Cross-Domain Information Extraction from Scientific Articles for Research Knowledge Graphs

Von der Fakultät für Elektrotechnik und Informatik

der Gottfried Wilhelm Leibniz Universität Hannover

zur Erlangung des akademischen Grades

Doktor der Ingenieurwissenschaften

Dr.-Ing.

genehmigte Dissertation von Herrn

M. Eng. Arthur Brack

2022

Hauptreferent:	Prof. Dr. Ralph Ewerth
Korreferent:	Prof. Dr. Kurt Schneider
Vorsitzender:	Prof. Dr.-Ing. Markus Fidler
Tag der Promotion:	19. Mai 2022

Abstract

Today's scholarly communication is a document-centred process and as such, rather inefficient. Fundamental contents of research papers are not accessible by computers since they are only present in unstructured PDF files. Therefore, current research infrastructures are not able to assist scientists appropriately in their core research tasks. This thesis addresses this issue and proposes methods to automatically extract relevant information from scientific articles for Research Knowledge Graphs (RKGs) that represent scholarly knowledge structured and interlinked.

First, this thesis conducts a requirements analysis for an Open Research Knowledge Graph (ORKG). We present literature-related use cases of researchers that should be supported by an ORKG-based system and their specific requirements for the underlying ontology and instance data. Based on this analysis, the identified use cases are categorised into two groups: The first group of use cases needs manual or semi-automatic approaches for knowledge graph (KG) construction since they require high correctness of the instance data. The second group requires high completeness and can tolerate noisy instance data. Thus, this group needs automatic approaches for KG population. This thesis focuses on the second group of use cases and provides contributions for machine learning tasks that aim to support them.

To assess the relevance of a research paper, scientists usually skim through titles, abstracts, introductions, and conclusions. An organised presentation of the articles' essential information would make this process more time-efficient. The task of *sequential sentence classification* addresses this issue by classifying sentences in an article in categories like *research problem*, *used methods*, or *obtained results*. To address this problem, we propose a novel unified cross-domain multi-task deep learning approach that makes use of datasets from different scientific domains (e.g. biomedicine and computer graphics) and varying structures (e.g. datasets covering either only abstracts or full papers). Our approach outperforms the state of the art on full paper datasets significantly while being competitive for datasets consisting of abstracts. Moreover, our approach enables the categorisation of sentences in a domain-independent manner.

Furthermore, we present the novel task of *domain-independent information extraction* to extract scientific concepts from research papers in a domain-independent manner. This task aims to support the use cases *find related work* and *get recommended articles*. For this purpose, we introduce a set of generic scientific concepts that are relevant over ten domains in

Science, Technology, and Medicine (STM) and release an annotated dataset of 110 abstracts from these domains. Since the annotation of scientific text is costly, we suggest an active learning strategy based on a state-of-the-art deep learning approach. The proposed method enables us to nearly halve the amount of required training data.

Then, we extend this domain-independent information extraction approach with the task of *coreference resolution*. Coreference resolution aims to identify mentions that refer to the same concept or entity. Baseline results on our corpus with current state-of-the-art approaches for coreference resolution showed that current approaches perform poorly on scientific text. Therefore, we propose a sequential transfer learning approach that exploits annotated datasets from non-academic domains. Our experimental results demonstrate that our approach noticeably outperforms the state-of-the-art baselines.

Additionally, we investigate the impact of coreference resolution on KG population. We demonstrate that coreference resolution has a small impact on the number of resulting concepts in the KG, but improved its quality significantly. Consequently, using our domain-independent information extraction approach, we populate an RKG from 55,485 abstracts of the ten investigated STM domains. We show that every domain mainly uses its own terminology and that the populated RKG contains useful concepts.

Moreover, we propose a novel approach for the task of *citation recommendation*. This task can help researchers improve the quality of their work by finding or recommending relevant related work. Our approach exploits RKGs that interlink research papers based on mentioned scientific concepts. Using our automatically populated RKG, we demonstrate that the combination of information from RKGs with existing state-of-the-art approaches is beneficial. Finally, we conclude the thesis and sketch possible directions of future work.

Keywords: scholarly communication, research knowledge graph, information extraction, deep learning, natural language processing, requirements analysis, sequential sentence classification, scientific concept extraction, coreference resolution, citation recommendation

Zusammenfassung

Die Kommunikation von Forschungsergebnissen erfolgt heutzutage in Form von Dokumenten und ist aus verschiedenen Gründen ineffizient. Wesentliche Inhalte von Forschungsarbeiten sind für Computer nicht zugänglich, da sie in unstrukturierten PDF-Dateien verborgen sind. Daher können derzeitige Forschungsinfrastrukturen Forschende bei ihren Kernaufgaben nicht angemessen unterstützen. Diese Arbeit befasst sich mit dieser Problemstellung und untersucht Methoden zur automatischen Extraktion von relevanten Informationen aus Forschungspapieren für Forschungswissensgraphen (Research Knowledge Graphs). Solche Graphen sollen wissenschaftliches Wissen maschinenlesbar strukturieren und verknüpfen.

Zunächst wird eine Anforderungsanalyse für einen Open Research Knowledge Graph (ORKG) durchgeführt. Wir stellen literaturbezogene Anwendungsfälle von Forschenden vor, die durch ein ORKG-basiertes System unterstützt werden sollten, und deren spezifische Anforderungen an die zugrundeliegende Ontologie und die Instanzdaten. Darauf aufbauend werden die identifizierten Anwendungsfälle in zwei Gruppen eingeteilt: Die erste Gruppe von Anwendungsfällen benötigt manuelle oder halbautomatische Ansätze für die Konstruktion eines ORKG, da sie eine hohe Korrektheit der Instanzdaten erfordern. Die zweite Gruppe benötigt eine hohe Vollständigkeit der Instanzdaten und kann fehlerhafte Daten tolerieren. Daher erfordert diese Gruppe automatische Ansätze für die Konstruktion des ORKG. Diese Arbeit fokussiert sich auf die zweite Gruppe von Anwendungsfällen und schlägt Methoden für maschinelle Aufgabenstellungen vor, die diese Anwendungsfälle unterstützen können.

Um die Relevanz eines Forschungsartikels effizient beurteilen zu können, schauen sich Forschende in der Regel die Titel, Zusammenfassungen, Einleitungen und Schlussfolgerungen an. Durch eine strukturierte Darstellung von wesentlichen Informationen des Artikels könnte dieser Prozess zeitsparender gestaltet werden. Die Aufgabenstellung der *sequenziellen Satzklassifikation* befasst sich mit diesem Problem, indem Sätze eines Artikels in Kategorien wie *Forschungsproblem*, *verwendete Methoden* oder *erzielte Ergebnisse* automatisch klassifiziert werden. In dieser Arbeit wird für diese Aufgabenstellung ein neuer vereinheitlichter Multi-Task Deep-Learning-Ansatz vorgeschlagen, der Datensätze aus verschiedenen wissenschaftlichen Bereichen (z. B. Biomedizin und Computergrafik) mit unterschiedlichen Strukturen (z. B. Datensätze bestehend aus Zusammenfassungen oder vollständigen Artikeln) nutzt. Unser Ansatz übertrifft State-of-the-Art-Verfahren der Literatur auf Benchmark-Datensätzen bestehend aus vollständigen Forschungsartikeln. Außerdem ermöglicht unser Ansatz die Klassifizierung von Sätzen auf eine domänenunabhängige Weise.

Darüber hinaus stellen wir die neue Aufgabenstellung *domänenübergreifende Informationsextraktion* vor. Hierbei werden, unabhängig vom behandelten wissenschaftlichen Fachgebiet, inhaltliche Konzepte aus Forschungspapieren extrahiert. Damit sollen die Anwendungsfälle *Finden von verwandten Arbeiten* und *Empfehlung von Artikeln* unterstützt werden. Zu diesem Zweck führen wir eine Reihe von generischen wissenschaftlichen Konzepten ein, die in zehn Bereichen der Wissenschaft, Technologie und Medizin (STM) relevant sind, und veröffentlichen einen annotierten Datensatz von 110 Zusammenfassungen aus diesen Bereichen. Da die Annotation wissenschaftlicher Texte aufwändig ist, kombinieren wir ein Active-Learning-Verfahren mit einem aktuellen Deep-Learning-Ansatz, um die notwendigen Trainingsdaten zu reduzieren. Die vorgeschlagene Methode ermöglicht es uns, die Menge der erforderlichen Trainingsdaten nahezu zu halbieren.

Anschließend erweitern wir unseren domänenunabhängigen Ansatz zur Informationsextraktion um die Aufgabe der *Koreferenzauflösung*. Die Auflösung von Koreferenzen zielt darauf ab, Erwähnungen zu identifizieren, die sich auf dasselbe Konzept oder dieselbe Entität beziehen. Experimentelle Ergebnisse auf unserem Korpus mit aktuellen Ansätzen zur Koreferenzauflösung haben gezeigt, dass diese bei wissenschaftlichen Texten unzureichend abschneiden. Daher schlagen wir eine Transfer-Learning-Methode vor, die annotierte Datensätze aus nicht-akademischen Bereichen nutzt. Die experimentellen Ergebnisse zeigen, dass unser Ansatz deutlich besser abschneidet als die bisherigen Ansätze.

Darüber hinaus untersuchen wir den Einfluss der Koreferenzauflösung auf die Erstellung von Wissensgraphen. Wir zeigen, dass diese einen geringen Einfluss auf die Anzahl der resultierenden Konzepte in dem Wissensgraphen hat, aber die Qualität des Wissensgraphen deutlich verbessert. Mithilfe unseres domänenunabhängigen Ansatzes zur Informationsextraktion haben wir aus 55.485 Zusammenfassungen der zehn untersuchten STM-Domänen einen Forschungswissensgraphen erstellt. Unsere Analyse zeigt, dass jede Domäne hauptsächlich ihre eigene Terminologie verwendet und dass der erstellte Wissensgraph nützliche Konzepte enthält.

Schließlich schlagen wir einen Ansatz für die Empfehlung von passenden Referenzen vor. Damit können Forschende einfacher relevante verwandte Arbeiten finden oder passende Empfehlungen erhalten. Unser Ansatz nutzt Forschungswissensgraphen, die Forschungsarbeiten mit in ihnen erwähnten wissenschaftlichen Konzepten verknüpfen. Wir zeigen, dass aktuelle Verfahren zur Empfehlung von Referenzen von zusätzlichen Informationen aus einem automatisch erstellten Wissensgraphen profitieren. Zum Schluss wird ein Fazit gezogen und ein Ausblick für mögliche zukünftige Arbeiten gegeben.

Stichworte: Forschungswissensgraph, Informationsextraktion, Deep Learning, Computerlinguistik, Anforderungsanalyse, Sequenzielle Satzklassifikation, Extraktion wissenschaftlicher Konzepte, Auflösung von Koreferenzen, Empfehlung von Referenzen

Acknowledgments

During my PhD study and the writing of this thesis, I have received a lot of support from various people and institutions, without which this work would not have been possible.

First and foremost, I would like to thank my supervisor, Prof. Dr. Ralph Ewerth, who made it possible to do my PhD study and supported me all the time. You put a lot of trust in me and coached me constantly to improve my research and writing skills. I would also like to express my sincere thanks to my co-supervisor Dr. Anett Hoppe. Without your help, it would have been tough to find my way through the research jungle. Furthermore, I thank Prof. Dr. Kurt Schneider and Prof. Dr.-Ing. Markus Fidler for their time and effort to examine this thesis. My special thanks go to my employer, SET GmbH in Hannover, especially the founder Till Dammermann, who gave me the freedom to do a PhD study. It would not have been possible without your support and trust. Moreover, I thank my colleagues at SET GmbH who watched my back during my PhD study, particularly Dr.-Ing. Tobias Baum who also gave me valuable feedback on this thesis.

I would also like to thank Prof. Dr. Sören Auer, who, as the director of the TIB and the founder of the ORKG, made it possible for me to work on this exciting project. Special thanks go to the entire ORKG team for the great collaboration, especially to Dr. Jennifer D'Souza, Dr. Markus Stocker, Dr. Oliver Karras, and Mohamad Yaser Jaradeh for their valuable contributions. In particular, Jennifer helped me with my first research paper. Furthermore, I thank Prof. Dr. Harald König from my former university, the FHDW Hannover. You encouraged me to start a PhD study and gave me valuable feedback on this thesis.

In my Visual Analytics research group, I had the opportunity to work with many great colleagues who were always helpful and gave me lots of valuable tips. Thank you very much for that. Special thanks go to my colleagues Dr. Sherzod Hakimov, Dr.-Ing. Eric Müller-Budack, and Matthias Springstein, and the students Daniel Uwe Müller and Pascal Buschermöhle, of whom I supervised the bachelor thesis, for their valuable contributions.

Finally, my wholeheartedly thanks go to my family. My beloved wife Kira, mother Alexandra, sister Jeanna, and my dear daughters Talissa, Lara, and Lina have always been great emotional support. It is only because of you that the PhD makes any sense at all. My father would certainly have been proud. This thesis is dedicated to you, my family.

To my family ...

Contents

List of Tables

List of Figures

Acronyms

ADAM Adaptive Moment Estimation. 25

API Application Programming Interface. 75

BALD Bayesian Active Learning by Disagreement. 111

BCE Binary Cross-Entropy. 23, 24

BERT Bidirectional Encoder Representations from Transformers. 37, 38, 53, 110, 125, 139

BFCR BERT for Coreference Resolution. 125, 128

Bi-GRU Bidirectional Gated Recurrent Unit. 83

Bi-LSTM Bidirectional Long Short-Term Memory. 82, 87, 110

Bi-RNN Bidirectional Recurrent Neural Network. 29

CE Cross-Entropy. 24

CheBi Chemical Entities of Biological Interest. 3, 57, 58

CNN Convolutional Neural Network. 82, 110

CRAFT Colorado Richly Annotated Full Text. 122

CRediT Contributor Roles Taxonomy. 13

CRF Conditional Random Field. 30, 83, 88, 110

CRMsci Scientific Observation Model. 58

CSO Computer Science Ontology. 57, 58

DOI Document Object Identifier. 3, 41, 42

DQ Data Quality. 63

DSKG Data Set Knowledge Graph. 57

DSR Design Science Research. 65, 151

ELMo Embeddings from Language Models. 37, 128

EXPO EXPeriments Ontology. 58

GloVe Global Vectors for Word Representation. 82, 139

GPT Generative Pre-training. 37

GRU Gated Recurrent Unit. 30

HSLN Hierarchical Sequential Labeling Network. 87

IRI International Resource Identifier. 41, 42

JSON-LD JavaScript Object Notation for Linked Data. 40

KG Knowledge Graph. 2, 40, 56, 103, 119

LSTM Long Short-Term Memory. 30

MAP Mean Average Precision. 51

MeSH Medical Subject Heading. 58

MNLP Maximum Normalized Log-Probability. 106, 111

NDCG Normalized Discounted Cumulative Gain. 51

NELL Never-Ending Language Learner. 63

NER Named Entity Recognition. 29, 45, 103, 106

NLP Natural Language Processing. 6, 10, 12, 18, 19, 53, 106

ORCID Open Researcher and Contributor ID. 41

ORKG Open Research Knowledge Graph. 2, 54, 76

OWL Web Ontology Language. 40, 42, 43

PCA Principal Component Analysis. 97, 99

PDF Portable Document Format. 1

PhySH Physics Subject Headings. 58

PICO Population, Intervention, Comparison, Outcome. 70

RDF Resource Description Framework. 40, 41

RDFS RDF Schema. 40, 42

ReLU Rectified Linear Unit. 20, 21

RKG Research Knowledge Graph. 1, 7, 19, 39, 53, 135, 143

RNN Recurrent Neural Network. 19, 26, 83

RQ Research Question. 5

S2ORC Semantic Scholar Open Research Corpus. 42, 56, 106

Scholix Scholarly Link Exchange. 57

SCIIE Scientific Information Extractor. 128

SHACL Shapes Constraint Language. 43

SPARQL SPARQL Protocol and RDF Query Language. 40

SPECTER Scientific Paper Embeddings using Citationinformed TransformERs. 137, 139

STM Science, Technology, and Medicine. 8, 10, 11, 104, 107, 117, 120, 133, 145

SVM Support Vector Machine. 83

ULMFiT Universal Language Model Fine-tuning. 37

UML Unified Modelling Language. 43, 66, 75

UMLS Unified Medical Language System. 57, 106

W3C World Wide Web Consortium. 40

XML Extensible Markup Language. 40

Notations

This chapter describes the notation used in this thesis that is based on the notation of Goodfellow et al. [101].

Scalars, Arrays, and Sets

a	A scalar (integer or real)
$\mathbf{a}$	A vector
$\mathbf{A}$	A matrix or tensor
$\mathbb{A}$	A set
$\lvert\mathbb{A}\rvert$	The number of items in set $\mathbb{A}$
$\mathbb{R}$	The set of real numbers
$\mathcal{G}$	A graph

Indexing and Operations

$\mathbf{a}^\intercal$ or $\mathbf{A}^\intercal$	Transpose of vector $\mathbf{a}$ or matrix $\mathbf{A}$
$\mathbf{a}_i$	Element with index i of vector $\mathbf{a}$
$\mathbf{A}_{i,j}$	Element in row i and column j of matrix $\mathbf{A}$
$[\mathbf{a_1}, ..., \mathbf{a_n}]$	Vertical concatenation of the vectors $\mathbf{a_1}, ..., \mathbf{a_n}$
$\mathbb{1}(p)$	Indicator function returning 1 if the predicate p is true and 0 otherwise
$\arg\min_{\mathbf{x}} f(\mathbf{x})$	Yields the vector $\mathbf{x}$ for which $f(\mathbf{x})$ attains a minimum
$\frac{\partial f(\mathbf{x})}{\partial \mathbf{x}_i}$	Partial derivative of f with respect to $\mathbf{x}_i$
$exp(x)$	returns e^x where e is the Euler's number

Datasets

$\mathbb{X}$	A set of training examples
$\mathbf{x}^{(i)}$	Input vector $\mathbf{x}^{(i)}$ of the i-th example
$\mathbf{y}^{(i)}$ or $y^{(i)}$	Expected vector $\mathbf{y}^{(i)}$ or scalar $y^{(i)}$ of the i-th example
$\hat{\mathbf{y}}^{(i)}$ or $\hat{y}^{(i)}$	Predicted vector $\hat{\mathbf{y}}^{(i)}$ or scalar $\hat{y}^{(i)}$ of the i-th example
$(\mathbf{x_1}, ..., \mathbf{x}_{\tau_\mathbf{x}})$	Sequence with length τ_x of input vectors of an example
$(\mathbf{y_1}, ..., \mathbf{y}_{\tau_\mathbf{y}})$	Sequence with length τ_y of expected vectors of an example
$(\hat{\mathbf{y}}_\mathbf{1}, ..., \hat{\mathbf{y}}_{\tau_\mathbf{y}})$	Sequence with length τ_y of predicted vectors of an example

1 Introduction

Research is an essential part of our society and is contributing significantly to innovation, progress, and new knowledge. Accordingly, in 2018, countries and companies invested over 2.23 trillion U.S. dollars globally into research and development. This compares to around one trillion U.S. dollars in 2005 and about 555 billion U.S. dollars in 1996 [269]. Consequently, the number of scientific publications has increased by 4% annually over the last decade leading to over 2.5 million papers published alone in 2018 [27, 188].

As stated by Auer et al. [8], in recent decades, modern information technologies have digitalised various industries. For instance, product catalogues were replaced by online shops and road maps by navigation systems. However, the way how research results are communicated in the research community has hardly changed over the past centuries [8]: In the 17th century, first research papers were published in paper form [208]; today, researchers communicate research results still through papers, but now electronically as files in Portable Document Format (PDF). As a result, the essential contents of research papers, such as research problems, applied or proposed methods, results, and contributions, are not directly accessible, i.e. machine-readable, by computers since they are "hidden" in unstructured text, tables, and figures. Therefore, the degree of digitalisation in scholarly communication corresponds to road maps as PDF files, and we are still on the lookout for an equivalent of nowadays navigation systems for research literature [8]. As a consequence, current research infrastructures such as academic search engines cannot assist researchers adequately, leading to rather inefficient scholarly communication. The explosion in the number of published articles [27, 188] aggravates this situation further: It becomes harder and harder to stay on top of current research, that is, to find relevant works, read, compare, and reproduce them and, later on, to make one's own contribution known for its quality.

> This thesis addresses the problem of scholarly communication by exploring automatic methods for *information extraction from scientific texts*. As depicted in Figure 1.1, the main objective of this thesis is to extract relevant information from research papers with machine learning approaches for Research Knowledge Graphs (RKGs).

The idea of RKGs is to represent scholarly knowledge in a structured and interlinked way. Therefore, they have the potential to reduce some of the current issues in scholarly communication so that relevant research could be easier to find, research field overviews automatically compiled, and own insights could be placed rapidly in the current ecosystem (see

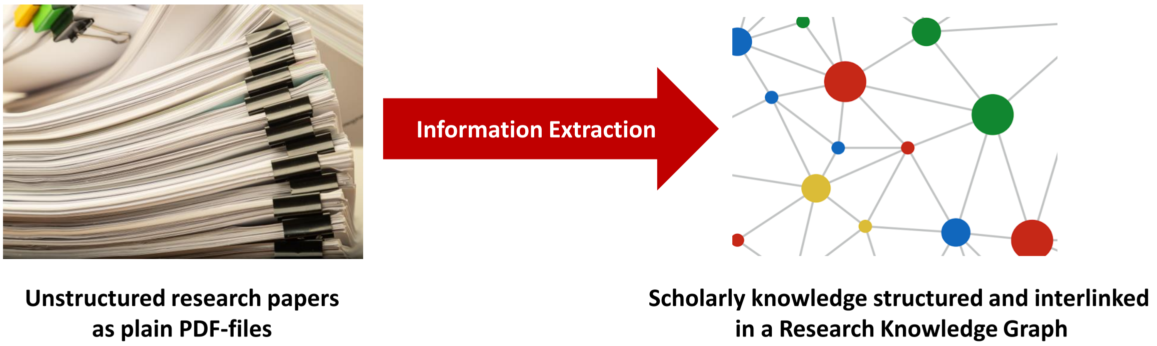

Information Extraction
Unstructured research papers as plain PDF-files
Scholarly knowledge structured and interlinked in a Research Knowledge Graph

Figure 1.1: The goal of this thesis is to extract information from unstructured research papers for Research Knowledge Graphs (RKGs) that represent and interlink scholarly knowledge in a structured manner. Image source: iStock.com

Chapter 3 for further details). The TIB – Leibniz Information Centre for Science and Technology in Hannover is currently developing an Open Research Knowledge Graph (ORKG) (`www.orkg.org`) and aims to publish the content online for the public good [130]. Moreover, the ORKG is not just an example of an RKG but is also a system that aims to offer applications for various use cases (e.g. comparative research field overviews). This thesis is also a part of this research and development project. However, this thesis refers mainly to the general idea of an ORKG [7] and not its concrete implementation at TIB.

In the following, Section 1.1 provides an overview of related work and research gaps that are the basis to formulate the addressed challenges and research questions which are described in the subsequent Section 1.2. Then, Section 1.3 summarises the contributions for the aforementioned research questions. Finally, Section 1.4 describes the overall structure of this thesis and Section 1.5 enumerates the papers that have been published in the context of this thesis.

1.1 Background

This section provides selected related work that serves as the basis for the research challenges and questions in the subsequent Section 1.2. In particular, we describe the applications of Knowledge Graphs (KGs) in general and in the research ecosystem, ontologies that conceptualise scholarly knowledge, and the construction of KGs in the scientific domain with machine learning approaches. The subsequent chapters, especially the requirements analysis in Chapter 3, provide a more comprehensive overview of related work for the respective topics.

Applications of Knowledge Graphs: A Knowledge Graph (KG) represents entities of interest and their relationships as a labelled graph of nodes and edges [117]. The instance data usually conform to a data model (e.g. ontology, schema, vocabulary), enabling machines to understand and reason over the represented knowledge (see Section 2.4 for more details). As stated by Hogan et al. [117], various knowledge-based applications use KGs as their backbone, for instance, in Web search (e.g. Google [261]), E-commerce (e.g. Amazon [158]), or social networks (e.g. LinkedIn [114]). The initiative `www.schema.org` creates and maintains schemas for unstructured data on the Internet to help search engines interpret the published content on websites. Moreover, KGs are used to structure encyclopedic knowledge (e.g. DBpedia [172], YAGO [275], Wikidata [285]), which are utilised in various applications such as semantic search engines [14], IBM Watson for answering questions in natural language [83], Babelfy [195] for natural language understanding, or academic search engines [294].

Knowledge Graphs in the Research Ecosystem: Some of the available infrastructures in the research ecosystem also use KGs to enhance their services. Academic search engines (e.g. `www.semanticscholar.org`), utilise metadata-based graph structures, such as the Literature Graph [3] or the Microsoft Academic Knowledge Graph [78], which link research articles based on citations, shared authors, venues, and keywords.

Recently, initiatives have promoted the usage of KGs in science communication, but on a deeper, semantic level [7, 110, 130, 184, 203, 215, 301]. They envision the transformation of the dominant document-centred knowledge exchange to knowledge-based information flows by representing and expressing knowledge through semantically rich, interlinked KGs. Indeed, they argue that a shared structured representation of scientific knowledge has the potential to alleviate some of science communication's current issues, such that related work could be easier to find and own contributions rapidly placed in the research ecosystem. Furthermore, such a powerful data structure could also encourage the interconnection of research artefacts such as datasets and source code much more than current approaches (e.g. Document Object Identifier (DOI)), allowing easier reproducibility and comparison.

Current approaches towards knowledge-based information flows enrich and interconnect research papers through machine-interpretable semantic content. For instance, initiatives such as Research Graph [5], Research Objects [17], and OpenAIRE [184] interlink research articles with artefacts such as datasets, source code, software, and video presentations. On a more semantic level, Papers With Code [206] is a community-driven effort to supplement machine learning papers with tasks, source code, and evaluation results to enable automatic construction of leaderboards for various benchmarks. Wikidata [285] complements articles in Wikipedia [176] with structured and interlinked content. The Gene Ontology Consortium [59] is a collaborative effort to develop a comprehensive, computational model of biological systems (e.g. genes, cells, molecules, biological processes), and the Chemical Entities of Biological Interest (CheBi) [65] initiative describes molecular entities of chemical compounds semantically. Although enrichment of research papers with semantic content is associated with additional efforts, Jaradeh et al.'s study [130] and the usage of Papers With Code [206] in the machine learning community indicate that authors are willing to contribute structured descriptions of their research articles.

Scientific Ontologies: However, it is far from being apparent *how* and *what* to provide in structured descriptions of research papers. Various ontologies have been developed to conceptualise scholarly knowledge. Some ontologies are more domain-independent, describing scholarly knowledge with terms like problem, method, activity, etc. [36, 110, 215], or focus on the primary research findings of papers [81, 281]. Other ontologies are more domain-specific, for instance, for mathematics [165] (e.g. definitions, assertions, proofs), machine learning [152, 191] (e.g. dataset, metric, model, experiment), physics [247] (e.g. formation, model, observation), or scientific experiments [264] (e.g. experiments, methods, results). This diversity in ontologies indicates that it is difficult to conceptualise scholarly knowledge comprehensively.

Construction of Knowledge Graphs: The construction of a KG involves the design of an ontology and the population with instance data using manual or automatic approaches (see Section 2.4.2 for further details). Several KGs were populated manually, e.g. Wikidata [285], YAGO [275], Papers With Code [206], Gene Ontology [97], and Chemical Entities of Biological Interest [65]. Manually populated KGs usually have high correctness (i.e. the degree to which the encoded information is correct), e.g. YAGO was found to be 95% correct [275]. However, the completeness (i.e. the degree to which all required information is present) is quite low. For instance, 69%-99% of entities in popular KGs such as YAGO [275] or DBpedia [172] do not have at least one property that other entities of the same class have [94, 274].

For automatic information extraction from scientific text with machine learning approaches, various datasets were annotated at the sentence or phrase level. Sentence level annotations [54, 68, 85, 147], enable the classificaton of sentences in categories like *objective*, *methods*, or *results*. Phrase level annotations [9, 90, 178, 229], enable (a) the recognition of scientific entities or concepts like *task*, *material*, or *method*, (b) their binary relations [9, 92, 178] such as *used-for*, *part-of*, or *evaluate-for*, or (c) n-ary relations [129, 131, 142] such as *drug-gene-mutation* interactions or *task-dataset-metric-score* tuples. Datasets for coreference resolution [58, 178] are used to identify mentions in text referring to the same entity or object. These datasets were usually annotated by domain experts and target specific domains, e.g. material sciences [90], computational linguistics [92, 229], computer science, material sciences, and physics [9], machine learning [178], or biomedicine [58, 131, 156].

Depending on the task's difficulty, machine learning approaches achieve different results. For instance, for sentence classification in abstracts from the biomedical domain, machine learning approaches achieve a high F1 score of over 93.0% [54]. However, for the extraction of *task-dataset-metric-score* tuples from machine learning papers, the best approach performs poorly with an F1 of only 28.7% [142]. The inter-coder agreement scores for these datasets range from 0.6 to 0.9 [9, 92, 178, 229] (in terms of Cohen's Kappa (κ) [57] or F1), indicating that these tasks are not only difficult for machines but also for humans. On the other hand, the automatically populated AI-KG from the artificial intelligence (AI) domain has comparatively a high estimated recall of 81.2% [70]. Thus, automatic approaches can help to achieve high completeness of KGs.

1.2 Challenges and Research Questions

As outlined in the previous section, representing scholarly knowledge via KGs is an active area of research. This section outlines several challenges and research questions with regard to RKG construction that are addressed in this thesis.

1.2.1 Diversity and Heterogeneity of Scholarly Knowledge

To represent scholarly knowledge structured and interlinked, we desire an RKG with a (a) domain-specific and (b) fine-grained ontology, as well as with instance data of (c) high completeness and (d) correctness. However, these *data quality requirements* are challenging and also conflicting. Scholarly knowledge is very heterogeneous and diverse, so it is quite impossible to conceptualise it comprehensively within an ontology. This claim is supported by the existence of numerous different scientific ontologies [36, 81, 110, 152, 191, 215, 247, 264]. Besides, the population of scientific ontologies is difficult and time-consuming [9, 92]. This is also indicated by the wide range of inter-coder agreement scores (0.6 to 0.9 in terms of Cohen's Kappa or F1) for scientific datasets [9, 92, 178, 229]. Thus, the population of complex ontologies with instance data requires domain and ontology experts. However, this is rather time-consuming and, therefore, not feasible to achieve high completeness, especially with the flood of new publications [27, 188]. Moreover, various studies have also shown that manually curated KGs have a low completeness [25, 75, 94, 274]. Current automatic approaches that could achieve high completeness can only populate relatively simple ontologies with rather low correctness (see Section 3.2.3.2). Thus, we are faced with the following problem statement:

On the one hand, we desire an ontology that can comprehensively capture scholarly knowledge and instance data with high correctness and completeness. On the other hand, we are faced with a "knowledge acquisition bottleneck".

Data quality is defined as *"fitness for use"* by a data consumer [289]. Therefore, to resolve the problem statement, the above requirements should be illuminated in the context of specific use cases that an RKG aims to support. This raises our first research question (RQ):

RQ1: *What are the main requirements for scholarly knowledge representation to support various use cases in an RKG?*

This research question also implies that we need to (a) identify use cases of researchers that should be supported by an RKG, (b) define data quality requirements for the underlying ontologies and instance data of the individual use cases, and (c) propose strategies (e.g. manual, automatic, semi-automatic) and approaches on how to construct such an RKG.

1.2.2 Lack of Labelled Data and Domain Experts

KGs such as DBpedia [172], YAGO [275], or Wikidata [285] were usually populated by human curators or from structured resources. Thus, these KGs have quite high correctness but rather low completeness [25, 75, 94, 274]. Furthermore, the introduction of new scientific concepts occurs at a faster pace than KG curation, resulting in a large gap in KG completeness of scientific concepts [3]. For instance, at the time of this writing, the task of *visual event recognition in videos* [48] from the computer vision field is neither present in Wikidata [285] nor in more specialised platforms like Papers With Code [206].

On the other hand, deep learning approaches have achieved astonishing results in computer vision [111] and Natural Language Processing (NLP) [113] benchmarks and could even surpass human-level performance on some tasks. Thus, deep learning approaches have the potential to populate an RKG automatically with high completeness and correctness. However, these powerful models were trained on large datasets from general (non-academic) domains (e.g. news, Wikipedia, magazines, etc.) and are not directly applicable to scientific text. Furthermore, the annotation process of scientific datasets is much more challenging and expensive than for the general domain counterpart [9, 92]: Understanding a research paper and determining its most essential statements demands certain expertise in the article's domain. Moreover, every domain is characterised by its specific terminology and phrasing, which is hard to grasp for a non-expert reader. Thus, the manual annotation of comprehensive and large datasets for different scientific domains is practically not feasible. This raises the RQ:

RQ2: *How can we modify machine learning methods for information extraction from scientific texts to be adaptable to new domains with few labelled data?*

This research question thus aims to reduce the efforts involved in annotating datasets to enable still automatic information extraction from scientific text with high correctness.

Furthermore, the population of an RKG from scientific text usually entails the involvement of domain experts to develop and design a specific extraction methodology and the annotation of datasets – and this for each scientific discipline. These are rather time-consuming and costly requirements. Existing datasets for information extraction from scientific text cover only a few scientific disciplines (see Tables 3.1, 3.2, and 3.3). This raises our next RQ:

> **RQ3:** *How can we automatically extract information from research papers from multiple scientific domains in a domain-independent manner?*

This research question involves the investigation of a generic domain-independent extraction approach since, by intuition, most research papers share certain core concepts, such as research tasks or methods.

As a consequence, an RKG that spans multiple scientific domains has not been populated yet. Current automatically populated RKGs cover only a single domain, for instance, artificial intelligence [70] or biomedicine [47]. However, during KG population, extracted mentions of scientific concepts in the text need to be collapsed to concept entities in the KG. Thus, it is not clear whether it is feasible to collapse mentions of scientific concepts across domains. Usually, terms within a scientific domain are unambiguous, but some terms can have different meanings across scientific disciplines (e.g. “neural network" has different meanings in computer science and medicine). Therefore, we investigate the following RQ:

> **RQ4:** *How can we automatically populate an RKG that covers multiple scientific domains?*

1.2.3 Automatically Populated Research Knowledge Graphs

Knowledge Graphs such as DBpedia [172], YAGO [275], or Wikidata [285] are well established. Since these KGs have been curated manually, they have rather high correctness but relatively low completeness [25, 75, 94, 274]. The usefulness of these KGs has been demonstrated in various applications [14, 83, 195, 261], especially in academic search engines [294]. However, the usefulness of *automatically* populated RKGs with high completeness but noisy data has not yet been demonstrated in downstream information retrieval tasks for research papers. In particular, current approaches for the task of citation recommendation to suggest relevant related work for a research paper do not exploit automatically populated RKGs [21, 40, 56, 132, 302]. Thus, we investigate the RQ:

> **RQ5:** *How can we exploit an automatically populated RKG to enhance the task of citation recommendation?*

1.3 Contributions

The previous section has revealed several challenges and research questions that need to be faced when constructing an RKG. In this section, we present our contributions that address these research questions. Our five main contributions can be summarised as follows:

1. **Requirements Analysis for an ORKG:** To address **RQ1**, we present literature-related use cases of researchers that should be supported by an ORKG [7] (e.g. *assess the relevance of research papers*, *find related work*, *get recommended articles*) and their specific requirements for the underlying ontology (granularity and domain-specialisation) and instance data (completeness and correctness). The requirements analysis builds the foundation for this thesis and shall guide further research.
2. **Multi-Task Learning for Sequential Sentence Classification:** We present a novel unified cross-domain multi-task learning approach for the task of *sequential sentence classification*. Our approach addresses **RQ2** by exploiting datasets from multiple scientific domains with different structures and text types (e.g. cover only abstracts or full papers, different sentence classes). Furthermore, to address **RQ3**, we present an approach to classify sentences in research papers in a domain-independent manner.
3. **Domain-Independent Information Extraction:** To address **RQ3**, we propose a domain-independent approach for scientific concept extraction and coreference resolution in ten different scientific domains from Science, Technology, and Medicine (STM). Our information extraction approach utilises active learning and transfer learning methods to reduce annotation costs (**RQ2**).
4. **Cross-Domain Research Knowledge Graph:** Using the above mentioned domain-independent information extraction approach, we automatically populate an RKG covering ten different scientific domains from over 55,000 abstracts of research papers (**RQ4**). The resulting RKG interconnects research papers through scientific concepts.
5. **Citation Recommendation via Knowledge Graphs:** To demonstrate the usefulness of our automatically populated RKG (**RQ5**), for the task of citation recommendation, we present a novel approach that can exploit such RKGs.

In the following, these contributions are described in more detail.

1.3.1 Requirements Analysis for an Open Research Knowledge Graph

As outlined in Section 1.2.1, the construction of an RKG is challenging and has several conflicting requirements. To illuminate these challenges, we perform a *requirements analysis*. We present a set of seven main use cases focusing on the literature-related tasks of scientists, namely *get research field overview* (#1), *find related work* (#2), *assess relevance of research papers* (#3), *extract relevant information from research papers* (#4), *get recommended articles* (#5), *obtain deep understanding of a research paper* (#6), and *reproduce results* (#7).

For each use case, we elaborate requirements for the underlying ontology (i.e. granularity and domain-specificness) and instance data (i.e. completeness and correctness). The identified use cases can be categorised into two groups: (1) The first group requires instance data with high correctness and rather fine-grained, domain-specific ontologies, and moderate completeness is acceptable. This group of use cases requires manual or semi-automatic approaches for KG construction. (2) The second group requires high completeness, but the ontologies can be kept rather simple and domain-independent, and moderate correctness of the instance data is sufficient. Thus, the construction of a KG for this group of use cases should be accomplished with automatic approaches.

This thesis focuses on the second set of use cases, i.e. to assist researchers in *assessing the relevance of research papers* (#3), *finding related work* (#2), and *recommending appropriate research papers* (#5). In the following, we present contributions for machine learning tasks that aim to assist these use cases.

1.3.2 Multi-Task Learning for Sequential Sentence Classification

To assess the relevance of a research paper, scientists usually skim through titles, abstracts, introductions, and conclusions [2, 133, 278]. An organised presentation of the articles' essential information would make this process more time-efficient. The task of *sequential sentence classification* addresses this problem by classifying sentences in an article in categories like *research problem*, *used methods*, or *obtained results* [68]. To target this problem, we present the following contributions:

Unified Deep Learning Approach: Current approaches for sequential sentence classification are designed either for abstracts or full papers only. Typically, deep learning is used for abstracts [54, 69, 100, 133, 295] whereas for full papers hand-crafted features and linear models have been suggested [6, 11, 85, 174]. We propose a unified deep learning approach that can be applied to various types of text with a different structure, e.g. abstracts as well as full papers.

