Saïd Assar, Oscar Pastor, and Haralambos Mouratidis, editors, 11th International Conference on Research Challenges in Information Science, RCIS 2017, Brighton, United Kingdom, May 10-12, 2017, pages 221–232. IEEE, 2017.
- [27] Chen Yang, Tingting Liu, Wenjie Yi, Xiaohong Chen, and Ben Niu. Identifying expertise through semantic modeling: A modified BBPSO algorithm for the reviewer assignment problem. Appl. Soft Comput., 94:106483, 2020.
- [28] Kai-Hsiang Yang, Tai-Liang Kuo, Hahn-Ming Lee, and Jan-Ming Ho. A reviewer recommendation system based on collaborative intelligence. In WI 2009, pages 564–567. IEEE Computer Society, 2009.
- [29] Dong Zhang, Shu Zhao, Zhen Duan, Jie Chen, Yanping Zhang, and Jie Tang. A multi-label classification method using a hierarchical and transparent representation for paper-reviewer recommendation. TOIS, 38:1–20, 02 2020.
- [30] Yongfeng Zhang and Xu Chen. Explainable recommendation: A survey and new perspectives. Found. Trends Inf. Retr., 14(1):1–101, 2020.

- [31] Shu Zhao, Dong Zhang, Zhen Duan, Jie Chen, Yan-Ping Zhang, and Jie Tang. A novel classification method for paper-reviewer recommendation. Scientometrics, 115(3):1293–1313, 2018.

9. Diverse Reviewer Suggestion for Extending Conference Program Committees

Outline

9.1	Introduction	245
9.2	Related Work	247
9.3	Problem Setting	250
9.3.1	Problem Statement	250
9.3.2	Notation	250
9.4	Method	251
9.4.1	Modelling Diversity	251
9.4.2	Algorithm	251
PR4All	253
DiveRS Subroutine: Reviewer Assignment	253
DiveRS Main Routine: Reviewer Suggestion for PC Extension	254
9.4.3	Practical Issues and Effects of Parameters	256
9.5	Experimental Setup	257
9.5.1	Datasets	257
9.5.2	Parameter Settings	258
9.5.3	Established Measures	258
9.5.4	Novel Measures	259
9.6	Experiments	260
9.6.1	Part 1: Reviewer Assignment	260
Automatic Evaluation	260
Manual Evaluation	261
Summary of Findings	262
9.6.2	Part 2: Reviewer Suggestion	263
9.7	Conclusion	264

Bibliographic Information

Kreutz, C. K., Balog, K., Schenkel, R. (2021). Diverse Reviewer Suggestion for Extending Conference Program Committees. In *WI-IAT 2021* (pp. 79–86). ACM. <https://doi.org/10.1145/3486622.3493931>.

Copyright Notice

©2021 ACM. This is an accepted but reformatted version of this article. Clarification of the copyright adjusted according to the guidelines of the publisher.

Keywords

Reviewer Assignment • Program Committee Extension • Reviewer Recommendation • Reviewer Coverage • Flow-Based Algorithm

Abstract

Automated reviewer recommendation for scientific conferences currently relies on the assumption that the program committee has the necessary expertise to handle all submissions. However, topical discrepancies between received submissions and reviewer candidates might lead to unreliable reviews or overburdening of reviewers, and may result in the rejection of high-quality papers. In this work, we present DiveRS, an explainable flow-based reviewer assignment approach, which automatically generates reviewer assignments as well as suggestions for extending the current program committee with new reviewer candidates. Our algorithm focuses on the diversity of the set of reviewers assigned to papers, which has been mostly disregarded in prior work. Specifically, we consider diversity in terms of professional background, location and seniority. Using two real world conference datasets for evaluation, we show that DiveRS improves diversity compared to both real assignments and a state-of-the-art flow-based reviewer assignment approach. Further, based on human assessments by former PC chairs, we find that DiveRS can effectively trade off some of the topical suitability in order to construct more diverse reviewer assignments.

9.1 Introduction

Scientific publishing heavily relies on peer review, which is typically performed by members of the program committee (PC) of a conference. In general, PCs need to grow and change each year: to keep up with the increasing number of submissions [24], to avoid tunnel vision as well as unchanging perceptions of good or bad concepts [5] (e.g., the ACM SIGSOFT policy recommends to change one third of the members each year [24]), and former PC members might become unavailable [6]. According to current practice, organisers compose the PC before the submission period of manuscripts ends. Once submissions are closed, each manuscript gets a number of PC members, also called reviewers, assigned by the PC chairs, either manually or automatically (based on bidding information or preferred topics, entered by reviewers) [21, 22, 13]. Importantly, to the best of our knowledge, current approaches to reviewer recommendation assume a perfectly composed PC, and do not consider modification or extension as a necessity.

To the best of our knowledge, there is currently no way of reliably estimating the topical composition or amount of incoming submissions. Therefore, a previously disregarded problem is the possible mismatch between the expertise of current PC members and the expertise required for the assessment of

all submissions. This problem may be further amplified by the ever-changing PC. Consequences of the mismatch might result in manuscripts tackling topics far from the PC’s interests being less favourably reviewed [16] and a general overburdening of reviewers. This, in turn, might lead to innovative and complex submissions being rejected solely due to low-quality reviews [1] or failure to find errors in submissions [20].

A solution for the above issues would be the inclusion of new and additional PC members after the submission period ended, but before the review assignments have been made. This can especially help to cover new or emerging research topics [6] and to ensure that under-represented groups can gain exposure and reviewing experience [19]. Identification of appropriate candidates is challenging as PC members should be diverse in localities, seniority [20, 16], research topics and gender [16]. Furthermore, suggested candidates should be explainable, in order to aid the conference chairs in effective and efficient decision making.

In this paper we focus not only on the automatic assignments of reviewers to submissions (i.e., reviewer assignment), but also introduce and address the problem of reviewer coverage: ensuring the assignment of suitable reviewers to all submissions. This gives rise to the novel task of reviewer suggestion for PC extension: given the current PC and all submitted manuscripts of a venue, recommend new reviewer candidates to be added to the PC. Note that these two tasks are interconnected: our reviewer assignment method identifies submissions that would not receive adequate reviewers using the current PC, which in turn triggers the suggestion of new reviewers to extend the PC. Those newly included persons should not only be capable of ideally assessing multiple manuscripts but also ensure diversity of the whole PC. Note that some gaps in the PC can be identified without requiring paper-reviewer assignments (e.g., not enough senior reviewers, reviewers from a given location or stark imbalance in academic vs. non-academic backgrounds of reviewers), while other gaps may only be identified once a (preliminary) assignment is done.

The main contribution of this work is a flow-based reviewer suggestion and PC extension approach, termed DiveRS. The main idea behind DiveRS is to iteratively identify submissions that are unlikely to get a set of suitable reviewers assigned. These problematic submissions and currently underrepresented diversity aspects (professional background, location or seniority) determine the reviewer candidates for inclusion in the PC to support a feasible reviewer assignment. We capture these characteristics in a constrained optimisation problem. At its core, DiveRS relies on a reviewer assignment method, which considers reviewers as a set for each paper, in order to satisfy diversity constraints. Additionally, reviewers’ individual upper and bounds

of the numbers of papers to review, and their conflicts of interests, also need to be respected.