Cross-Domain Multi-Task Learning: Transfer learning enables the combination of knowledge from multiple datasets to improve the classification performance and thus to reduce annotation costs. However, the field lacks studies on transfer learning for sequential sentence classification across domains [15, 46, 106, 168, 169, 178, 207]. For this purpose, we introduce a novel multi-task learning framework for sequential sentence classification that makes use of datasets from different scientific domains, with different annotation schemes, that contain abstracts or full papers. For datasets of full papers, our approach significantly outperforms the state of the art without any feature engineering, while being competitive for datasets consisting of abstracts only.

Semi-Automatic Analysis of Semantic Relatedness of Classes: Multiple annotation schemes have been developed for datasets from different scientific domains (e.g. [54, 68, 85, 147, 175]) that consist of varying associated sentence classes. An analysis of semantic relatedness of classes can help consolidate annotation schemes across domains. We propose a multi-task learning model to identify semantically related classes across annotation schemes semi-automatically. In contrast to prior work [175], our approach does not require the re-annotation of datasets with different annotation schemes. From the analysis of four datasets, we derived a domain-independent consolidated annotation scheme and compiled a domain-independent dataset. This allows for the classification of sentences in research papers with generic classes across disciplines.

1.3.3 Domain-Independent Information Extraction

We introduce the novel task of *domain-independent information extraction* from research papers, which aims to extract scientific concepts from research papers in a domain-independent manner. For this task, we present the following contributions:

Domain-Independent Extraction of Scientific Concepts: The extraction of scientific concepts from research papers is a first vital step towards a fine-grained RKG (see Section 2.4.2.2). However, most datasets focus on at most three scientific disciplines and rather domain-specific concept types (e.g. [9, 53, 139, 178, 229]). We introduce a set of generic scientific concepts that are relevant over ten domains in Science, Technology, and Medicine (STM), and release an annotated dataset of papers from these domains. Our experimental results with a state-of-the-art deep learning approach demonstrate that the domain-independent model noticeably outperforms the domain-specific ones, which indicates that the domain-independent model can generalise well across domains.

Active Learning for Scientific Concept Extraction: Active learning is an important technique to determine the optimal set of sufficiently distinct instances during the annotation of a training dataset [255]. Various studies [258, 259, 300] demonstrated that active learning can help reducing annotation costs for various NLP tasks. The annotation of scientific text is particularly costly since it demands expertise in the article's domain [9, 92]. However, to the best of our knowledge, active learning has not been applied to scientific text yet. We demonstrate that active learning enables us to nearly halve the amount of required training data on concept extraction from scientific text so that about five annotated abstracts per domain serving as training data are sufficient to build a performant model.

Coreference Resolution in Multiple Domains: Coreference resolution is the task of identifying mentions in a text which refer to the same entity or concept [117]. Most corpora for coreference resolution in research papers are limited to only a single domain (e.g.

biomedicine [58], artificial intelligence [178]). We extend our corpus for domain-independent scientific concept extraction with coreference annotations that covers ten STM domains.

Transfer Learning for Coreference Resolution: Datasets for coreference resolution in the general domain (e.g. news, magazines, etc.) are usually much larger than for the scientific domain and therefore machine learning approaches trained on these large datasets obtain impressive results (e.g. an F1 score of 79.6% for the OntoNotes 5.0 dataset [136]). However, current approaches for coreference resolution in scientific texts have not exploited these datasets yet [137, 170, 171, 178, 182]. We propose a sequential transfer learning approach for coreference resolution that takes advantage of such datasets by first pre-training a model with a large dataset from the general domain and fine-tuning the pre-trained model on a (smaller) dataset from the scientific domain. Our experimental results demonstrate that our approach noticeably outperforms the state-of-the-art baselines.

1.3.4 Cross-Domain Research Knowledge Graph

Using the above domain-independent information extraction approach, we automatically populate a fine-grained RKG covering ten STM domains. The RKG interconnects research papers through scientific concepts like materials, methods, and processes, and aims to support the *find related work* and *get recommended articles* use cases. We present the following contributions:

Impact of Coreference Resolution on KG Population: Coreference resolution is one of the main steps in the KG population pipeline [179, 226]. However, to date, it is not clear to what extent coreference resolution influences the quality of the populated KG [291]. We present an evaluation procedure for the clustering aspect (i.e. clustering mentions across documents to concept nodes) in the KG population pipeline and demonstrate that coreference resolution improves the quality of a populated KG significantly.

Automatically Populated KG from Multiple Domains: So far, automatically populated RKGs that interconnect scientific concepts with research papers comprise only a single domain (e.g. artificial intelligence [70], biomedicine [47]) so that an RKG covering multiple scientific domains does not exist yet. Based on our annotated corpora for domain-independent scientific concept extraction and coreference resolution, we have populated an RKG from 55,485 abstracts of the ten investigated STM domains. We have shown that the populated KG contains useful concepts and that every domain mostly uses its own terminology.

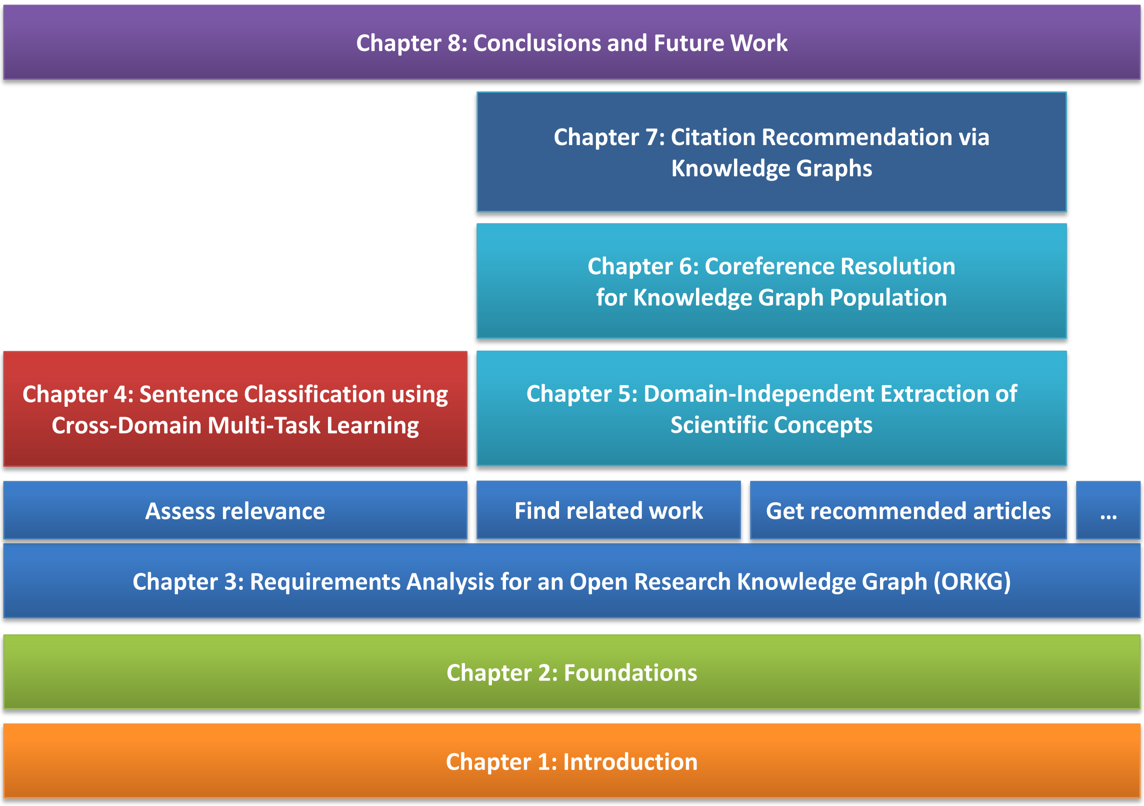

Chapter 8: Conclusions and Future Work
Chapter 7: Citation Recommendation via Knowledge Graphs
Chapter 6: Coreference Resolution for Knowledge Graph Population
Chapter 4: Sentence Classification using Cross-Domain Multi-Task Learning
Chapter 5: Domain-Independent Extraction of Scientific Concepts
Assess relevance
Find related work
Get recommended articles
...
Chapter 3: Requirements Analysis for an Open Research Knowledge Graph (ORKG)
Chapter 2: Foundations
Chapter 1: Introduction

Figure 1.2: Structure of this thesis.

1.3.5 Citation Recommendation via Knowledge Graphs

Citation recommendation for research papers can help researchers improve the quality of their work by finding or recommending relevant related work. Current approaches for the task of citation recommendation primarily rely on the text of the papers and the citation network [21, 40, 56, 132, 302]. We propose to exploit an additional source of information, namely RKGs that interlink research papers based on mentioned scientific concepts. Based on our automatically populated RKG, we show that the combination of information from RKGs with existing state-of-the-art approaches for citation recommendation is beneficial.

1.4 Thesis Structure

Figure 1.2 shows the overall structure of this thesis and the dependencies between the chapters. First, **Chapter 2** introduces the fundamental technology used in this thesis and gives an overview on Natural Language Processing (NLP) with deep learning approaches and KGs. Then, **Chapter 3** presents a comprehensive related work overview and analyses the requirements for an ORKG. The outcome of this chapter is a set of use cases, data quality requirements, and construction strategies for an ORKG. This thesis focuses on use cases that require automatic KG population approaches to achieve high completeness of the instance data, namely *assess relevance of research papers*, *find related work*, and *get recommended articles*. The subsequent chapters provide contributions for machine learning tasks that aim to support these use cases. **Chapter 4** addresses the task of *sequential sentence classification* that can contribute to the use case *assess relevance of research papers* since it enables to identify relevant sentences in research papers. Here, we propose a unified multi-task learning approach that makes use of datasets from different scientific domains and varying structures. The remaining chapters 5, 6, and 7 aim to support the use cases *find related work* and *get recommended articles*. For this purpose, **Chapter 5** presents an annotated corpus for the novel task of *domain-independent scientific concept extraction*, which aims at automatically extracting scientific concepts in a domain-independent manner. Furthermore, the chapter proposes an active learning approach for this task. Based on this work, **Chapter 6** extends the annotated corpus with coreference annotations. This corpus is then used to populate an RKG from multiple domains and evaluate the impact of coreference resolution on the quality of the populated RKG. Finally, **Chapter 7** proposes a novel approach for the task of citation recommendation that exploits automatically populated RKGs. The approach is evaluated with the RKG populated in the previous chapter. The last **Chapter 8** concludes this thesis, outlines limitations of our methods, and provides an outlook for potential areas of future work.

1.5 List of Publications

This section provides an overview of the publications that have been published in the context of this thesis. While I have been the main author of the papers that are the basis of this thesis, these publications were joint work with other co-authors. Therefore, throughout this thesis, the academic “we” is used. The keywords under **My Contributions** serve to identify my contributions to the respective papers using the Contributor Roles Taxonomy (CRediT) [34].

Five of the publications (underlying this thesis) were published at conferences ranked A* [29], A [28, 33], or B [30, 31] according to the *Australian Computing Research & Education* (CORE)[1] Conference Portal (source: CORE2021). Moreover, one paper [31] has been selected as one of the best papers at the *Theory and Practice of Digital Libraries* (TPDL) conferences in 2019 and 2020, and published as an extended article [32] in the *International Journal on Digital Libraries* (IJDL). In the following, we outline the publications that are part of this thesis.

The requirements analysis in Chapter 3 is based on the following two papers [31, 32]. The journal paper [32] is an extended version of the conference paper [31].

[1] `http://portal.core.edu.au/conf-ranks/`

- Arthur Brack, Anett Hoppe, Markus Stocker, Sören Auer, and Ralph Ewerth. "Requirements Analysis for an Open Research Knowledge Graph". In: *Digital Libraries for Open Knowledge - 24th International Conference on Theory and Practice of Digital Libraries, TPDL 2020, Lyon, France, August 25-27, 2020, Proceedings.* Ed. by Mark M. Hall, Tanja Mercun, Thomas Risse, and Fabien Duchateau. Vol. 12246. Lecture Notes in Computer Science. Springer, 2020, pp. 3–18. DOI: 10.1007/978-3-030-54956-5_1. URL: https://doi.org/10.1007/978-3-030-54956-5_1 [31]
- Arthur Brack, Anett Hoppe, Markus Stocker, Sören Auer, and Ralph Ewerth. "Analysing the requirements for an Open Research Knowledge Graph: use cases, quality requirements, and construction strategies". In: *Int. J. Digit. Libr.* 23.1 (2022), pp. 33–55. DOI: 10.1007/s00799-021-00306-x. URL: https://doi.org/10.1007/s00799-021-00306-x [32]

  Abstract: Current science communication has a number of drawbacks and bottlenecks which have been subject of discussion lately: Among others, the rising number of published articles makes it nearly impossible to get a full overview of the state of the art in a certain field, or reproducibility is hampered by fixed-length, document-based publications which normally cannot cover all details of a research work. Recently, several initiatives have proposed knowledge graphs (KG) for organising scientific information as a solution to many of the current issues. The focus of these proposals is, however, usually restricted to very specific use cases. In this paper, we aim to transcend this limited perspective and present a comprehensive analysis of requirements for an Open Research Knowledge Graph (ORKG) by (a) collecting and reviewing daily core tasks of a scientist, (b) establishing their consequential requirements for a KG-based system, (c) identifying overlaps and specificities, and their coverage in current solutions. As a result, we map necessary and desirable requirements for successful KG-based science communication, derive implications, and outline possible solutions.

  My Contributions: Conceptualisation, Methodology, Investigation, Resources, Writing – original draft, Writing – review & editing, Visualisation, Project administration

Chapter 4 presents a multi-task learning approach for sequential sentence classification and is based on the following conference paper:

- Arthur Brack, Anett Hoppe, Pascal Buschermöhle, and Ralph Ewerth. "Cross-Domain Multi-Task Learning for Sequential Sentence Classification in Research Papers". In: *JCDL '22: Proceedings of the ACM/IEEE Joint Conference on Digital Libraries in 2022, Cologne, Germany, June 20-24, 2022 (accepted for publication).* ACM, 2022. DOI: 10.1145/3529372.3530922. URL: https://doi.org/10.1145/3529372.3530922 [29]

Abstract: Sequential sentence classification deals with the categorisation of sentences based on their content and context. Applied to scientific texts, it enables the automatic structuring of research papers and the improvement of academic search engines. However, previous work has not investigated the potential of transfer learning for sentence classification across different scientific domains and the issue of different text structure of full papers and abstracts. In this paper, we derive seven related research questions and present several contributions to address them: First, we suggest a novel uniform deep learning architecture and multi-task learning for cross-domain sequential sentence classification in scientific texts. Second, we tailor two common transfer learning methods, sequential transfer learning and multi-task learning, to deal with the challenges of the given task. Semantic relatedness of tasks is a prerequisite for successful transfer learning of neural models. Consequently, our third contribution is an approach to semi-automatically identify semantically related classes from different annotation schemes and we present an analysis of four annotation schemes. Comprehensive experimental results indicate that models, which are trained on datasets from different scientific domains, benefit from one another when using the proposed multi-task learning architecture. We also report comparisons with several state-of-the-art approaches. Our approach outperforms the state of the art on full paper datasets significantly while being on par for datasets consisting of abstracts.

Source Code: `https://github.com/arthurbra/sequential-sentence-classification`

My Contributions: Conceptualisation, Methodology, Software, Validation, Investigation, Resources, Data Curation, Writing – original draft, Writing – review & editing, Visualisation, Supervision, Project administration

Chapter 5 introduces the novel task of domain-independent scientific concept extraction and is based on the following conference paper:

- Arthur Brack, Jennifer D'Souza, Anett Hoppe, Sören Auer, and Ralph Ewerth. "Domain-Independent Extraction of Scientific Concepts from Research Articles". In: *Advances in Information Retrieval - 42nd European Conference on IR Research, ECIR 2020, Lisbon, Portugal, April 14-17, 2020, Proceedings, Part I.* ed. by Joemon M. Jose, Emine Yilmaz, João Magalhães, Pablo Castells, Nicola Ferro, Mário J. Silva, and Flávio Martins. Vol. 12035. Lecture Notes in Computer Science. Springer, 2020, pp. 251–266. DOI: `10.1007/978-3-030-45439-5_17`. URL: `https://doi.org/10.1007/978-3-030-45439-5_17` [28]

  Abstract: We examine the novel task of *domain-independent scientific concept extraction from abstracts of scholarly articles* and present two contributions. First, we suggest a set of generic scientific concepts that have been identified in a systematic annotation process. This set of concepts is utilised to annotate a corpus of scientific abstracts from 10 domains of Science, Technology and Medicine at the phrasal level in a joint effort with domain experts. The resulting dataset is used in a set of benchmark experiments to (a) provide baseline performance for this task, (b) examine the transferability of concepts between domains. Second, we present a state-of-the-art deep learning baseline. Further, we propose the active learning strategy for an optimal selection of instances from among the various domains in our data. The experimental results show that (1) a substantial agreement is achievable by non-experts after consultation with domain experts, (2) the baseline system achieves a fairly high F1 score, (3) active learning enables us to nearly halve the amount of required training data.

Source Code: `https://gitlab.com/TIBHannover/orkg/orkg-nlp/tree/master/STM-corpus`

My Contributions: Conceptualisation, Methodology, Software, Validation, Investigation, Resources, Writing – original draft, Writing – review & editing, Visualisation

Chapter 6 extends the task of domain-independent scientific concept extraction with the task of coreference resolution, proposes and evaluates approaches for RKG population, and populates an RKG from multiple domains. This chapter is based on the following conference paper:

- Arthur Brack, Daniel Uwe Müller, Anett Hoppe, and Ralph Ewerth. "Coreference Resolution in Research Papers from Multiple Domains". In: *Advances in Information Retrieval - 43rd European Conference on IR Research, ECIR 2021, Virtual Event, March 28 - April 1, 2021, Proceedings, Part I.* ed. by Djoerd Hiemstra, Marie-Francine Moens, Josiane Mothe, Raffaele Perego, Martin Potthast, and Fabrizio Sebastiani. Vol. 12656. Lecture Notes in Computer Science. Springer, 2021, pp. 79–97. DOI: `10.1007/978-3-030-72113-8_6`. URL: `https://doi.org/10.1007/978-3-030-72113-8_6` [33]

  Abstract: Coreference resolution is essential for automatic text understanding to facilitate high-level information retrieval tasks such as text summarisation or question answering. Previous work indicates that the performance of state-of-the-art approaches (e.g. based on BERT) noticeably declines when applied to scientific papers. In this paper, we investigate the task of coreference resolution in research papers and subsequent knowledge graph population. We present the following contributions: (1) We annotate a corpus for coreference resolution that comprises 10 different scientific disciplines from Science, Technology, and Medicine (STM); (2) We propose transfer learning for automatic coreference resolution in research papers; (3) We analyse the impact of coreference resolution on knowledge graph (KG) population; (4) We release a research KG that is automatically populated from 55,485 papers in 10 STM domains. Comprehensive experiments show the usefulness of the proposed approach. Our transfer learning approach considerably outperforms state-of-the-art baselines on our corpus with an F1 score of 61.4 (+11.0), while the evaluation against a gold standard KG shows that coreference resolution improves the quality of the populated KG significantly with an F1 score of 63.5 (+21.8).

Source Code: `https://github.com/arthurbra/stm-coref`

My Contributions: Conceptualisation, Methodology, Software, Validation, Investigation, Resources, Data Curation, Writing – original draft, Writing – review & editing, Visualisation, Supervision, Project administration

The following paper was published during my PhD study and extends our annotated corpus of paper [28] with the task of entity linking. This paper is not part of this thesis. However, the extended corpus is used to evaluate the impact of coreference resolution on KG population in Chapter 6.

- Jennifer D'Souza, Anett Hoppe, Arthur Brack, Mohamad Yaser Jaradeh, Sören Auer, and Ralph Ewerth. "The STEM-ECR Dataset: Grounding Scientific Entity References in STEM Scholarly Content to Authoritative Encyclopedic and Lexicographic Sources". In: *Proceedings of The 12th Language Resources and Evaluation Conference, LREC 2020, Marseille, France, May 11-16, 2020.* Ed. by Nicoletta Calzolari, Frédéric Béchet, Philippe Blache, Khalid Choukri, Christopher Cieri, Thierry Declerck, Sara Goggi, Hitoshi Isahara, Bente Maegaard, Joseph Mariani, Hélène Mazo, Asunción Moreno, Jan Odijk, and Stelios Piperidis. European Language Resources Association, 2020, pp. 2192–2203. URL: `https://www.aclweb.org/anthology/2020.lrec-1.268/` [61]

  Abstract: We introduce the STEM (Science, Technology, Engineering, and Medicine) Dataset for Scientific Entity Extraction, Classification, and Resolution, version 1.0 (STEM-ECR v1.0). The STEM-ECR v1.0 dataset has been developed to provide a benchmark for the evaluation of scientific entity extraction, classification, and resolution tasks in a domain-independent fashion. It comprises abstracts in 10 STEM disciplines that were found to be the most prolific ones on a major publishing platform. We describe the creation of such a multidisciplinary corpus and highlight the obtained findings in terms of the following features: 1) a generic conceptual formalism for scientific entities in a multidisciplinary scientific context; 2) the feasibility of the domain-independent human annotation of scientific entities under such a generic formalism; 3) a performance benchmark obtainable for automatic extraction of multidisciplinary scientific entities using BERT-based neural models; 4) a delineated 3-step entity resolution procedure for human annotation of the scientific entities via encyclopedic entity linking and lexicographic word sense disambiguation; and 5) human evaluations of Babelfy returned encyclopedic links and lexicographic senses for our entities. Our findings cumulatively indicate that human annotation and automatic learning of multidisciplinary scientific concepts as well as their semantic disambiguation in a wide-ranging setting as STEM is reasonable.

Source Code: `https://gitlab.com/TIBHannover/orkg/orkg-nlp/-/tree/master/STEM-ECR-v1.0`

My Contributions: Software, Investigation

The approach for citation recommendation via KGs presented in Chapter 7 is based on the following conference paper:

- Arthur Brack, Anett Hoppe, and Ralph Ewerth. "Citation Recommendation for Research Papers via Knowledge Graphs". In: *Linking Theory and Practice of Digital Libraries - 25th International Conference on Theory and Practice of Digital Libraries, TPDL 2021, Virtual Event, September 13-17, 2021, Proceedings.* Ed. by Gerd Berget, Mark Michael Hall, Daniel Brenn, and Sanna Kumpulainen. Vol. 12866. Lecture Notes in Computer Science. Springer, 2021, pp. 165–174. DOI: `10.1007/978-3-030-86324-1_20`. URL: `https://doi.org/10.1007/978-3-030-86324-1_20` [30]

  Abstract: Citation recommendation for research papers is a valuable task that can help researchers improve the quality of their work by suggesting relevant related work. Current approaches for this task rely primarily on the text of the papers and the citation network. In this paper, we propose to exploit an additional source of information, namely research knowledge graphs (KGs) that interlink research papers based on mentioned scientific concepts. Our experimental results demonstrate that the combination of information from research KGs with existing state-of-the-art approaches is beneficial. Experimental results are presented for the STM-KG (STM: Science, Technology, Medicine), which is an automatically populated knowledge graph based on the scientific concepts extracted from papers of ten domains. The proposed approach outperforms the state of the art with a mean average precision of 20.6% (+0.8) for the top-50 retrieved results.

  Source Code: `https://github.com/arthurbra/citation-recommendation-kg`

  My Contributions: Conceptualisation, Methodology, Software, Validation, Investigation, Resources, Data Curation, Writing – original draft, Writing – review & editing, Visualisation, Supervision, Project administration

◊◊◊

This chapter has motivated the topic of this thesis, outlined the addressed research challenges and research questions, and summarised its contributions. The next chapter introduces the foundations of this thesis, namely Natural Language Processing (NLP) and KGs.

2 Foundations

This thesis deals with information extraction from scientific papers for Research Knowledge Graph (RKG) population. In this chapter, we introduce the foundations for this thesis, namely selected state-of-the-art approaches for information extraction from text and the construction of a KG. Since current approaches for Natural Language Processing (NLP) are based on artificial neural networks, Section 2.1 first introduces the basics of them. Then, Section 2.2 describes neural network architectures for sequence processing that are used for various NLP tasks, namely Recurrent Neural Networks (RNNs) and the transformer architecture. The building blocks of these architectures form the foundation for the approaches proposed in this thesis. Subsequently, Section 2.3 presents text representation approaches using word embeddings that employ neural networks and architectures for sequence processing. Word embeddings are utilised in all proposed approaches of this thesis. Knowledge Graphs and the tasks involved to construct them automatically are introduced in Section 2.4. Finally, Section 2.5 describes evaluation methods and metrics that are used to evaluate the approaches presented in this thesis. We refer to the notation chapter at the beginning of this thesis that describes the mathematical formulas used in this thesis.

2.1 Basics of Artificial Neural Networks

The following description of neural networks is mainly based on the book of Jurafsky and Martin [138] unless otherwise stated. Artificial neural networks (abbreviated as neural networks) are a set of machine learning algorithms that are inspired by the brain's structure. The origin of them lies in the McCulloch-Pitts neuron [190], which was proposed in 1943 as a simplified model for the human neuron. Many current state-of-the-art approaches in NLP and other domains such as computer vision are based on neural networks [101].

A neural network is a function $\hat{\mathbf{y}} = f(\mathbf{x})$ that takes as input a vector $\mathbf{x}$ and outputs (or predicts) a scalar $\hat{y}$ or an output vector $\hat{\mathbf{y}}$ [138]. The neural network consists of small computation units (called neurons) whereas each neuron takes a vector as input and produces a single output value. The output values of the neurons can serve as inputs to further neurons that enable the neural network to approximate complex non-linear functions. Each neuron has a set of parameters (also called weights) used to compute the output value based on the input values. The values of these parameters are determined during the neural network

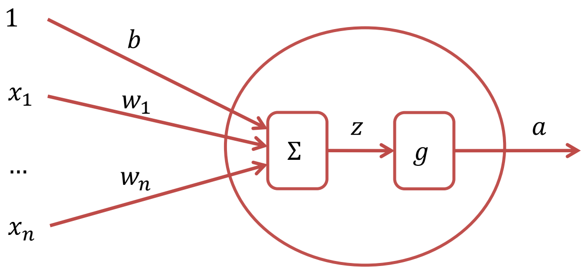


Figure 2.1: Illustration of a neural unit. It takes as input the values $\mathbf{x}_1, ..., \mathbf{x}_n$ and calculates a weighted sum z using the weights $\mathbf{w}_1, ..., \mathbf{w}_n$ and the bias value b. The sum z is passed to an activation function g that forms the final output a of the neuron. Illustration is based on [138].

training on a given set of training examples. Each training example consists of the input vector and an expected output vector (also called ground truth and referred to as $\mathbf{y}$ or y). In the following, we describe neural units (Section 2.1.1), feed-forward neural networks (Section 2.1.2), and the training procedure (Section 2.1.3) of neural networks in more detail.

2.1.1 Neural Units

Neural units (also called neurons) are the building blocks of neural networks and illustrated in Figure 2.1 [138]. It takes a vector of n values $\mathbf{x} = (\mathbf{x}_1, ..., \mathbf{x}_n)^\intercal$ as input, performs computations on them, and produces a single output value $a \in \mathbb{R}$. First, the neuron calculates a weighted sum z over the input values using a linear function with the weights $\mathbf{w} = (\mathbf{w}_1, ..., \mathbf{w}_n)^\intercal$ and the bias value b:

$$z = \sum_{i=1}^{n} \mathbf{w}_i \mathbf{x}_i + b = \mathbf{w}^\intercal \mathbf{x} + b \tag{2.1}$$

Then, the weighted sum z is passed to a non-linear activation function $g(.)$ to produce the final output a of the neuron.

$$a = g(z) \tag{2.2}$$

Since neural units are composed to form a neural network such that the outputs of the units are used as input to further units, activation functions enable the network to learn complex non-linear functions [138]. Without activation functions, neural networks would be the composition of multiple linear functions, forming a linear function again. Figure 2.2 illustrates the most common activation functions used in neural units: sigmoid, hyperbolic tangent (tanh), and Rectified Linear Unit (ReLU). These activation functions have different useful properties [138]: For instance, the sigmoid function maps the input into the range $[0, 1]$ that is useful in binary classification tasks. Furthermore, outliers are squashed towards 0 or 1.

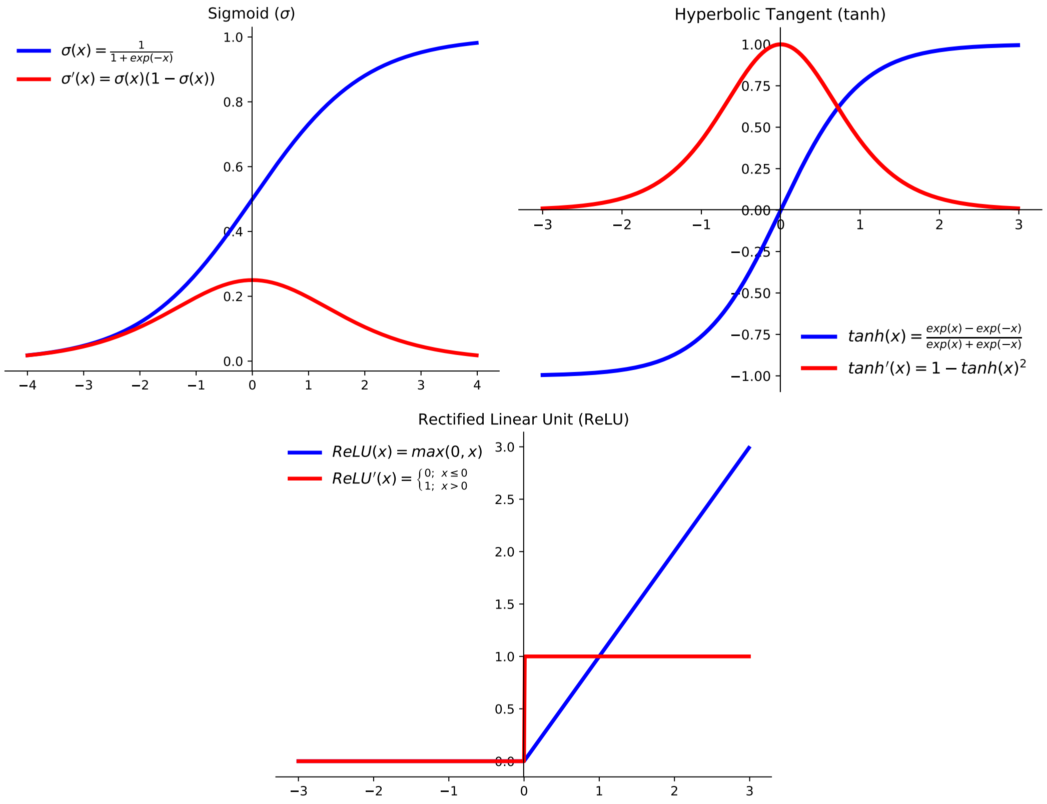


Figure 2.2: Common activation functions (blue lines) and their derivatives (red lines), namely sigmoid (σ), hyperbolic tangent (tanh), and Rectified Linear Unit (ReLU).

The tanh function maps the input values into the range $[-1, 1]$ so that the mean of the output values are more centred around zero, which helps the network to learn easier. However, for very large and very low input values, the results of sigmoid and tanh are saturated so that the gradients become very small, resulting in the *vanishing gradient* problem. In the vanishing gradient problem, the gradients become smaller and smaller so that the neural network cannot learn (see Section 2.1.3.3 for further details). The ReLU activation function mitigates this problem since for high input values, the gradient is always 1.

2.1.2 Feed-Forward Neural Networks

A feed-forward neural network consists of an input layer, multiple hidden layers, and an output layer [138]. The input layer represents the input values $\mathbf{x} = (\mathbf{x}_1, ..., \mathbf{x}_{n_x})^\intercal$ and the output layer represents the last layer that computes the final output value $\hat{\mathbf{y}} = (\hat{\mathbf{y}}_1, ..., \hat{\mathbf{y}}_{n_y})^\intercal$ that may be a single scalar (e.g. in regression tasks), or a vector representing a probability distribution such as in classification tasks. Figure 2.3 illustrates a feed-forward neural network. In the following, we describe the hidden and output layers in more detail.

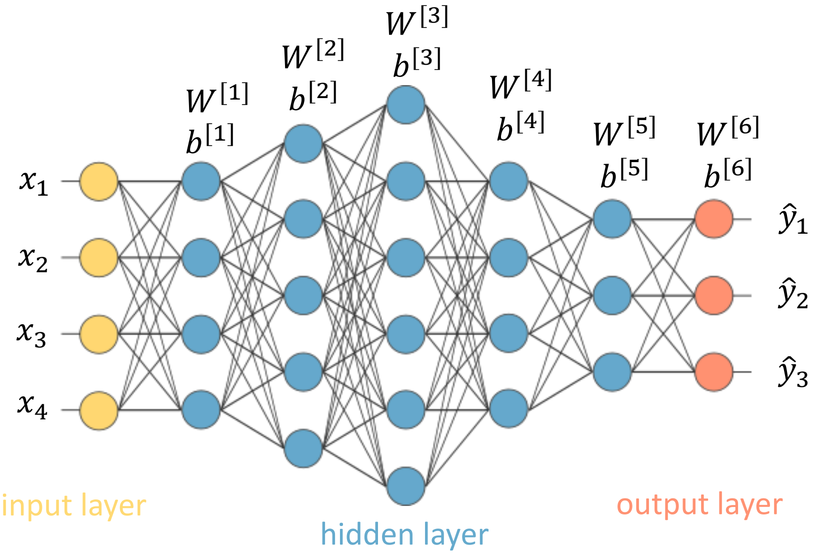


Figure 2.3: Illustration of a fully-connected feed-forward neural network consisting of the input layer (yellow), the hidden layers (blue), and the output layer (red). A circle represents a neural unit, and the matrix $W^{[l]}$ and vector $b^{[l]}$ with $1 \leq l \leq 6$ represent the parameters of the corresponding layers. Illustration based on [138].

Hidden Layer: Hidden layers are the core of a neural network that consist of multiple neural units (also called hidden units) [138]. The hidden layers can be composed to form a fully-connected feed-forward neural network such that each unit in a layer takes as input the outputs from all the units in the previous layer. A weight matrix and a bias vector can represent the parameters of all units in a hidden layer, which enables calculating all output values of a layer using efficient matrix multiplication. More formally, when $\mathbf{a}^{[\mathbf{0}]} = \mathbf{x}$, the output vector $\mathbf{a}^{[\mathbf{l}]}$ of a hidden layer l is computed recursively as follows:

$$\mathbf{z}^{[\mathbf{l}]} = \mathbf{W}^{[\mathbf{l}]}\mathbf{a}^{[\mathbf{l-1}]} + \mathbf{b}^{[\mathbf{l}]} \tag{2.3}$$

$$\mathbf{a}^{[\mathbf{l}]} = g^{[l]}(\mathbf{z}^{[\mathbf{l}]}) \tag{2.4}$$

Here, the weight matrix $\mathbf{W}^{[\mathbf{l}]} \in \mathbb{R}^{n_l \times n_{l-1}}$ and the bias vector $\mathbf{b}^{[\mathbf{l}]} \in \mathbb{R}^{n_l}$ represent the parameters of the l-th layer, n_l denotes the number of neural units in layer l, and $g^{[l]}(.)$ is the activation function in layer l.

Output Layer: The output layer is the last layer that produces the final output value $\hat{y}$ or vector $\hat{\mathbf{y}}$ of the neural network [138]. Depending on the task, various output layer types exist. For instance, in a binary classification task, the output layer consists of a single neuron that outputs a probability distribution in the range $0 \leq \hat{y} \leq 1$ (e.g. how likely is a given email a spam email). Since the output layers and the associated loss functions are coupled, they are described in Section 2.1.3.1 in more detail.

2.1.3 Training Neural Networks

To determine the values of the parameters of the neural network, we first need a training set $\mathbb{X} = ((\mathbf{x^{(1)}}, \mathbf{y^{(1)}}), ..., (\mathbf{x^{(m)}}, \mathbf{y^{(m)}}))$ of m training examples each consisting of the input $\mathbf{x^{(i)}}$ and the expected output $\mathbf{y^{(i)}}$ (also called *ground truth*) [138]. Now, a neural network is defined as a function $\hat{\mathbf{y}} = f(\mathbf{x}, \Theta)$ that takes as input a training sample $\mathbf{x}$ and the parameters of the network $\Theta = \{\mathbf{W^{[i]}}, \mathbf{b^{[i]}} | 1 \leq i \leq L\}$ with L layers, and outputs a vector $\hat{\mathbf{y}}$.

2.1.3.1 Loss Functions

Given the training set $\mathbb{X}$ and the neural network $f(\mathbf{x}, \Theta)$, the goal of the training procedure is to learn the parameters Θ such that for each training sample $\mathbf{x}^{(i)}$ the predicted output $\hat{\mathbf{y}}^{(i)}$ is as close as possible to the expected output $\mathbf{y}^{(i)}$ [138]. The difference between the predicted and the expected output is denoted as the loss $L(\hat{\mathbf{y}}^{(i)}, \mathbf{y}^{(i)})$. The loss function should return a value close to zero when the predicted and expected output values are close to each other, and a higher value otherwise. More formally, let $J(\Theta)$ be the function of the network parameters Θ that is the loss averaged over all training examples:

$$J(\Theta) = \frac{1}{m} \sum_{i=1}^{m} L(f(\mathbf{x^{(i)}}, \Theta), \mathbf{y^{(i)}}) \tag{2.5}$$

Now, the objective is to find the parameter values $\hat{\Theta}$ at which $J(\hat{\Theta})$ attains a minimum:

$$\hat{\Theta} = \arg\min_{\Theta} J(\Theta) \tag{2.6}$$

Depending on the task, different output layers and loss functions are required. Since this thesis deals with classification tasks, in the following, we describe the corresponding output layers and loss functions in more detail.

Binary Classification: The output layer in a binary classification task consists of a single neuron with the sigmoid activation function that outputs a value $0 \leq \hat{y} \leq 1$ [101]. The expected output $y \in \{0, 1\}$ is encoded as a binary value, e.g. in a spam-email classification task, value 1 indicates a spam email and 0 a non-spam email. As a loss function, Binary Cross-Entropy (BCE) can be used:

$$L_{BCE}(\hat{y}, y) = -y \log(\hat{y}) + (1 - y) \log(1 - \hat{y}) \tag{2.7}$$

Thus, when the predicted and the expected output are close to each other, the binary cross-entropy loss function returns a value close to zero. Otherwise, when the predicted and expected output differ, the loss increases rapidly.