We evaluate DiveRS on real-world conference datasets in two parts. First, we compare it on the task of reviewer assignment against the current state-of-the-art, PR4All [21], and against real assignments, in terms of both established measures (mean number of papers assigned, fairness, and textual diversity of reviewer sets) as well as novel measures (diversity and dependency between reviewers). We show that DiveRS achieves fairness that is on par with PR4All, while being superior in terms of diversity. Second, we evaluate the reviewer suggestion task by asking actual PC chairs to assess the generated suggestions for PC extension in terms of relevance, usefulness, and accompanying explanation. Our results indicate that DiveRS can effectively trade off topical suitability in order to improve the diversity of the assigned reviewer sets.

In summary, we make the following contributions:

- We propose the reviewer coverage problem as an extension of the reviewer assignment problem, where we no longer assume the current PC to be perfectly suitable for all submissions. We define the extension of the PC, to accommodate possibly ill-covered submissions, as part of the objective.
- We present DiveRS¹, a novel reviewer assignment and PC extension approach. It incorporates previously overlooked diversity aspects in terms of professional background, location and seniority of reviewer candidates directly in the assignment process, and generates explainable suggestions for extending the PC.
- We propose new measures for evaluating the diversity and dependency of reviewer sets.
- We automatically evaluate our approach on two real-world datasets and demonstrate its suitability in manual evaluations with the actual PC chairs of these conferences.

9.2 Related Work

Areas related to our work are reviewer assignment, which corresponds to the typical reviewer assignment problem, as well as the general field of *program*

¹DiveRS implementation: <https://github.com/kreutzch/DiveRS>

committee construction, which relates to the extension of PCs. For conference organisers there are many systems supporting the bidding and reviewer assignment process but “[e]xtending PCs based on submitted papers” as identified as a future objective by Price and Flach [17] has not yet been tackled to the best of our knowledge. There have been efforts to expand expert sets to hold more persons similar to the ones already contained in the set [25] but these approaches differ from our research objective: instead of finding more similar experts, our goal is to suggest an unbiased and diverse set of reviewer candidates to better cover the topical composition of incoming submissions.

Reviewer Assignment. There is a multiplicity of author-topic models to capture topical relationships between authors and (their) papers [8, 9, 15, 18, 23]. We refrain from discussing them in detail or utilising them here, as our focus within assigning reviewers to submissions lies not only on topical similarity of the two, but more on diversity aspects.

Conry et al. [4] tackle the reviewer assignment problem with given bidding information as an optimisation problem with global criteria. They extend bidding data by predicting new preferences of reviewers, and utilise manuscript as well as reviewer similarities. Liu et al. [13] recommend n reviewers for each manuscript which are dependent on each other. They model reviewers’ expertise, authority and diversity in a graph, which they traverse with random walk with restart. The number of co-authorships is modelled as authority. Tang et al. [22] propose a constraint-based optimisation framework that proposes sets of reviewers for query manuscripts and user feedback, if available. They incorporate expertise matching, authority aspects based on seniority, load balance and aim to maximise the topic coverage between reviewer sets and manuscripts, using LDA. Long et al. [14] study topic coverage and fairness of manuscript-reviewer assignments. They maximise the numbers of different topics of manuscripts in which the assigned reviewer set is knowledgeable. Additionally, they define and regard the influence of different conflict of interest types, such as the competitor relationship, in the assignment. Kou et al. [11] build upon [14] and instead observe a weighted topic coverage score. Their approach calculates the assignment resulting in the approximate maximum weight-coverage group-based scores, while fulfilling workload and reviewer set size constraints.

Jecmen et al. [7] provide a solution for the reviewer assignment problem, which focuses on supporting the integrity of the peer review process. The approach prevents reviewers’ manipulation efforts in the assignment to either submit overly positive or negative feedback as well as de-anonymise the reviewing process. Here, the similarity between manuscripts and reviewers’ profiles (expertise) is a critical factor in the randomised assignment. Kobren et al. [10] introduce a paper-reviewer-assignment strategy which incorporates

upper and lower load bounds per reviewer, guarantees a minimal required expertise in the area of the submission from all assigned reviewers and optimises a global objective. They present a linear programming and min-cost flow-based heuristic approach.

The Toronto Paper Matching System (TPMS) [3] conducts automatic reviewer assignment for all manuscripts submitted to a conference by using either word count representation or LDA topics, but can also incorporate reviewers' bids on submissions. TPMS supports some constraints: papers must be reviewed by three reviewers, and reviewers are assigned not more than a certain limit of papers. Reviewers for manuscripts are determined based on expertise extracted from their published papers and maximising the similarity between reviewers and manuscripts. Stelmakh et al. [21] use TPMS in PR4All; they propose an approach utilising a max-flow algorithm to identify the top- k papers submitted to conferences, which should be accepted. They focus on fairly assigning suitable reviewer sets to all submissions via TPMS, especially those which received low similarity with all reviewer candidates. This approach is considered as the state of the art for flow-based reviewer assignment [10].

We note that the datasets used in related work are mostly not available online and even fewer contain all submissions of a conference, i.e., include rejected papers. Those that remain either do not contain the real reviewers (ICLR 2018 [7, 21]) or do not contain names of both reviewers and authors (MIDL, CVPR and CVPR2018 [10]). Thus, to the best of our knowledge, there is no publicly available dataset including rejected papers, and non-anonymised reviewer and author names from a real conference. Therefore, we create our own datasets based on real conference data in §9.5.1.

Program Committee Construction. Han et al. [6] recommend PC members for conferences based on the previous year's PC and core authors, preferring candidates socially close to current chairs. They build a language model for a conference by aggregating previously published papers and compare it to PC candidates' publications. Authoritativeness of candidates influences the recommendations. Sekar [19] introduces EZ-PC, a tool to define constraining factors and help automate the PC formation process as an integer linear programming problem. Several factors are considered: topical coverage, diversity of the PC, avoiding over-representation of groups and keeping the PC size manageable. The main differences between their work and ours are that diversity constraints in EZ-PC are on the PC level, and they do not support reviewer assignment.

9.3 Problem Setting

9.3.1 Problem Statement

We define the reviewer coverage problem (RCP) as an extension of the reviewer assignment problem (RAP) for scientific manuscripts. Both problems have the underlying goal of finding suitable sets of reviewers for each manuscript. These sets need to be constructed such that (i) reviewer expertise is sufficient for the topics of the respective manuscript, (ii) there are no conflicts of interests between authors of submissions and reviewers, and (iii) overall reviewer load constraints are met. Contrasting with RAP, RCP does not assume that the current PC is perfect (i.e., has sufficient coverage), but explicitly allows for its extension by adding reviewer candidates from an extended reviewer candidate pool (ERC). So the immediate goal for RCP is the suggestion of new PC members, which leads to sufficient reviewer expertise for all submissions, while also ensuring diversity in the PC in terms of (i) seniority, (ii) location, and (iii) industrial/academic affiliation.² An additional desirable condition for the inclusion of new PC members is their ability to review multiple papers. Formally, the output of RCP is twofold: (1) a ranked list of reviewer suggestions to include in the PC and (2) an assignment of reviewer sets to submissions.

9.3.2 Notation

M describes the set of submissions to a conference for which reviewers from the program committee PC need to be assigned. A single reviewer is addressed as $r_i, i \in \{0, \dots, |PC| - 1\}$ or only by their index i . We address a single submission as $m_j, j \in \{0, \dots, |M| - 1\}$ or only by their index j . An assignment is feasible if all submissions are assigned a predefined number of reviewers λ , the number of submissions a reviewer is assigned lies between a predefined lower (μ_i^l) and upper bound (μ_i^u), which is specific for each reviewer i , and conflicts of interests (COI) are not violated by the assignment. The reviewer set assigned to a submission j under a feasible assignment A is denoted by $R_A(j)$. We store similarities of reviewers and submissions in $S \in [0, 1]^{|PC| \times |M|}$; the similarity S_{ij} of reviewer i with submission j is seen as a proxy for expected review quality [21] and can be determined, e.g., by the cosine similarity between TF-IDF representations of j 's and i 's profiles, composed of their papers. In case of a COI between i and j , we set $S_{ij} = -1$.