Multiclass Classification: In a multiclass classification task with more than two classes (e.g. categorise news by genre such as entertainment, sports, politics, etc.), the output layer produces a probability distribution over the set of classes $\mathbb{C}$ where $k = |\mathbb{C}|$ is the number of classes [101]. For this purpose, the output layer consists of k neurons without an activation function that produces a vector of values $(\mathbf{z}_1, ..., \mathbf{z}_k)^\intercal$. Each value $\mathbf{z}_i$ represents an unnormalised score for a class (the higher the score, the more probable the class). Then, the softmax function is used to obtain a probability distribution:

$$\hat{\mathbf{y}}_i = softmax(\mathbf{z}_i) = \frac{\exp(\mathbf{z}_i)}{\sum_{j=1}^{k} \exp(\mathbf{z}_j)} \tag{2.8}$$

Here, each value $\hat{\mathbf{y}}_i$ is in the range of $[0, 1]$ and all the values sum up to 1. Consequently, $\hat{\mathbf{y}}_i$ represents the probability of the input vector belonging to class i. Thus, the class with the highest probability is the predicted output of the neural network.

The expected output is encoded as a one-hot vector $onehot(c_i) \in \mathbb{R}^k$ for a class $c_i \in C$ that is defined as a vector of dimension k (i.e. the number of classes) in which the i-th component equals 1 and all remaining components are 0 [138]. So, the one-hot vector for class c_i indicates that the expected probability for class c_i is 1 and all other classes 0. As loss function, the Cross-Entropy (CE) loss is used as the generalisation of the Binary Cross-Entropy (BCE) loss function where $\mathbf{y}_i \in \{0, 1\}$ and $0 \leq \hat{\mathbf{y}}_i \leq 1$:

$$L_{ce}(\hat{\mathbf{y}}, \mathbf{y}) = \sum_{i=1}^{k} -\mathbf{y}_i \log(\hat{\mathbf{y}}_i) \tag{2.9}$$

Hence, the output of the cross-entropy loss is the logarithm of the predicted probability corresponding to the expected class. When the predicted probability for the expected class is high, then the loss is close to zero, otherwise, when the predicted probability for the expected class is low, then the loss becomes a large value.

2.1.3.2 Gradient Descent

The function $J(\Theta)$ in Equation 2.6 is a function of the network parameters Θ and the objective is to minimise this function. *Gradient descent* is a popular method that finds a minimum of a differentiable function by taking repeated steps in the opposite direction of the function's gradient at the current point [138]. The parameters of the network are first initialised with random values or with values of a pre-trained model (e.g. in a transfer learning scenario, see Section 4.2.2 for further details). Then, the parameters $\mathbf{w}_i \in \Theta$ are updated iteratively with a specified learning rate $\eta \in \mathbb{R}$ according to the following equation:

$$\mathbf{w}_i \leftarrow \mathbf{w}_i + \eta \frac{\partial J(\Theta)}{\partial \mathbf{w}_i} \tag{2.10}$$

Since a neural network is a composition of basic differentiable functions, the partial derivatives $\frac{\partial J(\Theta)}{\partial \mathbf{w}_i}$ can be computed efficiently with the *backpropagation algorithm* [240], that exploits the chain rule in calculus.

To train a neural network with many parameters and layers is a non-convex optimisation problem [138]. Therefore, the gradient descent algorithm can, for instance, be stuck in local minima or fail to converge. Adaptive learning rate algorithms such as Adaptive Moment Estimation (ADAM) [148] try to mitigate such problems. Furthermore, the computation of the loss requires a lot of memory and computation resources when the training set has many examples. To solve this, the loss can also be calculated on mini-batches with a few (e.g. 32) random training examples, which is also called *mini-batch training* [138].

2.1.3.3 Further Techniques for Training

In this section, we introduce further important techniques for training neural networks.

Regularisation: Neural networks are prone to *overfitting* since they usually have many parameters [138]. In overfitting, the neural network "memorises" the training data (i.e. leading to a very low loss during training) and thus cannot generalise sufficiently to unseen data. To avoid overfitting, various *regularisation techniques* have been proposed that can be applied during training. The most popular regularisation techniques are the following [101]: (1) $L2$ regularisation that encourages the network to keep the learned parameter values small, and (2) *dropout* in which some units and their connections are dropped during training.

Vanishing and Exploding Gradients: Additionally, deep neural networks consisting of many hidden layers can suffer from *exploding gradients* or *vanishing gradients*, i.e. the gradients can become very large or very small [138]. The reason for this is that the derivative of a composed function is a product of derivatives. For instance, if the neural network consists of many layers, the product may consist of many factors. Small factors multiplied with each other become even smaller, large factors multiplied with each other become even larger. Small gradients slow down the training and large gradients can lead to numerical overflows. Exploding gradients can be solved via *gradient clipping* [209], i.e. the length of the gradients is truncated. To address the vanishing gradients problem, *residual connections* [112] between layers can be introduced to enable the gradients to flow through layers directly without passing non-linear activation functions. A further technique is *batch normalisation* [128] where the input vectors to a layer are standardised for each mini-batch so that values have the mean zero and a standard deviation of one, whereas in *layer normalisation* [10] the input vectors to a layer are normalised for each training sample.

Hyperparameter Tuning: Furthermore, there are various *hyperparameters* that cannot be learned with gradient descent. Hyperparameters are chosen by the algorithm designer, such as the number of hidden layers, number of neurons in each layer, activation functions, learning rate, or regularisation techniques [138]. To enable hyperparameter tuning, we need a further set of annotated examples, called *validation set*. First, the training set is used to train neural network models with different hyperparameters, and for each trained model, the loss on the validation set is computed. Finally, the model with the lowest loss on the validation set is chosen for prediction.

2.2 Neural Network Architectures for Sequence Processing

In many NLP tasks, we require to process sequential data where the input and output length can vary. For instance, in the English-to-German machine translation task [283], the input is a sequence of words in English, and the output is a sequence of words in German. Another example is part-of-speech tagging [213], where the input is a sequence of words, and the output is a sequence of classes where each class denotes the part-of-speech of the word (e.g. noun, verb). However, classic feed-forward neural networks, as introduced in Section 2.1 expect input and output vectors of a fixed size. In the following, we introduce *Recurrent Neural Networks* (RNNs) and the *transformer* architecture. Both approaches can process a sequence of input vectors $(\mathbf{x_1}, ..., \mathbf{x_{\tau_x}})$ (e.g. sequence of words) and produce an output sequence $(\mathbf{\hat{y}_1}, ..., \mathbf{\hat{y}_{\tau_y}})$ (e.g. sequence of words or classes). Here, τ_x denotes the length of the input sequence and τ_y the length of the output sequence. For approaches to represent words as vectors, we refer to Section 2.3. The presented architectures in this section form the basis for the proposed approaches of this thesis, namely for the tasks of sequential sentence classification (Chapter 4), scientific concept extraction (Chapter 5), and coreference resolution (Chapter 6). The following descriptions of RNNs and the transformer architecture are mainly based on Jurafsky and Martin [138] unless otherwise stated.

2.2.1 Recurrent Neural Networks (RNNs)

As depicted in Figure 2.4, an RNN processes the input data sequentially, i.e. at each time step t ranging from 1 to τ_x an RNN takes as input the value $\mathbf{x_t}$ and the previous hidden state $\mathbf{h_{t-1}}$ of the RNN, and produces a new hidden state $\mathbf{h_t}$ [138]. The previous hidden state $\mathbf{h_{t-1}}$ encodes information about the input data until time step $t-1$ and thus represents the "memory" of the RNN. More formally, an RNN is defined recursively as follows while the initial hidden state $\mathbf{h_0}$ is a zero vector:

$$\mathbf{h_t} = rnn(\mathbf{h_{t-1}}, \mathbf{x_t}) \tag{2.11}$$

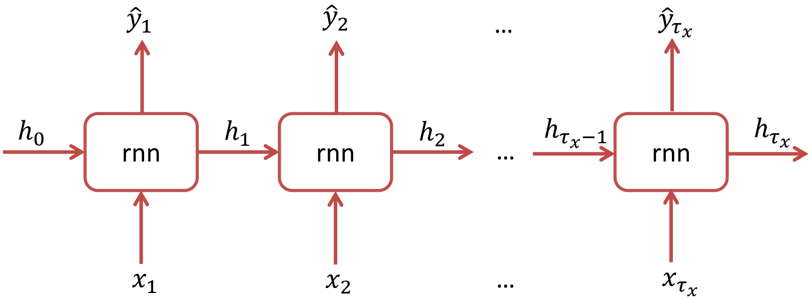


Figure 2.4: Illustration of an RNN that takes as input the sequence $(\mathbf{x_1}, ..., \mathbf{x_{\tau_x}})$ and produces the output sequence $(\mathbf{\hat{y}_1}, ..., \mathbf{\hat{y}_{\tau_x}})$ based on the hidden states $(\mathbf{h_1}, ..., \mathbf{h_{\tau_x}})$. Illustration is based on [138].

The function *rnn* corresponds to a neural unit and can use any non-linear activation function, for instance:

$$rnn(\mathbf{h_{t-1}}, \mathbf{x_t}) = \tanh(\mathbf{W}\mathbf{h_{t-1}} + \mathbf{U}\mathbf{x_t} + \mathbf{b}) \tag{2.12}$$

Here, **W** and **U** are the weight matrices and **b** is the bias vector.

2.2.1.1 Common RNN Architectures

Figure 2.5 shows four popular RNN architectures and their typical use cases in NLP tasks [138, 143]. The architectures depend mainly on the expected input τ_x and output length τ_y. Depending on the architectures, different approaches are used to calculate the output sequence $(\mathbf{\hat{y}_1}, ..., \mathbf{\hat{y}_{\tau_y}})$. Please note, that in the following, we describe the forward pass of RNNs during the *training phase*. The generation of the output sequence in the *prediction phase* is described in Section 2.2.1.4. Furthermore, we consider only classification tasks where the elements of the output sequence represent categorical data such as words or classes. We refer to Section 2.3 on how to represent words as vectors. Next, we describe approaches to compute the output sequence [138]:

1. *Many-to-one*: Here, we have only one output vector. This architecture is used in text classification tasks such as spam email classification. The output is computed using the last hidden state $\mathbf{h_{\tau_x}}$ using softmax (see Equation 2.8):

   $$\mathbf{\hat{y}} = softmax(\mathbf{V}\mathbf{h_{\tau_x}} + \mathbf{c}) \tag{2.13}$$

   Here, **V** is the weight matrix and **c** is a bias vector.

2. *Many-to-many with* $\tau_x = \tau_y$: In this architecture, we have an output value for each input value, such as in part-of-speech tagging or named entity recognition. The output value can be computed at each time step t ranging from 1 to τ_x based on the

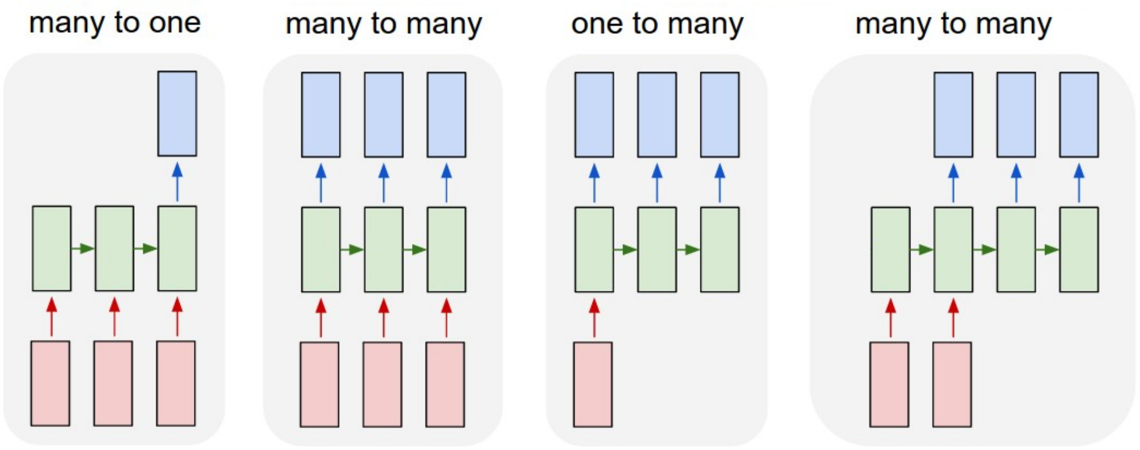


Figure 2.5: RNN architectures (from left to right): (a) *many to one* with $\tau_y = 1$ (e.g. document classification [298]), (b) *many to many* with $\tau_x = \tau_y$ (e.g. part-of-speech tagging [213]), (c) *one to many* with $\tau_x = 1, \tau_y \geq 1$ (e.g. text generation [37]), and (d) *many to many* with $\tau_x \neq \tau_y$ (e.g. machine translation [283]). Each rectangle is a vector and arrows represent functions (e.g. matrix multiplication). Input vectors are in red, output vectors are in blue, and green vectors hold the RNN's hidden state. Illustration is based on [143].

corresponding hidden state $\mathbf{h_t}$ using $softmax$ (see Equation 2.8):

$$\mathbf{\hat{y}_t} = softmax(\mathbf{V}\mathbf{h_t} + \mathbf{c}) \tag{2.14}$$

3. *One-to-many*: Here, we have only one input value and multiple output values. This architecture is used to generate output sequences, for example, in text or music generation. To compute the hidden states, the input values $\mathbf{x_t}$ with $t > 1$ correspond to the previously expected outputs $\mathbf{y_{t-1}}$ and the output $\mathbf{\hat{y}_t}$ is computed using softmax:

$$\mathbf{y_0} = \mathbf{x_1} \tag{2.15}$$

$$\mathbf{h_t} = rnn_{one-to-many}(\mathbf{h_{t-1}}, \mathbf{y_{t-1}}) \tag{2.16}$$

$$\mathbf{\hat{y}_t} = softmax(\mathbf{V}\mathbf{h_t} + \mathbf{c}) \tag{2.17}$$

4. *Many-to-many with $\tau_x \neq \tau_y$*: This architecture is also called sequence-to-sequence architecture. The input and output length can differ as it is the case in machine translation. Although a single RNN can be used to accomplish this, Cho et al. [49] proposed an improved architecture that consists of two separate RNNs, namely an encoder and a decoder. An encoder first encodes the entire input sequence into a summary vector $c = \mathbf{h_{\tau_x}}$ that is the last hidden state of the encoder, and a decoder generates the output sequence using that summary vector. At each time step t ranging from 1 to τ_y, the decoder produces a new decoder hidden state $\mathbf{s_t}$ based on the previous

hidden state $\mathbf{s_{t-1}}$, the previous expected output $\mathbf{y_{t-1}}$, and the summary vector $\mathbf{c}$. The output $\mathbf{\hat{y}_t}$ is also computed using softmax:

$$\mathbf{s_t} = rnn_{decoder}(\mathbf{s_{t-1}}, \mathbf{y_{t-1}}, \mathbf{c}) \tag{2.18}$$

$$\mathbf{\hat{y}_t} = softmax(\mathbf{V}\mathbf{h_t} + \mathbf{c}) \tag{2.19}$$

2.2.1.2 Composing RNNs

Recurrent Neural Networks can be further composed in different ways to enable solving complex tasks [138].

Left-to-Right and Right-to-Left RNNs: As introduced above, an RNN processes the input data from left to right. However, for some tasks, it is better to process the data from right to left (e.g. for the Arabic language). Such RNNs are called left-to-right respective right-to-left RNNs [138].

Bidirectional RNNs: In tasks such as Named Entity Recognition (NER) [183] it is important to consider the whole context within a sentence to predict the class for a single word [138]. For instance, a left-to-right RNN might not recognise the person name "Mercedes Benz" in the sentence "Mercedes Benz is a nice woman" since "Mercedes Benz" mainly refers to an automotive brand. To address this issue, a Bidirectional Recurrent Neural Network (Bi-RNN) can be used that consists of a left-to-right and a right-to-left RNN. The output $\mathbf{\hat{y}_t}$ of a Bi-RNN is the concatenation of both hidden states at time step t as in Equation 2.14.

Stacked RNNs: Stacked RNNs consist of multiple RNNs where the output hidden states of an RNN serve as the input for a subsequent RNN [138]. Bi-RNNs can also be stacked in the same way. Stacking enables the model to learn representations at initial layers that can serve as useful abstractions for subsequent layers.

2.2.1.3 Training RNNs

In the following, we describe the loss function for RNNs and introduce gated RNNs.

Loss Function: Given the ground truth output sequence $(\mathbf{y_1}, ..., \mathbf{y_{\tau_y}})$ and the predicted sequence $(\mathbf{\hat{y}_1}, ..., \mathbf{\hat{y}_{\tau_y}})$ for a single training example, the loss function $L(.)$ of all time steps is defined based on the loss (e.g. cross-entropy) at every time step as follows [138]:

$$L((\mathbf{y_1}, ..., \mathbf{y_{\tau_y}}), (\mathbf{\hat{y}_1}, ..., \mathbf{\hat{y}_{\tau_y}})) = \frac{1}{\tau_y} \sum_{t=1}^{\tau_y} L(\mathbf{y_t}, \mathbf{\hat{y}_t}) \tag{2.20}$$

Thus, RNNs can be trained in the same way as neural networks (see Section 2.1.3).

Gated RNNs: An RNN, as introduced in Equation 2.12, is not used in this form in practice since during training, it suffers from *exploding gradients* and *vanishing gradients* (see Section 2.1.3.3), especially for long input or output sequences [138]. To resolve these problems, techniques presented in Section 2.1.3 can also be applied to RNNs. Additionally, the RNN types Gated Recurrent Unit (GRU) [50] and Long Short-Term Memory (LSTM) [116] address the vanishing gradient problem using "gates" that allow gradients to be back-propagated unchanged.

2.2.1.4 Prediction of the Output Sequence

For the prediction of the output sequence in a classification task, we are usually interested in the output sequence that has the highest conditional joint probability $P(\mathbf{y_1}, ..., \mathbf{y_{\tau_y}} | \mathbf{x_1}, ..., \mathbf{x_{\tau_x}})$. Depending on the output length τ_y, various approaches exist [138]:

- Case $\tau_y = 1$: When we have a single output vector, the class in $\mathbf{y_{\tau_x}}$ with the highest probability can be used as prediction.
- Case $\tau_y \geq 1$: When we have a sequence of output vectors, various strategies exist to generate the output sequence:
    1. Greedy approach: For each output step t, take in $\mathbf{y_t}$ the class with the highest probability. However, this strategy may not yield the "best" output sequence since the decision for the actual class at each output step is independent of the previous decisions.
    2. Viterbi algorithm: For the case $\tau_x = \tau_y$ and when the number of classes is quite low (e.g. in named entity recognition), a Conditional Random Field (CRF) [163] can be used to generate the optimal output sequence using the Vitebi algorithm [88]. A CRF is introduced in Section 4.3.1 in more detail.
    3. Beam search: For the case $\tau_x \neq \tau_y$ or when the number of output classes is high, the beam search strategy can be used to *estimate* the optimal output sequence [89]. It generates the output for each time step from left to right while keeping a fixed number (beam size) of active candidates at each time step. This enables the identification of the most promising output paths by their cumulative likelihood.

2.2.2 Transformer Architecture

RNNs can be used to realise many different NLP tasks and are widely used. However, due to their recurrent nature, RNNs process the input sequence sequentially so that the processing

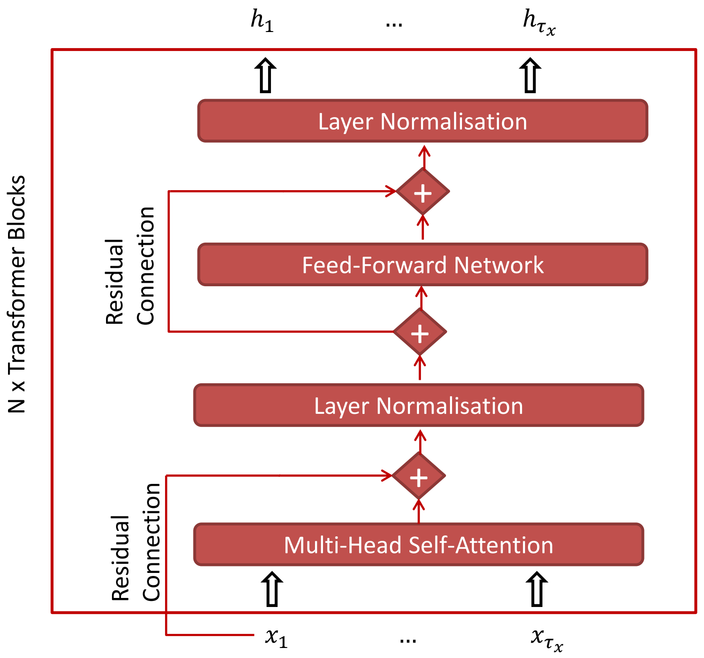


Figure 2.6: Illustration of a transformer block and its layers (based on [138]).

cannot be parallelised [138]. Furthermore, RNNs suffer from vanishing gradients due to recurrent connections, which makes them difficult to train [116].

To mitigate the problems of RNNs, Vaswani et al. [283] introduced the transformer architecture. As described in Jurafsky and Martin [138], a transformer encodes an input sequence $(\mathbf{x_1}, ..., \mathbf{x_{\tau_x}})$ to a sequence of representations $(\mathbf{h_1}, ..., \mathbf{h_{\tau_x}})$ of the same length. A transformer is a stack of N transformer blocks consisting of linear layers, feed-forward neural networks, residual connections [112], and layer normalisation layers [10]. Figure 2.6 illustrates a transformer block. However, the key innovation of transformers is the use of *self-attention* layers that allow the model to focus on important parts of the input sequence depending on the context without the need to process the input sequence sequentially [138]. This enables training parallelisation and allows the model to learn long-range dependencies within the input sequence. Therefore, using the transformer, models can be trained on larger datasets with less training time than with RNNs.

It should be mentioned, that the transformer architecture introduced by Vaswani et al. [283] has been proposed for sequence-to-sequence tasks (i.e. machine translation) and consists of two components, an encoder and a decoder. Similarly to RNNs, an encoder encodes the input sequence, and a decoder generates the output sequence (see Section 2.2.1.1). Both components are similar. However, since this thesis only relies on the encoder (see Section 2.3.4.2), we focus here on the description of the transformer encoder and refer to Vaswani et al. [283], who also describe the specifics of the decoder. In the following, we introduce the self-attention mechanism and positional encodings. The training and prediction

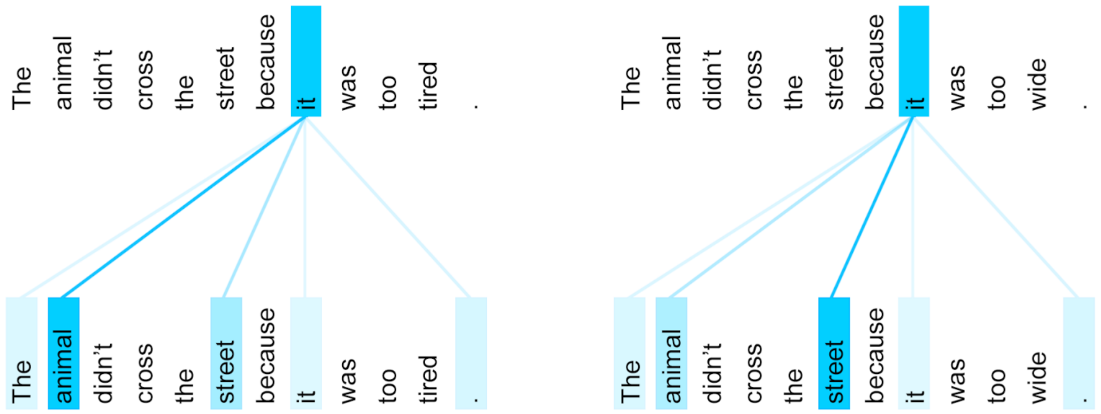


Figure 2.7: Illustration of the self-attention distributions of the weights for the word “it” in two different sentences. Illustration is from [280].

of transformer-based models can be performed similarly to RNNs (see Sections 2.2.1.3 and 2.2.1.4, respectively).

2.2.2.1 Self-Attention

Conceptually, self-attention computes a weighted sum over all input elements $(\mathbf{x_1}, ..., \mathbf{x_{\tau_x}})$ to obtain a hidden representation $\mathbf{h_i}$ for the i-th input element [138]:

$$\mathbf{h_i} = \sum_{j=1}^{\tau_x} \alpha_{i,j} \mathbf{x_j} \tag{2.21}$$

The weight $\alpha_{i,j} \in \mathbb{R}$ represents a comparison score between the i-th and the j-th input element, i.e. how relevant is the j-th element for the current element i [138]. The weights are computed using a score function (see below) whose parameters are learned during training. Figure 2.7 shows the weight distributions for the word “it” of a self-attention layer in two example sentences: In the first sentence, the word “animal” has a high weight for the pronoun “it” and in the second sentence, the word “street”.

Various approaches were proposed to calculate the weights $\alpha_{i,j}$ [283]. The transformer uses the *scaled dot-product attention* mechanism that is described below. A self-attention layer uses three weight matrices $\mathbf{W^Q}, \mathbf{W^K}$, and $\mathbf{W^V}$ whose parameters are learned during training. These weight matrices are used to compute linear transformations of the input values [138]:

$$\mathbf{q_i} = \mathbf{W^Q x_i}; \quad \mathbf{k_i} = \mathbf{W^K x_i}; \quad \mathbf{v_i} = \mathbf{W^V x_i} \tag{2.22}$$

The projections $\mathbf{q_i}, \mathbf{k_i}$, and $\mathbf{v_i}$ stand for *query*, *key*, and *value*. A *query* represents the current focus of attention i.e. the current element. The *key* projections are compared against the query using the dot-product to calculate unnormalised comparison scores between the query and the keys [138]:

$$score(\mathbf{x_i}, \mathbf{x_j}) = \frac{\mathbf{q_i} \cdot \mathbf{k_j}}{\sqrt{d}} \tag{2.23}$$

Here, d is the dimension of keys, values, and queries and the score is divided by the scaling factor $\sqrt{d}$ to avoid small gradients [283]. These scores are then normalised using softmax to obtain the final weights $\alpha_{i,j}$ [138]:

$$\alpha_{i,j} = \frac{\exp(score(\mathbf{x_i}, \mathbf{x_j}))}{\sum_{k=1}^{\tau_x} \exp(score(\mathbf{x_i}, \mathbf{x_k}))} \tag{2.24}$$

Instead of using the input representations directly as in Equation 2.21, the transformer uses the *value* projections to compute the weighted sum over the input sequence [138]:

$$\mathbf{h_i} = \sum_{j=1}^{\tau_x} \alpha_{i,j} \mathbf{v_j} \tag{2.25}$$

In a self-attention layer, in contrast to RNNs, each hidden representation $\mathbf{h_i}$ can be computed independently of all other hidden representations allowing to parallelise the computations via matrix multiplications [138]. Besides, the path length from the current element to all other elements - independent of its position - is always constant. Thus, the transformer can learn long-range dependencies between the input elements better than RNNs [138].

A transformer block employs multiple self-attention layers (called *multi-head attention layer*) that enable the model to learn different kinds of relationships between the words in a sentence (e.g. syntactic or semantic relationships) [138]. A self-attention layer in a multi-head attention layer is also called a *head*. The outputs of the heads are concatenated and projected with a linear layer to the dimension of the inputs. Such a projection to the original dimension of the input elements enables to utilise residual connections [112].

2.2.2.2 Positional Embedding

A self-attention layer cannot make use of the order of the input elements since it just computes a weighted sum over the input elements irrespective of their order. Therefore, the transformer adds positional embeddings to the input vectors that can capture the absolute and relative position of an input element [283]. A positional embedding can be computed with a static function that maps an integer to a vector. For this purpose, the transformer uses a combination of sine and cosine functions [283].

2.3 Text Representation using Word Embeddings

In the previous Section 2.2, we have introduced neural network architectures for sequence processing that can be applied to NLP tasks. However, neural networks can only process data encoded as vectors. Since this thesis deals with information extraction from scientific text, we need to encode text into vectors. This section first outlines the *preprocessing* of text (Section 2.3.1) and *one-hot encoding* for text representation (Section 2.3.2). The subsequent Section 2.3.3 and Section 2.3.4 present approaches for *word embeddings*. The basic idea of word embeddings is to represent words as low-dimensional dense vectors such that the similarity between vectors of words with similar meaning should be high, and low otherwise [138]. This allows machine learning models, for instance, to generalise well with less training data since points close to each other are more likely to share the same label. Word embeddings can be learned automatically from a large text corpus [138]. The assumption is that words occurring in similar contexts tend to have similar meanings, also known as the *distributional hypothesis* [109]. Word embeddings are a fundamental part of all proposed approaches in this thesis. The following descriptions of text representation approaches are based on the book of Jurafsky and Martin [138] unless otherwise stated.

2.3.1 Text Preprocessing

In the following, we introduce the tasks involved in text preprocessing.

Tokenisation: Tokenisation is the task of segmenting a text into a sequence of tokens [138]. A token usually represents a word or punctuation. The set of all unique tokens in a (large) text corpus is also called the vocabulary that is the basis for representing tokens as vectors. However, a fixed vocabulary of words has the drawback that out-of-vocabulary words may exist. These are words in a new text that were not present in the original text corpus used to build the vocabulary. To resolve this issue, the vocabulary can also be built upon single characters or subwords. Current state-of-the-art approaches for word representations utilise subwords as the basic unit of the vocabulary [37, 71], which can be determined automatically with algorithms such as byte-pair encoding [254], unigram language modelling [160], and word piece tokenisation [251]. Thus, subword-tokenisation enables a good balance between the flexibility of a character-based and a word-based vocabulary [138].

Text Normalisation: Text normalisation is the task of putting tokens into a standard format to help to reduce the vocabulary size [138]. This can include lemmatising words to determine the root of words (e.g. “ran” and “running” have the same root “run”) using dictionaries or stemming algorithms such as the Porter stemmer [220]. Another technique is the removal of stop words that represent unimportant words. However, text normalisation has the drawback of losing some information (e.g. verb tense in lemmatisation), which might be important in downstream tasks.

Sentence Segmentation: In sentence segmentation, the text is segmented into sentences by punctuation marks like periods, question marks, and exclamation points [138]. Since punctuation marks might be ambiguous (e.g. an abbreviation can also end with a period), rule-based or machine-learning approaches are used for this task [150].

2.3.2 One-hot Encoding of Text

One-hot encoding is a simple and effective approach to encode tokens as vectors. The assumption is that tokens of a fixed vocabulary represent categorical data in which each token has no ordinal relationship to other tokens [138]. Let $\mathbb{V}$ be the vocabulary, i.e. the set of tokens. The one-hot vector $onehot(w_i) \in \mathbb{R}^{|\mathbb{V}|}$ for a token $w_i \in \mathbb{V}$ is defined as a vector of dimension $|\mathbb{V}|$ (i.e. the number of words in the corpus) in which the i-th component equals 1 and all remaining components are 0. Thus, each one-hot vector is unique for each token. To encode a sequence of tokens (e.g. the text in an email) into a vector, the one-hot encoded token vectors of the sequence can be summed. Such a vector is also called the *bag of words*.

One-hot encoding is used to encode the *output text* in NLP tasks, such as in machine translation or text generation. However, using one-hot encoding to represent the *input text* has several drawbacks [138]. For instance, the vectors are sparse and high-dimensional, so that models with many parameters are required. Furthermore, all one-hot vectors are orthogonal to each other so that the vectors of semantically similar words (e.g. "bed" and "couch") are always dissimilar. Therefore, training models using one-hot encoding for input text requires a large amount of variant training data to enable the models to generalise sufficiently. In the subsequent sections, we introduce *word embeddings* that can capture the semantics of similar words better than one-hot encoded vectors and are therefore better suited to represent words as input values.

2.3.3 Static Word Embeddings

Static word embeddings are pre-trained models that can assign a vector (also known as *word embedding*) $\mathbf{e}$ to a single word x [138]. Mikolov et al. [193] propose a popular approach named *word2vec* that uses a neural network to learn static word embeddings from a large text corpus that acts as supervised training data. Two variants of word2vec exist [193]: (1) continuous bag of words and (2) the continuous skip-gram model. Both variants are similar and have various advantages and disadvantages, e.g. skip-gram can better represent rare words but is slower to train [193]. In the following, we present the continuous skip-gram model in more detail.

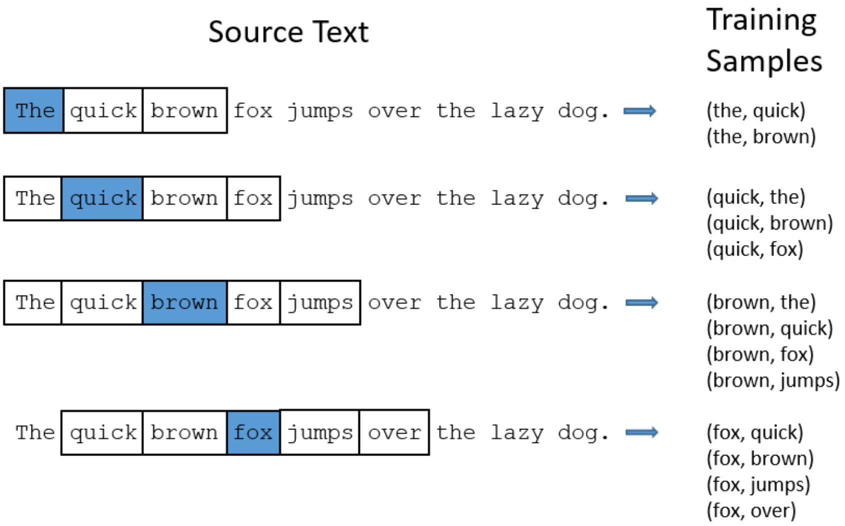


Figure 2.8: Illustration of the process of generating training samples from a source text with radius $r = 2$ for the skip-gram model of word2vec. A blue box represents the centre word and a white box the output word. Illustration is from [189].

Let $(w_1, ..., w_T)$ be the sequence of words of a large text corpus (e.g. all Wikipedia articles) and $\mathbb{V}$ the vocabulary. Given a centre word w_t, the objective of word2vec is to estimate the probability $P(w_{t+j}|w_t)$ of an output word w_{t+j} with $-r \leq j \leq r, j \neq 0$ occurring in the context of the centre word w_t with radius r. Thus, the objective is to maximise the following loss function [193]:

$$L = \frac{1}{T} \sum_{t=1}^{T} \sum_{-r \leq j \leq r, j \neq 0} \log P(w_{t+j}|w_t) \tag{2.26}$$

The model that estimates the probability distribution $P(w_{t+j}|w_t)$ can be learned with a neural network using a softmax classification layer. Figure 2.8 illustrates the process of generating training data consisting of (x, y) pairs from a source text where x denotes the centre word and y the expected output word.

Let $\mathbf{x}$ be the one-hot vector for the centre word x and $\mathbf{y}$ the one-hot vector for the expected output word y. Now, we define the following neural network [138]:

$$\mathbf{e} = \mathbf{E}\mathbf{x} \tag{2.27}$$

$$\mathbf{z} = \mathbf{O}\mathbf{e} \tag{2.28}$$

$$\hat{\mathbf{y}} = softmax(\mathbf{z}) \tag{2.29}$$

The matrices $\mathbf{E} \in \mathbb{R}^{d \times |\mathbb{V}|}, \mathbf{O} \in \mathbb{R}^{|\mathbb{V}| \times d}$ represent the weights of the model that can be learned with the gradient descent algorithm using the cross-entropy loss while d is the dimension of the word embeddings (e.g. $d = 300$). The matrix $\mathbf{E}$ contains the learned word embeddings and a word embedding $\mathbf{e}$ for a word $\mathbf{x}$ can be obtained via $\mathbf{e} = \mathbf{Ex}$.

The word embeddings of word2vec have some interesting properties [193]: For instance, the cosine similarity of similar words is usually high, e.g. $word2vec(Man) \approx word2vec(Boy)$. Furthermore, word2vec enables the calculation of analogies with simple vector operations, e.g. "Man is to Woman what King is to __" can be solved using the approximation $word2vec(Queen) \approx word2vec(King) - word2vec(Man) + word2vec(Woman)$.

However, word2vec is not able to handle words that were not present in the vocabulary during training (out-of-vocabulary words). Bojanowski et al. [24] proposed *fastText* as an extension of word2vec, where each word is represented by the sum of the vector representations of its character n-grams. Character n-grams are all subwords of size n for a given word. They enable fastText to better handle rare and out-of-vocabulary words.

2.3.4 Contextual Word Embeddings

Static word embeddings such as word2vec and fastText assign a pre-trained vector $\mathbf{e} = f(\mathbf{x})$ to a word $\mathbf{x}$ regardless of the word's context. Therefore, static word embeddings cannot capture word semantics in different contexts [138]. For instance, the word "bank" is polysemous such as in "I lend money from a *bank*" and "I am sitting on a *bank*", but the word "bank" is always represented with the same vector – regardless of its context. On the other hand, contextual word embeddings can consider the context of the words so that the word "bank" has different representations in both sentences. Thus, contextual word embeddings are functions of the form $(\mathbf{e_1}, ..., \mathbf{e}_\tau) = f(\mathbf{x_1}, ..., \mathbf{x}_\tau)$ [138].

In the following, we introduce contextual word embedding approaches that are based on language models and the popular state-of-the-art approach Bidirectional Encoder Representations from Transformers (BERT) [71].

2.3.4.1 Word Embeddings from Language Models

Many state-of-the-art approaches for contextual word embeddings are based on a *language model* [138]. A language model enables an estimation of the probability of a token sequence. This is relevant in many applications such as speech recognition or machine translation [186]. To train a language model, the model learns to predict the next token $\mathbf{x_t}$ for a given input sequence $(\mathbf{x_1}, ..., \mathbf{x_{t-1}})$ using a large text corpus (e.g. all Wikipedia articles). The transformer architecture and RNNs can be used to train such language models. For instance, Embeddings from Language Models (ELMo) [217] and Universal Language Model Fine-tuning (ULMFiT) [124] use Bi-RNNs to train a left-to-right (i.e. predict the next token) and a right-to-left (i.e. predict the previous token) language model. The more recent approach Generative Pre-training (GPT) [37] uses the transformer architecture to learn a left-to-right

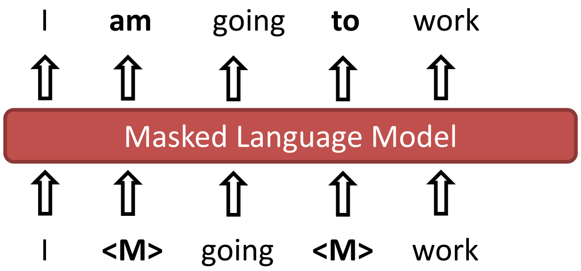


Figure 2.9: Illustration of *masked language modelling*. The <M> represents masked tokens that have to be predicted based on the context.

language model. To obtain contextual word embeddings from a trained language model, a given sentence $(\mathbf{x_1}, ..., \mathbf{x}_{T_\mathbf{x}})$ is processed via the model and the hidden representations $(\mathbf{h_1}, ..., \mathbf{h}_{T_\mathbf{x}})$ serve as the word representations. Since the meaning of a word depends on the surrounding words, language models can capture word semantics in different contexts better than static word embeddings [138].

2.3.4.2 Bidirectional Encoder Representations from Transformers

This section introduces Bidirectional Encoder Representations from Transformers (BERT) [71], which is a popular state-of-the-art approach for contextual word embeddings. Some variants of BERT are used in the proposed approaches of this thesis. First, we present the pre-training objectives of BERT. Then, we describe how BERT can be utilised for downstream tasks and some important variants of BERT.

Masked Language Modelling: The approach BERT is a language model trained using the transformer encoder (see Section 2.2.2). The key innovation in BERT is to use a *masked language modelling* training objective instead of learning to predict the next, respective, previous token as in standard language modelling [138]. Moreover, BERT uses wordpiece-tokenisation [251] to better handle rare and out-of-vocabulary words (see also Section 2.3.1). As illustrated in Figure 2.9, in masked language modelling, some tokens are randomly masked from the input, and the objective is to predict the masked tokens based only on their context. Unlike left-to-right or right-to-left language model training, where the representations are conditioned either on the left or the right context, the masked language modelling objective using the transformer architecture enables to learn representations that are jointly conditioned on both the left and the right context [71]. For instance, in the sentence “I went to the bank to sit down”, in a left-to-right or right-to-left language model, the word “bank” is represented either with “I went to the ...” or with "... to sit down". In BERT, the word “bank” is represented using both its previous and its next context: “I went to the ... to sit down” [138].

Next Sentence Prediction: Furthermore, BERT also employs the *next sentence prediction* task [71]. Given two sentences, the model has to predict whether both sentences are consecutive or not. This enables the model to better learn relationships between sentences. In the following, we show an example for two input sentences and the expected labels:

> **Sentence 1:** The men went to the $[MASK]_1$.
> **Sentence 2:** He bought a $[MASK]_2$ of milk.
> **Labels:** $[MASK]_1$=store; $[MASK]_2$=gallon; is next sentence=yes

Applying BERT to Downstream Tasks: Two approaches exist for applying the BERT model to downstream tasks, such as text classification or named entity recognition: *feature-based* and *fine-tuning* [71]. The *feature-based* approach uses task-specific architectures that include the word representations from BERT as additional features. In the *fine-tuning* approach, only a minimal set of task-specific parameters are introduced, and all parameters of the pre-trained BERT model are fine-tuned during the training of the downstream task. Although fine-tuning is simple and widely used, the approach has the disadvantage that all parameters of the BERT model have to be fine-tuned, which is resource-intensive since BERT has several hundred million parameters. In the feature-based approach, only the task-specific parameters have to be learned. However, the feature-based approach requires the design of custom architectures. In this thesis, we employ both variants, i.e. the feature-based approach for sequential sentence classification (Chapter 4) and scientific concept extraction (Chapter 5), and the fine-tuning approach for coreference resolution (Chapter 6).

Variants of BERT: Based on the BERT approach, several different models have been pre-trained for different domains and languages. For instance, the original BERT model has been pre-trained on the BooksCorpus [303] and on English Wikipedia, thus covering language of common discourse [71]. The model SciBERT [19], that is used in this thesis, has been pre-trained on scientific text. For German text, GBERT [44] can be used while M-BERT [71] has been pre-trained on monolingual corpora in 104 languages. Furthermore, various transformer-based language models were proposed that use modified training objectives. For instance, SpanBERT [136], which is used for coreference resolution in Chapter 6, can better represent text spans such as proper nouns that may consist of multiple words.

2.4 Knowledge Graphs

One of the objectives of this thesis is the automatic population of an Research Knowledge Graph (RKG). Thus, in this section, we introduce the foundations of KGs. First, we describe how KGs can be modelled and represented (Section 2.4.1). This is relevant background information for the requirements analysis in Chapter 3. Then, we outline the construction of KGs (Section 2.4.2), including ontology creation and KG population from unstructured text. The KG population approaches relevant for this thesis are based on NLP methods that were introduced in the previous sections. The following descriptions of KGs and their construction are mainly based on Hogan et al. [117] and Kejriwal et al. [144].

2.4.1 Modelling and Representing Knowledge Graphs

The term Knowledge Graph (KG) has become popular in the industry and the research community after the announcement of the Google Knowledge Graph in 2012 [261] that is the backbone of the Google Search Engine [117]. The core idea is to represent knowledge as a graph. However, the underlying concepts and technologies are based on the Semantic Web movement that envisions data on the Internet to be machine-interpretable [20]. There are multiple definitions of the term KG in the research community (see [117] for a discussion). In this thesis, we refer to the definition of Hogan et al. [117], who state that a KG consists of the following main components:

1. The *instance data* (also known as a graph of data) that is "*intended to accumulate and convey the knowledge of the real world, whose nodes represent entities of interest and whose edges represent relations between these entities*" [117].
2. An optional *ontology* (also known as schema) describing a conceptual model for a certain domain to prescribe the high-level structure for the instance data.
3. An optional component to infer new knowledge with deductive or inductive methods.

Although the definition of a KG by Hogan et al. [117] does not pose any prerequisites for the implementation of KGs, various technologies exist that support their operation. The Resource Description Framework (RDF) [232] is a standardised data model proposed by the World Wide Web Consortium (W3C) to represent information on the Web (see Section 2.4.1.1 for more details) structured and interlinked. Modelling languages such as RDF Schema (RDFS) [104] or Web Ontology Language (OWL) [159] enable the definition of ontologies (see Section 2.4.1.2). Multiple serialisation formats exist to represent an RDF-KG in a file or for transmission over a network, such as RDF/XML [249] that uses Extensible Markup Language (XML) [267], Turtle [224], a more compact serialisation format than RDF/XML, or JavaScript Object Notation for Linked Data (JSON-LD) [145]. Graph databases such as Neo4J [126] and GraphDB [127] enable to persist and query KGs efficiently. They also support graph query languages such as SPARQL Protocol and RDF Query Language (SPARQL) [108] to query a KG with graph patterns flexibly.

In the following, we describe the three components of KGs outlined above in more detail.

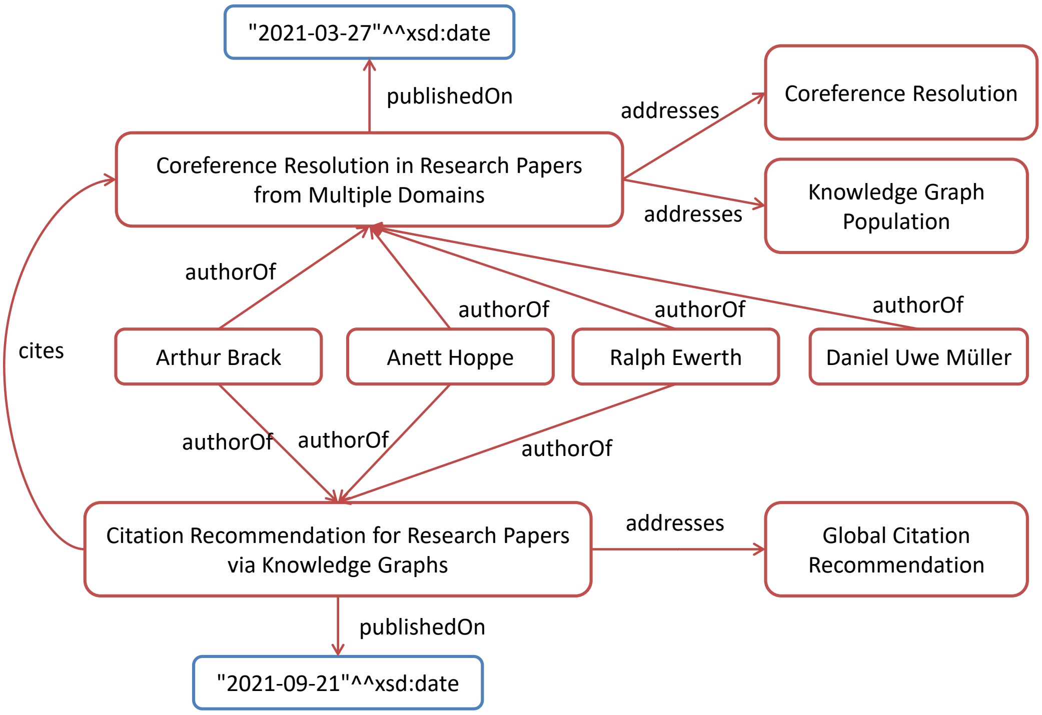


Figure 2.10: An example KG depicting two research papers [30, 33], the addressed tasks of the research papers, the citations, and the authors. The blue nodes are literals with their corresponding data type in RDF syntax. For better readability, the text of the nodes denotes human-readable labels (property `rdfs:label`), and not IRIs. For instance, the IRI of a research paper would be a DOI and the IRI of an author an ORCID (which stands for Open Researcher Contributor Identification Initiative).

2.4.1.1 Instance Data

A *directed edge-labelled graph* is the most popular formalism to describe the instance data of a KG [117]. The Resource Description Framework (RDF) [232] is based on this formalism. A directed edge-labelled graph is a tuple $\mathcal{G} = (\mathbb{V}, \mathbb{E}, \mathbb{L})$ where $\mathbb{V}$ is a set of nodes, $\mathbb{L}$ is a set of edge labels (also known as predicates, properties, or relation types), and $\mathbb{E} \subseteq \mathbb{V} \times \mathbb{L} \times \mathbb{V}$ is a set of edges [117]. An edge $(s, p, o) \in \mathbb{E}$ is also called a (subject (s), predicate (p), object (o)) triplet. For instance, the statement “Bob is married with Alice” can be formally described as $(Bob, marriedWith, Alice)$ where $Bob \in \mathbb{V}$ is the subject and $Alice \in \mathbb{V}$ the object in the statement, and $marriedWith \in \mathbb{L}$ is an edge label. Figure 2.10 depicts an example of a KG describing the relationships between two research papers, authors, and tasks.

The RDF data model consists of the following components [232]:

1. *Triples:* A triple describes a statement about resources i.e. it consists of a *subject*, *predicate*, and an *object* (see above). A resource can be anything including documents, people, physical objects, and abstract concepts.

2. *IRIs:* An International Resource Identifier (IRI) is a string that identifies a resource globally on the Web. For instance, the DOI `https://doi.org/10.1007/978-3-030-54956-5_1` is a unique IRI for the research paper [31]. Also, predicates are identified by an IRI.
3. *Literals:* Literals are basic values that are strings with an optional datatype (e.g. number, date) or language tag. Literals are used to describe the attributes of a resource (e.g. date of birth) or to provide human-readable labels in a specific language. Literals can only appear in the object position of a triple.
4. *Blank nodes:* Blank nodes are anonymous resources without an IRI. They are used to describe n-ary relationships between resources where a relationship is represented as a node without a globally unique IRI. For instance, a blank node can be defined for a "marriedWith" relation that can have further attributes such as the date or venue of the marriage. Since blank nodes do not have an IRI, external data sources cannot refer to them.

Identity links can be used to state that a resource has the same identity as another coreferent entity in a different source [232]. For instance, the IRI `https://api.semanticscholar.org/v1/paper/CorpusID:218763261` refers to the same paper [31] in the Semantic Scholar Open Research Corpus (S2ORC) [177]. The Web Ontology Language (OWL) [159] standard defines the `owl:sameAs` property to enable the definition of identity links.

2.4.1.2 Ontologies

Ontologies (or schemata) prescribe the high-level structure and semantics that instance data should or must follow. Two common schema types exist to describe an ontology [117]:

Semantic Schema: A *semantic schema* defines high-level terms (also known as vocabulary or terminology) such as classes and properties [117]. With the `type` predicate it is possible to define the semantics of nodes. For instance, the statement (Bob, type, Person) denotes that "Bob" is an instance of the class "Person". Furthermore, a semantic schema enables to define a hierarchy on classes (also known as *taxonomy*) and properties. For instance, the class "Student" can be defined as a subclass of "Person" so that all instances of "Student" are also instances of "Person". Besides, it is possible to define the domain and the range of properties where the domain indicates the class of the subject (i.e. the source node), and the domain the class of the object (i.e. the target node). Table 2.1 shows the definitions of these basic modelling features in a semantic schema. Standards such as RDF Schema (RDFS) [104] and Web Ontology Language (OWL) [159] are used to model a semantic schema which provide also further powerful modelling features.

Table 2.1: Definitions for `subclass`, `subproperty`, `domain`, and `range` features in a semantic schema [117].

Feature	Definition	Condition
`subclass`	$(c, subclass, d)$	$(x, type, c) \Rightarrow (x, type, d)$
`subproperty`	$(p, subproperty, q)$	$(x, p, y) \Rightarrow (x, q, y)$
`domain`	$(p, domain, c)$	$(x, p, y) \Rightarrow (x, type, c)$
`range`	$(p, range, d)$	$(x, p, y) \Rightarrow (y, type, d)$

Validating Schema: A *validating schema* allows for the validation of the instance data in a KG so that applications using the data can ensure that it contains the minimal information required [117]. For instance, we can define that a research paper must have a title, at least one author, and a publication date in a date format. Furthermore, we can state that the author of a research paper must be a "Person" rather than inferring it. The Shapes Constraint Language (SHACL) [153] can be used to model such constraints and class diagrams such as in Unified Modelling Language (UML) [26] enable to visualise a validating schema [117].

2.4.1.3 Deductive and Inductive Knowledge

Knowledge Graphs enable to infer new knowledge that is not explicitly modelled in the graph. Hogan et al. [117] distinguish between deductive and inductive knowledge. Although inferring new knowledge in KGs is not covered by this thesis, we briefly describe their approaches in the following.

Deductive Knowledge: Deductive knowledge *"is characterised by precise, logical consequences"* [117]. Semantic schema modelling features like those depicted in Table 2.1 (e.g. subclass, subproperty, domain, range) allow deducing new knowledge by following the rules. Furthermore, schema languages such as OWL [87, 159] enable to define custom inferencing rules that encode IF-THEN consequences as graph patterns. For instance, we can define a rule that the location of an event can be inferred from the location of the respective city [117].

Inductive Knowledge: While in deductive knowledge the conclusion from a given set of observations is certain, the conclusion might be imprecise in inductive knowledge [117]. For instance, it is possible to classify nodes in a graph (e.g. to predict the gender of a person in a social network), to predict new links between nodes (e.g. to suggest potential friends in a social network), or to cluster nodes in a graph (e.g. identify communities in a social network). Hogan et al. [117] and Kejriwal et al. [144] provide a comprehensive overview of popular inductive techniques for KGs.

2.4.2 Knowledge Graph Construction

The construction of a KG involves two phases: (1) *ontology creation* and (2) *population* with instances [144]. Both phases are described in the following. Approaches for ontology creation serve as background information for the requirements analysis in Chapter 3 while the KG population part is the basis for various methods proposed in this thesis.

2.4.2.1 Ontology Creation

An ontology for a KG is usually designed manually by humans. *Ontology engineering* methods, including ontology requirements engineering and ontology design patterns [117], can be used to develop meaningful ontologies that target specific domains and applications. An ontology can be designed by a closed group of experts or collaboratively, such as in Wikidata [285]. However, approaches also exist for *ontology learning* that is (semi-)automatic ontology creation from text [117, 216]. Ontology learning approaches can assist humans in an ontology design process, such as detecting new relation types or classes. For *taxonomy population*, i.e. populating a hierarchy of classes, Salatino et al. [244] provide an overview of methods based on rule-based NLP, clustering, and statistical methods.

2.4.2.2 Knowledge Graph Population

Nickel et al. [202] classify KG population methods into four groups:

1. *Curated approaches:* Triples are created manually by a closed group of experts.
2. *Collaborative approaches:* Triples are created manually by an open group of volunteers.
3. *Automated semi-structured approaches:* Triples are extracted automatically from semi-structured text via hand-crafted rules.
4. *Automated unstructured approaches:* Triples are extracted automatically from unstructured text.

In the following, we describe the fourth group, namely KG population from unstructured text in more detail since it is most relevant for this thesis.

KG population from unstructured text can be subdivided into two phases [117, 144, 226]: (a) *information extraction* to extract entities and their relations from text, and (b) *graph construction* to clean and complete the extracted graph, as it is usually ambiguous, incomplete, and inconsistent. Both phases and their involved tasks are depicted in Figure 2.11. In this thesis, we address the information extraction part. Therefore, in the following, the tasks involved in information extraction from unstructured text are described in more detail while we provide only a brief overview of the involved tasks in graph construction.

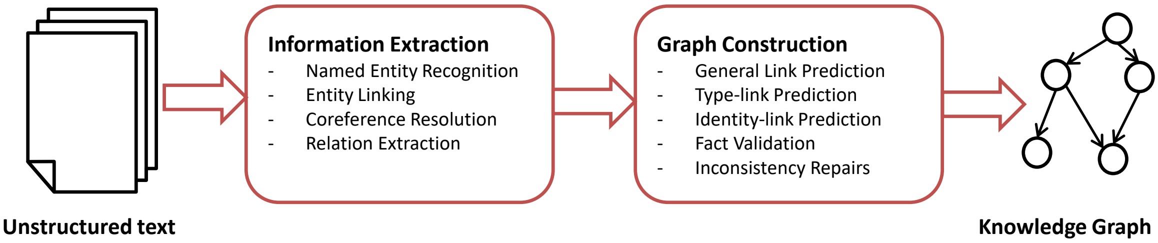


Figure 2.11: The KG population pipeline from unstructured text consisting of the two phases (1) *information extraction* and (2) *graph construction* and their involved tasks.

Information Extraction: Information extraction from text involves various tasks which are introduced in the following [117, 144]:

1. Named Entity Recognition (NER): In this task, the objective is to identify mentions of named entities in text such as people, organisations, or locations [144]. Current state-of-the-art approaches use pre-trained language models based on the transformer architecture (see Section 2.2.2) for this task [19, 71]. For a given set of classes $\mathbb{L}$ and a token sequence $(\mathbf{x_1}, ..., \mathbf{x}_\mathcal{T})$, the objective is to predict the corresponding label sequence $(\hat{\mathbf{y}}_\mathbf{1}, ..., \hat{\mathbf{y}}_\mathcal{T})$ with $\hat{\mathbf{y}}_\mathbf{i} \in L$ with the highest conditional joint probability $P(\hat{\mathbf{y}}_\mathbf{1}, ..., \hat{\mathbf{y}}_\mathcal{T} | \mathbf{x_1}, ..., \mathbf{x}_\mathcal{T})$.
2. *Entity Linking*: Entities identified by the NER task can be used as new candidate entities for a KG (also known as emerging entities) or linked to existing entities in a KG via entity linking approaches [144]. However, the recognised mentions are ambiguous (for instance, "apple" may refer to the fruit "apple" or to the company "Apple Inc.") and there are also various ways how an entity can be mentioned in a text. Therefore, entity linking considers a disambiguation phase, where mentions are associated with candidate nodes which are ranked using the context of the mentions in text and the context of the nodes in the KG [117, 154].
3. *Coreference Resolution*: The objective of coreference resolution is to identify mentions of an entity in a text which refer to the same entity. In particular, this task includes the resolution of pronouns to the referred entity [144]. For instance, the text "Bob is... He is married with Alice...", has two coreferent mentions "Bob" and "He" that refer to the same person "Bob". Current approaches for coreference resolution are usually ranking-based models [66, 170, 187] that simultaneously rank all candidate antecedents (i.e. preceding mention candidates) to identify the most probable antecedent.
4. *Relation Extraction:* The objective of relation extraction is to extract relations between entities from the text considering a fixed set of relation types [144]. For instance, we can extract the statement $(Bob, marriedWith, Alice)$ from the text *"Bob is... He is married with Alice..."*. In *binary relation extraction* [9, 92, 178], the model predicts a relation between two entities, such as between "Bob" and "Alice". However, in *n-ary*

relation extraction [129, 131, 142], the model extracts relationships between multiple entities. For instance, the *marriedWith* relationship can be represented as a node with further extracted information such as the date and the location of the marriage.

While the above tasks can be performed in sequence, various approaches exist that can *jointly* perform multiple tasks to improve their performance [178, 246, 290].

Graph Construction: The extracted knowledge from text is usually ambiguous, incomplete, and inconsistent. Graph completion and correction methods can complete and clean the extracted graph from text [117, 144]. This includes the prediction of additional links (i.e. edges or triples) between nodes, the identification of incorrect links, and repairing inconsistencies. In the following, we introduce the involved tasks in KG completion and correction while we refer to Hogan et al. [117] and Kejriwal et al. [144] for a more comprehensive overview about approaches for these tasks:

1. *General Link Prediction:* Predict missing general links between nodes.
2. *Type-link prediction:* Predict the `type` predicate between an entity and a class. This task is also known as *node classification* [144].
3. *Identity-link prediction:* Identify nodes which refer to the same entity, i.e. predict the `owl:sameAs` relation between nodes. This task is analogous to *entity resolution* [276] and *instance matching* [144], and related to *entity linking* (see above).
4. *Fact validation:* Assign a plausibility or veracity score for a link.
5. *Inconsistency repairs:* Repair introduced inconsistencies that violate rules or axioms defined in the ontology.

2.5 Evaluation Methods

This section describes the evaluation procedures and evaluation metrics that are used in this thesis. First, Section 2.5.1 introduces the most common evaluation procedures. Then, Section 2.5.2 presents basic evaluation metrics of machine learning models in classification tasks and specific metrics for the task of named entity recognition, coreference resolution, and information retrieval that are addressed in this thesis. Finally, Section 2.5.3 describes metrics to measure the inter-rater agreement.

2.5.1 Evaluation Procedures

As described in Section 2.1.3, during the training of a neural network model, a training set is is used to determine the optimal weights via gradient descent, whereas a validation set is required for hyperparameter tuning. In the following, we describe the evaluation procedure of machine learning models using a test set and cross-validation.

Test Set: To evaluate the performance of the final model (i.e. the one that performs best on the validation set), we need an additional set of annotated examples, namely the *test set*. A test set is independent of the training and validation set to enable unbiased evaluation but should follow the same distribution [138].

Cross-Validation: When the test set is small, it might not be representative enough to obtain robust evaluation results. Therefore, *k-fold cross-validation* can be applied to mitigate this issue [138]. In k-fold cross-validation, the annotated dataset is partitioned into k disjoint subsets (e.g. $k = 10$). Then, for each subset i $(1 \leq i \leq k)$ a model is trained using the data of the other subsets and the model performance is tested on the subset i. The test set performance for each subset are then summed up or averaged to obtain the final evaluation result.

2.5.2 Evaluation Metrics

In this section, we first describe basic evaluation metrics such as accuracy, precision, recall, and F1. In the subsequent subsections we introduce specific evaluation metrics for the task of named entity recognition, coreference resolution, and information retrieval that are used in the Chapter 5, Chapter 6, and Chapter 7, respectively.

2.5.2.1 Basic Evaluation Metrics

The metric *accuracy* measures the proportion of correctly predicted examples [138]:

$$accuracy = \frac{\sum_{i=1}^{m} \mathbb{1}(\mathbf{\hat{y}^{(i)}} = \mathbf{y^{(i)}})}{m} \tag{2.30}$$

Here, m is number of examples, $\mathbf{\hat{y}^{(i)}}$ is the predicted and $\mathbf{y^{(i)}}$ the expected output for the i-th example, and $\mathbb{1}(p)$ is the indicator function that returns 1 if the predicate p is true and 0 otherwise.

However, the accuracy metric is not appropriate when the dataset is unbalanced. For instance, in the spam email classification example, 95% of all examples may not be spam and a classifier that always classifies emails as not being spam would already have a high accuracy of 95%, but not for the intended purpose. Therefore, the metrics precision, recall, and F1 are often used to evaluate the performance of classification tasks. To calculate these metrics, we first need to define the terms *true positives (tp)*, *false positives (fp)*, *false negatives (fn)*, and *true negatives (tn)*. In a binary classification task, these terms are defined as follows [138]:

$$tp = \sum_{i=1}^{m} \mathbb{1}(\mathbf{y^{(i)}} = 1 \wedge \mathbf{\hat{y}^{(i)}} = 1); \quad fp = \sum_{i=1}^{m} \mathbb{1}(\mathbf{y^{(i)}} = 0 \wedge \mathbf{\hat{y}^{(i)}} = 1) \tag{2.31}$$

$$fn = \sum_{i=1}^{m} \mathbb{1}(\mathbf{y^{(i)}} = 1 \wedge \mathbf{\hat{y}^{(i)}} = 0); \quad tn = \sum_{i=1}^{m} \mathbb{1}(\mathbf{y^{(i)}} = 0 \wedge \mathbf{\hat{y}^{(i)}} = 0) \tag{2.32}$$

For instance, true positives are those emails that are annotated as spam in the dataset and also detected as spam by the model. Using these scores, we can now define precision and recall. Precision measures the proportion of positive identifications that are actually correct, and recall measures the proportion of actual positives that were identified correctly [138]:

$$precision = \frac{tp}{tp + fp}; \quad recall = \frac{tp}{tp + fn} \tag{2.33}$$

In our above example of spam email classification, a model that always returns "not spam" would have a precision and recall of 0% (if spam is the "positive" class to be detected). To incorporate precision and recall into a single metric, the F1 score represents the harmonic mean of both measures [138]:

$$\mathrm{F1} = 2 \times \frac{precision \times recall}{precision + recall} \tag{2.34}$$

In a multiclass classification task with $k > 2$ classes, we first compute the scores for tp_i, fp_i, fn_i, tn_i, $precision_i$, $recall_i$, $\mathrm{F1}_i$ for each class i as described above. Then, we need to aggregate these scores into a single metric. Three common approaches exist to calculate the F1 score in a multiclass classification setting [105]: (1) *macro-averaged* F1, (2) *micro-averaged* F1, and (3) *weighted* F1.

The *macro-averaged* F1 is an arithmetic mean over the F1 scores of each class so that all classes are weighted equally [105]:

$$\textit{macro-averaged}\ \mathrm{F1} = \frac{\sum_{i=1}^{k} \mathrm{F1}_i}{k} \tag{2.35}$$

In the *micro-averaged* F1, there is no weighting of the classes so that larger classes dominate. For this purpose, first, the scores tp_i, fp_i, and fn_i are summed [105]:

$$tp = \sum_{i=1}^{k} tp_i; \quad fp = \sum_{i=1}^{k} fp_i; \quad fn = \sum_{i=1}^{k} fn_i \tag{2.36}$$

Then, using these scores, we calculate precision and recall as described above and can obtain the *micro-averaged* F1 using the formula in Equation 2.34 [105].

In the *weighted* F1 score, each class is weighted by the number of examples m_i annotated with the class i so that, comparable to the *micro-averaged* F1, larger classes dominate [105]:

$$\textit{weighted}\ \mathrm{F1} = \frac{\sum_{i=1}^{k} m_i \mathrm{F1}_i}{m} \tag{2.37}$$

2.5.2.2 Evaluation Metrics for Named Entity Recognition

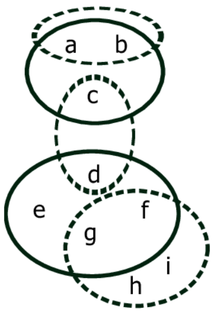
The standard metrics precision, recall, and F1 are used for the evaluation of an NER model [138]. To compute these metrics, we need the set of ground truth mentions and the set of mentions found by the NER model. Since a mention usually represents a span of multiple consecutive words (e.g. the location "New York"), it is represented uniquely as a tuple $(\mathit{begin}, \mathit{end}, \mathit{class})$ consisting of the begin and end position of the mention in the text and the corresponding class. Now, precision is the ratio of mentions found by the NER model that are correct, and recall is the ratio of mentions present in the corpus that are found by the NER model. The F1 is the harmonic mean of precision and recall as in Equation 2.34. A found mention is correct only if it is an exact match of the ground truth mention [245].

Architectures for sequence processing introduced in Section 2.2 can only classify single tokens or words. Therefore, tagging schemes are used that encode the beginning and the end of spans [183]. In the IOB tagging scheme (short for inside, outside, beginning), a token is additionally classified as a "B" (begin) if it denotes the first token of a mention, as an "I" (inside) if it is a token inside a mention, and as an "O" (outside or other) if the token is not a mention. Thus, the location "New York" can be encoded with the tagging sequence (B-Location, I-Location).

2.5.2.3 Evaluation Metrics for Coreference Resolution

The task of coreference resolution is to extract mentions of entities and cluster those mentions that refer to the same entity. Thus, for evaluation, the ground truth clusters (also called *key clusters*) have to be compared against the predicted clusters (also called *response clusters*) [181]. Figure 2.12 shows an example of ground truth and predicted clusters.

A cluster (also called entity) is a *coreference chain* that is a sequence of mentions sorted according to their occurrence in the text [181]. For instance, the text "Alice is a friend of Bob. She knows him for a long time since he is her neighbour." has two coreference chains: ($\mathit{Alice} \leftarrow \mathit{she} \leftarrow \mathit{her}$) and ($\mathit{Bob} \leftarrow \mathit{him} \leftarrow \mathit{he}$). A path from a mention to a preceding mention (called antecedent) within a coreference chain is called a *coreference link*. For instance, the chain ($\mathit{Alice} \leftarrow \mathit{she} \leftarrow \mathit{her}$) has three links: ($\mathit{Alice} \leftarrow \mathit{she}$), ($\mathit{she} \leftarrow \mathit{her}$), and ($\mathit{Alice} \leftarrow \mathit{her}$).



$$K = \overbrace{\{a, b, c\}}^{K_1}\,\overbrace{\{d, e, f, g\}}^{K_2}$$

$$R = \overbrace{\{a, b\}}^{R_1}\,\overbrace{\{c, d\}}^{R_2}\,\overbrace{\{f, g, h, i\}}^{R_3}$$

Figure 2.12: Example of two key clusters K_1 and K_2 (solid lines), and three response clusters R_1, R_2, and R_3 (dashed lines). The letters a to i represent mentions. The illustration is from [221].

There are various methods for making the comparison between the ground truth and the predicted clusters. The three popular metrics reported for coreference resolution models are *MUC* [284] that was proposed for the Message Understanding Conference, B^3 [12], and $CEAFe_{\phi 4}$ [180] that stands for Constrained Entity-Aligned F-Measure. Each of them represents different evaluation aspects [67, 118, 181, 194] and reports a precision, recall, and F1 score. To aggregate these metrics, the *CoNLL* precision, recall, and F1 scores are the arithmetic means of MUC's, B^3's and $CEAFe_{\phi 4}$'s respective precision, recall, and F1 scores. The CoNLL metrics were proposed for the conference on Computational Natural Language Learning (CoNLL) shared tasks on coreference resolution [221]. Pradhan et al. [221] provide a reference implementation for them. In the following, we introduce the computation of precision and recall of each metric. The F1 score is then the harmonic mean of precision and recall as defined in Equation 2.34. For a more comprehensive discussion and description of these metrics, we refer to Luo and Pradhan [181].

MUC: The *MUC* [284] score is based on coreference links and considers the minimum set of links that are required to connect all mentions in a cluster [181]. Thus, a cluster C has $|C| - 1$ coreference links. Precision is the number of correctly predicted links divided by the number of predicted links, and recall is the number of correctly predicted links divided by the number of ground truth links. However, the *MUC* metric favours models that produce fewer clusters and cannot cope with singleton clusters since they do not have links [180].

B^3: The B^3 [12] metric is based on mentions rather than links and addresses the issues of the *MUC* metric. To compute the recall, first, a score is calculated for each ground truth mention. The score equals the number of correctly predicted mentions in the predicted cluster containing the mention, divided by the number of mentions in the corresponding ground truth cluster. The recall is then the sum over these scores normalised by the number of ground truth clusters. The precision is computed in the same way, except switching the role of the ground truth and predicted mentions [221]. However, B^3 also has some issues, e.g. a cluster is used multiple times during the calculation of the scores for a mention [180].

$CEAFe_{\phi4}$: The $CEAFe_{\phi4}$ [180] metric is based on clusters rather than mentions or links and tries to mitigate some issues of B^3 and MUC metrics. First, we need to define a similarity metric between a ground truth (key) and a predicted cluster (response). Luo [180] proposes different metrics for that. The most popular similarity metric is $\phi_4(K, R)$ that measures how many common mentions a key (K) and a response (R) cluster share:

$$\phi_4(K, R) = \frac{2 \times |K \cap R|}{|K| + |R|} \tag{2.38}$$

Using this similarity metric, we can compute an optimal alignment mapping between the response and the key clusters such that the sum of the similarity of all aligned pairs is maximised. For instance, in the example of Figure 2.12, R_1 aligns with K_1, R_3 with K_2, and R_2 remains unaligned. The alignment problem under the constraint that a cluster can be aligned at most once corresponds to a bipartite matching problem and can be solved efficiently. We refer to Luo [180] for a description of the alignment algorithm. Given the optimal alignment mapping, we can now compute precision and recall. The precision is the fraction of the sum of the similarity scores between the aligned pairs and the number of predicted clusters. Thus, the precision of a model is penalised if it returns too many clusters. The recall is the fraction of the sum of the similarity scores between the aligned pairs and the number of ground truth clusters. Thus, the recall of a model is penalised if it returns too few clusters. However, $CEAFe_{\phi4}$ is not reliable for recall-precision analysis [194].

2.5.2.4 Evaluation Metrics for Information Retrieval

An information retrieval system such as a search engine yields a ranked list of items (e.g. found websites) relevant to a query (e.g. the search query). The objective is that the top-ranked items are most relevant for the user. Various metrics have been proposed to evaluate the effectiveness of an information retrieval system, such as Mean Average Precision (MAP) and Normalized Discounted Cumulative Gain (NDCG) [185]. The NDCG score is designed for non-binary notions of relevance. In the following, we describe MAP [162] in more detail since it is a widely used metric in information retrieval tasks and utilised in Chapter 7. The metric assumes that a user is interested in finding many relevant items.

Let $\{(q_1, \{d_{1,1}, ..., d_{1,m_1}\}), ..., (q_n, \{d_{n,1}, ..., d_{n,m_n}\})\}$ be a test set collection consisting of queries q_i and the set of relevant items $\{d_{i,1}, ..., d_{i,m_i}\}$ for the respective query. First, we need to define the notion of $Precision@k(q_i)$ that is the fraction of relevant items among the top k retrieved items for the query q_i. Then, for each query q_i, we calculate the Average Precision@k ($AP@k(q_i)$) as the average of $Precision@k$ values ranging from 1 to k if the retrieved item is available at position k [162]:

$$AP@k(q_i) = \frac{\sum_{k'=1}^{k} Precision@k'(q_i) \times rel(k')}{m_i} \tag{2.39}$$

Here, $rel(k)$ equals 1 if the item at position k is relevant, 0 otherwise. Thus, $AP@k$ penalises systems that are not able to rank the retrieved items such that the top-ranked items are relevant. The value $AP@k$ is 1 if all relevant items are ranked first, and 0 if all top k retrieved items are not relevant. Now, $MAP@k$ is the arithmetic mean of the $AP@k$ scores over the queries $\{q_1, ..., q_n\}$ [162]:

$$MAP@k = \frac{1}{n}\sum_{i=1}^{n} AP@k(q_i) \tag{2.40}$$

2.5.3 Inter-Rater Agreement

The annotation of datasets for a specific task is an essential prerequisite to train and evaluate machine learning models. However, the annotation of datasets by humans can be biased by their interpretation [227]. To limit the scope for interpretation, annotation guidelines should be developed and refined during the annotation process. They contain clear instructions for the annotators, such as the description of the classes and annotation examples [227].

Inter-rater agreement (also known as inter-annotator agreement) metrics can be used to evaluate the consistency of annotation results across multiple annotators and thus also the adequacy and quality of the annotation guidelines. These metrics do not measure only the actual agreement between the annotators but also consider the fact that agreement may happen solely by chance. The most used metrics are Cohen's Kappa [57], Fleiss's Kappa [86], and Krippendorff's Alpha [157]. We refer to Artstein and Poesio [4] for a discussion of these metrics in computational linguistics. In the following, we describe Cohen's Kappa in more detail since it applies to most annotation tasks in NLP [227]. Furthermore, we use this metric in the Chapters 5 and 6.

Cohen's Kappa (κ) [57] measures the inter-rater agreement between two annotators for categorical items. It is calculated using the observed agreement p_o and the expected hypothetical agreement by chance p_e as follows [4]:

$$\kappa = \frac{p_o - p_e}{1 - p_e} \tag{2.41}$$

The value $1 - p_e$ measures how much agreement above chance is attainable while the value $p_o - p_e$ tells us the agreement beyond chance actually found [4]. The observed agreement p_o is the proportion of items on which two annotators agree:

$$p_o = \frac{\#\text{agreements}}{n} \tag{2.42}$$

Table 2.2: The level of agreement for interpreting Cohen's Kappa (κ) score [164].

κ	Agreement level
≤ 0	poor
0.01-0.20	slight
0.21-0.40	fair
0.41-0.60	moderate
0.61-0.80	substantial
0.81-1.00	perfect

Here, n is the total number of items and #agreements is the number of items where the annotators were consistent with their annotations. The expected agreement by chance p_e is calculated as follows [4]:

$$p_e = \frac{1}{n^2} \sum_{k \in \mathbb{C}} n_{k_1} n_{k_2} \tag{2.43}$$

Here, $\mathbb{C}$ is the set of classes, and n_{k_1} and n_{k_2} denote the number of items annotator 1 respectively annotator 2 selected class k. Thus, assuming that the annotations of the annotators are independent, the term $\frac{n_{k_1} n_{k_2}}{n^2}$ estimates the joint probability that both annotators has classified the same item with class k.

The κ score typically yields a result in the interval $[0, 1]$. If the result is negative, then both annotators disagree, i.e. they are worse than randomly annotating the items. Landis and Koch [164] provide an interpretation of the κ score that is shown in Table 2.2.

2.6 Summary

Since this thesis deals with information extraction from research papers to populate an Research Knowledge Graph (RKG), this chapter presented an overview of the state of the art for Natural Language Processing (NLP) and KGs, as well as related evaluation methods.