²Gender would also be a desirable diversity aspect for PCs [16], but we consciously refrain from touching this subject due to the challenges involved in collecting potentially personal information from reviewers for inclusion in our datasets.

We store dependencies between reviewers in $dep \in \{0, 1\}^{|PC| \times |PC|}$; dependencies such as recent co-authorships between reviewers i and k are expressed by $dep_{ik} = 1$ if there is a dependency and 0 otherwise.

9.4 Method

We introduce **DiveRS**, a **D**iverse **R**ewiever **S**uggestion system for extending conference program committees. It focuses not only on fairness of reviewer assignments but also considers diversity in professional background, location of reviewer candidates and their seniority. We build on and extend a previous state-of-the-art flow-based approach [21], by explicitly modelling diversity as a layer in the flow-graph; see Fig. 9.1.

9.4.1 Modelling Diversity

We focus on diversity in three different areas: professional background, location and seniority. We integrate these properties of the assignment in a specific layer in our flow network between papers and reviewers (diversity layer **L4** in Fig. 9.1). Diversity in professional background means that each reviewer set has to contain at least one reviewer working in academia and one reviewer (possibly the same one) working in industry. For diversity in location it would be desirable to include reviewers in a reviewer set with locations from completely different geographical locations. The goal here is to not have all reviewers in a set being located on the same continent. We achieve diversity in seniority by enforcing each reviewer set to contain at least one senior researcher [3, 22]. Meanwhile, overburdening of reviewers from underrepresented backgrounds can be prevented by decreasing their possible reviewing load. Satisfying all diversity constraints might lead to an increase of the PC size.

9.4.2 Algorithm

DiveRS identifies submissions with high probability of not obtaining enough suitable (topically fitting and diverse from each other) reviewers and adds new reviewers to the PC accordingly. It then constructs suitable reviewer sets for all submissions from the extended PC. Our reviewer suggestion approach is inspired by PR4All [21], the current state-of-the-art in flow-based reviewer assignment [10]. However, PR4All tackles the reviewer assignment problem (RAP), which is only one element of the larger reviewer coverage problem (RCP) that we are addressing (cf. §9.3.1). We do not only construct suitable

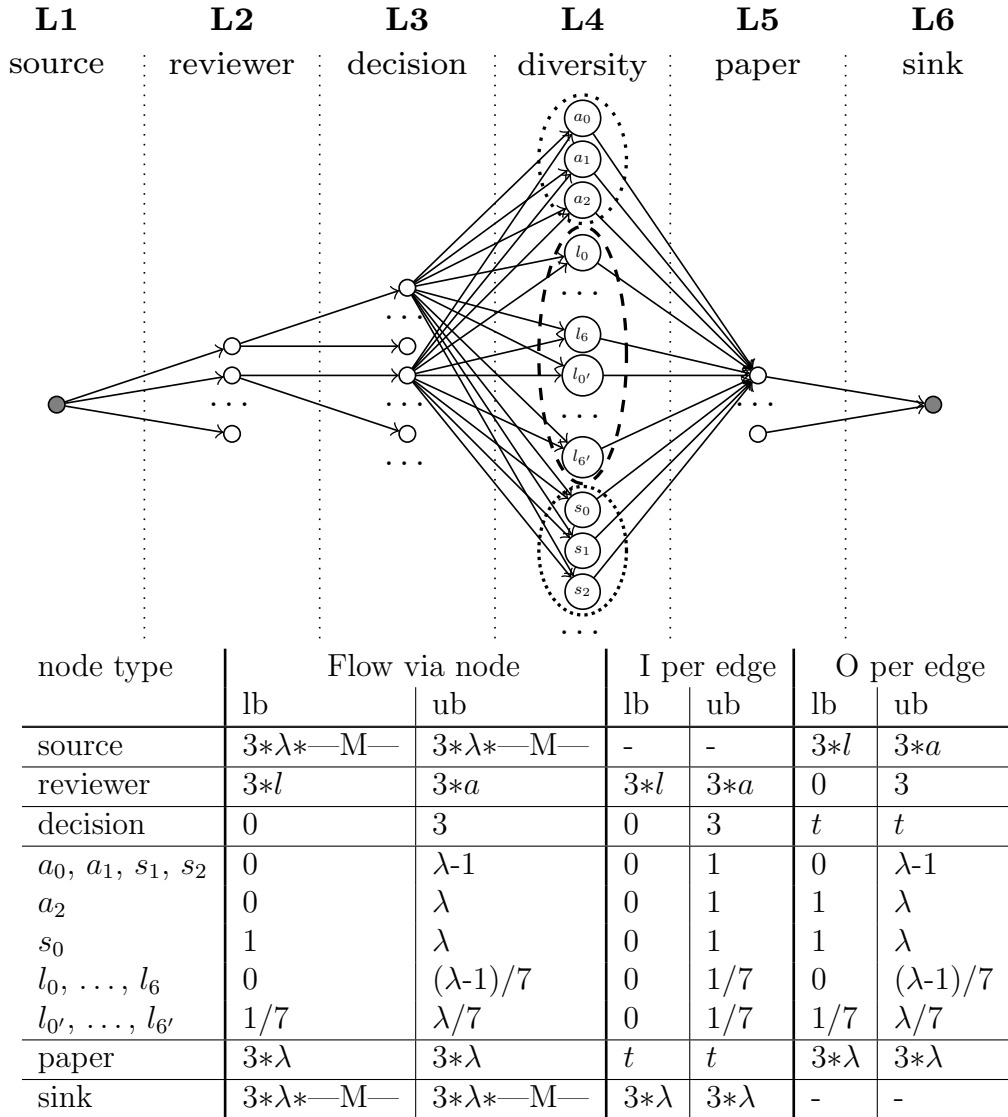


Figure 9.1: **Top:** A simplified version of the flow network constructed by DiveRS. Only the depicted edges between neighbouring layers allow flow. Background nodes in the dotted ellipse are used to ensure diversity in the professional background, those in the dashed ellipse are used to enforce diversity in the continent of the assigned reviewers and those in the densely dotted ellipse guarantee diversity in seniority.

Bottom: Lower (lb) and upper bounds (ub) of incoming (I) and outgoing flow (O) per edge as well as the general flow via a specific node type with ability a , demand λ , lowest load l and amount of flow depending on the node type t .

assignments but also identify possibly problematic papers and actively extend the PC to ensure diverse reviewer sets.

We first discuss the limitations of PR4All in §9.4.2, followed by the introduction of DiveRS’s reviewer assignment subroutine in §9.4.2 and its main routine in §9.4.2 which is responsible for identifying problematic submissions and suitable reviewer candidates.

PR4All

The goal of PR4All [21] is the fair assignment of suitable reviewer sets for all submissions with a focus on the most disadvantaged ones. The iterative approach fixes one reviewer set for the worst off submission in each iteration. Each iteration constructs partial reviewer sets for all unassigned submissions consisting of the most similar reviewers. This is their central optimisation problem. The partial sets are merged and considered a possible assignment. One assignment resulting in the highest fairness is computed out of several of these possible assignments. From the best overall merged assignment, the worst off paper is finally assigned its reviewers. Fixed (worst off) papers are disregarded in the next iterative assignment and merge steps until all papers are fixed.