Current state-of-the-art approaches for information extraction are based on neural networks. Therefore, we first introduced the basics of artificial neural networks and architectures for sequence processing such as RNNs and the transformer architecture that enable the application of neural networks on text data. Furthermore, we described text representation approaches using static and contextual word embeddings. In particular, contextual word embeddings based on the transformer architecture such as BERT are fundamental for current state-of-the-art approaches for various NLP tasks. Then, we introduced technologies for modelling and representing KGs and provided an overview of the involved tasks and methods to construct a KG. State-of-the-art approaches for KG population from unstructured text are based on neural networks. Finally, we described evaluation methods and metrics that are used to evaluate the proposed approaches of this thesis.

◊◊◊

This chapter presented the foundations for NLP, KGs, and evaluation methods. In the following chapters, we present the contributions of this thesis. First, we begin with a requirements analysis for an Open Research Knowledge Graph (ORKG).

3 Requirements Analysis for an Open Research Knowledge Graph

This chapter presents a requirements analysis that addresses the first research question introduced in Section 1.2.1, namely:

> **RQ1:** *What are the main requirements for scholarly knowledge representation to support various use cases in an RKG?*

The requirements analysis is conducted for an ORKG [7] as an example for an RKG. An ORKG aims to represent scholarly knowledge in a structured and interlinked manner and offer applications to support various use cases.

In the following, Section 3.1 introduces the requirements analysis. Section 3.2 summarises related work on RKGs, scientific ontologies, KG population, data quality requirements, and systematic literature reviews. The requirements analysis is presented in Section 3.3. It includes the identified use cases that should be supported by an ORKG and data quality requirements for the underlying ontologies and instance data. Based on the identified use cases and requirements, Section 3.4 discusses implications and possible approaches for ORKG construction. Finally, Section 3.5 summarises this chapter.

3.1 Introduction

As motivated in Chapter 1, fundamental contents of research papers such as addressed research problems, applied or proposed methods, and obtained results are not machine-interpretable. However, representing scholarly knowledge in an RKG structured and interlinked has the potential to enhance some of the researchers' core tasks.

Section 1.1 outlined various available infrastructures in the research ecosystem that already use KGs to enhance their services, and some initiatives also promote the usage of KGs in scholarly communication. However, current proposals usually focus on a specific use case (e.g. enhancing academic search [78], reproducing research results [184]), although the researcher's work is manifold. This chapter presents a detailed analysis of common literature-related tasks in a scientist's daily life and investigates how an ORKG could support them. Our requirements analysis aims to obtain a common understanding of the objectives of an

ORKG to guide the research and development. For this purpose, our analysis concentrates on the following aspects:

1. To elicit use cases that an ORKG should support.
2. To identify required user and machine interfaces, and the interdependence with external systems.
3. To define data quality requirements for the underlying KG to support these use cases:
    3.1. The required granularity of information representation and degree of domain-specialisation for the ontologies.
    3.2. The required degree of completeness and correctness of the instance data.
4. To elaborate construction strategies (human vs. machine) to populate an ORKG.

3.2 Related Work

This section gives a brief overview of (a) existing RKGs, (b) ontologies for scholarly knowledge, (c) approaches for KG population in the research ecosystem, (d) quality dimensions of KGs, and (e) processes in systematic literature reviews.

3.2.1 Research Knowledge Graphs

As stated in Section 2.4.1, the research community lacks a common definition of the term Knowledge Graph (KG). In this thesis, we refer to the broad definition of Hogan et al. [117] who state that a KG represents entities of interest and their relationships as a labelled graph of nodes and edges. The data in the KG usually conform to a data model so that machines can understand and reason over the represented knowledge. In the research ecosystem, examples for entities of interest are nodes that represent research papers, artefacts (e.g. source code, datasets), scientific concepts (e.g. tasks, methods, metrics), authors, or venues. Examples for relationships of interest are structured machine-readable links such as citations between the papers, associations between a paper and the corresponding source code or an addressed task, or relationships between concepts. Thus, we refer to platforms that represent such kind of knowledge with structured and interlinked content as RKGs.

Academic search engines (e.g. Google Scholar, Microsoft Academic, Semantic Scholar) exploit graph structures such as the Microsoft Academic Knowledge Graph [78], SciGraph [296], the Literature Graph [3], or the Semantic Scholar Open Research Corpus (S2ORC) [177]. These graphs interlink research articles through metadata, e.g. citations, authors, affiliations, grants, journals, or keywords.

To help reproduce research results, initiatives such as Research Graph [5], Research Objects [17], and OpenAIRE [184] interlink research articles with research artefacts such as datasets, source code, software, and video presentations. Scholarly Link Exchange (Scholix) [38] aims to create a standardised ecosystem to collect and exchange links between research artefacts and literature.

Some approaches connect articles at a more semantic level: The ORKG [130] (this thesis is part of this project) aims to organise the communicated content of research papers in a structured and interlinked manner. Papers With Code [206] is a community-driven effort to supplement machine learning articles with source code, tasks, datasets, metrics, and evaluation results to construct leaderboards. The Gene Ontology [59] and Chemical Entities of Biological Interest (CheBi) [65] are KGs for genes and molecular entities. Ammar et al. [3] link entity mentions in abstracts with DBpedia [172] and Unified Medical Language System (UMLS) [23]. Cohan et al. [53] and Jurgens et al. [139] extend the citation graph with citation intents (e.g. citation as background or used method) from computer science, medicine, and computational linguistics.

Various RKGs have also been populated automatically. The Computer Science Ontology (CSO) is a taxonomy of computer science research fields [244]. The AI-KG was automatically generated from 330,000 research papers in the artificial intelligence (AI) domain [70]. It contains five entity types (tasks, methods, metrics, materials, others) linked by 27 relation types. Kannan et al. [141] create a multimodal KG for deep learning papers from text and images and the corresponding source code while Färber and Lamprecht [91] populate a Data Set Knowledge Graph (DSKG) that links research papers with datasets. Zhang et al. [301] employ a rule-based approach to mine research problems and proposed solutions from research papers. The COVID-19 KG [47] has been populated from the Covid-19 Open Research Dataset [288] and contains various biological concept entities.

Various scholarly applications benefit from semantic content representation, e.g. academic search engines by exploiting general-purpose KGs [294], and graph-based research paper recommendation systems [18] that utilise citation graphs and mentioned entities.

3.2.2 Scientific Ontologies

Various ontologies have been proposed to model metadata such as bibliographic resources and citations [214]. Ruiz-Iniesta and Corcho [239] reviewed ontologies to describe scholarly articles. In the following, we describe some ontologies that conceptualise the semantic content in research articles.

Several ontologies focus on rhetorical [60, 103, 286] (e.g. background, methods, results, conclusion), argumentative [175, 279] (e.g. claims, contrastive and comparative statements about other work), or activity-based structure [215] (e.g. sequence of research activities) of research articles. Others describe scholarly knowledge with linked entities such as problem, method, theory, statement [36, 110], or focus on the main research findings and characteristics of research articles described in surveys with concepts such as problems, approaches, implementations, and evaluations [81, 281].

Various domain-specific ontologies exist, for instance, for mathematics [165] (e.g. definitions, assertions, proofs), machine learning [152, 191] (e.g. dataset, metric, model, experiment), and physics [247] (e.g. formation, model, observation). The EXPeriments Ontology (EXPO) is a core ontology for scientific experiments that conceptualises experimental design, methodology, and results [264], while the Scientific Observation Model (CRMsci) is an ontology of metadata about scientific observations, processed data, and measurements in descriptive and empirical sciences (e.g. biodiversity, geology, geography, archaeology) [73]. Various repositories provide access to several ontologies such as Open Biological and Biomedical Ontologies (OBO) Foundry [262] for the domain of life sciences or Linked Open Vocabularies [282] for Web data.

Taxonomies for domain-specific research areas support the characterisation and exploration of a research field. Salatino et al. [244] give an overview, e.g. Medical Subject Heading (MeSH), Physics Subject Headings (PhySH), and Computer Science Ontology (CSO).

3.2.3 Knowledge Graph Population

Section 2.4.2 introduced the involved tasks in KG construction, i.e. ontology creation and KG population. This section provides an overview of related work for manual and automatic KG population approaches in the research ecosystem. For a more general overview of ontology population systems, we refer to the review of Lubani et al. [179].

3.2.3.1 Manual Approaches

Wikidata [285] is one of the most popular KGs with semantically structured, encyclopaedic knowledge curated manually by a community. As of December 2021, Wikidata comprises over 96 million entities curated by almost 24.000 active contributors[2]. The community also maintains a taxonomy of categories and "infoboxes" which define common properties of certain entity types. Furthermore, Papers With Code [206] is a community-driven effort to interlink machine learning articles with tasks, source code, and evaluation results. Knowledge Graphs such as Gene Ontology [59], CheBi [65], or Wordnet [82] are curated by domain experts, and research article submission portals such as EasyChair (`https://wwww.easychair.org/`) enforce the authors to provide machine-readable metadata. Moreover, librarians and publishers tag new articles with keywords and subjects [296]. Furthermore, virtual research environments enable the execution of data analysis on interoperable infrastructure and store the data and results in KGs [272].

[2] `https://www.wikidata.org/wiki/Wikidata:Statistics`

3.2.3.2 Automatic Approaches

Nasar et al. [197] survey methods on information extraction from scientific text. Beltagy et al. [19] present benchmarks for several scientific datasets and Peng et al. [211] especially for the biomedical domain. Table 3.1, Table 3.3, and Table 3.2 show comparative overviews for selected datasets from research papers of various disciplines for the tasks of *sentence classification*, *relation extraction*, and *concept extraction*, respectively. These comparisons aim to provide an overview of the current state of the art on information extraction methods in scientific texts to get a sense about which kind of information can be extracted with which accuracy. In the following, we summarise these approaches.

As shown in Table 3.1, there are datasets that are annotated at *sentence level* for several domains, e.g. biomedical [68, 147], computer graphics [85], computer science [54], chemistry and computational linguistics [279], or algorithmic metadata [243]. They cover either only abstracts [54, 62, 68, 100, 147, 270] or full articles [85, 175, 243]. The datasets differentiate between five and eleven concept classes (e.g. Background, Objective, Results). Machine learning approaches for datasets consisting of abstracts achieve an F1 score ranging from 66.0% to 92.9% and for datasets with full papers F1 scores range from 51.6% to 78.5%.

More recent corpora shown in Table 3.2 and Table 3.3, that are annotated at *phrasal level* (e.g. noun phrases), aim at constructing a fine-grained KG from scholarly abstracts with the tasks of concept extraction [9, 74, 90, 173, 178, 229, 230], binary relation extraction [9, 92, 156, 178, 230], n-ary relation extraction [129, 131, 142], and coreference resolution [42, 58, 178, 248]. They cover several domains, e.g. material sciences [90]; computational linguistics [92, 229]; computer science, material sciences, and physics [9]; machine learning [178]; or biomedicine [58, 131, 156]. The datasets differentiate between four to seven concept classes (like Task, Method, Tool) and between two to seven binary relation types (like used-for, part-of, evaluate-for). The extraction of n-ary relations involves extraction of relations among multiple concepts such as drug-gene-mutation interactions in medicine [131], experiments related to solid oxide fuel cells with involved material and measurement conditions in material sciences [90], or task-dataset-metric-score tuples for leaderboard construction for machine learning tasks [142].

Approaches for concept extraction achieve F1 scores ranging from 56.6% to 96.9% (see Table 3.2), for coreference resolution F1 scores range from 46.0% to 61.4% [58, 178], and for binary relation extraction from 28.0% to 83.6% (see Table 3.3). The task of n-ary relation extraction with an F1 score from 28.7% to 56.4% [131, 142] is especially challenging, since such relationships usually span beyond sentences or even sections and thus, machine learning models require an understanding of the whole document. The inter-coder agreement in terms of Cohen's Kappa (κ) or F1 for the task of concept extraction ranges from 0.6 to 0.96 (Table 3.2), for relation extraction from 0.6 to 0.9 (see also Table 3.3), while for coreference

Table 3.1: Characteristics of datasets and performance measures for sentence classification in research papers.

Dataset	Domains	# Papers	Coverage	Sentence Classes	Inter-coder Agreement	Performance
PubMed-20k [68]	Biomedicine	20,000	abstracts	Background Objective, Methods Results, Conclusion	n/a	92.9% F1 [54]
NICTA-PIBOSO [147]	Biomedicine	1,000	abstracts	Background Intervention, Study Population Outcome, Other	0.62 κ	84.7% F1 [54]
CSABSTRUCT [54]	Computer Science	2,189	abstracts	Background Objective Method Result, Other	0.75 κ	83.1% F1 [54]
CS-Abstracts [100]	Computer Science	654	abstracts	Background Objective, Methods Results, Conclusions	n/a	74.6% F1 [100]
Emerald 100k [270]	Management Information Science Engineering	103,457	abstracts	Purpose Design/methodology/approach Findings Originality/value Social implications Practical implications Research limitations/implications	n/a	n/a
MAZEA [62]	Physics Engineering Life and Health Sciences	1,335	abstracts	Background Gap, Purpose Method Result, Conclusion	0.59 κ	66.0% accuracy [62]
Safder et al. [243]	Computer Science	92	full text	Algorithmic Efficiency Dataset Description Algorithmic Time Complexity Other	n/a	78.5% accuracy [243]
Dr. Inventor [85]	Computer Graphics	40	full text	Background Challenge, Approach Outcome, Future Work	0.67 κ	72.5% accuracy [11]
ART/CoreSC [175]	Chemistry Computational Linguistic	225	full text	Background Motivation, Goal Hypothesis, Object Model, Method Experiment, Result Observation, Conclusion	0.57 κ	51.6% F1 [174]

Table 3.2: Characteristics of datasets and performance measures for scientific concept extraction in research papers. * For SOFC-Exp corpus, performance values were obtained with ground truth sentences describing experiments.

Dataset	**Domains**	**# Papers**	**# Concepts**	**Coverage**	**Concept Types**	**Inter-coder Agreement**	**Performance**
SemEval17 [9]	Computer Science Material Sciences Physics	500	9,946	abstract	Process Task Material	0.60 κ	56.9% F1 [207]
SciERC [178]	Art. Intelligence	500	8,089	abstract	Task Method Metric Material Other Generic	0.77 κ	75.2% F1 [207]
ACL2 [230]	Comp. Linguistics	300	6,818	abstract	Method Tool Language Resource (LR) LR product Model Measures/Measurements Other	63.0% F1	69.9% F1 [207]
B5CDR [173]	Biomedicine	1500	28,785	abstract	Chemical Disease	91.8% F1	88.9% F1 [19]
NCBI-disease [74]	Biomedicine	793	6,892	abstract	Disease	88.0% F1	96.9% F1 [19]
SOFC-Exp [90]	Material Sciences	45	4,004	full text	Material Device Value	95.8% F1	81.5% F1* [90]

Table 3.3: Characteristics of datasets and performance measures for binary and n-ary relation extraction in research papers. Relation types in brackets denote a tupel. * For SOFC-Exp corpus, performance values were obtained with ground truth concept mentions.

Dataset	Domains	# Papers	Coverage	Cardinality	Relation Types	Scope	Inter-coder Agreement	# Relations	Performance
SemEval17 [9]	Computer Science Material Sciences Physics	500	abstract	binary	synonym-of hyponym-of	intra-sentence	0.60 κ	672	28.0% F1 [9]
SemEval18 [230]	Comp. Linguistics	500	abstract	binary	usage result model part-whole topic comparison	intra-sentence	90.8% F1	1595	49.3% F1 [230]
ChemProt [156]	Biomedicine	2482	abstract	binary	Upregulator Activator Downregulator Inhibitor Agonist Antagonist Substrate	intra-sentence	n/a	10,031	83.64% F1 [19]
SciERC [178]	Art. Intelligence	500	abstract	binary	hyponym-of compare part-of conjunction evaluate-for feature-of used-for	cross-sentence	0.68 κ	4,716	39.3% F1 [178]
PWC [142]	Art. Intelligence	731	full text	n-ary	(Task, Dataset, Metric, Score)	document-level	n/a	2,295	28.7% F1 [142]
CKB [131]	Biomedicine	343	full text	n-ary	(Drug, Gene, Mutation)	document-level	n/a	2,025	52.8% F1 [131]
SOFC-Exp [90]	Material Sciences	45	full text	n-ary	(AnodeMaterial, CathodeMaterial, Device, ElectrolyteMaterial, FuelUsed, InterlayerMaterial, OpenCircuitVoltage, PowerDensity, Resistance, WorkingTemperature)	document-level	n/a	n/a	56.4% F1* [90]

resolution the value of 0.68 κ was reported in Luan et al. [178]. The results suggest that these tasks are not only difficult for machines but also for humans in most cases.

3.2.4 Quality of Knowledge Graphs

Knowledge Graphs may contain billions of machine-readable facts about the world or a certain domain. However, do the KGs also have an appropriate quality? Data Quality (DQ) is defined as *"fitness for use"* by a data consumer [289]. Thus, to evaluate data quality, it is important to know the needs of the data consumer since, in the end, the consumer judges whether a product is fit for use. Wang et al. [289] propose a data quality evaluation framework for information systems consisting of 15 dimensions grouped into four categories:

1. *Intrinsic DQ*: accuracy, objectivity, believability, and reputation.
2. *Contextual DQ*: value-added, relevancy, timeliness, completeness, and an appropriate amount of data.
3. *Representational DQ*: interpretability, ease of understanding, representational consistency, and concise representation.
4. *Accessibility DQ*: accessibility and access security.

Bizer [22] and Zaveri et al. [299] propose further dimensions for the Linked Data context like consistency, offensiveness, licensing, and interlinking. Pipino et al. [219] subdivide completeness into *schema completeness*, i.e. the extent to which classes and relations are missing in the ontology to support a certain use case, *column completeness* (also known as *Partial Closed World Assumption* [93]), i.e. the extent to which facts are not missing, and *population completeness*, i.e. the extent to which instances for a certain class are missing. Färber et al. [79] comprehensively evaluate and compare the data quality of popular KGs (e.g. DBpedia [172], Freebase [25], Wikidata [285], YAGO [275]) using such dimensions.

To evaluate the correctness of instance data (also known as *precision*), the facts in the KG have to be compared against a ground truth. For that, humans annotate a set of facts as true or false. For instance, YAGO was found to be 95% correct [275]. The automatically populated AI-KG has a precision of 79% [70]. The KG automatically populated by the Never-Ending Language Learner (NELL) has a precision of 74% [41].

To evaluate the *completeness of instance data* (also known as *coverage and recall*), small collections of ground-truth capturing *all* knowledge for a certain ontology is necessary, that are usually difficult to obtain [291]. However, some studies estimate the completeness of several KGs. Galarrage et al. [94] suggest a rule mining approach to predict missing facts. In Freebase [25], 71% of people have an unknown place of birth, and 75% have an unknown nationality [75]. Suchanek et al. [274] report that 69%-99% of instances in popular KGs (e.g. YAGO [275], DBpedia [172]) do not have at least one property that other instances of the same class have. The AI-KG has an estimated recall of 81.2% [70].

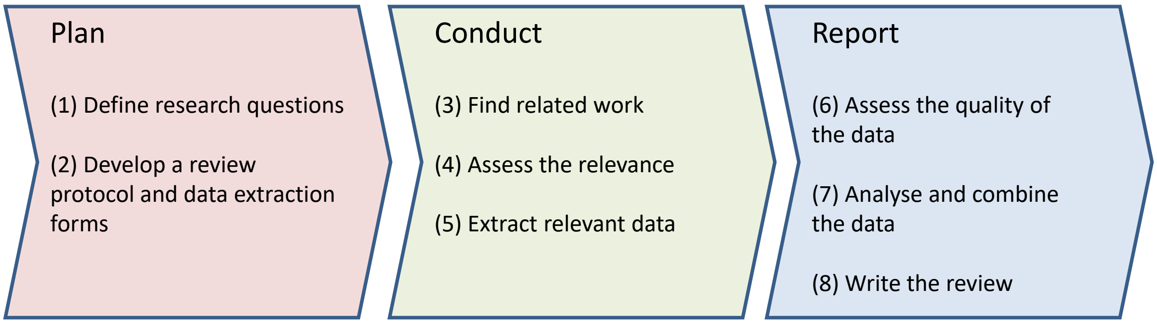

Plan
(1) Define research questions
(2) Develop a review protocol and data extraction forms
Conduct
(3) Find related work
(4) Assess the relevance
(5) Extract relevant data
Report
(6) Assess the quality of the data
(7) Analyse and combine the data
(8) Write the review

Figure 3.1: Activities within a systematic literature review. Based on [151].

3.2.5 Systematic Literature Reviews

Literature reviews are one of the main tasks of researchers since a clear identification of a contribution to the present scholarly knowledge is a crucial step in scientific work [115]. This requires a comprehensive elaboration of the present scholarly knowledge for a certain research question. Furthermore, systematic literature reviews help to identify research gaps and to position new research activities [151].

A literature review can be conducted systematically or in a non-systematic, narrative way. Following Fink's [84] definition, a systematic literature review is *"a systematic, explicit, comprehensive, and reproducible method identifying, evaluating, and synthesising the existing body of completed and recorded work"*. Guidelines for systematic literature reviews have been suggested for several scientific disciplines, e.g. for software engineering [151], for information systems [204], and for health sciences [84]. A systematic literature review consists typically of the activities depicted in Figure 3.1 subdivided into the phases *plan*, *conduct*, and *report*. The activities may differ in detail for the specific scientific domains [84, 151, 204]. In particular, a *data extraction form* defines which data has to be extracted from the reviewed papers. Data extraction requirements vary from review to review so that the form is tailored to the specific research questions investigated in the review.

3.3 Requirements Analysis

As the discussion of related work reveals, existing knowledge graphs for research information focus on specific use cases (e.g. improve search engines, help to reproduce research results) and mainly manage metadata and research artefacts about articles. We envision a KG in which research articles are linked through a deep semantic representation of their content to enable further use cases. In the following, Section 3.3.1 first states the problem and describes our research method. This motivates our use case analysis in Section 3.3.2, from which we derive requirements for an ORKG described in Section 3.3.3.

3.3.1 Problem Statement and Research Method

This section frames the problem statement and describes our research method.

Problem Statement: As described in Section 1.2.1, we are faced with several conflicting requirements when constructing an ORKG. On the one hand, we desire an ontology that can comprehensively capture scholarly knowledge, and instance data with high correctness and completeness. On the other hand, we are faced with a "knowledge acquisition bottleneck" because instance data of comprehensive ontologies and with high correctness can only be populated manually by ontology and domain experts, which, however, prevents instance data with high completeness. Current automatic approaches that could achieve high completeness of the instance data can only populate rather simple ontologies with moderate accuracy.

Research Method: To illuminate the problem statement, we perform a *requirements analysis*. The development of an ORKG should follow the Design Science Research (DSR) methodology [35, 121]. The objective of DSR, in general, is the innovative, rigorous, and relevant design of information systems for solving important practical problems or the improvement of existing solutions [35, 115]. The requirements analysis is a central phase in DSR, as it is the basis for design decisions and selection of methods to construct effective solutions systematically [35]. A requirements analysis (also known as requirements engineering), is *"the systematic and disciplined approach to the specification and management of requirements with the goal of understanding the stakeholders' desires and needs and minimizing the risk of delivering a system that does not meet these desires and needs"* [99]. One fundamental aspect of requirements engineering is that all involved people obtain a shared understanding of the problem and the corresponding requirements [241]. Moreover, the major tasks in requirements engineering are elicitation, documentation, validation, and management of requirements [99, 241].

To elicit requirements, we studied guidelines for (a) systematic literature reviews (see Section 3.2.5), (b) data quality requirements for information systems (see Section 3.2.4), and (c) interviewed and discussed with members of the ORKG and Visual Analytics team at TIB[3] (referred to as requirements analysis group). The members of the requirements analysis group are experienced researchers in the fields of computer science and environmental sciences (two professors, three postdoctoral researchers, and me as a PhD student). In Section 3.3.2 and Section 3.3.3, we describe and discuss the elicited requirements. Based on the requirements, we elaborate on possible approaches to construct an ORKG, which were identified through a literature review (see Section 3.2.3). To verify our assumptions on the presented requirements and approaches, members of the requirements analysis group

[3] `https://projects.tib.eu/orkg/project/team/`, `https://www.tib.eu/en/research-development/visual-analytics/staff`

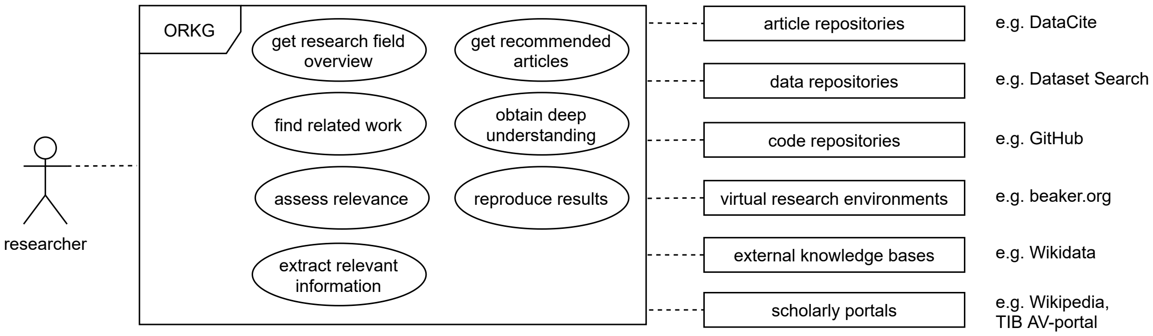


Figure 3.2: UML use case diagram for the main use cases between a researcher, an Open Research Knowledge Graph (ORKG), and external systems.

reviewed them in an iterative refinement process. Although the requirements analysis group members may be biased towards the fields of computer science and environmental sciences, we have encouraged them to define the requirements in a domain-independent manner as possible.

3.3.2 Overview of the Use Cases

We define functional requirements with use cases which are a popular technique in software requirements engineering [26, 52, 241]. A use case describes the interaction between a user and the system from the *user's perspective* to achieve a certain goal. Furthermore, a use case provides a motivating scenario to guide the design of a supporting ontology and the use case analysis helps to figure out which kind of information is necessary [64].

During our requirements analysis, we elicited many use cases (e.g. literature reviews, reproducing research results, plagiarism detection, peer reviewer suggestion) and several stakeholders (e.g. researchers, librarians, peer reviewers, practitioners) that may benefit from an ORKG. The primary source for the elicitation were the guidelines for systematic literature reviews (see Section 3.2.5) and discussions within the the requirements analysis group. However, a complete analysis of all possible use cases of graph-based knowledge management systems in the research environment is far beyond the scope of this requirements analysis. Therefore, we focus in the requirements analysis only on use cases that support the stakeholder *researcher* in the following tasks:

(a) Conducting literature reviews (see also Section 3.2.5).

(b) Obtaining a deep understanding of a research article.

(c) Reproducing research results.

We decided on this selection of tasks after discussions within the requirements analysis group since, in our view, they represent the most important tasks of researchers that an ORKG should support.

Figure 3.2 depicts the main identified seven use cases. The first five use cases, namely (1) *get research field overview*, (2) *find related work*, (3) *assess relevance*, (4) *extract relevant information*, and (5) *get recommended articles* aim to support researchers in the task *conducting literature reviews* (see also Figure 3.1). The last two use cases, namely (6) *obtain deep understanding* and (7) *reproduce results* aim to support the researchers in the tasks *obtaining a deep understanding of a research article* and *reproducing research results*, respectively. In the following, the use cases are described briefly in the casual form at the user goal level so that a variety of audiences can understand them [52]. Please note that we focus on how *semantic content* can improve these use cases and not further metadata.

Get Research Field Overview: Survey articles provide an overview of a particular research field, e.g. a certain research problem or a family of approaches. The results in such surveys are sometimes summarised in structured and comparative tables (an approach usually followed in domains such as computer science, but not as systematically practised in other fields). However, once survey articles are published, they are no longer updated. Moreover, they usually represent only the perspective of the authors, i.e. very few researchers of the field. To support researchers to obtain an up-to-date overview of a research field, the system should maintain such surveys in a structured way, and allow for dynamics and evolution. A researcher interested in such an overview should be able to search or browse the desired research field in a user interface for ORKG access. Then, the system should retrieve related articles and available overviews, e.g. in a table or a leaderboard chart.

While an ORKG user interface should allow for showing tabular leaderboards or other visual representations, the backend should semantically represent information to allow for the exploitation of overlaps in conceptualisations between research problems or fields. Furthermore, faceted drill-down methods based on the properties of semantic descriptions of research approaches could empower researchers to quickly filter and zoom into the most relevant literature.

Find Related Work: Finding relevant research articles is a core activity of researchers. The primary goal of this use case is to find research articles which are relevant to a certain research question. A broad research question is often broken down into smaller, more specific sub-questions which are then converted to search queries [84]. For instance, in this requirements analysis, we explored the following questions during our literature review in Section 3.2:

(a) Which ontologies do exist to represent scholarly knowledge?

(b) Which scientific knowledge graphs do exist, and which information do they contain?

(c) Which datasets do exist for scientific information extraction?

(d) What are current state-of-the-art methods for scientific information extraction?

(e) Which approaches do exist to construct a KG?

An ORKG should support the answering of queries related to such kind of questions, which can be fine-grained or broad search intents. Preferably, the system should support natural language queries as approached by semantic search and question answering engines [14]. The system has to return a set of relevant articles.

Assess Relevance: Given a set of relevant articles, the researcher has to assess whether the articles match the criteria of interest. Usually, researchers skim through the title and abstract [2, 133, 278]. Often, also the introduction and conclusions have to be considered, which is cumbersome and time-consuming. If only the most important paragraphs in the article are presented to the researcher in a structured way, this process can be boosted. Such information snippets might include, for instance, text passages that describe the problem tackled in the research work, the main contributions, the employed methods or materials, or the yielded results.

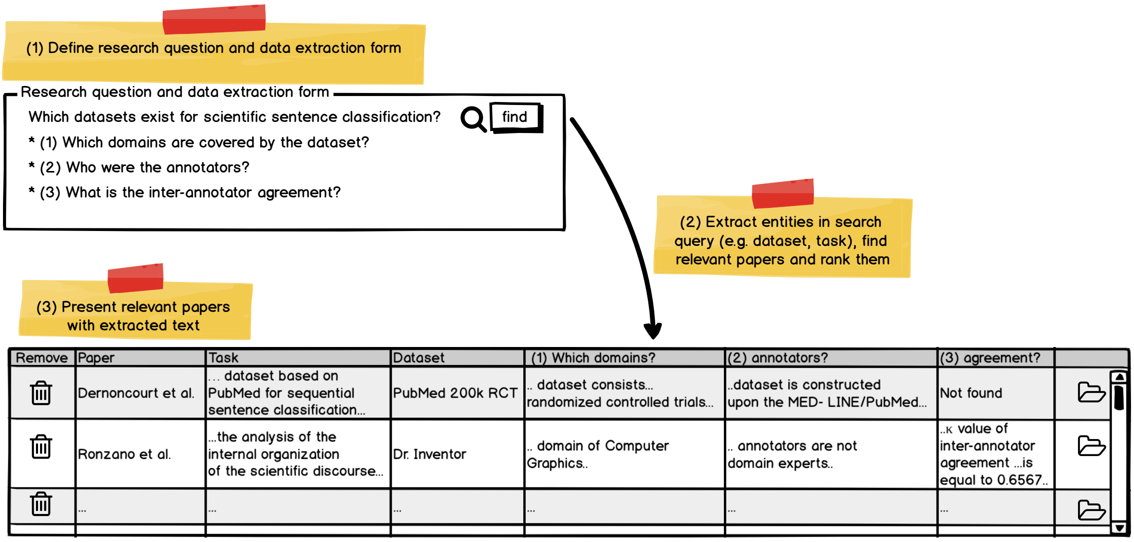
Extract Relevant Information: To tackle a particular research question, the researcher has to extract relevant information from research articles. In a systematic literature review, the information to be extracted can be defined through a *data extraction form* (see Section 3.2.5). Such extracted information is usually compiled in written text or comparison tables in a related work section or survey articles. For instance, for the question *"Which datasets do exist for scientific sentence classification?"* a researcher who focuses on a new annotation study could be interested in (a) domains covered by the dataset and (b) the inter-coder agreement (see Table 3.1 as an example). Another researcher might follow the same question but focusing on machine learning, and thus could be more interested in (c) evaluation results and (d) feature types used.

The system should support the researcher with tailored information extraction from a set of research articles:

1. The researcher defines a data extraction form as proposed in systematic literature reviews (e.g. the fields (a)-(d) above).
2. The system presents the extracted information as suggestions for the corresponding data extraction form and articles in a comparative table.

Figure 3.3 illustrates a data extraction form with corresponding fields in form of questions, and a possible approach to visualise the extracted text passages from the articles for the respective fields in a tabular form.

Get Recommended Articles: When the researcher focuses on a particular article, further related articles could be recommended by the system utilising an ORKG, for instance, articles that address the same research problem or apply similar methods.



(1) Define research question and data extraction form

Research question and data extraction form

Which datasets exist for scientific sentence classification?

* (1) Which domains are covered by the dataset?

* (2) Who were the annotators?

* (3) What is the inter-annotator agreement?

(2) Extract entities in search query (e.g. dataset, task), find relevant papers and rank them

(3) Present relevant papers with extracted text

Remove	Paper	Task	Dataset	(1) Which domains?	(2) annotators?	(3) agreement?	
🗑	Dernoncourt et al.	... dataset based on PubMed for sequential sentence classification...	PubMed 200k RCT	.. dataset consists... randomized controlled trials...	..dataset is constructed upon the MED- LINE/PubMed...	Not found	📂
🗑	Ronzano et al.	...the analysis of the internal organization of the scientific discourse...	Dr. Inventor	.. domain of Computer Graphics..	.. annotators are not domain experts..	..κ value of inter-annotator agreement ...is equal to 0.6567..	📂
🗑	...	...	...	...	...	...	📂

Figure 3.3: An example research question with a corresponding data extraction form, and the extracted text passages from relevant research articles for the respective (data extraction form) fields presented in a tabular form.

Obtain Deep Understanding: The system should help the researcher to obtain a deep understanding of a research article (e.g. equations, algorithms, diagrams, datasets). For this purpose, the system should connect the article with artefacts such as conference videos, presentations, source code, datasets, etc., and visualise the artefacts appropriately. Also, text passages can be linked, e.g. to explanations of methods in Wikipedia, source code snippets of an algorithm implementation, or equations described in the article.

Reproduce Results: The system should offer researchers links to all necessary artefacts to help to reproduce research results, e.g. datasets, source code, virtual research environments, materials describing the study, etc. Furthermore, the system should maintain semantic descriptions of domain-specific and standardised evaluation protocols and guidelines such as in machine learning reproducibility checklists [218] and bioassays in the medical domain.

3.3.3 Knowledge Graph Requirements

As outlined in Section 3.2.4, data quality requirements should be considered within the context of a particular use case ("fitness for use"). In this section, we first describe dimensions we used to define non-functional requirements for an ORKG. Then, we discuss these requirements within the context of our identified use cases.

3.3.3.1 Dimensions for Knowledge Graph Requirements

In the following, we describe the dimensions that we use to define the requirements for ontology design and instance data. We selected these dimensions since we assume they are most relevant and most challenging to construct an ORKG with appropriate data to support the various use cases.

For *ontology design*, i.e. how comprehensively should an ontology conceptualise scholarly knowledge to support a certain use case, we use the following dimensions:

A) *Domain specialisation of the ontology:* How domain-specific should the concepts and relation types be in the ontology? An ontology with *high domain specialisation* targets a specific (sub-)domain and uses domain-specific terms. An ontology with *low domain specialisation* targets a broad range of domains and uses rather domain-independent terms. For instance, Pertsas and Constantopoulos [215] propose domain-independent concepts (e.g. activity, method, assertion). In contrast, Klampanos et al. [152] present a very domain-specific ontology for artificial neural networks.

B) *Granularity of the ontology:* Which granularity of the ontology is required to conceptualise scholarly knowledge? An ontology with *high granularity* conceptualises scholarly knowledge with a lot of classes that have very detailed and many fine-grained properties and relations. An ontology with a *low granularity* has only a few classes and relation types. For instance, the annotation schemes for scientific corpora (see Section 3.2.3) have a rather low granularity, as they do not have more than 10 classes and 10 relation types. In contrast, various ontologies (e.g. [110, 215]) with more than 20 to 35 classes and over 20 to 70 relations and properties are fine-grained and have a relatively high granularity.

Although there is usually a correlation between domain specialisation and granularity of the ontology (e.g. an ontology with high domain-specialisation has also a high granularity), there exist also rather domain-independent ontologies with a high granularity, e.g. Scholarly Ontology [215], and ontologies with high domain-specialisation and low granularity, e.g. the PICO criterion in Evidence Based Medicine [147, 236]) which stands for population (P), intervention (I), comparison (C), and outcome (O). Thus, we use both dimensions independently. Furthermore, a high domain specialisation requirement for a use case implies that each sub-domain requires a separate ontology for the specific use case. These domain-specific ontologies can be organised in a taxonomy.

For the *instance data*, we use the following dimensions:

C) *Completeness of the instance data:* Given an ontology, to which extent do *all* possible instances (i.e. instances for classes and links for relation types) in *all* research articles have to be represented in the KG? *Low completeness:* it is tolerable for the use case when a considerable amount of instance data is missing for the respective ontology.

High completeness: it is mandatory for the use case that for the respective ontology, a considerable amount of instances are present in the instance data. For instance, given an ontology with a class “Task” and a relation type “subTaskOf” to describe a taxonomy of tasks, the instance data for that ontology would be complete if all tasks mentioned in all research articles are present (population completeness) and “subTaskOf” links between the tasks are not missing (column completeness).

D) *Correctness of the instance data:* Given an ontology, which correctness is necessary for the corresponding instances? *Low correctness:* it is tolerable for the use case that some instances (e.g. 30%) are not correct. *High correctness:* it is mandatory for the use case that instance data must not be wrong, i.e. all present instances in the KG must conform to the ontology and reflect the content of the research articles properly. For instance, an article is correctly assigned to the task addressed in the article, the F1 score in the evaluation results are correctly extracted, etc.

It should be noted that the completeness and correctness of instance data can be evaluated only for a given ontology. For instance, let A be an ontology having the class “Deep Learning Model” without properties, and let B be an ontology that also has a class “Deep Learning Model” and additionally further relation types describing the properties of the deep learning model (e.g. drop-out, loss functions, etc.). In this example, the instance data of ontology A would be considered to have high completeness, if it covers most of the important deep learning models. However, for ontology B, the completeness of the same instance data would be rather low since the properties of the deep learning models are missing. The same holds for correctness: If ontology B has, for instance, a sub-type “Convolutional Neural Network”, then the instance data would have rather low correctness for ontology B if all “Deep Learning Model” instances are typed only with the generic class “Deep Learning Model”.