Due to the merge step, PR4All cannot introduce new conditions for the single reviewers and reviewer sets on the final level only, e.g., lower bounds (μ^l) for the number of assigned submissions for each reviewer or that each set must contain at least one reviewer from industry and one from academia. Instead, these lower bounds for reviewers and conditions for reviewer sets would be applied during all parts of the assignment process. Overcoming this issue is non-trivial as all partial assignments which are then merged fulfilling the new conditions could also lead to violated upper bounds (μ^u) and an excess of industry reviewers per final reviewer set. For their initial run with sets of size 1, the one reviewer would be required to represent both professional backgrounds which is hard to find. Conditions that only merged assignments have to fulfil cannot be realised in the described optimisation problem. So, PR4All prevents definition of desirable properties for final assignments that surpass mere similarity, such as diversity in certain properties.

DiveRS Subroutine: Reviewer Assignment

We strive to overcome some of the weaknesses in reviewer assignment encountered in PR4All: we introduce individual upper (μ_i^u) and lower bounds (μ_i^l) of reviewing abilities for each reviewer i [10]. The lower bound describes the number of submissions, a reviewer has to review at least. Additionally,

we allow for the definition of dependencies between reviewers (e.g., in case of shared current affiliations or recent collaborations), as reviewers in a set should have distinct affiliations to make sure that their opinions are sufficiently independent from each other [7]. The resulting constraint can mathematically be described by the expression $con_I : \sum_{j \in M} \sum_{i, k \in R_A(j), i \neq k} dep_{ik} == 0$.

Our goal is to assign reviewers to the best fitting submissions to maximise the overall similarity between assigned reviewers and submissions. The following equation formulates the optimisation objective: $maximise J_f^S(A) := \sum_{i \in R_A(j), j \in M} f(S_{ij})$ while all submissions receive λ reviewers, dependencies between reviewers, COIs, diversity constraints of reviewer sets as well as reviewers' lower and upper abilities are not violated. f is a monotonically increasing function used to transform similarity values $[0 : 1] \rightarrow [0 : \infty]$ [21].

Algorithm 1 (main routine) and Algorithm 2 (subroutine) depict the pseudo code of our approach. In the subroutine we construct our flow network such that reviewers review a number of submissions limited by their upper and lower bounds. Submissions are reviewed by λ reviewers. Each reviewer set for a submission is diverse in professional background (at least one from industry and one from academia), location (not all from the same continent) and seniority (at least one senior reviewer). We only allow the allocation of reviewers to submissions, if this combination is contained in *pairs*. The decision if a reviewer is assigned to a submission is contained in **L3**, if there is flow over an edge $(i, i * |M| + j)$ between **L2** (reviewer i) and **L3** (decision to review submission j), i is assigned as reviewer for j . If we can compute a max flow, we find an feasible assignment.

DiveRS Main Routine: Reviewer Suggestion for PC Extension

In the main routine we generally first check if the original PC contains enough reviewers such that each submission can be assigned someone from both professional backgrounds as well as one senior reviewer. Otherwise, we include new reviewers with missing diversity properties in the PC from *ERC* (l. 1). The *ERC* could, e.g., be composed of authors of former instances of a conference. The similarity threshold θ heavily influences DiveRS, it defines the minimal similarity between submissions and assigned reviewers [10] (l. 3-5, 11). If $\theta=0$, the algorithm often finds a solution to the reviewer assignment problem, computed by the subroutine after including new reviewers based on underrepresented diversity aspects (l. 7), and does not need to identify possibly problematic papers (l. 8-9). Those problematic papers (l. 9) are submissions which have a high probability of not getting assigned reviewers (i.e., where runs of the sub-routine oftentimes fail if they are part of M_θ).

Algorithm 1 DiveRS main routine: reviewer suggestion for PC extension.

Input: $\lambda, M, PC, S, \mu^l, \mu^u, dep, acaInd, location, seniority, tries, S_ERC, \mu^l_ERC, \mu^u_ERC, dep_ERC, acaInd_ERC, location_ERC, seniority_ERC, \theta, \kappa$

Output: Reviewer assignment A , problematic papers $M_{<\theta}$

- 1: while PC is not able to produce assignment based on ability and seniority or professional background and —ERC— $\neq 0$: include new reviewers from underrepresented aspects with highest average similarities to all manuscripts
 - 2: if abilities of PC are not enough to find assignment: terminate with error
 - 3: \forall reviewer-submission pairs from S and S_{ERC} set similarity = -1 if similarity $\leq \theta$ (equivalent to COI)
 - 4: $M_\theta = M$ w/o submissions with all similarities $< \theta$
 - 5: delete reviewers r from PC where $\mu_r^l \leq$ number of submissions with which they have similarity $\geq \theta$
 - 6: $pairs =$ compute all pairs of reviewers in PC and papers in M_θ
 - 7: **while** $sub(\lambda, M_\theta, PC, S, [0]^{|PC|}, \mu^u, dep, acaInd, location, seniority, pairs)$ does not produce assignment **do**
 - 8: $fewCandidatePapers =$ papers with $\leq \lambda$ reviewers w/o COI
 - 9: run sub multiple times w/o $fewCandidatePapers$ and w/o predefined % of submissions to identify (possibly problematic) submissions where run fails, i.e., for which no assignment can be computed due to ill-fitting or few reviewers in the submission's area; adjust $pairs$ and M_θ for runs, papers with highest probability of failed run are $problemPapers$
 - 10: insert up to κ reviewers in PC from ERC fitting $fewCandidatePapers$ + most problematic $problemPapers$ and underrepresented background variables best
 - 11: delete papers from M as out of scope for which not enough reviewer ($< \lambda$) candidates with similarity $\geq \theta$ can be found
 - 12: $pairs =$ compute all pairs of reviewers and papers
 - 13: **end while**
 - 14: $f_A = []$ // list of all feasible assignments
 - 15: **for** $try = 0, try \leq tries, try ++$ **do**
 - 16: $pairs_c =$ drop predefined percentage of $pairs$
 - 17: $A_{curr} = sub(\lambda, M_\theta, PC, S, \mu^l, \mu^u, dep, acaInd, location, seniority, pairs_c)$
 - 18: if $A_{curr} \neq \emptyset : f_A.append(A_{curr})$
 - 19: **end for**
 - 20: return most diverse assignment from $f_A, M - M_\theta$
-

Algorithm 2 DiveRS subroutine: reviewer assignment step *sub*.

Input: $\lambda, M, PC, S, \mu^l, \mu^u, dep, acaInd, location, seniority, pairs, \theta$

Output: Computed reviewer assignment A_{curr} , \emptyset if unfeasible

- 1: **Initialization:** flow network (see Figure 9.1):
 - L1** (source, 1 vertex)
 - L2** (reviewers, vertex $\forall i \in PC$)
 - L3** (reviewer paper decision, vertex $\forall j \in M * \forall i \in PC$)
 - L4** (diversity, 3 vertex types, vertex $\forall x \in M * 20$, see Figure 9.1)
 - L5** (papers, vertex $\forall j \in M$)
 - L6** (sink, 1 vertex)
 - 2: Reset flow constraints for all vertices in the network: source, reviewer, decision, diversity, papers, sink
 - 3: $\forall (ij) \in pairs$: insert edge (i, j) between **L2** and **L3** (i.e., set $capacity[i, i * |M| + j] = 3$), adjust flow constraints
 - 4: Compute max flow, create assignment A_{curr} corresponding to max flow $\forall (ij)$: **if** flow on edge $(i, i * |M| + j)$ between **L2** and **L3** **then** assign reviewer i to submission j
 - 5: return A_{curr}
-

The higher the value of θ , the more difficult it is to find a feasible assignment. A value κ describes the number of fitting inserted reviewers per iteration (l. 10). If a feasible assignment (l. 7) has been found for a reviewer set, we randomly exclude reviewers from reviewing submissions in order to find the most diverse assignment (l. 15-20).