3.3.3.2 Discussion of the Knowledge Graph Requirements

Next, we discuss the seven main use cases regarding the required level of ontology domain specialisation and granularity, as well as the completeness and correctness of instance data. Table 3.4 summarises the requirements for the use cases along the four dimensions at an ordinal scale. These requirements were discussed and consolidated within the requirements analysis group. The aim of the chosen scale and the selected values is to indicate the direction of future research and development efforts for the respective use cases. The use cases are grouped if they have (1) similar justifications for the requirements and (2) a high overlap in ontology concepts and instances.

Extract Relevant Information & Get Research Field Overview: The information to be extracted from relevant research articles for a data extraction form within a literature review is very heterogeneous and depends highly on the intent of the researcher and the

Table 3.4: Requirements and approaches for the main use cases. The upper part describes the minimum requirements for the ontology (domain specialisation and granularity) and the instance data (completeness and correctness). The bottom part lists possible approaches for manual, automatic and semi-automatic curation of the KG for the respective use cases. "X" indicates that the approach is suitable for the use case while "(x)" denotes that the approach is only appropriate with human supervision. The left part (delimited by the vertical triple line) groups use cases suitable for manual, and the right side for automatic approaches. Vertical double lines group use cases with similar requirements.

		Extract relevant info	*Research field overview*	*Deep understanding*	*Reproduce results*	*Find related work*	*Recommend articles*	*Assess relevance*
Ontology	Domain specialisation	high	high	med	med	low	low	med
	Granularity	high	high	med	med	low	low	low
Instance data	Completeness	low	med	low	med	high	high	med
	Correctness	med	high	high	high	low	low	med
Manual curation	Maintain terminologies	-	X	-	-	X	X	-
	Define templates	X	X	-	-	-	-	-
	Fill in templates	X	X	X	X	-	-	-
	Maintain overviews	X	X	-	-	-	-	-
Automatic curation	Entity/relation extr.	(x)	(x)	(x)	(x)	X	X	X
	Entity linking	(x)	(x)	(x)	(x)	X	X	X
	Sentence classification	(x)	-	(x)	-	X	-	X
	Template-based extr.	(x)	(x)	(x)	(x)	-	-	-
	Cross-modal linking	-	-	(x)	(x)	-	-	-

research questions. Thus, the ontology has to be domain-specific and fine-grained to offer all possible kinds of desirable information. However, missing information for certain questions in the KG may be tolerable for a researcher. Furthermore, it might be acceptable if some of the extracted suggestions for a data extraction form are wrong since the researcher can correct them.

Research field overviews are usually the result of a literature review. The data in such an overview has also to be very domain-specific and fine-grained. Also, this information must have high correctness, e.g. an F1 score of an evaluation result must not be wrong. Furthermore, an overview of a particular research field should have medium completeness and must not miss any important research papers. However, it is acceptable when overviews for some research fields are missing.

Obtain Deep Understanding & Reproduce Results: The information required for these use cases has to achieve a high level of correctness (e.g. accurate links to dataset, source code, videos, articles, research infrastructures). An ontology for the representation of default artefacts can be rather domain-independent (e.g. Scholix [38]). However, semantic representation of evaluation protocols requires domain-dependent ontologies (e.g. EXPO [264]). Missing information is tolerable for these use cases.

Find Related Work & Get Recommended Articles: When searching for related work, it is essential not to miss relevant articles. Previous studies revealed that more than half of the search queries in academic search engines refer to scientific entities [294]. However, the coverage of scientific entities in general-purpose KGs (e.g. Wikidata) is rather low, since the introduction of new concepts in research literature occurs at a faster pace than KG curation [3]. Despite the low completeness, Xiong et al. [294] could improve the ranking of search results in academic search engines by exploiting general-purpose KGs. Hence, the instance data for the *find related work* use case should have high completeness with fine-grained scientific entities. However, semantic search engines leverage latent representations of KGs and text (e.g. graph and word embeddings) [14]. Since a non-perfect ranking of the search results is tolerable for a researcher, lower correctness of the instance data could be acceptable. Furthermore, due to latent feature representations, the ontology can be kept rather simple and domain-independent.

Graph- and content-based research paper recommendation systems [18] have similar requirements since they also leverage latent feature representations and require fine-grained scientific entities. Also, non-perfect recommendations are tolerable for a researcher.

Assess Relevance: To help the researcher to assess the relevance of an article according to her needs, the system should highlight the most essential zones in the article to get a quick overview. The completeness and correctness of the presented information must not be too low, as otherwise, the user acceptance may suffer. However, it can be suboptimal since it is acceptable for a researcher when some of the highlighted information is not essential or when some important information is missing. The ontology to represent essential information should be rather domain-independent and quite simple (cf. ontologies for scientific sentence classification in Section 3.2.3.2).

3.4 ORKG Construction Strategies

In this section, we discuss the implications for the design and construction of an ORKG and outline possible approaches, which are mapped to the use cases in Table 3.4. The possible approaches were identified through a literature review (see Section 3.2.3). Based on the discussion in the previous section, we can subdivide the use cases into two groups:

1. Use cases requiring high correctness and high domain specialisation with rather low requirements on the completeness (left side in Table 3.4).
2. Use cases requiring high completeness with rather low requirements on the correctness and domain specialisation (right side in Table 3.4).

The first group requires manual approaches for KG construction, while the KG construction for the second group could be accomplished with fully-automatic approaches. To ensure

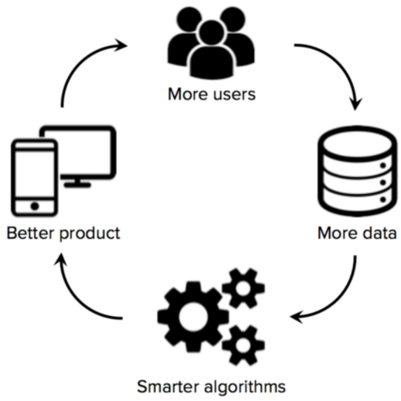


Figure 3.4: The virtuous cycle of data network effects by combining manual and automatic data curation approaches [39].

trustworthiness, data records should contain provenance information, i.e. who or what system curated the data.

Manually curated data can also support use cases with automatic approaches, and vice versa. Furthermore, automatic approaches can complement manual approaches by providing suggestions in user interfaces. Such synergy between humans and algorithms may lead to a "data flywheel" (also known as data network effects, see Figure 3.4): Users produce data which enable to build a smarter product with better algorithms so that more users use the product and thus produce more data, and so on.

3.4.1 Manual Approaches

This section describes possible manual approaches for ontology design and KG population.

Ontology Design: The first group of use cases requires rather domain-specific and fine-grained ontologies. We suggest developing novel or reusing ontologies that fit the respective use case and the specific domain (e.g. EXPO [264] for experiments). Moreover, appropriate and simple user interfaces are necessary for an efficient and easy population.

However, such ontologies can also evolve with the help of the community, as demonstrated by Wikidata and Wikipedia with "infoboxes" (see Section 3.2.3). Therefore, the system should enable the maintenance of *templates*, which are pre-defined and very specific forms consisting of fields with certain types (see Figure 3.5). For instance, to automatically generate leaderboards for machine learning tasks, a template would have the fields task, model, dataset, metric, and score, which can then be filled in by a curator for articles providing such kind of results in a user interface generated from the template. Such an approach is based on *meta-modelling* [26], as the meta-model for templates enables the definition of concrete templates, which are then instantiated for articles.

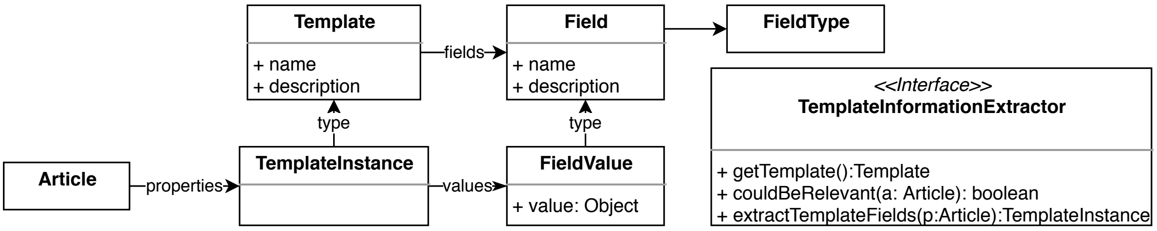


Figure 3.5: Conceptual meta-model in UML for templates and interface design for an external template-based information extractor.

Knowledge Graph Population: Several user interfaces are required to enable manual population:

1. Populate semantic content for a research article by (1a) choosing relevant templates or ontologies and (1b) fill in the values.
2. Terminology management (e.g. domain-specific research fields).
3. Maintain research field overviews by (3a) assigning relevant research articles to the research field, (3b) define corresponding templates, and (3c) fill in the templates for the relevant research articles.

Furthermore, the system should also offer Application Programming Interfaces (APIs) to enable population by third-party applications, e.g.

- Submission portals such as `https://www.easychair.org/` during submission of an article.
- Authoring tools such as `https://www.overleaf.com/` during writing.
- Virtual research environments [272] to store evaluation results and links to datasets and source code during experimenting and data analysis.

To encourage stakeholders like researchers, librarians, and crowd workers to contribute content, we see the following options:

- *Top-down enforcement* via submission portals and publishers.
- *Incentive models*: Researchers want their articles to be cited; semantic content helps other researchers to find, explore and understand an article. This is also related to the concept of *enlightened self-interest*, i.e. act to further interests of others to serve the own self-interest [125].
- Provide *public acknowledgements* for curators.
- Bring together *experts* (e.g. librarians, researchers from different institutions) who curate and organise content for specific research problems or disciplines.

3.4.2 (Semi-)automatic Approaches

This section describes possible ontology design methods for use cases requiring rather simple ontologies, and semi-automatic approaches for KG population.

Ontology design: The second group of use cases requires high completeness, while a relatively low correctness and domain specialisation are acceptable. For these use cases, rather simple or domain-independent ontologies should be developed or reused. Although approaches for automatic ontology learning exist (see Section 3.2.3), the quality of their results is not sufficient to generate a meaningful ORKG with complex conceptual models and relations. Therefore, meaningful ontologies should be designed by human experts.

Knowledge Graph Population: Various approaches can be used to (semi-)automatically populate an ORKG. Methods for *entity and relation extraction* (see Section 3.2.3) can help to populate fine-grained KGs with high completeness and *entity linking* approaches can link mentions in text with entities in KGs. For cross-modal linking, Singh et al. [260] suggest an approach to detect URLs to datasets in research articles automatically, while the Scientific Software Explorer [120] connects text passages in research articles with code fragments. To extract relevant information at the sentence level, approaches for *sentence classification* in scientific text can be applied (see Section 3.2.3). To support the curator fill in templates semi-automatically, *template-based extraction* can (1) suggest relevant templates for a research article and (2) pre-fill fields of templates with appropriate values. For pre-filling, approaches such as n-ary relation extraction [90, 122, 131, 142] or end-to-end question answering [71, 233] could be applied.

Furthermore, the system should allow to plugin *external information extractors*, developed for certain scientific domains to extract specific types of information. For instance, as depicted in Figure 3.5, an external template information extractor has to implement an interface with three methods. This enables the system to (1) filter relevant template extractors for an article and (2) extract field values from an article.

3.5 Summary

In this chapter, we have presented a requirements analysis for an Open Research Knowledge Graph (ORKG) that addresses the first research question **RQ1** (requirements for scholarly knowledge representation) of this thesis. An ORKG should represent the content of research articles semantically to enhance or enable a wide range of use cases. Our requirements analysis identified literature-related core tasks of a researcher that can be supported by an ORKG and formulated them as use cases. For each use case, we discussed specificities and requirements for the underlying ontology and the instance data. In particular, we identified two groups of use cases:

1. The first group requires instance data with high correctness and rather fine-grained, domain-specific ontologies, but moderate completeness of the instance data might be sufficient.
2. The second group requires high completeness of the instance data, but the ontologies can be kept rather simple and domain-independent, and moderate correctness of the instance data might be acceptable.

Based on the requirements, we have described possible manual and semi-automatic approaches (necessary for the first group), and automatic approaches (appropriate for the second group) for KG construction. In particular, we have proposed a framework using lightweight ontologies (called templates) that can evolve by community curation. Furthermore, we have described the interdependence with external systems, user interfaces, and APIs for third-party applications to populate an ORKG.

◊◊◊

The requirements analysis of this chapter has built the foundation for this thesis and shall guide further research. In this thesis, we focus on the second group of use cases, namely to assist researchers in (1) *assessing relevance of research papers*, (2) *finding related work*, and (3) *recommending appropriate research papers.* The following chapters present contributions for machine learning tasks that can enhance these use cases. The next Chapter 4 proposes a novel cross-domain multi-task learning approach for the task of sequential sentence classification that aims to support the use case *assess relevance of research papers.* Chapters 5 and 6 present a domain-independent information extraction approach for the tasks of scientific concept extraction and coreference resolution. Using this information extraction approach, in Chapter 6, we populate an RKG that aims to support the use cases *find related work* and *recommend appropriate research papers.* Finally, in Chapter 7, we present an approach for citation recommendation that leverages our populated RKG.

4 Sentence Classification using Cross-Domain Multi-Task Learning

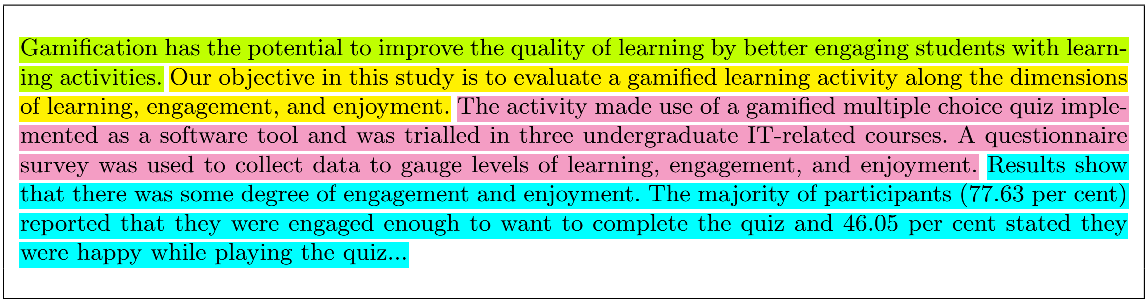
As described in the previous Chapter 3, a structured presentation of the most important paragraphs or sentences of an article can assist researchers in the assessment of a research paper's relevance. The task of *sequential sentence classification* enables categorising sentences into a predefined set of categories and thus to structure research papers. This chapter explores **RQ2** and **RQ3** which are related to this task, namely:

> **RQ2:** *How can we modify machine learning methods for information extraction from scientific texts to be adaptable to new domains with few labelled data?*
>
> **RQ3:** *How can we automatically extract information from research papers from multiple scientific domains in a domain-independent manner?*

We propose a unified cross-domain multitask learning approach that can exploit datasets from different scientific domains with different structures. Furthermore, we present an approach to classify sentences in research papers in a domain-independent manner. In the following, Section 4.1 first motivates our approaches. Then, Section 4.2 summarises related work on sentence classification in research papers and transfer learning in NLP. Our proposed approaches are presented in Section 4.3. The setup and results of our experimental evaluation are reported in Section 4.4 and 4.5, while Section 4.6 summarises this chapter.

4.1 Introduction

To search relevant research papers for a particular field is a core activity of researchers. Scientists usually use academic search engines and skim through the text of the found articles to assess their relevance. The task of *sequential sentence classification* targets the categorisation of sentences by their semantic content or function. In research papers, this can be used to classify sentences by their contribution to the article's content, e.g. to determine if a certain sentence contains information about the research work's objective, methods, or results [68]. Figure 4.1 shows an example of an abstract with classified sentences. Such a semantification of sentences can help algorithms focus on relevant elements of text and thus assist information retrieval systems [198, 242]. The task is called *sequential* to distinguish



Gamification has the potential to improve the quality of learning by better engaging students with learning activities. Our objective in this study is to evaluate a gamified learning activity along the dimensions of learning, engagement, and enjoyment. The activity made use of a gamified multiple choice quiz implemented as a software tool and was trialled in three undergraduate IT-related courses. A questionnaire survey was used to collect data to gauge levels of learning, engagement, and enjoyment. Results show that there was some degree of engagement and enjoyment. The majority of participants (77.63 per cent) reported that they were engaged enough to want to complete the quiz and 46.05 per cent stated they were happy while playing the quiz...

Figure 4.1: An annotated abstract taken from the CSABSTRUCT dataset [54], where the sentences are coloured according to their respective category: *background* (green), *objectives* (yellow), *methods* (magenta), and *results* (cyan).

it from the general *sentence classification* task where a sentence is classified in isolation, i.e. without using local context. However, in research papers, the meaning of a sentence is often informed by the context from neighbouring sentences, e.g. sentences that describe the methods usually precede sentences about results.

In previous work, several approaches have been proposed for *sequential sentence classification* (e.g. [6, 133, 257, 295]), and several datasets were annotated for various scientific domains (e.g. [68, 85, 100, 270]). The datasets contain either abstracts or full papers and were annotated with domain-specific sentence classes. However, research infrastructures usually support multiple scientific domains. Therefore, stakeholders of digital libraries are interested in a uniform solution that enables the combination of these datasets to improve the overall prediction accuracy. For this purpose, this chapter explores the following research questions.

First, although some approaches propose transfer learning for the scientific domain [15, 106, 207], the field lacks a comprehensive empirical study on transfer learning across different scientific domains for *sequential sentence classification*. Transfer learning enables the combination of knowledge from multiple datasets to improve classification performance and thus to reduce annotation costs. The annotation of scientific text is particularly costly since it demands expertise in the article's domain [9, 92]. However, studies revealed that the success of transferring neural models largely depends on the relatedness of the tasks, and transfer learning with unrelated tasks may even degrade performance [196, 205, 238, 253]. Two tasks are related if there exists some implicit or explicit relationship between the feature spaces [205]. On the other hand, every scientific domain is characterised by its specific terminology and phrasing, which yields different feature spaces. Thus, it is not clear to what extent datasets from different scientific disciplines are related. This raises the following sub-questions of **RQ2** (few labelled data) for the task of sequential sentence classification:

#Q1: To which extent are datasets from different scientific domains semantically related?

#Q2: Which transfer learning approach works best?

#Q3: Which neural network layers are transferable under which constraints?

#Q4: Is it beneficial to train a multi-task model with multiple datasets?

Normally, every dataset has a domain-specific annotation scheme that consists of a set of associated sentence classes. This raises the second set of research questions with regard to the consolidation of these annotation schemes. Prior work [175] annotated a dataset multiple times with different schemes, and analysed the multivariate frequency distributions of the classes. They found that the investigated schemes are complementary and should be combined. However, annotating datasets multiple times is costly and time-consuming. To support the consolidation of different annotation schemes across domains, we examine the following sub-questions of **RQ3** (domain-independent extraction):

#Q5: Can a model trained with multiple datasets recognise the semantic relatedness of classes from different annotation schemes?

#Q6: Can we derive a consolidated, domain-independent annotation scheme and use that scheme to compile a new dataset to train a domain-independent model?

Finally, current approaches for sequential sentence classification are designed either for abstracts or full papers. One reason is that these text types follow rather different structures: In abstracts, different sentence classes directly follow one another normally. The general paper text, however, exhibits longer passages without change of the semantic sentence class. Typically, deep learning is used for abstracts [54, 69, 100, 133, 257, 295] since presumably more training data are available, whereas for full papers, also called *zone identification*, hand-crafted features and linear models have been suggested [6, 11, 85, 174]. However, deep learning approaches have also been applied successfully to full papers in related tasks such as argumentation mining [167], scientific document summarisation [1, 55, 63, 98], or n-ary relation extraction [90, 131, 140]. Thus, the potential of deep learning has not been fully exploited yet for sequential sentence classification on full papers, and no unified solution for abstracts as well as full papers exists. This raises the following sub-question of **RQ2** (adaptable to new domains):

#Q7: Can a unified deep learning approach be applied to text types with very different structures like abstracts or full papers?

This chapter investigates these research questions and presents the following contributions:

1. We introduce a novel multi-task learning framework for sequential sentence classification.
2. Furthermore, we propose and evaluate an approach to semi-automatically identify semantically related classes from different annotation schemes and present an analysis of four annotation schemes. Based on the analysis, we suggest a domain-independent

annotation scheme and compile a new dataset that enables to classify sentences in a domain-independent manner.

3. Our proposed unified deep learning approach can handle both text types, abstracts and full papers despite their structural differences.
4. To facilitate further research, we make our source code publicly available: `https://github.com/arthurbra/sequential-sentence-classification`.

Comprehensive experimental results demonstrate that our multi-task learning approach successfully makes use of datasets from different scientific domains, with different annotation schemes, that contain abstracts or full papers. In particular, we outperform state-of-the-art approaches for full paper datasets significantly, while obtaining competitive results for datasets of abstracts.

4.2 Related Work

In Section 3.2.3.2 and Table 3.1 we outline datasets for sentence classification in scientific texts. In this section, we describe machine learning methods for this task. Furthermore, we briefly review transfer learning methods.

4.2.1 Sequential Sentence Classification in Scientific Text

In the following, we review machine learning approaches for sentence classification in abstracts and full papers.

4.2.1.1 Approaches for Abstracts

Deep learning has been the preferred approach for sentence classification in abstracts in recent years [54, 69, 100, 133, 257, 295]. These approaches follow a common *hierarchical sequence labelling architecture*: (1) a word embedding layer encodes tokens of a sentence to word embeddings, (2) a sentence encoder transforms the word embeddings of a sentence to a sentence representation, (3) a context enrichment layer enriches all sentence representations of the abstract with context from surrounding sentences, and (4) an output layer predicts the label sequence.

As depicted in Table 4.1, the approaches vary in different implementations of the layers. They use different kinds of word embeddings, e.g. Global Vectors for Word Representation (GloVe) [212], Word2Vec [193], or SciBERT [19] that is BERT [71] pre-trained on scientific text. For sentence encoding, a Bi-LSTM [116] or a Convolutional Neural Network (CNN) with various pooling strategies are utilised, while Yamada et al. [295] and Shang et al. [257] use the classification token ([CLS]) of BERT or SciBERT. To enrich sen-

Table 4.1: Comparison of deep learning approaches for sequential sentence classification in abstracts.

Approach	**Word embedding**	**Sentence encoding**	**Context enrichment**	**Output layer**
Dernoncourt and Lee (2016) [69]	Character Emb. + GloVe	Bi-LSTM with concatenation	-	CRF
Jin and Szolovits (2018) [133]	Bio word2vec	Bi-LSTM with attention pooling	Bi-LSTM	CRF
Cohan et al. (2019) [54]	SciBERT	SciBERT-[SEP]	SciBERT-[SEP]	softmax
Gonçalves et al. (2020) [100]	GloVe	CNN with max pooling	Bi-GRU	softmax
Yamada et al. (2020) [295]	BERT from PubMed	BERT-[CLS]	Bi-LSTM	Semi-Markov CRF
Shang et al. (2021) [257]	SciBERT	SciBERT-[CLS]	Bi-LSTM/ attention	CRF

tences with further context, an RNN such as a Bi-LSTM or Bi-GRU [50] is used. Shang et al. [257] additionally exploit an attention-mechanism across sentences; however, it introduces quadratic runtime complexity that depends on the number of sentences. A Conditional Random Field (CRF) [163] is mostly used as an output layer to capture the interdependence between classes (e.g. *results* usually follow *methods*). Yamada et al. [295] form spans of sentence representations and Semi-Markov CRFs to predict the label sequence by considering all possible span sequences of various lengths. Thus, their approach can better label longer continuous sentences but is computationally more expensive than a CRF. Cohan et al. [54] obtain contextual sentence representations directly by fine-tuning SciBERT and utilising the separation token ([SEP]) of SciBERT. However, their approach can process only about 10 sentences at once since BERT supports sequences of up to 512 tokens only.

4.2.1.2 Approaches for Full Papers

For full papers, Logistic Regression, Support Vector Machines (SVMs), and Conditional Random Fields (CRFs) with hand-crafted features have been proposed [6, 11, 85, 174, 277, 279]. They represent a sentence with various syntactic and linguistic features, such as n-grams, part-of-speech tags, or citation markers, which were engineered for the respective datasets. Asadi et al. [6] also exploit semantic features obtained from knowledge bases such as Wordnet [82]. To incorporate contextual information, each sentence representation also contains the label of the previous sentence ("history feature") and the sentence position in the document ("location feature"). To better consider the interdependence between labels, some approaches apply CRFs [163], while Asadi et al. [6] suggest fusion techniques within a dynamic window of sentences. However, some approaches [6, 11, 85] exploit the *ground* *truth label* instead of the predicted label of the preceding sentence ("history feature") during prediction, which has a significant impact on the performance.

Related tasks also classify sentences in full papers with deep learning methods, e.g. for citation intent classification [53, 161] or algorithmic metadata extraction [243] but without exploiting context from surrounding sentences. Comparable to us, Lauscher et al. [167] utilise a hierarchical deep learning architecture for argumentation mining in full papers but evaluate it only on one corpus.

To the best of our knowledge, a unified approach for the task of sequential sentence classification for abstracts as well as full papers has not been proposed and evaluated yet.

4.2.2 Transfer Learning

The idea of transfer learning is to exploit knowledge from a source task to improve prediction accuracy in a target task. The tasks can have training data from different domains and vary in their objectives. According to Ruder's taxonomy for transfer learning [238], we investigate inductive transfer learning in this study since the target training datasets are labelled. Inductive transfer learning can be further subdivided into multi-task learning, where tasks are learned simultaneously, and sequential transfer learning (also referred to as parameter initialisation), where tasks are learned sequentially. Since there are so many applications for transfer learning, we focus on the most relevant cases for sentence classification in scientific texts. For a more comprehensive overview, we refer to [205, 238, 292].

Fine-tuning a pre-trained language model is a popular approach for sequential transfer learning in NLP [37, 71, 113, 124]. Here, the source task involves learning a language model (or a variant of it) using a large unlabelled text corpus. Then, the model parameters are fine-tuned with labelled data of the target task. Edwards et al. [76] evaluate the importance of domain-specific unlabelled data on pre-training word embeddings for text classification in the general domain (i.e. data such as news, phone conversations, magazines, etc.). Pruksachatkun et al. [225] improve these language models by *intermediate task transfer learning* where a language model is fine-tuned on a data-rich intermediate task before fine-tuning on the final target task. Park and Caragea [207] provide an empirical study on intermediate transfer learning from the non-academic domain to scientific keyphrase identification. They show that SciBERT in combination with related tasks such as sequence tagging improves performance, while BERT or unrelated tasks degrade the performance.

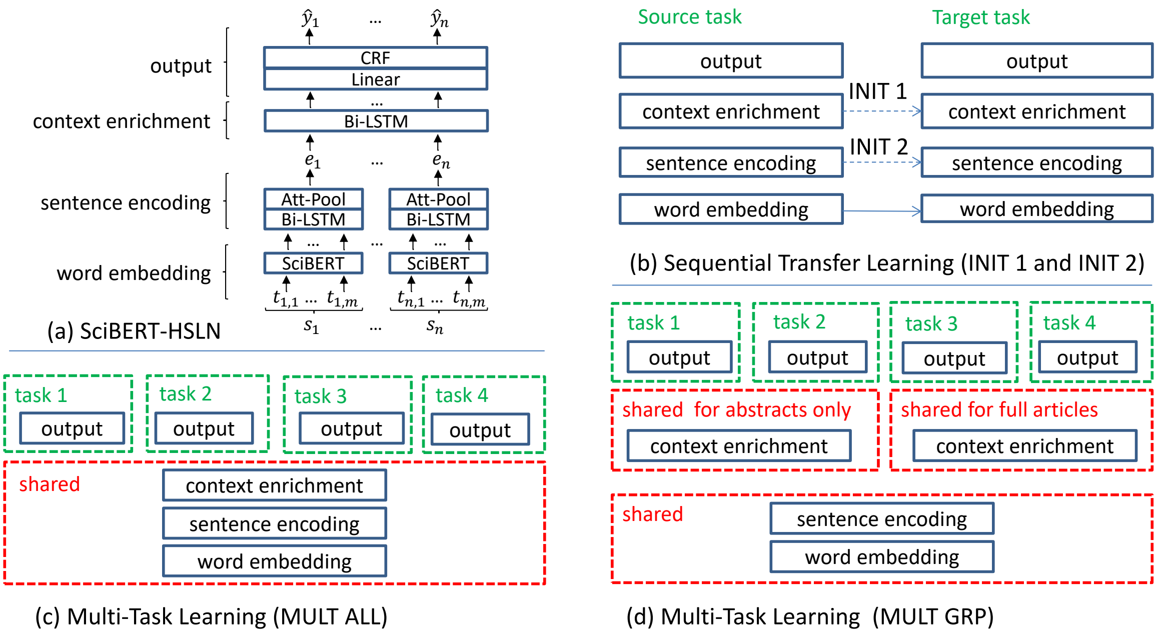
For *sequence tagging*, Yang et al. [297] investigate multi-task learning in the general domain with cross-domain, cross-application, and cross-lingual transfer. In particular, target tasks with few labelled data benefit from related tasks. Lee et al. [169] successfully transfer pre-trained parameters from a big dataset to a small dataset in the biological domain. Schulz et al. [250] evaluate multi-task learning for argumentation mining with multiple datasets in the general domain and could show that performance improves when training data for the tasks is sparse.

For *sentence classification*, Mou et al. [196] compare (1) transferring parameters from a source dataset to a target dataset against (2) training one model with two datasets in the non-academic domain. They demonstrate that semantically related tasks improve while unrelated tasks degrade the performance of the target tasks. Semwal et al. [253] investigate the extent of task relatedness for product reviews and sentiment classification with sequential transfer learning. Su et al. [273] study multi-task learning for sentiment classification in product reviews from multiple domains. Lauscher et al. [168] evaluate multi-task learning on scientific texts, however, only on one dataset with different annotation layers. Banerjee et al. [15] apply sequential transfer learning from the medical to the computer science domain for discourse classification, however, only for two domains and on abstracts, whereas Spangher et al. [266] explore this task on news articles with multi-task learning using multiple datasets. Gupta et al. [106] utilise a multi-task learning with two scaffold tasks to detect contribution sentences in full papers, however, only in one domain and with limited sentence context.

Several approaches have been proposed to *train multiple tasks jointly*: Luan et al. [178] train a model on three tasks (coreference resolution, entity and relation extraction) using one dataset of research papers. Sanh et al. [246] introduce a multi-task model that is trained on four tasks (mention detection, coreference resolution, entity and relation extraction) with two different datasets. Wei et al. [290] utilise a multi-task model for entity recognition and relation extraction on one dataset in the non-academic domain. Comparable to us, Changpinyo et al. [46] analyse multi-task training with multiple datasets for sequence tagging. In contrast, we investigate sequential sentence classification across multiple science domains.

4.3 Cross-Domain Multi-Task Learning for Sequential Sentence Classification

On the one hand, the discussion of related work shows that several approaches and datasets from various scientific domains have been introduced for sequential sentence classification. On the other hand, while transfer learning has been applied to various NLP tasks, it is known that the success depends largely on the relatedness of the tasks [196, 205, 238]. However, the field lacks an empirical study on transfer learning between different scientific domains for sequential sentence classification that cover either only abstracts or entire papers. Furthermore, previous approaches investigated transfer learning for one or two datasets only. To the best of our knowledge, a unified approach for different types of texts that differ noticeably by their structure and semantic context of sentences, as it is the case for abstracts and full papers, has not been proposed yet.



(a) SciBERT-HSLN

(b) Sequential Transfer Learning (INIT 1 and INIT 2)

(c) Multi-Task Learning (MULT ALL)

(d) Multi-Task Learning (MULT GRP)

Figure 4.2: Proposed approaches for sequential sentence classification: (a) unified deep learning architecture *SciBERT-HSLN* for datasets of abstracts and full papers; (b) sequential transfer learning approaches, i.e. INIT 1 transfers all possible layers, INIT 2 only the sentence encoding layer; (c) and (d) are the multi-task learning approaches, i.e. in MULT ALL all possible layers are shared between the tasks, in MULT GRP the context enrichment is shared between tasks with the same text type.

In this section, we suggest a unified cross-domain multi-task learning approach for sequential sentence classification. Our tailored transfer learning approaches, depicted in Figure 4.2, exploit multiple datasets comprising different text types in form of abstracts and full papers. The unified approach without transfer learning is described in Section 4.3.1 while Section 4.3.2 introduces the sequential transfer learning and multi-task learning approaches. Finally, in Section 4.3.3, we present an approach to semi-automatically identify the semantic relatedness of sentence classes between different annotation schemes.

4.3.1 Unified Deep Learning Approach

Given a paper with the sentences $(\mathbf{s_1}, ..., \mathbf{s_n})$ and the set of dataset specific classes $\mathbb{L}$ (e.g. Background, Methods), the task of *sequential sentence classification* is to predict the corresponding label sequence $(\mathbf{y_1}, ..., \mathbf{y_n})$ with $\mathbf{y_i} \in \mathbb{L}$. For this task, we propose a unified deep learning approach as depicted in Figure 4.2(a), which is applicable to both abstracts *and* full papers. The core idea is to enrich sentence representations with context from surrounding sentences.

Our approach (denoted as *SciBERT-HSLN*) is based on the Hierarchical Sequential Labeling Network (HSLN) [133]. In contrast to Jin and Szolovits [133], we utilise SciBERT [19] as word embeddings and evaluate the approach on abstracts *as well as* full papers. We have chosen HSLN as the basis since it is better suited for full papers: It has no limitations on text length (in contrast to the approach of Cohan et al. [54]), and is computationally less expensive than the more recent approaches [257, 295]. Furthermore, their implementation is publicly available. The goal of this study is not to beat state-of-the-art results but rather to provide an empirical study on transfer learning and offer a uniform solution. Our *SciBERT-HSLN* architecture has the following layers:

Word Embedding: Input is a sequence of tokens $(\mathbf{t_{i,1}}, ..., \mathbf{t_{i,m}})$ of sentence $\mathbf{s_i}$, while output is a sequence of contextual word embeddings $(\mathbf{w_{i,1}}, ..., \mathbf{w_{i,m}})$.

Sentence Encoding: Input $(\mathbf{w_{i,1}}, ..., \mathbf{w_{i,m}})$ is transformed via a Bidirectional Long Short-Term Memory (Bi-LSTM) [116] into the hidden token representations $(\mathbf{h_{i,1}}, ..., \mathbf{h_{i,m}})$ $(\mathbf{h_{i,t}} \in \mathbb{R}^{d^h})$ which are enriched with contextual information within the sentence. Then, attention pooling [133, 298] with r heads produces a sentence vector $\mathbf{e_i} \in \mathbb{R}^{rd^u}$. An attention head produces a weighted average over the token representations of a sentence. Multiple heads enable to capture several semantics of a sentence. Formally, at first, a token representation $\mathbf{h_{i,t}}$ is transformed via a feed-forward network into a further hidden representation $\mathbf{a_{i,t}}$ with the learned weight matrix $\mathbf{W^{[S]}}$ and bias vector $\mathbf{b^{[S]}}$:

$$\mathbf{a_{i,t}} = FFN(\mathbf{h_{i,t}}) = \tanh(\mathbf{W^{[S]}}\mathbf{h_{i,t}} + \mathbf{b^{[S]}}) \tag{4.1}$$

Then, for each attention head k with $1 \leq k \leq r$ the learned token level context vector $\mathbf{u_k} \in \mathbb{R}^{d^u}$ is used to compute importance scores for all token representations which are then normalised by $softmax$:

$$\alpha_{k,i,t} = \frac{\exp(\mathbf{u_k^\intercal a_{i,t}})}{\sum_{t'} \exp(\mathbf{u_k^\intercal a_{i,t'}})} \tag{4.2}$$

An attention head $\mathbf{e_{k,i}} \in \mathbb{R}^{d^h}$ is computed as a weighted average over the token representations and all heads are concatenated to form the final sentence representation $\mathbf{e_i} \in \mathbb{R}^{rd^h}$:

$$\mathbf{e_{k,i}} = \sum_{t'} \alpha_{k,i,t'} \mathbf{h_{i,t'}} \tag{4.3}$$

$$\mathbf{e_i} = [\mathbf{e_{1,i}}, ..., \mathbf{e_{r,i}}] \tag{4.4}$$

Context Enrichment: This layer takes as input all sentence representations $(\mathbf{e_1}, ..., \mathbf{e_n})$ of the paper and outputs contextualised sentence representations $(\mathbf{c_1}, ..., \mathbf{c_n})$ $(\mathbf{c_i} \in \mathbb{R}^{d^h})$ via a Bi-LSTM. Thus, each sentence representation $\mathbf{c_i}$ contains contextual information from surrounding sentences.

Output Layer: This layer transforms sentence representations $(\mathbf{c_1}, ..., \mathbf{c_n})$ via a linear transformation to the logits $(\mathbf{l_1}, ..., \mathbf{l_n})$ with $\mathbf{l_i} \in \mathbb{R}^{|\mathbb{L}|}$. Each component of vector $\mathbf{l_i}$ contains a score for the corresponding label:

$$\mathbf{l_i} = \mathbf{W^{[O]}} \mathbf{c_i} + \mathbf{b^{[O]}} \tag{4.5}$$

Finally, the logits serve as input for a Conditional Random Field (CRF) [163] that predicts the label sequence $(\mathbf{\hat{y}_1}, ..., \mathbf{\hat{y}_n})$ $(\mathbf{\hat{y}_i} \in \mathbb{L})$ with the highest joint probability. A CRF captures linear (one step) dependencies between the labels (e.g. Methods are usually followed by Methods or Results). Therefore, a CRF learns a transition matrix $\mathbf{T} \in \mathbb{R}^{|\mathbb{L}| \times |\mathbb{L}|}$, where $\mathbf{T}_{l_1, l_2}$ represents the transition score from label l_1 to label l_2, and two vectors $\mathbf{b}, \mathbf{e} \in \mathbb{R}^{|\mathbb{L}|}$, where $\mathbf{b}_l$ and $\mathbf{e}_l$ represent the score of beginning and ending with label l, respectively. The objective is to find the label sequence with the highest conditional joint probability $P(\mathbf{\hat{y}_1}, ..., \mathbf{\hat{y}_n} | \mathbf{l_1}, ..., \mathbf{l_n})$. For this purpose, we define a score function for a label sequence $(\mathbf{\hat{y}_1}, ..., \mathbf{\hat{y}_n})$, that is a sum of the scores of the labels and the transition scores:

$$\mathit{score}((\mathbf{\hat{y}_1}, ..., \mathbf{\hat{y}_n}), (\mathbf{l_1}, ..., \mathbf{l_n})) = \mathbf{b}_{\hat{y}_1} + \sum_{t=1}^{n} \mathbf{l}_{\mathbf{t}, \hat{y}_t} + \sum_{t=1}^{n-1} \mathbf{T}_{\hat{y}_t, \hat{y}_{t+1}} + \mathbf{e}_{\hat{y}_m} \tag{4.6}$$

Then, the score is transformed to a probability value with $softmax$:

$$Z(\mathbf{l_1}, ..., \mathbf{l_n}) = \sum_{\mathbf{y'_1}, ..., \mathbf{y'_n}} \exp(\mathit{score}((\mathbf{y'_1}, ..., \mathbf{y'_n}), (\mathbf{l_1}, ..., \mathbf{l_n}))) \tag{4.7}$$

$$P(\mathbf{\hat{y}_1}, ..., \mathbf{\hat{y}_n} | \mathbf{l_1}, ..., \mathbf{l_n}) = \frac{\exp(\mathit{score}((\mathbf{\hat{y}_1}, ..., \mathbf{\hat{y}_n}), (\mathbf{l_1}, ..., \mathbf{l_n})))}{Z(\mathbf{l_1}, ..., \mathbf{l_n})} \tag{4.8}$$

The denominator $Z(.)$ represents a sum of the scores of all possible label sequences for the given logits. The Viterbi algorithm [88] is used to efficiently calculate the sequence with the highest score and the denominator (both with time complexity $O(|\mathbb{L}|^2 \cdot n)$).

During training, the CRF maximises $P(\mathbf{y_1}, ..., \mathbf{y_n} | \mathbf{l_1}, ..., \mathbf{l_n})$ of the ground truth labels for all m training samples $((\mathbf{x^{(1)}}, \mathbf{y^{(1)}}), ..., (\mathbf{x^{(m)}}, \mathbf{y^{(m)}}))$, where $\mathbf{x^{(i)}}$ represents the sentences of paper i and $\mathbf{y^{(i)}}$ the corresponding ground truth label sequence. Thus, the objective is to minimise the following loss function:

$$L = -\frac{1}{m} \sum_{i=1}^{m} \log P(\mathbf{y^{(i)}} | \mathbf{l^{(i)}}) \tag{4.9}$$

For regularisation, we use dropout after each layer. The SciBERT model is not fine-tuned since it requires the training of 110 Mio. additional parameters.

4.3.2 Transfer Learning Methods

For sequential sentence classification, we tailor and evaluate the following transfer learning methods.

Sequential Transfer Learning (INIT): The approach first trains the model for the source task and uses its tuned parameters to initialise the parameters for the target task. Then, the parameters are fine-tuned with the labelled data of the target task. As depicted in Figure 4.2(b), we propose two types of layer transfers. *INIT 1*: transfer parameters of *context enrichment* and *sentence encoding*; *INIT 2*: transfer parameters of *sentence encoding.* Other layers, except *word embedding*, of the target task are initialised with random values.

Multi-Task Learning (MULT): Multi-task learning (MULT) aims for a better generalisation by simultaneously training samples in all tasks and sharing parameters of certain layers between the tasks. As depicted in Figure 4.2(c,d), we propose two multi-task learning architectures. The *MULT ALL* model shares all layers between the tasks except the *output layers* so that the model learns a common feature extractor for all tasks. However, full papers are much longer and have a different rhetorical structure compared to abstracts. Therefore, it is not beneficial to share the context enrichment layer between both dataset types. Thus, in the *MULT GRP* model, the *context enrichment layers* are only shared between datasets with the same text type. Formally, the objective is to minimise the following loss functions:

$$L_{\text{MULT ALL}} = \sum_{t \in \mathbb{T}^{\mathbb{A}} \cup \mathbb{T}^{\mathbb{F}}} L_t(\Theta^S, \Theta^C, \Theta_t^O) \tag{4.10}$$

$$L_{\text{MULT GRP}} = \sum_{t \in \mathbb{T}^{\mathbb{A}}} L_t(\Theta^S, \Theta^{C^A}, \Theta_t^O) + \sum_{t \in \mathbb{T}^{\mathbb{F}}} L_t(\Theta^S, \Theta^{C^F}, \Theta_t^O) \tag{4.11}$$

where $\mathbb{T}^{\mathbb{A}}$ and $\mathbb{T}^{\mathbb{F}}$ are the tasks for datasets containing abstracts and full papers, respectively; L_t is the loss function for task t; the parameters Θ^S are for sentence encoding, Θ^C, Θ^{C^A}, Θ^{C^F} for context enrichment, and Θ_t^O for the output layer of task t.

Furthermore, we propose the variants *MULT ALL SHO* and *MULT GRP SHO* that are applicable if all tasks share the same (domain-independent) set of classes. *MULT ALL SHO* shares all layers among all tasks. *MULT GRP SHO* shares the context enrichment and output layer only between tasks with the same text type. The loss functions are defined as:

$$L_{\text{MULT ALL SHO}} = \sum_{t \in \mathbb{T}^{\mathbb{A}} \cup \mathbb{T}^{\mathbb{F}}} L_t(\Theta^S, \Theta^C, \Theta^O) \tag{4.12}$$

$$L_{\text{MULT GRP SHO}} = \sum_{t \in \mathbb{T}^{\mathbb{A}}} L_t(\Theta^S, \Theta^{C^A}, \Theta^{O^A}) + \sum_{t \in \mathbb{T}^{\mathbb{F}}} L_t(\Theta^S, \Theta^{C^F}, \Theta^{O^F}) \tag{4.13}$$

4.3.3 Semantic Relatedness of Classes

Datasets for sentence classification have different domain-specific annotation schemes, that is different sets of pre-defined classes. Intuitively, some classes have a similar meaning across domains, e.g. the classes Model and Experiment in the ART corpus are semantically related to Methods in PubMed-20k (PMD) (see Table 4.2). An analysis of semantic relatedness can help consolidate different annotation schemes.

We propose machine learning models to support the identification of semantically related classes according to the following idea: if a model trained for PMD recognises sentences labelled with ART:Model as PMD:Method, and vice versa, then the classes ART:Model and PMD:Method can be assumed to be semantically related.

Let $\mathbb{T}$ be the set of all tasks, $\mathbb{L}$ the set of all classes in all tasks, $m_t(s)$ the label of sentence s predicted by the model for task t, and $\mathbb{S}^l$ the set of sentences with the ground truth label l. For each class $l \in \mathbb{L}$ the corresponding semantic vector $\mathbf{v_l} \in \mathbb{R}^{|\mathbb{L}|}$ is defined as:

$$\mathbf{v}_{\mathbf{l},l'} = \frac{\sum_{t \in \mathbb{T}, s \in \mathbb{S}^l} \mathbb{1}(m_t(s) = l')}{|\mathbb{S}^l|} \tag{4.14}$$

where $\mathbf{v}_{\mathbf{l},l'} \in \mathbb{R}$ is the component of the vector $\mathbf{v_l}$ for class $l' \in \mathbb{L}$ and $\mathbb{1}(p)$ is the indicator function that returns 1 if p is true and 0 otherwise. Intuitively, the semantic vectors concatenated vertically to a matrix represent a "confusion matrix" (see Figure 4.4 as an example).

Now, we define the semantic relatedness of two classes $k, l \in \mathbb{L}$ using cosine similarity:

$$\text{semantic\_relatedness}(k, l) = \cos(\mathbf{v_k}, \mathbf{v_l}) = \frac{\mathbf{v_k^{\intercal}} \cdot \mathbf{v_l}}{||\mathbf{v_k}|| \cdot ||\mathbf{v_l}||} \tag{4.15}$$

4.4 Experimental Setup

This section describes the experimental evaluation of the proposed approaches, i.e. used datasets, implementation details, and evaluation methods.

Table 4.2: Characteristics of the benchmark datasets. The row "State of the art" depicts the best results for approaches that do not exploit the ground truth label of the preceding sentence during prediction: for PMD [295], for NIC [257], for DRI [11] (cf. Table 7), and for ART [174].

	PMD	**NIC**	**DRI**	**ART**
Domains	Biomedicine	Biomedicine	Computer Graphics	Chemistry, Computational Linguistic
Text Type	Abstract	Abstract	Full paper	Full paper
# Papers	20.000	1.000	40	225
# Sentences	235.892	9.771	8.777	34.680
∅ # Sentences	12	10	219	154
# Classes	5	6	5	11
Classes	Background Objective Methods Results Conclusion	Background Intervention Study Population Outcome Other	Background Challenge Approach Outcome FutureWork	Background Motivation Hypothesis Goal Object Experiment Model Method Observation Result Conclusion
State of the art	93.1	86.8	72.5	51.6
Original metric	weighted F1	weighted F1	weighted F1	accuracy

4.4.1 Investigated Datasets

Table 4.2 summarises the characteristics of the investigated datasets, namely PubMed-20k (PMD) [68], NICTA-PIBOSO (NIC) [147], ART [175], and Dr. Inventor (DRI) [85]. The four datasets are publicly available and provide a good mix to investigate the transferability: They represent four different scientific domains; PMD and NIC cover abstracts and are from the same domain but have different annotation schemes; DRI and ART cover full papers but are from different domains and have different annotation schemes; NIC and DRI are rather small datasets, while PMD and ART are about 20 and 3 times larger, respectively; ART has a much finer annotation scheme compared to other datasets. As denoted in Table 4.2, the state-of-the-art results for ART are the lowest ones since ART has more fine-grained classes than the other datasets. In contrast, best results are obtained for PMD: It is a large dataset sampled from PubMed, where authors are encouraged to structure their abstracts. Therefore, abstracts in PMD are more uniformly structured than in other datasets, leading to better classification results.

4.4.2 Implementation

Our approaches are implemented in PyTorch [210]. The Adaptive Moment Estimation (ADAM) optimiser [148] with 0.01 weight decay and an exponential learning rate decay of 0.9 after each epoch is used for training. To speed up training, sentences longer than 128 tokens are truncated since the computational cost for the attention layers in BERT is quadratic in sentence length [283]. To reproduce the results of the original HSLN architecture, we tuned SciBERT-HSLN for PMD and NIC with hyperparameters as proposed in other studies [71, 133]. The following parameters performed best on the validation sets of PMD and NIC: learning rate 3e-5, dropout rate 0.5, Bi-LSTM hidden size $d^h = 2 \cdot 758$, $r = 15$ attention heads of size $d^u = 200$. We used these hyperparameters in all our experiments.

For each dataset, we grouped papers to mini-batches without splitting them, if the mini-batch does not exceed 32 sentences. Thus, for full papers a mini-batch may consist of sentences from only one paper. During multi-task training we switched between the mini-batches of the tasks by proportional sampling [246]. After a mini-batch, only task-related parameters are updated, i.e. the associated output layer and all the layers below.

4.4.3 Evaluation

To be consistent with previous results and due to non-determinism in deep neural networks [234], we repeated the experiments and averaged the results. According to [54] we performed three random restarts for PMD and NIC and used the same train/validation/test sets. For DRI and ART, we performed 10-fold and 9-fold cross-validation, respectively, as in the original papers [85, 174]. Within each fold the data is split into train/validation/test sets with the proportions $\frac{k-2}{k}/\frac{1}{k}/\frac{1}{k}$ where k is the number of folds. For multi-task learning, the experiment was repeated with the maximum number of folds of the datasets used, but at least three times. All models were trained for 20 epochs. The test set performance within a fold and restart, respectively, was calculated for the epoch with the best validation performance.

We compare our results only with approaches which do not exploit *ground truth labels* of the preceding sentence as a feature *during prediction* (see Section 4.2.1). This has a significant impact on the performance: Using the ground truth label of the previous sentences as a sole input feature to a SVM classifier already yields an accuracy of 77.7 for DRI and 55.5 for ART (compare also results for the “history” feature in [11], cf. Table 5). Best reported results using ground truth labels as input features have an accuracy of 84.15 for DRI and 65.75 for ART [6]. In contrast, we pursue a realistic setting by exploiting the *predicted* (not ground truth) label of neighbouring sentences during prediction.

Moreover, we provide additional results for three strong deep learning baselines: (1) fine-tuning SciBERT using the [CLS] token of individual sentences as in [71] (referred to

Table 4.3: Experimental results for the proposed approaches (in perent): our SciBERT-HSLN model without transfer learning, parameter initialisation (INIT), and multi-task learning (MULT ALL and MULT GRP). Previous state of the art (see Table 4.2), SciBERT-[CLS], original HSLN approach of Jin and Szolovits [133], and the approach of Cohan et al. [54] are the baseline results. For PMD (P), NIC (N), and DRI (D) we report weighted F1 score and for ART (A) accuracy. The average of all scores is denoted by ∅. *Italics* depicts whether the result is better than the baseline, **bold** whether the transfer method improves SciBERT-HSLN, underline the best overall result.

	PMD	**NIC**	**DRI**	**ART**	∅
Previous state of the art	[295] 93.1	[257] 86.8	[11] 72.5	[174] 51.6	76.0
SciBERT-[CLS]	89.6	78.4	69.5	51.5	72.3
Jin and Szolovits [133] (HSLN)	92.6	84.7	75.3	49.3	75.5
Cohan et al. [54]	92.9	84.8	74.3	54.3	76.6
SciBERT-HSLN	92.9	*84.9*	*78.0*	*58.0*	*78.5*
INIT 1 PMD to *T*	-	84.8	***81.2***	*57.7*	
INIT 2 PMD to *T*	-	84.8	***80.1***	*58.0*	
INIT 1 NIC to *T*	92.9	-	***81.9***	*57.6*	
INIT 2 NIC to *T*	92.9	-	***79.6***	*57.2*	
INIT 1 DRI to *T*	92.9	83.5	-	*57.8*	
INIT 2 DRI to *T*	92.9	83.8	-	*57.6*	
INIT 1 ART to *T*	**93.0**	84.7	***82.2***	-	
INIT 2 ART to *T*	92.9	84.7	***81.0***	-	
MULT ALL	**93.0**	**86.0**	***81.8***	*57.7*	***79.6***
PMD, NIC	**93.0**	**86.1**	-	-	
PMD, DRI	92.9	-	***80.6***	-	
PMD, ART	**93.0**	-	-	*58.0*	
NIC, DRI	-	84.2	***80.7***	-	
NIC, ART	-	84.4	-	*57.9*	
DRI, ART	-	-	***82.0***	*57.6*	
PMD, NIC, DRI	**93.0**	**86.2**	***81.0***	-	
PMD, NIC, ART	**93.0**	**86.3**	-	*58.0*	
PMD, DRI, ART	**93.0**	-	***82.7***	*57.8*	
NIC, DRI, ART	-	84.7	***82.0***	*57.7*	
MULT GRP	**93.0**	**86.1**	***83.4***	***58.8***	***80.3***
P,N,D,A	92.9	**85.4**	***84.4***	*58.0*	***80.2***
(P,D),(N,A)	**93.0**	**86.0**	***81.1***	***58.5***	***79.7***
(P,A),(N,D)	92.9	**85.8**	***83.6***	*58.0*	***80.1***
(P,N,D),(A)	92.9	**86.0**	***80.6***	***58.2***	***79.4***
(P,N,A),(D)	**93.0**	**86.0**	***84.1***	***58.1***	***80.3***
(P,D,A),(N)	92.9	**85.5**	***82.2***	*58.0*	***79.6***
(N,D,A),(P)	92.9	**85.9**	***83.3***	***58.5***	***80.1***

as SciBERT-[CLS]), (2) original implementation of Jin and Szolovits [133], and (3) the SciBERT-based approach of Cohan et al. [54]. We cannot provide baseline results for DRI and ART of the approaches [257, 295] since their implementations are not publicly available.

4.5 Results and Discussion

In this section, we present and discuss the experimental results for our proposed cross-domain multi-task learning approach for sequential sentence classification. The results for different variations of our approach, the respective baselines, and for several state-of-the-art methods are depicted in Table 4.3. The results are discussed in the following three subsections regarding the unified approach without transfer learning (Section 4.5.1), with sequential transfer learning (Section 4.5.2), and multi-task learning (Section 4.5.3). Section 4.5.4 analyses the semantic relatedness of classes for the four annotation schemes.

4.5.1 Unified Approach without Transfer Learning (SciBERT-HSLN)

For the full paper datasets DRI and ART, our SciBERT-HSLN model significantly outperforms the previously reported best results, and the deep learning baselines SciBERT-[CLS], Jin and Szolovits [133], and Cohan et al. [54]. The previous state of the art approaches for DRI and ART [11, 174] require feature engineering and a sentence is enriched only with the context of the previous sentence. In SciBERT-[CLS], each sentence is classified in isolation. The original HSLN architecture [133] uses shallow word embeddings pre-trained on biomedical texts. Thus, the incorporation of SciBERT's contextual word embeddings into HSLN helps improve performance for the DRI and ART datasets. The approach of Cohan et al. [54] can process only about 10 sentences at once since SciBERT supports sequences of up to 512 tokens only. Thus, long text has to be split into multiple chunks. Our deep learning approach can process *all* sentences of a paper at once so that all sentences are enriched with context from surrounding sentences.

For the PMD dataset, our SciBERT-HSLN results are equivalent [295] to the current state of the art, while for NIC, they are below [257]. Thus, our proposed approach is competitive with the current approaches for sequential sentence classification in abstracts.

Our unified deep learning approach is applicable to datasets consisting of different text types, i.e. abstracts and full papers, without any feature engineering (#Q7).

4.5.2 Sequential Transfer Learning (INIT)

Using the INIT approach, we can only improve the baseline results for the DRI dataset in all settings. The approach INIT 1 performs better than INIT 2 in most cases which indicates that transferring all parameters is more effective.

However, the results suggest that sequential transfer learning is not a very effective transfer method for sequential sentence classification (#Q2).

4.5.3 Multi-Task Learning (MULT)

Next, we discuss the results of our multi-task learning approach, and the effects of multi-task learning on smaller datasets and individual sentence classes.

MULT ALL Model: In this setting, all tasks were trained jointly sharing all possible layers. Except for the ART task, all results are improved using the SciBERT-HSLN model. For the PMD task, the improvement is marginal since the baseline results (F1 score) were already on a high level. Pairwise MULT ALL combinations show that the models for PMD and NIC, respectively, benefit from the (respective) other dataset, and the DRI model especially from the ART dataset. The PMD and NIC datasets are from the same domain, and both contain abstracts, so the results are as expected. Furthermore, DRI and ART datasets both contain full papers, and DRI has more coarse-grained classes. However, ART is a larger dataset with fine-grained classes and presumably therefore the model for ART does not benefit from other datasets. In triple-wise MULT ALL combinations the models for PMD and DRI, respectively, benefit from all datasets, and the model for NIC only if the PMD dataset is present.

The results suggest that sharing all possible layers between multiple tasks is effective except for bigger datasets with more fine-grained classes (#Q3, #Q4).

MULT GRP Model: In this setting, the models for all tasks were trained jointly, but only models for the same text type share the *context enrichment layer*, i.e. (PMD, NIC) and (DRI, ART). Here, all models benefit from the other datasets. In our ablation study, we also provide results for sharing only the *sentence encoding layer*, referred to as MULT GRP P,N,D,A, and all pairwise and triple-wise combinations sharing the *context enrichment layer*. Other combinations also yield good results. However, MULT GRP is effective for *all* tasks.

Our results indicate that sharing the sentence encoding layer between multiple models is beneficial. Furthermore, sharing the context enrichment layer only between models for the same text type is an even more effective strategy (#Q3, #Q4).

Effect of Dataset Size: The NIC and DRI models benefit more from multi-task learning than PMD and ART. However, PMD and ART are bigger datasets than NIC and DRI. The ART dataset has also more fine-grained classes than the other datasets. This raises the following question:

How would the models for PMD and ART benefit from multi-task learning if they were trained on smaller datasets?

Table 4.4: Experimental results (in percent) for μPMD, NIC, DRI, and μART with our SciBERT-HSLN model and our proposed multi-task learning approaches.

	μPMD	**NIC**	**DRI**	**μART**	∅
SciBERT-HSLN	90.9	84.9	78.0	52.2	76.5
MULT ALL	**91.1**	**85.7**	**81.0**	**53.8**	**77.9**
MULT GRP	**91.1**	**85.9**	**82.2**	**55.1**	**78.6**

To answer this question, we created smaller variants of PMD and ART, referred to as μPMD and μART, with a comparable size with NIC and DRI. Within each fold we truncated the training data to $\frac{1}{20}$ for μPMD and $\frac{1}{3}$ for μART while keeping the original size of the validation and test sets. As shown in Table 4.4, all models benefit from the other datasets, whereas the MULT GRP model again performs best.

The results indicate that models for small datasets benefit from multi-task learning independent of the difference in the granularity of the classes (#Q1).

Effect for each Class: Figure 4.3 shows the F1 scores per class for the investigated approaches. Classes, which are intuitively highly semantically related (*:Background, *:Results, *:Outcome), and classes with few examples (DRI:FutureWork, DRI:Challenge, ART:Hypothesis, NIC:Study Design) tend to benefit significantly from multi-task learning. The classes ART:Model, ART:Observation, and ART:Result have worse results than SciBERT-HSLN when using MULT ALL, but MULT GRP yields better results. This can be attributed to sharing the *context enrichment layers* only between datasets with the same text type.

The analysis suggests that especially semantically related classes and classes with few examples benefit from multi-task learning (#Q1).

4.5.4 Semantic Relatedness of Classes across Annotation Schemes

In this section, we first evaluate our proposed approach for the semi-automatical identification of semantically related classes in the datasets PMD, NIC, DRI, and ART. Based on the analysis, we identify six clusters of semantically related classes. Then, we present a new dataset that is compiled from the investigated datasets and is based on the identified clusters. As a possible down-stream application, this multi-domain dataset with a generic set of classes could help to structure research papers in a domain-independent manner, supporting, for instance, the development of academic search engines.

Analysis of Semantic Relatedness of Classes: Based on the annotation guidelines of the investigated datasets PMD [68], NIC [147], DRI [85], and ART [175], we identified six

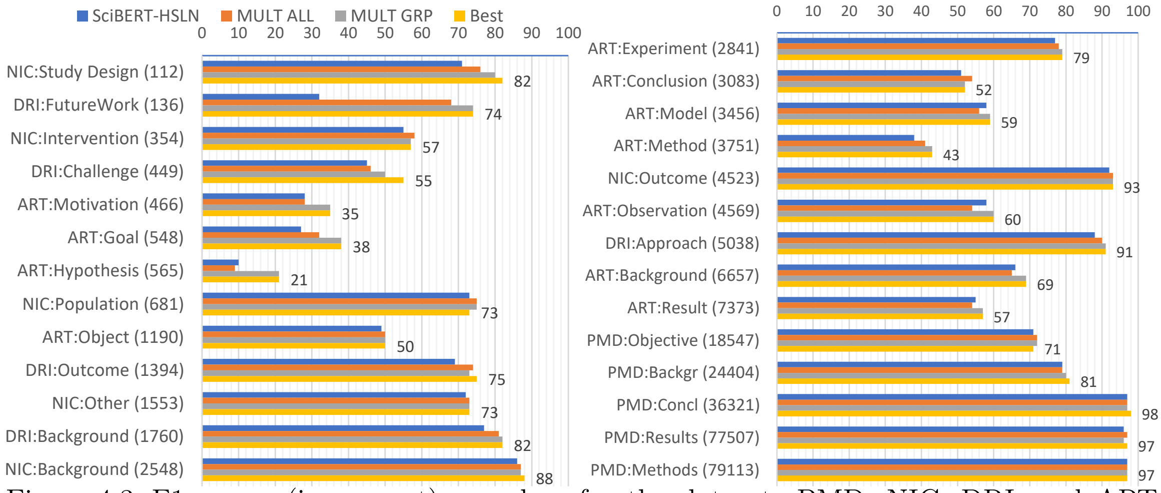
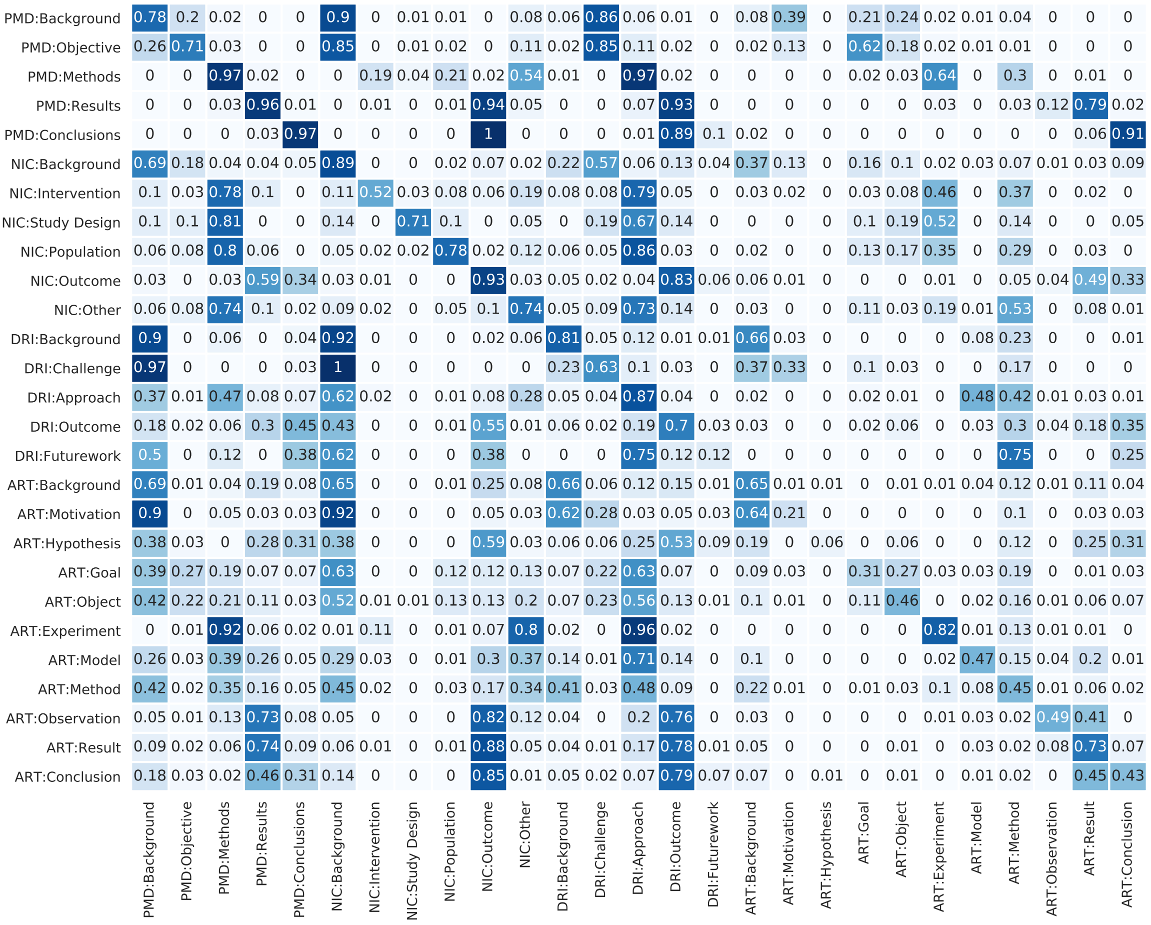


Figure 4.3: F1 scores (in percent) per class for the datasets PMD, NIC, DRI, and ART for the approaches SciBERT-HSLN, MULT ALL, MULT GRP, and the best combination for the respective dataset. Numbers at the bars depict the F1 scores of the best classifiers and in brackets the number of examples for the given class. The classes are ordered by the number of examples.

clusters of semantically related classes, which are depicted in Figure 4.5. The identification process of the clusters followed the intuition, that most research papers independent of the scientific domain (1) investigate a research problem (Problem), (2) provide background information for the problem (Background), (3) apply or propose certain methods (Methods), (4) yield results (Results), (5) conclude the work (Conclusions), and (6) outline future work (Future Work).

Figure 4.4 shows the semantic vectors for all classes computed with the MULT ALL model. It can be observed that some semantic vectors look similar, e.g. PMD:Background and DRI:Background. We computed the semantic vectors also with SciBERT-HSLN and MULT GRP and projected them onto a 2D space using Principal Component Analysis (PCA) [135], as shown in Figure 4.5. It can be seen that already for the SciBERT-HSLN classifiers our approach enables to identify semantically related classes (e.g. see Results cluster). However, the MULT ALL model yields more meaningful clusters. Except Problem, all clusters for semantically related classes are well identifiable in Figure 4.5(c). Although MULT GRP performs best, the clusters are not consistent in Figure 4.5(b). The semantic vector for ART:Hypothesis is an outlier in the Problem cluster in Figure 4.5(c), because ART:Hypothesis is confused mostly with ART:Conclusion and ART:Result (see Figure 4.4) and has also a very low F1 score (see Figure 4.3).

Table 4.5 shows the Silhouette scores [237] for each cluster. A positive Silhouette score indicates that objects lie well within the cluster, and a negative score that the objects are merely somewhere in between clusters. As a distance metric, we use semantic_relatedness as defined in Equation 4.15. The Silhouette scores also confirm that MULT ALL forms better clusters than SciBERT-HSLN and MULT GRP. We hypothesise that MULT ALL



	PMD:Background	PMD:Objective	PMD:Methods	PMD:Results	PMD:Conclusions	NIC:Background	NIC:Intervention	NIC:Study Design	NIC:Population	NIC:Outcome	NIC:Other	DRI:Background	DRI:Challenge	DRI:Approach	DRI:Outcome	DRI:Futurework	ART:Background	ART:Motivation	ART:Hypothesis	ART:Goal	ART:Object	ART:Experiment	ART:Model	ART:Method	ART:Observation	ART:Result	ART:Conclusion
PMD:Background	0.78	0.2	0.02	0	0	0.9	0	0.01	0.01	0	0.08	0.06	0.86	0.06	0.01	0	0.08	0.39	0	0.21	0.24	0.02	0.01	0.04	0	0	0
PMD:Objective	0.26	0.71	0.03	0	0	0.85	0	0.01	0.02	0	0.11	0.02	0.85	0.11	0.02	0	0.02	0.13	0	0.62	0.18	0.02	0.01	0.01	0	0	0
PMD:Methods	0	0	0.97	0.02	0	0	0.19	0.04	0.21	0.02	0.54	0.01	0	0.97	0.02	0	0	0	0	0.02	0.03	0.64	0	0.3	0	0.01	0
PMD:Results	0	0	0.03	0.96	0.01	0	0.01	0	0.01	0.94	0.05	0	0	0.07	0.93	0	0	0	0	0	0	0.03	0	0.03	0.12	0.79	0.02
PMD:Conclusions	0	0	0	0.03	0.97	0	0	0	0	1	0	0	0	0.01	0.89	0.1	0.02	0	0	0	0	0	0	0	0	0.06	0.91
NIC:Background	0.69	0.18	0.04	0.04	0.05	0.89	0	0	0.02	0.07	0.02	0.22	0.57	0.06	0.13	0.04	0.37	0.13	0	0.16	0.1	0.02	0.03	0.07	0.01	0.03	0.09
NIC:Intervention	0.1	0.03	0.78	0.1	0	0.11	0.52	0.03	0.08	0.06	0.19	0.08	0.08	0.79	0.05	0	0.03	0.02	0	0.03	0.08	0.46	0	0.37	0	0.02	0
NIC:Study Design	0.1	0.1	0.81	0	0	0.14	0	0.71	0.1	0	0.05	0	0.19	0.67	0.14	0	0	0	0	0.1	0.19	0.52	0	0.14	0	0	0.05
NIC:Population	0.06	0.08	0.8	0.06	0	0.05	0.02	0.02	0.78	0.02	0.12	0.06	0.05	0.86	0.03	0	0.02	0	0	0.13	0.17	0.35	0	0.29	0	0.03	0
NIC:Outcome	0.03	0	0.03	0.59	0.34	0.03	0.01	0	0	0.93	0.03	0.05	0.02	0.04	0.83	0.06	0.06	0.01	0	0	0	0.01	0	0.05	0.04	0.49	0.33
NIC:Other	0.06	0.08	0.74	0.1	0.02	0.09	0.02	0	0.05	0.1	0.74	0.05	0.09	0.73	0.14	0	0.03	0	0	0.11	0.03	0.19	0.01	0.53	0	0.08	0.01
DRI:Background	0.9	0	0.06	0	0.04	0.92	0	0	0	0.02	0.06	0.81	0.05	0.12	0.01	0.01	0.66	0.03	0	0	0	0	0.08	0.23	0	0	0.01
DRI:Challenge	0.97	0	0	0	0.03	1	0	0	0	0	0	0.23	0.63	0.1	0.03	0	0.37	0.33	0	0.1	0.03	0	0	0.17	0	0	0
DRI:Approach	0.37	0.01	0.47	0.08	0.07	0.62	0.02	0	0.01	0.08	0.28	0.05	0.04	0.87	0.04	0	0.02	0	0	0.02	0.01	0	0.48	0.42	0.01	0.03	0.01
DRI:Outcome	0.18	0.02	0.06	0.3	0.45	0.43	0	0	0.01	0.55	0.01	0.06	0.02	0.19	0.7	0.03	0.03	0	0	0.02	0.06	0	0.03	0.3	0.04	0.18	0.35
DRI:Futurework	0.5	0	0.12	0	0.38	0.62	0	0	0	0.38	0	0	0	0.75	0.12	0.12	0	0	0	0	0	0	0	0.75	0	0	0.25
ART:Background	0.69	0.01	0.04	0.19	0.08	0.65	0	0	0.01	0.25	0.08	0.66	0.06	0.12	0.15	0.01	0.65	0.01	0.01	0	0.01	0.01	0.04	0.12	0.01	0.11	0.04
ART:Motivation	0.9	0	0.05	0.03	0.03	0.92	0	0	0	0.05	0.03	0.62	0.28	0.03	0.05	0.03	0.64	0.21	0	0	0	0	0	0.1	0	0.03	0.03
ART:Hypothesis	0.38	0.03	0	0.28	0.31	0.38	0	0	0	0.59	0.03	0.06	0.06	0.25	0.53	0.09	0.19	0	0.06	0	0.06	0	0	0.12	0	0.25	0.31
ART:Goal	0.39	0.27	0.19	0.07	0.07	0.63	0	0	0.12	0.12	0.13	0.07	0.22	0.63	0.07	0	0.09	0.03	0	0.31	0.27	0.03	0.03	0.19	0	0.01	0.03
ART:Object	0.42	0.22	0.21	0.11	0.03	0.52	0.01	0.01	0.13	0.13	0.2	0.07	0.23	0.56	0.13	0.01	0.1	0.01	0	0.11	0.46	0	0.02	0.16	0.01	0.06	0.07
ART:Experiment	0	0.01	0.92	0.06	0.02	0.01	0.11	0	0.01	0.07	0.8	0.02	0	0.96	0.02	0	0	0	0	0	0	0.82	0.01	0.13	0.01	0.01	0
ART:Model	0.26	0.03	0.39	0.26	0.05	0.29	0.03	0	0.01	0.3	0.37	0.14	0.01	0.71	0.14	0	0.1	0	0	0	0	0.02	0.47	0.15	0.04	0.2	0.01
ART:Method	0.42	0.02	0.35	0.16	0.05	0.45	0.02	0	0.03	0.17	0.34	0.41	0.03	0.48	0.09	0	0.22	0.01	0	0.01	0.03	0.1	0.08	0.45	0.01	0.06	0.02
ART:Observation	0.05	0.01	0.13	0.73	0.08	0.05	0	0	0	0.82	0.12	0.04	0	0.2	0.76	0	0.03	0	0	0	0	0.01	0.03	0.02	0.49	0.41	0
ART:Result	0.09	0.02	0.06	0.74	0.09	0.06	0.01	0	0.01	0.88	0.05	0.04	0.01	0.17	0.78	0.01	0.05	0	0	0	0.01	0	0.03	0.02	0.08	0.73	0.07
ART:Conclusion	0.18	0.03	0.02	0.46	0.31	0.14	0	0	0	0.85	0.01	0.05	0.02	0.07	0.79	0.07	0.07	0	0.01	0	0.01	0	0.01	0.02	0	0.45	0.43

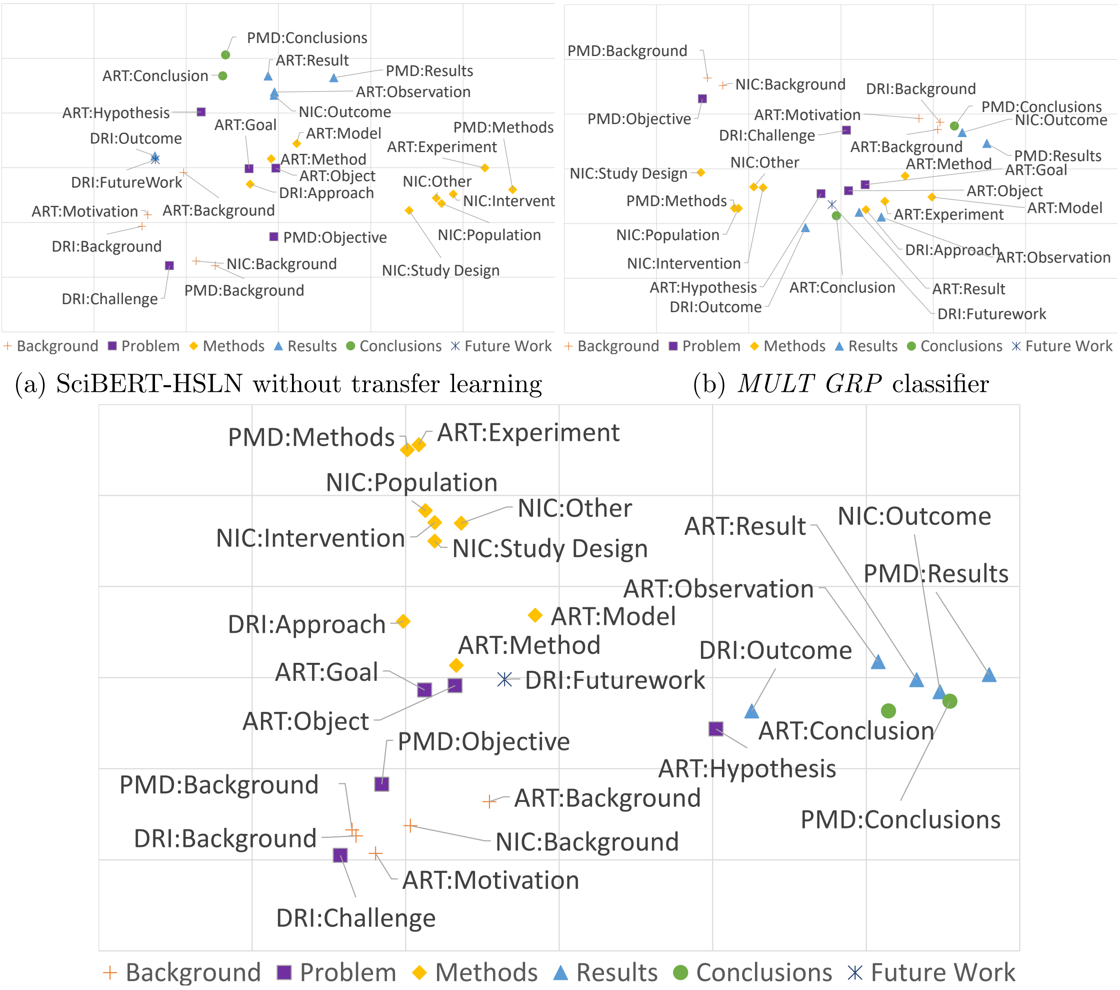
Figure 4.4: Each row represents a semantic vector representation as described in Section 4.3.3 for a class computed with the *MULT ALL* classifier.