Figure 9.1 depicts our network and associated flow constraints. Nodes a_x indicate the professional background of a reviewer ($a_0 = \text{industry}$, $a_1 = \text{academia}$, $a_2 = \text{both}$). The different l_x indicate the location background of reviewers (l_y indicates the presence of a continent in continents associated with a specific reviewer while $l_{y'}$ indicates the continent's absence in their continents; $l_0 = \text{South America}$, $l_1 = \text{Africa}$, $l_2 = \text{Antarctica}$, $l_3 = \text{Asia}$, $l_4 = \text{Oceania}$, $l_5 = \text{North America}$, $l_6 = \text{Europe}$). Nodes s_x indicate the different levels of seniority of researchers ($s_0 = \text{senior}$, $s_1 = \text{advanced}$, $s_2 = \text{junior}$).

In our implementation, we utilise Gurobi³, a commonly used [21, 10] solver software for mathematical optimisation.

9.4.3 Practical Issues and Effects of Parameters

Our approach tackles several practical issues which arise in PC extension and reviewer assignment:

³<https://www.gurobi.com/>

- Submissions for which no suitable reviewers can be found as their topics might be out of scope of a current conference can be identified and considered manually.
- Reviewers that are part of the original PC should all be assigned at least one submission out of courtesy, even if they might no longer fit the topical composition of the conference. For subsequent instances of the conference, these individuals may no longer be invited to the PC. Our approach identifies such reviewers and is able to assign them to current submissions nevertheless.

Running time constraints influence the choice of parameters:

- The higher the similarity threshold θ is set, the more iterations (l. 7-12 A. 1) are required until a feasible assignment is found. The higher the number of included new reviewers per run κ is set (l. 10 A. 1), the longer one single run of the assignment step (A. 2) takes but in total less iterations might be needed. If κ is high, the total review load will be distributed among the many new candidates. In order to keep the PC comparably small, we advise to have a low κ and more iterations in total.
- The higher the bias towards incorporation of reviewers with underrepresented background variables (l. 10 A. 1), the less focus is put on similarity of reviewers and submissions. In consequence, fairness of assignments decreases while diversity increases.

Note that we do not separately handle sub- or meta-reviewing but DiveRS can be used in these steps with different parametrisation.

9.5 Experimental Setup

We present our experimental setup, introducing two new datasets (§9.5.1), our parameter settings (§9.5.2), an overview of established measures (§9.5.3), and novel ones for reviewer assignment assessment, namely diversity and dependency (§9.5.4).

9.5.1 Datasets

We evaluate on two real-world conference datasets based on the International Conference on the Theory of Information Retrieval (ICTIR) in 2019 (I'19)

and 2020 (I’20). The data was made available to us by the conference organisers upon request and signing an NDA. The datasets include all manuscripts submitted to the conferences, not only the accepted ones, authors of submissions, reviewers, real reviewer-submission assignments, and constructed extended reviewer candidate pools. I’19 (I’20) contains 78 (65) papers submitted by 201 (184) authors. 43 (30) papers were accepted and 36 (35) rejected. There were 43 (67) reviewers. The extended reviewer candidate pool consists of 6,445 (5,692) authors from papers which appeared in CIKM, ECIR, ICTIR and SIGIR in the previous five instances of the conferences.

For all *reviewers*, we retrieved their DBLP key [12], COIs and dependencies (collaborators from the previous five years and persons with current shared affiliations), seniority, location, current affiliation(s) as well as information on whether they are working in industry and/or academia. Demographics were automatically derived from their affiliations and earliest published paper. Additionally, we collected the titles of their publications up until the year of the conference, and abstracts from the previous five years for papers which appeared with Springer or ACM. We performed further manual post-processing to ensure high data quality.

9.5.2 Parameter Settings

We assign each submission to three reviewers, following the practice of the I’19 and I’20 conferences. Similarity between submissions and reviewers (a concatenation of their publications’ titles and abstracts) is taken to be the cosine similarity of TF-IDF-weighted document representations, thus all similarity values lie in $[0,1]$. We utilise $f(S_{ij}) = \frac{1}{1-S_{ij}}$ if $S_{ij} < 1$ and $1 * e^6$ otherwise [21]. For DiveRS we set $tries = 25$, $\kappa = 10$ and $\theta > 0$ to .25 for I’20 and .15 for I’19.⁴ We set $\mu^u = 9$ for I’19 and $= 7$ for I’20 according to the real number of maximal assigned submissions per reviewer candidate.

For the manual evaluations we obtain reliable human assessments by asking respective PC chairs (3 from I’19 and 2 from I’20) to fill out a questionnaire.

9.5.3 Established Measures

The following established measures describe the *quality* of reviewer assignments: mean *number of papers* assigned to single reviewers [10], *fairness* of the assignment [21, 10], and *average textual diversity* of reviewer sets [13].

⁴Different values for θ had to be chosen to find feasible assignments, as θ is highly dependent on topical fit between the submissions and the PC.

Fairness of an assignment A is defined as the minimal summed similarity between any submission j and its reviewers $R_A(j)$ [21]: $\Gamma_f^S(A) = \min_{j \in M} \left(\sum_{i \in R_A(j)} f(S_{ij}) \right)$, with f being a monotonically increasing function $[0, 1] \rightarrow [0, \infty]$. *Average textual diversity* of reviewer sets is calculated by the average Kullback-Leibler (KL) divergence between pairs of reviewers assigned to the submissions [13]: $KL(A) = \text{avg}_{j \in M} \left(\frac{\sum_{i,k \in R_A(j), i \neq k} KL\text{Divergence}(i,k)}{|R_A(j)| * (|R_A(j)| - 1) / 2} \right)$. We calculate this value on an unigram language model of the reviewer’s publication information. Higher values for average KL-divergence indicate less similar reviewers in reviewer sets. Desirable complementary reviewers [3] produce a high value.

9.5.4 Novel Measures

We present a novel measure for quantifying the diversity of backgrounds of reviewers. We define diversity for reviewers that are part of a feasible assignment A , as a linear combination of background-, location-, and seniority-based diversity scores (each in $[0, 1]$): $Div(A) = \text{avg}_{j \in M} (Div_{BG}(j) + Div_L(j) + Div_S(j))$. Diversity can take values in $[0, 3]$, where higher values are more desirable. Note that diversity of one single reviewer set $R_A(j)$ can be computed using the same formula by setting $M = \{j\}$.

The component-level diversity scores are estimated as:

$$Div_{BG}(j) = 1 - \frac{|\sum_{i \in R_A(j)} profBG[i]|}{\lambda}$$

$$Div_L(j) = 1 - \frac{1}{\binom{2}{\lambda}} * \sum_{i,k \in R_A(j), i \neq k} \frac{|location[i] \cap location[k]|}{|location[i] \cup location[k]|}$$

$$Div_S(j) = \sum_{val \in \{0,1,2\}} \mathbb{1}(\exists i \in R_A(j) : seniority[i] == val) * \frac{1}{3},$$

where for each reviewer i , $profBG[i]$ indicates the professional background (0 if both, -1 if industry, 1 if academia), $location[i]$ denotes the distinct locations associated with i , and $seniority[i]$ describing the seniority level (0 if senior, 1 if advanced, 2 if junior).

We further quantify the dependency of an assignment as the percentage of reviewer sets with violated dependencies between reviewers i, k : $Dep(A) = \frac{\sum_{j \in M} \mathbb{1}(\exists i,k \in R_A(j) : i \neq k, dep_{ik} == 1)}{|M|} * 100$.

Example. Given: $M = \{j\}$, $R_A(j) = \{i$ (both, senior), k (academia, senior) $\}$, i and k from different locations, $dep_{ik} = 0$. We can then compute $Div(A) = (1 - \frac{1}{2}) + (1 - \frac{1}{1} * 0) + (\frac{1}{3}) = \frac{11}{6}$ and $Dep(A) = 0$.

9.6 Experiments

Recall that the output of RCP is twofold: (1) an assignment of reviewer sets to submissions and (2) a ranked list of reviewer suggestions to include in the PC. We thus divide our evaluation into two parts: an examination of reviewer assignments in §9.6.1, using both automatic (§9.6.1) and manual evaluation (§9.6.1), followed by an evaluation of reviewer suggestions using human assessments by the respective PC chairs in §9.6.2.

9.6.1 Part 1: Reviewer Assignment

For evaluating the reviewer set construction properties of our approach (conducted by our subroutine in §9.4.2) we compare different variants of our DiveRS (D_θ) algorithm against (1) assignments produced by a state-of-the-art flow-based reviewer assignment system, PR4ALL [21], and (2) the real reviewer assignments.

Automatic Evaluation

In our automatic evaluation, we report the established measures for reviewer assignment from §9.5.3, the newly introduced measures from §9.5.4, and the number of unused reviewers from the original PC.

In addition to the DiveRS default setting, we also report on a restrictive setting, where each reviewer i from the original PC who can review at least one submission (i.e., similarity $\geq \theta$) needs to be used in the final assignment ($\mu_i^l = 1$). This setting is desirable to prevent displeasing reviewers who have already been invited to the PC by not assigning them to a submission. In PR4All such an option is not given, including a lower bound for numbers of assignments is impossible as the approach merges assignment sets.

Table 9.1 reports the results of the automatic evaluation. DiveRS achieves the highest diversity scores regardless of the setting. Real assignments are worse in fairness and diversity than the automatically constructed sets. Usage of $D_{\theta>0}$ leads to fairer and mostly more diverse results compared to the $D_{\theta=0}$ variants. KL-divergence does not seem to change much between configurations, but PR4All produces sets with the highest score. Introduction of new PC members naturally reduces the mean workload per reviewer. With the restrictive DiveRS variants to include all reviewers from the original PC in the assignment (marked with *), we achieve fairness, KL, and diversity values comparable to the unrestricted variants. For unrestricted DiveRS versions, the number of unused reviewers from the original PC also lies around the value produced by PR4All. Of all methods, it is only DiveRS that pre-

Table 9.1: Reviewer assignment results for the automatic evaluation in terms of mean workload per reviewer (mW/R) and all initial PC members (/PC), number of unused initial PC members (U) as well as dependency (Dep), fairness (Γ_f^S), average textual diversity (KL), and diversity (Div) of assignments per dataset and method. Methods marked with * correspond to the restrictive setting.

method	d.set	mW/R (/PC)	U	Dep	Γ_f^S	KL	Div
real	I'19	6.16 (5.44)	5	15.38	2.12	.45	1.51
PR4All	I'19	7.09 (5.44)	10	48.72	3.51	.52	1.58
$D_{\theta=0}$	I'19	6.69 (5.09)	11	0	3.31	.46	2.16
$D_{\theta=0^*}$	I'19	5.09 (5.09)	0	0	3.07	.45	2.13
$D_{\theta>0}$	I'19	6.16 (4.78)	11	0	3.68	.45	2.15
$D_{\theta>0^*}$	I'19	4.98 (4.78)	2	0	3.68	.45	2.13
real	I'20	3.73 (3.12)	11	24.62	2.4	.45	1.57
PR4All	I'20	4.88 (2.91)	27	47.69	3.62	.52	1.55
$D_{\theta=0}$	I'20	4.53 (2.87)	25	0	3.5	.47	2.04
$D_{\theta=0^*}$	I'20	2.87 (2.87)	0	0	3.18	.47	2.05
$D_{\theta>0}$	I'20	3.16 (2.03)	32	0	4.05	.44	2.12
$D_{\theta>0^*}$	I'20	2.23 (2.13)	4	0	4.05	.44	2.09

vents the generation of assignments with dependencies between reviewers in sets.

For I'20 with $D_{\theta=.25}$ we found four papers as well as four original reviewers which were out of scope of the conference. For I'19 with $D_{\theta=.15}$ we found two original reviewers which were out of scope of the conference.

Manual Evaluation

We set up an online questionnaire where the two respective groups of PC chairs assessed the suitability of reviewer sets for ten randomly drawn submissions for their conferences. We presented them with four reviewer sets:⁵ the real assignment as well as three automatic assignments produced by PR4All, $D_{\theta=0}$, and $D_{\theta>0}$. For each assignment, PC chairs indicated the set's suitability on a four-point scale (*no reviewers are suitable*, *two reviewers need to be replaced*, *one reviewer needs to be replaced*, *suitable assignment*) and justified their decision in a free-text field.

Figure 9.2 shows the average diversity against the number of suitable reviewers, for the two datasets combined. Both $D_{\theta=0}$ and $D_{\theta>0}$ produce reviewer sets with fewer suitable reviewers than the real assignment and

⁵If sets produced from different methods are identical, we only depict it once.

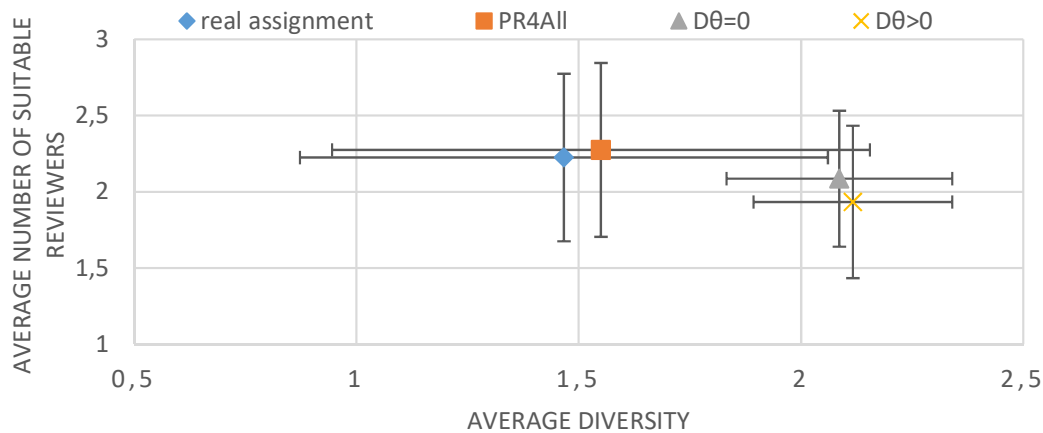


Figure 9.2: Reviewer assignment results using manual evaluation, displaying average diversity (x-axis) against the number of suitable reviewers (y-axis). The error bars correspond to the standard deviation per method. Results are reported on the two datasets combined.

PR4All—on the other hand, they produce much more diverse assignments. We observed low agreement between PC chairs when asked about the suitability of reviewer sets, as reflected in the standard deviations. It suggests that there are additional factors that may need to be considered in the reviewer assignment task; the free text comments, however, did not allow us to identify any common patterns.

Summary of Findings

DiveRS achieves fairness values which are comparable to those achieved by PR4All, without specifically focusing on this aspect of the problem. Additionally, our approach introduces more options to control reviewer load and to ensure the independence of reviewers. The resulting diversity values for DiveRS are much better than those of the real assignments or PR4All.

In our experiments, we found that there is a high probability of not assigning papers to all reviewers from the initial PC. Some members might have been included in a PC solely due to their reputation, not because of current interests or expertise in the fields of the submissions [2]. Unlike other methods, DiveRS offers the possibility of enforcing the involvement of all (fitting) PC members.

Manual evaluation showed the difficulty of objectively assessing the suitability of reviewer assignments, as we observed a high degree of disagreement between PC chairs. A comparison of diversity against the number of suitable reviewers revealed DiveRS’ tendency to sacrifice some suitability in order to

Table 9.2: Reviewer suggestion results, listing average values for relevance of explanation (r), confidence (f), usefulness (u), convincingness (c), as well as suggestion ranking (NDCG) per dataset. Usefulness and convincingness are further subdivided (in parentheses) to cases with relevance below 3 ($u_{<}$, $c_{<}$) and above 3 ($u_{>}$, $c_{>}$).

d.set	r	f	u ($u_{<}/u_{>}$)	c ($c_{<}/c_{>}$)	NDCG
I'19	2.22	4.06	2.56 (2.15/3.67)	2.06 (1.69/3)	.7967
I'20	2.65	3.65	2.2 (1.89/2.17)	2.25 (1.56/3.17)	.9105

achieve high diversity.

9.6.2 Part 2: Reviewer Suggestion

In the second part of the evaluation, we measure the quality of reviewer suggestions for inclusion in the PC; this corresponds to our main routine (§9.4.2). We consider up to ten reviewer candidates suggested by DiveRS (6 for I'19 and 10 for I'20⁶). PC chairs are given a list of reviewers that could be invited. Each candidate is presented by their name, link to their DBLP profile, their main diversity attributes (professional background, location, seniority) as well as an explanation why they would be useful for an exemplary submission (e.g., *non-academia and academia background, topically fitting*). Additionally, other submissions in which the suggested candidate could help are listed. PC chairs are then asked to rate the relevance of the suggestion, their confidence in their assessment, as well as the usefulness of the explanation and how convincing it is on a Likert scale from 1 (not at all) to 5 (very).

The PC chairs' agreement on the relevance of suggestions is low for both datasets, which leads us to believe that this task is also very difficult to evaluate. The average values for assessed quality dimensions of suggested reviewer candidates are listed in Table 9.2. In general, relevancy for suggested reviewers is low, usefulness and convincingness of explanations increase drastically if only relevant (relevancy>3) are considered. We also evaluate suggestions as a ranked list in terms of NDCG, and observe high scores, especially for I'20. This can be interpreted as our method's ability to estimate the confidence of the recommendations and rank them accordingly. Our results hint at difficulties in suggestions' quality assessment, which should be investigated further to make findings more conclusive.

⁶DiveRS introduces different numbers of reviewers based on the dataset as well as θ .

9.7 Conclusion

In this paper we introduced the novel reviewer coverage problem and proposed DiveRS, a flow-based reviewer assignment and PC member suggestion approach to solve it. DiveRS constructs diverse and fair reviewer set assignments for submissions and also suggests new reviewer candidates for inclusion in the PC. Our evaluation on two real world datasets showed DiveRS' superior diversity compared to both real assignments and the current state-of-the-art. Our experiments also highlighted the inherent difficulties of the reviewer assignment task, as evidenced by the low inter-annotator agreement between former PC chairs.

Future work could include utilising bidding information, when available, to identify papers with insufficient coverage. Requiring junior reviewers to be part of each reviewer set may be desirable at times. Also, candidate suggestions may be subjected to stricter requirements, e.g., they should be able to review multiple submissions or not be considered at all. Additionally, creating a reusable dataset for reviewer suggestion will be a challenge in itself. Finally, there are further gains to be made by employing more advanced methods for determining the similarity between reviewers and submissions.

Acknowledgements

We want to thank the conference general chairs of ICTIR'19 and ICTIR'20 for kindly providing us with the conference data, as well as the PC chairs for evaluating our automatic recommendations.

Bibliography

- [1] Ken Birman and Fred B. Schneider. Viewpoint - program committee overload in systems. Commun. ACM, 52(5):34–37, 2009.
- [2] Georgeta Bordea, Toine Bogers, and Paul Buitelaar. Benchmarking domain-specific expert search using workshop program committees. In CompSci@CIKM '13, pages 19–24, 2013.
- [3] Laurent Charlin, Richard S. Zemel, and Craig Boutilier. A framework for optimizing paper matching. In UAI '11, pages 86–95, 2011.
- [4] Don Conry, Yehuda Koren, and Naren Ramakrishnan. Recommender systems for the conference paper assignment problem. In RecSys '09, pages 357–360, 2009.
- [5] Fred Douglis. Best practices for the care and feeding of a program committee, and other thoughts on conference organization. In WOWCS '08, 2008.
- [6] Shuguang Han, Jiepu Jiang, Zhen Yue, and Daqing He. Recommending program committee candidates for academic conferences. In CompSci@CIKM '13, pages 1–6, 2013.
- [7] Steven Jecmen, Hanrui Zhang, Ryan Liu, Nihar B. Shah, Vincent Conitzer, and Fei Fang. Mitigating manipulation in peer review via randomized reviewer assignments. In NeurIPS '20, 2020.
- [8] Jian Jin, Qian Geng, Haikun Mou, and Chong Chen. Author-subject-topic model for reviewer recommendation. J. Inf. Sci., 45(4), 2019.
- [9] Noriaki Kawamae. Author interest topic model. In SIGIR '10, pages 887–888, 2010.
- [10] Ari Kobren, Barna Saha, and Andrew McCallum. Paper matching with local fairness constraints. In SIGKDD '19, pages 1247–1257, 2019.
- [11] Ngai Meng Kou, Leong Hou U, Nikos Mamoulis, and Zhiguo Gong. Weighted coverage based reviewer assignment. In SIGMOD '15, pages 2031–2046, 2015.
- [12] Michael Ley. DBLP - some lessons learned. Proc. VLDB Endow., 2(2):1493–1500, 2009.

- [13] Xiang Liu, Torsten Suel, and Nasir D. Memon. A robust model for paper reviewer assignment. In RecSys '14, pages 25–32, 2014.
- [14] Cheng Long, Raymond Chi-Wing Wong, Yu Peng, and Liangliang Ye. On good and fair paper-reviewer assignment. In ICDM '13, pages 1145–1150, 2013.
- [15] Haikun Mou, Qian Geng, Jian Jin, and Chong Chen. An author subject topic model for expert recommendation. In AIRS '15, pages 83–95, 2015.
- [16] Fabio Pacheco, Igor Wiese, Bruno Cartaxo, Igor Steinmacher, and Gustavo Pinto. Analyzing the evolution and diversity of SBES program committee. CoRR, 2020.
- [17] Simon Price and Peter A. Flach. Computational support for academic peer review: a perspective from artificial intelligence. Commun. ACM, 60(3):70–79, 2017.
- [18] Michal Rosen-Zvi, Thomas L. Griffiths, Mark Steyvers, and Padhraic Smyth. The author-topic model for authors and documents. In UAI '04, pages 487–494, 2004.
- [19] Vyas Sekar. EZ-PC: program committee selection made easy. Comput. Commun. Rev., 46(3):6:1–6:3, 2016.
- [20] Anna Severin and Joanna Chataway. Overburdening of peer reviewers. a multi-disciplinary and multi-stakeholder perspective on causes, effects and potential policy implications. bioRxiv, 2021.
- [21] Ivan Stelmakh, Nihar B. Shah, and Aarti Singh. Peerreview4all: Fair and accurate reviewer assignment in peer review. CoRR, 2018.
- [22] Wenbin Tang, Jie Tang, and Chenhao Tan. Expertise matching via constraint-based optimization. In WI '10, pages 34–41, 2010.
- [23] Yuancheng Tu, Nikhil Johri, Dan Roth, and Julia Hockenmaier. Citation author topic model in expert search. In COLING '10, pages 1265–1273, 2010.
- [24] Bogdan Vasilescu, Alexander Serebrenik, Tom Mens, Mark G. J. van den Brand, and Ekaterina Pek. How healthy are software engineering conferences? Sci. Comput. Program., 89:251–272, 2014.

- [25] Thanasis Vergoulis, Serafeim Chatzopoulos, Theodore Dalamagas, and Christos Tryfonopoulos. Veto: Expert set expansion in academia. In TPDL '20, pages 48–61, 2020.

Appendix

Curriculum Vitae (as of Jan 2022)	269
Complete List of Publications (as of Jan 2022)	271

A. Curriculum Vitae (as of Jan 2022)

Christin Katharina Kreutz

kreutzch (at) uni-trier.de • <https://kreutzch.github.io/>

Academic Position

since 11.2016 **Research Fellow/PhD Student** TRIER UNIVERSITY, Trier (DE)
chair Databases and Information Systems, supervisor Prof. Dr. Ralf Schenkel
Research areas: bibliographic metadata, scholarly recommendation, scientometrics, domain-specific query languages

Education

since 11.2016 **PhD Studies in Computer Science** TRIER UNIVERSITY, Trier (DE)

2014–2016 **MSc Computer Science** TRIER UNIVERSITY, Trier (DE)
Grade: 1.3, Immersion: Data Structures and Efficient Algorithms

2011–2015 **BSc Computer Science** TRIER UNIVERSITY, Trier (DE)
Grade: 2.4, Elective course: Business Sciences

Research Visit

3 months **with Prof. Dr. Krisztian Balog** at the Department of Electrical Engineering and Computer Science, UNIVERSITY OF STAVANGER, Stavanger (NO), Jan 2021 - Apr 2021

Academic Administration

- 2018 - 2021 PhD student representation for department IV, TRIER UNIVERSITY, Trier (DE)
- 2020 Member of search committee for a professorship (W1) on Algorithmics, TRIER UNIVERSITY, Trier (DE)
- 2018 Member of search committee for a professorship (W1) on HCI, TRIER UNIVERSITY, Trier (DE)
- 2017 - 2018 Substitutional PhD student representation for department IV, TRIER UNIVERSITY, Trier (DE)
- 2013 - 2016 Member of the student representatives in computer science, TRIER UNIVERSITY, Trier (DE)

Reviewing Activity

- 2019 Scientometrics

Grants, Awards & Honours

- 2020/2021 **DAAD IFI grant**, financial support for a three-month research visit at University of Stavanger, Stavanger (NO)
- 2020 **Best Paper Award**, ICADL 2020
- 2019 **Winning Group**, Wein-Mosel-Hackathon 2019
- 2017 **Best Poster Award**, FDIA@ESSIR 2017
- 2017 **PROMOS travel grant**, financial support for the participation at ESSIR 2017

B. Complete List of Publications (as of Jan 2022)

Journal Papers

- *RevASIDE: Evaluation of Assignments of Suitable Reviewer Sets for Publications from Fixed Candidate Pools*
Christin Katharina Kreutz, Ralf Schenkel
Journal of Data Intelligence (to appear)
- *SchenQL: in-depth analysis of a query language for bibliographic meta-data*
Christin Katharina Kreutz, Michael Wolz, Jascha Knack, Benjamin Weyers, Ralf Schenkel
International Journal of Digital Libraries 23(2): 113-132 (2022)
Link: <https://doi.org/10.1007/s00799-021-00317-8>
- *Evaluating semantometrics from computer science publications*
Christin Katharina Kreutz, Premtim Sahitaj, Ralf Schenkel
Scientometrics 125(3): 2915-2954 (2020)
Link: <https://doi.org/10.1007/s11192-020-03409-5>

Conference Papers

- *Diverse Reviewer Suggestion for Extending Conference Program Committees*
Christin Katharina Kreutz, Krisztian Balog, Ralf Schenkel
WI-IAT 2021: 79–86
Link: <https://doi.org/10.1145/3486622.3493931>
- *RevASIDE: Assignment of Suitable Reviewer Sets for Publications from Fixed Candidate Pools*
Christin Katharina Kreutz, Ralf Schenkel
iiWAS 2021: 57–68
Link: <https://doi.org/10.1145/3487664.3487673>

- *SchenQL: Evaluation of a Query Language for Bibliographic Metadata*
Christin Katharina Kreutz, Michael Wolz, Benjamin Weyers, Ralf Schenkel
ICADL 2020: 323-339
Link: https://doi.org/10.1007/978-3-030-64452-9_30
- *Segmenting and Clustering Noisy Arguments*
Lorik Dumani, Christin Katharina Kreutz, Manuel Biertz, Alex Witry, Ralf Schenkel
LWDA 2020: 23-34
Link: http://ceur-ws.org/Vol-2738/LWDA2020_paper_24.pdf
- *SchenQL: A Concept of a Domain-Specific Query Language on Bibliographic Metadata*
Christin Katharina Kreutz, Michael Wolz, Ralf Schenkel
ICADL 2019: 239-246
Link: https://doi.org/10.1007/978-3-030-34058-2_22
- *Reevaluating Semantometrics from Computer Science Publications*
Christin Katharina Kreutz, Premtim Sahitaj, Ralf Schenkel
BIRNDL@SIGIR 2019: 42-55
Link: <http://ceur-ws.org/Vol-2414/paper5.pdf>
- *FacetSearch: A Faceted Information Search and Exploration Prototype*
Christin Katharina Kreutz, Peter Boesten, Alex Witry, Ralf Schenkel
LWDA 2018: 215-226
Link: <http://ceur-ws.org/Vol-2191/paper26.pdf>
- *A Hybrid Approach for Dynamic Topic Models with Fluctuating Number of Topics*
Christin Katharina Kreutz
Grundlagen von Datenbanken 2018: 35-40
Link: <http://ceur-ws.org/Vol-2126/paper5.pdf>

Extended Abstract

- *Trend Mining on Bibliographic Data*
Christin Katharina Kreutz
FDIA 2017
Link: <https://doi.org/10.14236/ewic/FDIA2017.11>

Preprints

- *Diverse Reviewer Suggestion for Extending Conference Program Committees*
Christin Katharina Kreutz, Krisztian Balog, Ralf Schenkel
Link: <https://arxiv.org/pdf/2201.11030.pdf>
- *Scientific Paper Recommendation Systems: a Literature Review of recent Publications*
Christin Katharina Kreutz, Ralf Schenkel
Link: <https://arxiv.org/pdf/2201.00682.pdf>
- *RevASIDE: Assignment of Suitable Reviewer Sets for Publications from Fixed Candidate Pools*
Christin Katharina Kreutz, Ralf Schenkel
Link: <https://arxiv.org/pdf/2110.02862.pdf>
- *SchenQL - A Domain-Specific Query Language on Bibliographic Metadata*
Christin Katharina Kreutz, Michael Wolz, Ralf Schenkel
Link: <https://arxiv.org/pdf/1906.06132.pdf>