Table 4.5: Silhouette scores per cluster and overall computed for the semantic vectors with the SciBERT-HSLN, MULT GRP, and MULT ALL classifiers.

	SciBERT HSLN	MULT GRP	MULT ALL
Background	0.45	0.18	**0.48**
Problem	-0.27	**-0.04**	-0.29
Methods	0.19	-0.03	**0.31**
Results	-0.38	0.01	**0.32**
Conclusions	**0.92**	-0.49	0.02
Future Work	0.00	0.00	0.00
Overall	0.10	-0.02	**0.20**

can better capture the semantic relatedness of classes than the other approaches since it is enforced to learn a generic feature extractor across multiple datasets.

The multi-task learning approach sharing all possible layers can recognise semantically related classes (#Q5).



(a) SciBERT-HSLN without transfer learning

(b) *MULT GRP* classifier

(c) *MULT ALL* classifier

Figure 4.5: Semantic vectors of classes computed by (a) SciBERT-HSLN model without transfer learning, (b) *MULT GRP* model, and (c) *MULT ALL* model, and projected to 2D space using PCA. The semantic vectors are assigned to generic clusters of semantically related labels.

Domain-Independent Sentence Classification: Based on the identified clusters, we compile a new dataset *G-PNDA* from the investigated datasets PMD, NIC, DRI, and ART. The labels of the datasets are collapsed according to the clusters in Figure 4.5. Table 4.6 summarises the characteristics of the compiled dataset. To prevent a bias towards bigger datasets, we truncate PMD to $\frac{1}{20}$ and ART to $\frac{1}{3}$ of their original size.

Table 4.7 depicts our experimental settings and results for the generic dataset *G-PNDA*. We train a model for each dataset part, and the multi-task learning models MULT ALL and MULT GRP. Since we have common sentence classes now, we train also models that share the output layers between the dataset parts, referred to as MULT ALL SHO and MULT GRP SHO (see Section 4.3.2). For training and evaluation, we split each dataset part into

Table 4.6: Characteristics of the domain-independent dataset *G-PNDA* that was compiled from the origin datasets *PMD*, *NIC*, *DRI*, and *ART*.

	G-PMD	**G-NIC**	**G-DRI**	**G-ART**
Text Type	Abstract	Abstract	Full paper	Full paper
# Papers	1.000	1.000	40	67
# Sentences	11.738	9.771	8.777	9.528
∅ # Sentences	11	10	219	142
Background	1.220	2.548	1.760	1.657
Problem	953	0	449	529
Methods	3.927	2.700	5.038	2.752
Results	3.760	4.523	1.394	3.672
Conclusions	1.878	0	0	918
Future Work	0	0	136	0

Table 4.7: Experimental results in terms of F1 scores (in percent) for our proposed approaches for the generic dataset *G-PNDA*: baseline model SciBERT-HSLN with one separate model per dataset and the multi-task learning models MULT ALL SHO, MULT ALL, MULT GRP SHO, and MULT GRP. **Bold** depicts whether the approach improves the baseline, underline the best overall result.

	G-PMD	G-NIC	G-DRI	G-ART	∅
SciBERT-HSLN (one model per dataset)	90.1	89.3	81.7	70.8	83.0
MULT ALL SHO (shared output layer)	89.8	89.1	**83.5**	67.1	82.4
MULT ALL (separate output layer)	**90.5**	**89.8**	**84.9**	70.5	**83.9**
MULT GRP SHO (shared output layer)	90.0	**89.9**	**86.1**	70.4	**84.1**
MULT GRP (separate output layer)	**90.6**	**89.7**	**87.2**	**71.0**	**84.6**

train/validation/test sets with the portions 70/10/20, average the results over three random restarts and use the same hyperparameters as before (see Section 4.4.2).

Table 4.7 shows that the proposed MULT GRP model outperforms all other settings. Surprisingly, sharing the output layer impairs the performance in all settings. We can attribute this to the fact that the output layer learns different transition distributions between the classes.

Thus, in a domain-independent setting a separate output layer per dataset part helps the model to capture the individual rhetorical structure present in the domains (#Q3, #Q6).

4.6 Summary

In this chapter, we explored **RQ2** (adaptable to new domains with few labelled data) and **RQ3** (domain-independent extraction) for the task of sequential sentence classification, which were divided into the sub-questions #Q1-#Q7. This is an important task that can help researchers to assess the relevance of research papers more effectively (see also the use case *assess relevance of research articles* in Chapter 3).

To investigate **RQ2**, we presented a unified deep learning architecture for the task of sequential sentence classification. The unified approach can be applied to different types of text with a differing structure, e.g. abstracts as well as full papers. For datasets of full papers, our approach significantly outperforms the state of the art without any feature engineering (#Q7).

Then, we have tailored two common transfer learning approaches to sequential sentence classification and compared their performance. We found that training a multi-task model with multiple datasets works better than sequential transfer learning (#Q2). Our comprehensive experimental evaluation with four different datasets offers useful insights under which conditions transferring or sharing of specific layers is beneficial or not (#Q3). In particular, it is always beneficial to share the sentence encoding layer between datasets from different domains. However, it is most effective to share the context enrichment layer, which encodes the context of neighbouring sentences, only between datasets with the same text type. This can be attributed to different rhetorical structures in abstracts and full papers.

Our tailored multi-task learning approach makes use of multiple datasets and yields new state-of-the-art results for two full paper datasets, i.e. DRI [85] with 84.4% F1 (+11.9% absolute improvement) and ART [175] with 58.8% accuracy (+7.1% absolute improvement) (#Q4). In particular, models for tasks with small datasets and classes with few labelled examples benefit significantly from models of other tasks. Our study suggests that the classes of the different dataset annotation schemes are semantically related, even though the datasets come from different domains and have different text types (#Q1). This semantic relatedness is an important prerequisite for transfer learning in NLP tasks [196, 205, 238].

Finally, to address **RQ3**, we have proposed an approach to semi-automatically identify semantically related classes from different datasets to support manual comparison and inspection of different annotation schemes across domains. We demonstrated the usefulness of the approach with an analysis of four annotation schemes. This approach can support the investigation of annotation schemes across disciplines without re-annotating datasets (#Q5). From the analysis, we have derived a domain-independent consolidated annotation scheme and compiled a domain-independent dataset. This allows for the classification of sentences in research papers with generic classes across disciplines, which can support, for instance, academic search engines (#Q6).

◊◊◊

While this chapter has proposed an approach to extract information from research papers at the *sentence* level, we propose a domain-independent information extraction approach for scientific concepts at the *phrasal* level in the subsequent Chapters 5 and 6. This approach aims at constructing a fine-grained RKG that covers multiple scientific domains to support the uses cases *find related work* and *recommend articles* (see Chapter 3).

5 Domain-Independent Extraction of Scientific Concepts

The task of Named Entity Recognition (NER) is the first vital part in the KG population pipeline (see Section 2.4.2.2). Since research papers usually mention scientific concepts rather than named entities, in the following, we refer to this task as *scientific concept extraction*. In this chapter, we address the research questions **RQ2** and **RQ3** introduced in Section 1.2.2 for this task, namely:

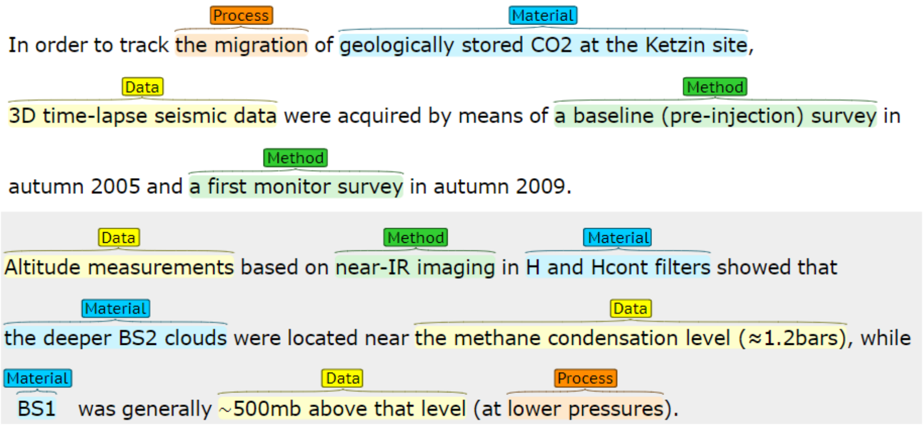
> **RQ2:** *How can we modify machine learning methods for information extraction from scientific texts to be adaptable to new domains with few labelled data?*
>
> **RQ3:** *How can we automatically extract information from research papers from multiple scientific domains in a domain-independent manner?*

In the following, Section 5.1 first motivates our approach for *domain-independent scientific concept extraction*. Section 5.2 provides related work on scientific concept extraction, active learning, and applications for domain-independent information extraction from scientific text. Then, Section 5.3 introduces a new corpus for domain-independent scientific concept extraction. Section 5.4 proposes a deep learning approach and an active learning based strategy for this task, while Section 5.5 presents the experimental results. Finally, Section 5.6 summarises this chapter.

5.1 Introduction

The task of *scientific concept extraction* enables the identification of scientific concepts in research papers (see Figure 5.1 for an example). This task is analogous to Named Entity Recognition (NER) introduced in Section 2.4.2.2 but focuses on scientific concepts rather than real-world entities such as persons or locations. Thus, scientific concept extraction is a first vital step towards a fine-grained RKG in which research papers are described and interconnected through entity types like tasks, materials, and methods.

As stated in Section 1.2.2, information extraction from scientific texts, obviously, differs from its general domain counterpart. In consequence, the extraction of scientific concepts from scientific texts would entail the involvement of domain experts and a specific design



In order to track the migration of geologically stored CO2 at the Ketzin site, 3D time-lapse seismic data were acquired by means of a baseline (pre-injection) survey in autumn 2005 and a first monitor survey in autumn 2009.

Altitude measurements based on near-IR imaging in H and Hcont filters showed that the deeper BS2 clouds were located near the methane condensation level (≈1.2bars), while BS1 was generally ~500mb above that level (at lower pressures).

Figure 5.1: Two example sentences taken from abstracts in Earth Science and Astronomy, that are annotated with generic scientific concepts.

of an extraction methodology for each scientific discipline – both requirements are rather time-consuming and costly.

At present, a systematic study of these assumptions is missing. We thus present the task of *domain-independent scientific concept extraction.* We examine the intuition that most research papers share certain core concepts such as the mentions of research tasks or methods. If so, these would allow for a domain-independent information extraction method to support RKG population, which does not reach all semantic depths of the analysed article, but still provides some science-specific structure.

In this chapter, we introduce a set of common scientific concepts that we find are relevant over a set of ten examined domains from Science, Technology, and Medicine (STM). Figure 5.1 shows two annotated example sentences from abstracts in two scientific domains. The generic concepts have been identified in a systematic, joint effort of domain experts and non-domain experts. The inter-coder agreement is measured to ensure the adequacy and quality of concepts. A set of research abstracts has been annotated using these concepts and the results are discussed with experts from the corresponding fields. The resulting dataset serves as a basis to train two baseline deep learning classifiers. In particular, we present an active learning approach to reduce the amount of required training data. The systems are evaluated in different experimental setups.

Our main contributions can be summarised as follows:

1. We introduce the novel task of *domain-independent scientific concept extraction* which aims at automatically extracting scientific entities in a domain-independent manner.
2. We release a new corpus that comprises 110 abstracts of ten STM domains annotated at the phrasal level.
3. We present and evaluate a state-of-the-art deep learning approach for this task. Additionally, we propose active learning for an optimal selection of instances, which to our

knowledge, is demonstrated for the first time on scholarly text. We find that strategic instance selection gives us the same performance with only about half of the training data.

4. We have made our corpora and source code publicly available to facilitate further research: `https://gitlab.com/TIBHannover/orkg/orkg-nlp/tree/master/STM-corpus`

5.2 Related Work

This section gives a brief overview of existing annotated datasets for scientific information extraction, followed by related work on some exemplary applications for domain-independent information extraction from scientific papers.

5.2.1 Scientific Corpora

Sentence-level Annotation: Early approaches for semantic structuring of research papers focused on sentences as the basic unit of analysis. As depicted in Table 3.1, annotated datasets exist for several domains. However, most datasets cover only a single domain, while few other datasets cover three domains.

Phrase-level Annotation: As described in Section 3.2.3.1 and depicted in Table 3.2 and Table 3.3, more recent corpora have been annotated at phrasal level to enable the construction of fine-grained KGs with scientific concepts with the tasks of concept extraction, coreference resolution, and relation extraction. These datasets differentiate between four to seven concept classes. However, each corpus covers at most three domains.

Experts vs. Non-Experts Annotation: The aforementioned datasets were usually annotated by domain experts [9, 68, 147, 175, 178, 229]. In contrast, Teufel et al. [279] explicitly use non-experts in their annotation tasks, arguing that text understanding systems can use general, rhetorical, and logical aspects also when qualifying scientific text. According to this line of thought, more researchers used (presumably cheaper) non-expert annotation as an alternative [43, 85].

Snow et. al. [263] provide a study on expert versus non-expert performance for general, non-scientific annotation tasks. They state that about four non-experts (Mechanical Turk workers, in their case) were needed to rival the experts' annotation quality. However, systems trained on data generated by non-experts showed to benefit from annotation diversity and to suffer less from annotator bias. Pustu-Iren et al. [228] examines the agreement between experts and non-experts for visual concept classification and person recognition in historical

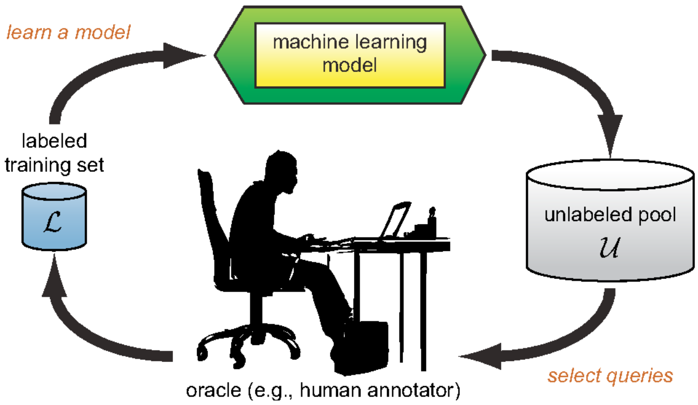


Figure 5.2: The active learning cycle. Illustration from [255].

video data. For the task of face recognition, training with expert annotations lead to an increase of only 1.5 % in classification accuracy.

Active Learning in Natural Language Processing: Active learning aims to minimise annotation costs. As depicted in Figure 5.2, the idea is that an annotator shall annotate those instances from which the model can learn most [255]. To the best of our knowledge, active learning has not been utilised in classification approaches for scientific text yet. Recent publications demonstrate the effectiveness of active learning for Natural Language Processing (NLP) tasks such as Named Entity Recognition (NER) [258] and sentence classification [300]. Siddhant and Lipton [259] and Shen et. al. [258] compare several sampling strategies on NLP tasks and show that *Maximum Normalized Log-Probability (MNLP)* based on uncertainty sampling performs well in NER.

5.2.2 Applications for Domain-Independent Information Extraction

Academic Search Engines: Academic search engines such as Google Scholar [102], Microsoft Academic [192], and Semantic Scholar [252] specialise in search of scholarly literature. They exploit graph structures such as the Microsoft Academic Knowledge Graph [78], SciGraph [268], or the Semantic Scholar Open Research Corpus (S2ORC) [3, 177]. These graphs interlink the papers through meta-data such as citations, authors, venues, and keywords, but not through deep semantic representation of the articles' content.

However, first attempts towards a more semantic representation of article content exist: Ammar et al. [3] interlink the Semantic Scholar Corpus with DBpedia [172] and Unified Medical Language System (UMLS) [23] using entity linking techniques. Yaman et al. [296] connect SciGraph with DBpedia [172] person entities. Xiong et al. [294] demonstrate that academic search engines can greatly benefit from exploiting general-purpose knowledge bases such as Freebase [25]. However, the coverage of science-specific concepts is rather low [3].

Research Paper Recommendation Systems: Beel et al. [18] provide a comprehensive survey about research paper recommendation systems. Such systems usually employ different strategies (e.g. content-based and collaborative filtering) and several data sources (e.g. text in the documents, ratings, feedback, stereotyping). Graph-based systems, in particular, exploit citation graphs and genes mentioned in the papers [13, 166]. Beel et al. [18] conclude that it is not possible to determine the most effective recommendation approach at the moment. However, we believe that a fine-grained RKG can improve such systems. Although Papers With Code [206] is not a typical recommendation system, it allows researchers to browse easily for papers from the field of machine learning that address a certain task.

5.3 Corpus for Domain-Independent Scientific Concept Extraction

In this section, we introduce the novel task of *domain-independent extraction of scientific concepts* and present an annotated corpus. As the discussion of related work reveals, the annotation of scientific resources is not a novel task. However, most researchers focus on at most three scientific disciplines and on expert-level annotations. In this work, we explore the domain-independent annotation of lexical phrasal units indicating scientific knowledge, i.e. scientific concepts, in abstracts from ten different science domains. Since other studies have also shown that non-expert annotations are feasible for the scientific domain, we go for a cost-efficient middle course: annotations by non-experts with scientific proficiency, and consultation with domain-experts. Finally, we explore how well a state-of-the-art deep learning model performs on this novel information extraction task and whether active learning can help to reduce the amount of required training data. Our novel corpus and the annotation process are described below.

5.3.1 OA-STM Corpus

The OA-STM corpus [77] is a set of open access (OA) articles from various domains in Science, Technology, and Medicine (STM). It was published in 2017 as a platform for benchmarking methods in scholarly article processing, amongst other scientific information extraction. The dataset contains a selection of 110 articles from ten domains, namely Agriculture (*Agr*), Astronomy (*Ast*), Biology (*Bio*), Chemistry (*Che*), Computer Science (*CS*), Earth Science (*ES*), Engineering (*Eng*), Materials Science (*MS*), Mathematics (*Mat*), and Medicine (*Med*). This annotation study focuses on the articles' abstracts as they contain a condensed summary of the article.

Table 5.1: The four generic scientific concepts that were derived in this study.

Process Natural phenomenon or activities, e.g. growing (*Bio*), reduction (*Mat*), flooding (*ES*).
Method A commonly used procedure that acts on entities, e.g. powder X-ray (*Che*), the PRAM analysis (*CS*), magnetoencephalography (*Med*).
Material A physical or abstract entity used in scientific experiments or proofs, e.g. soil (*Agr*), the moon (*Ast*), the carbonator (*Che*).
Data The data themselves, measurements, or quantitative or qualitative characteristics of entities, e.g. rotational energy (*Eng*), tensile strength (*MS*), 3D time-lapse seismic data (*ES*).

5.3.2 Annotation Process

The OA-STM Corpus is used as a base for (a) the identification of potential domain-independent concepts and (b) a first annotated corpus for baseline classification experiments. The annotation task was mainly performed by two postdoctoral researchers with a background in Computer Science (acting as non-expert annotators) using the BRAT annotation tool [271]. Their basic annotation assumptions were checked by domain experts. The annotation procedure consists of the following four phases:

1. *Pre-annotation:* A literature review of annotation schemes [9, 60, 174, 175] provided a seed set of potential candidate concepts. Both non-experts independently annotated a subset of the STM abstracts with these concepts (non-overlapping) and discussed the outcome. In a three-step process, the concept set was pruned to only contain those which seemed suitably transferable between domains. Our set of *generic* scientific concepts consists of Process, Method, Material, and Data (see Table 5.1 for their definitions). We also identified Task [9], Object [174], and Results [60], however, in this study we do not consider nested span concepts, hence we leave them out since they were almost always nested with the other scientific entities (e.g. a Result may be nested with Data).
2. *Phase I:* Five abstracts per domain (i.e. 50 abstracts) were annotated by both annotators and the inter-annotator agreement was computed using Cohen's κ [57] (see Section 2.5.3). Results showed a moderate inter-annotator agreement of 0.52 κ.
3. *Phase II:* The annotations were then presented to subject specialists who each reviewed (a) the choice of concepts and (b) annotation decisions on the respective domain corpus. The interviews mostly confirmed the concept candidates as generally applicable. The experts' feedback on the annotation was even more valuable: The comments allowed for a more precise reformulation of the annotation guidelines, including illustrating examples from the corpus.
4. *Consolidation:* Finally, the 50 abstracts from phase I were re-annotated by the non-experts. Based on the revised annotation guidelines, a substantial agreement

Table 5.2: Per-domain and overall inter-annotator agreement (Cohen's Kappa κ) for PROCESS, METHOD, MATERIAL, and METHOD scientific concept annotation.

	Med	*MS*	*CS*	*ES*	*Eng*	*Che*	*Bio*	*Agr*	*Mat*	*Ast*	*Overall*
κ	0.94	0.90	0.85	0.81	0.79	0.77	0.75	0.60	0.58	0.57	0.76

Table 5.3: The annotated corpus characteristics containing 11 abstracts per domain in terms of size and the number of scientific concept phrases.

	Ast	*Agr*	*Eng*	*ES*	*Bio*	*Med*	*MS*	*CS*	*Che*	*Mat*
∅ # Tokens/Abstract	382	333	303	321	273	274	282	253	217	140
# PROCESS	241	252	248	243	281	244	178	220	149	56
# METHOD	19	28	27	9	15	33	27	66	27	7
# MATERIAL	296	292	208	249	291	191	231	102	188	51
# DATA	235	169	258	197	62	132	138	165	119	183
# Concept phrases	791	741	741	698	649	600	574	553	483	297
# Unique concept phrases	663	631	618	633	511	518	493	482	444	287

of 0.76 κ could be reached (see Table 5.2). Similar annotation tasks for scientific entities, e.g. SciERC [178] considering one domain and ScienceIE-17 [9] considering three domains achieved agreements of 0.76 κ and 0.6 κ, respectively. Subsequently, the remaining 60 abstracts (six per domain) were annotated by one annotator. This phase also involved reconciliation of the previously annotated 50 abstracts to obtain a gold standard corpus.

5.3.3 Corpus Characteristics

Table 5.3 shows some characteristics of the resulting corpus. The corpus has a total of 6,127 scientific entities, including 2,112 PROCESS, 258 METHOD, 2,099 MATERIAL, and 1,658 DATA concept entities. The number of entities per abstract in our corpus directly correlates with the length of the abstracts (Pearson's R 0.97). Among the concepts, PROCESS and MATERIAL directly correlate with abstract length (R 0.8 and 0.83, respectively), while DATA has only a slight correlation (R 0.35) and METHOD has no correlation (R 0.02). The domains *Bio*, *CS*, *Ast*, and *Eng* contain the most of PROCESS, METHOD, MATERIAL, and DATA concepts, respectively.

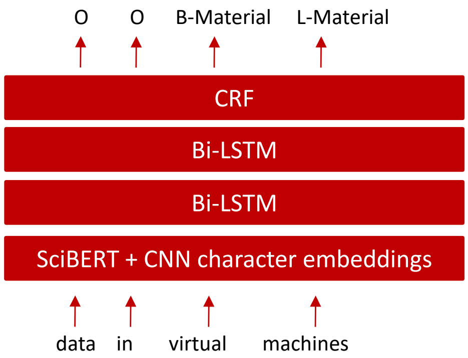


Figure 5.3: Scientific concept extraction system of Beltagy et al. [19] consisting of (1) a token embedding layer, (2) a token-level encoder with two stacked Bi-LSTMs, and (3) a CRF based tag decoder with BILOU tagging scheme as an output layer.

5.4 Automatic Domain-Independent Scientific Concept Extraction

The current state-of-the-art for scientific concept extraction is Beltagy et al.'s deep learning system with SciBERT word embeddings [19], which were pre-trained on scientific texts using the BERT [71] architecture. As depicted in Figure 5.3, it consists of three components:

1. A token embedding layer comprising a per-sentence sequence of tokens, where each token is represented as a concatenation of SciBERT word embedding and character embeddings based on Convolutional Neural Networks (CNNs) [183].
2. A token-level encoder with two stacked Bi-LSTMs [116].
3. A Conditional Random Field (CRF) based tag decoder [183] with BILOU (beginning, inside, last, outside, unit) tagging scheme.

This deep learning architecture is implemented in AllenNLP [96]. We use spaCy [119] for text preprocessing, i.e. for tokenisation and sentence-splitting.

5.4.1 Supervised Learning with Full Training Dataset

Using the above mentioned architecture, we train one model with data from all domains combined. We refer to this model as the *domain-independent* classifier. Similarly, we train ten models for each domain in our corpus – the *domain-specific* classifier.

To obtain a robust evaluation of models, we perform five-fold cross-validation experiments (see also Section 2.5). In each fold experiment, we train a model on 8 abstracts per domain (i.e. 80 abstracts), tune hyperparameters on 1 abstract per domain (i.e. 10 abstracts), and test on the remaining 2 abstracts per domain (i.e. 20 abstracts) ensuring that the data splits are not identical between the folds. All results reported in the paper are averaged over the five folds. We still obtain reliably trained domain-specific classifiers since on average they are trained on 400 concepts.

5.4.2 Active Learning with Training Data Subset

In this setting, we employ an active learning strategy [259, 300] to train a new *domain-independent* classifier. Active learning is usually applied to determine the optimal set of sufficiently distinct instances to minimise annotation costs. With our application of active learning, we find which proportion of our annotations suffice for training a robust classifier. We decide to use the Maximum Normalized Log-Probability (MNLP) [258] sampling strategy. We prefer it over its contemporary, Bayesian Active Learning by Disagreement (BALD) [123] since it has less computational requirements. The MNLP objective involves greedy sampling of sentences preferring those with the least logarithmic likelihood of the predicted tag sequence output by the CRF tag decoder, normalised by the number of tokens to avoid preferring longer sentences. Specifically, every sentence receives an MNLP score as follows:

$$score_{MNLP}(\mathbf{x_1}, ..., \mathbf{x}_\tau) = \frac{1}{\tau} \log P(\hat{\mathbf{y}}_\mathbf{1}, ..., \hat{\mathbf{y}}_\tau | \mathbf{x_1}, ..., \mathbf{x}_\tau) \tag{5.1}$$

Here, $(\hat{\mathbf{y}}_\mathbf{1}, ..., \hat{\mathbf{y}}_\tau)$ is the CRF decoder output for the input tokens $(\mathbf{x_1}, ..., \mathbf{x}_\tau)$ normalised by the number of tokens τ.

In our experiments, we found that adding 4% of the data to be the most discriminative selection of classifier performance. Therefore, we run 25 iterations of active learning in each stage adding 4% training data. We perform five-fold cross validation as before and the per-fold models are retrained after data resampling. The models use the same hyperparameters as for the domain-independent classifier.

5.5 Experimental Results and Discussion

In this section, we discuss the results obtained with our trained classifiers and the correlation analysis between inter-annotator agreement and performance of the classifiers.

5.5.1 Domain-Independent and Domain-Specific Classifiers with Full Training Dataset

Table 5.4 shows an overview of the *domain-independent* classifier results. The system achieves an overall F1 of 65.5% and has low standard deviation 1.26 across the five folds. For this classifier, Material was the easiest concept with an F1 of 71% (± 1.88), whereas Method was the hardest concept with an F1 of 43% (± 6.30). Method is also the most

Table 5.4: The *domain-independent* classifier results in terms of Precision (P), Recall (R), and F1-score on scientific concepts, respectively, and *Overall*.

	Process	Method	Material	Data	*Overall*
P	65.5 (± 4.22)	45.8 (± 13.50)	69.2 (± 3.55)	60.3 (±4.14)	64.3 (± 1.73)
R	68.3 (± 1.93)	44.1 (± 8.73)	73.2 (± 4.27)	60.0 (± 4.84)	66.7 (± 0.92)
F1	66.8 (± 2.07)	43.0 (± 6.30)	71.0 (± 1.88)	59.8 (± 1.75)	65.5 (± 1.26)

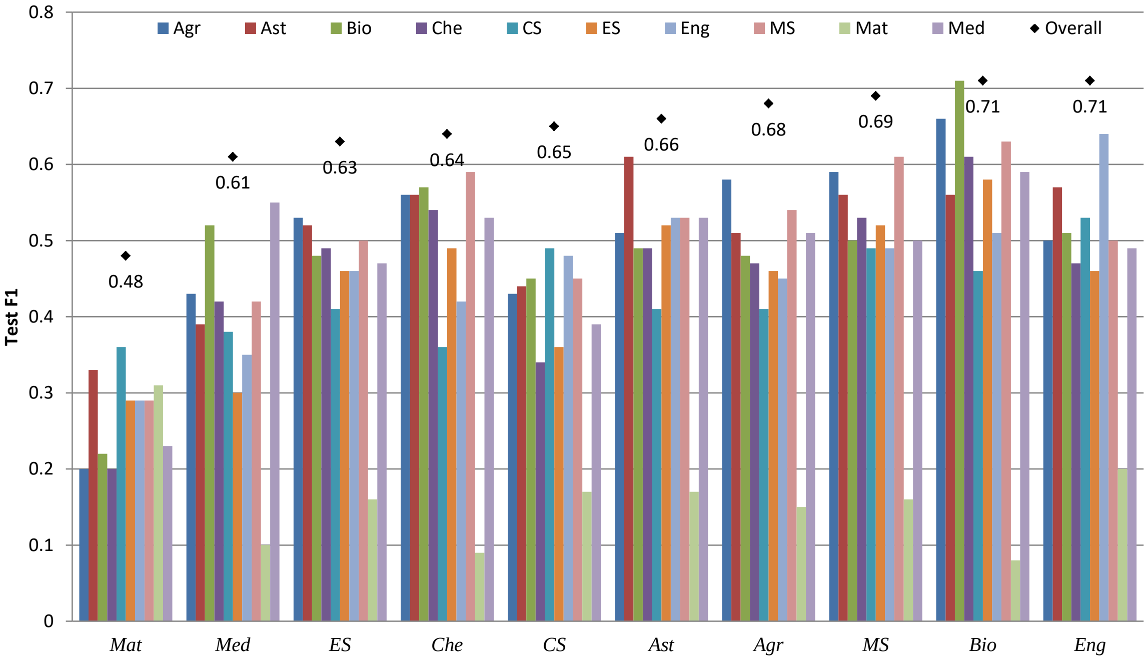
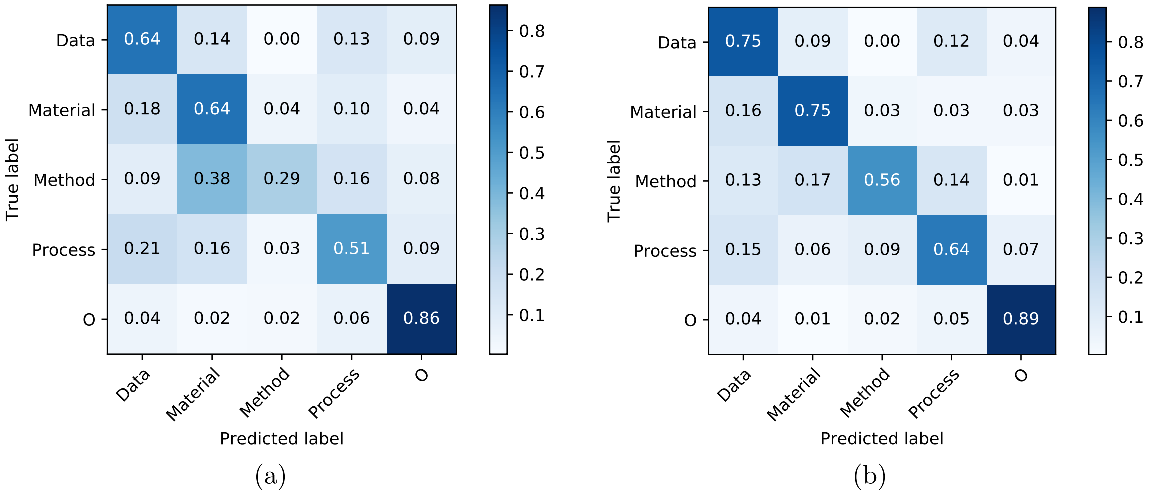


Figure 5.4: F1 per domain of the ten *domain-specific* classifiers (as bar plots) and of the *domain-independent* classifier (as scatter plots) for scientific concept extraction; the x-axis represents the ten test domains.

underrepresented in our corpus, which partly explains the poor extraction performance. Best reported results for similar datasets, ScienceIE17 [9] and SciERC [178] (both have 500 abstracts), have an F1 score of 65.6% [19] and 44.7% [178], respectively, indicating that the size of our dataset with only 110 abstracts is sufficient.

Next, we compare and contrast the ten *domain-specific* classifiers (see Figure 5.4) by their capability to extract the concepts from their own domains and in other domains.

Most Robust Domain: *Bio* (third bar in each domain in Figure 5.4) extracts scientific concepts from its own domain at the same performance as the *domain-independent* classifier with an F1 score of 71% (± 9.0), thus demonstrating a robust domain. It comprises only 11% of the overall data, yet the *domain-independent* classifier trained on all data does not outperform it.



(a)

(b)

Figure 5.5: Confusion matrix for (a) the *CS* classifier and (b) *domain-independent classifier* on *CS* domain predicting concept-type of tokens.

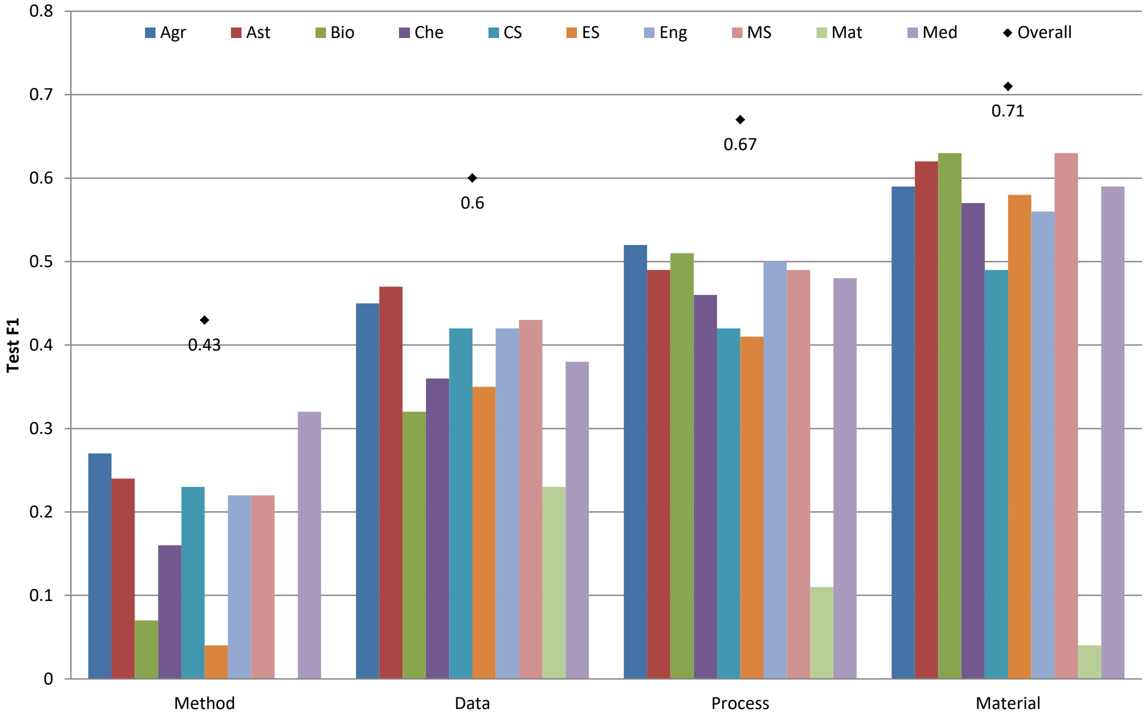


Figure 5.6: F1 scores of the ten *domain-specific* classifiers (bar plots) and the *domain-independent* classifier (scatter plots) for extracting each scientific concept; the x-axis represents the evaluated concepts.

Most Generic Domain: *MS* (the third last bar in each domain in Figure 5.4) exhibits a high degree of domain independence since it is among the top 3 classifiers for seven of the ten domains (viz. *ES*, *Che*, *CS*, *Ast*, *Agr*, *MS*, and *Bio*).

Most Specialised Domain: *Mat* (the second last bar in each domain in Figure 5.4) shows the lowest performance in extracting scientific concepts from all domains except itself. Hence it shows to be the most specialised domain in our corpus. Notably, a characteristic feature of this domain is that it has short abstracts (nearly a third of the size of the longest abstracts), so it is also the most underrepresented in our corpus. Also, distinct from the other domains, *Mat* has triple the number of Data entities compared to each of its other concepts, where in the other domains Process and Material are consistently predominant.

Medical and Life Science Domains: The *Med*, *Agr*, and *Bio* domains show strong domain relatedness. Their respective *domain-specific* classifiers show top five system performances among the three domains, when applied to another domain. For instance, the *Med* domain shows the strongest domain relatedness and is classified best by *Med* (last bar), followed by *Bio* (third bar) and *Agr* (first bar).

Domain-Independent vs. Domain-Specific Classifier: Except for *Bio*, the *domain-independent* classifier clearly outperforms the *domain-specific* one in extracting concepts from their respective domains. We attribute this, in part, to the improved span-detection performance. Span-detection merely relies on syntactic regularity; thus the *domain-independent* classifier can benefit from more training data of other domains. e.g. the CS classifier shows an improvement from 49.5% F1 with the *domain-specific* classifier to 65.9% F1 with the *domain-independent* classifier, which is supported by the enhanced span-detection performance from 73.4% to 82.0% in F1. Accuracy on token-level also improves from 67.7% to 77.5% F1 for CS, that is correct labelling of the tokens also benefits from other domains. This is also supported by the results in the confusion matrix depicted in Figure 5.5 for the *CS* and the *domain-independent* classifier on token-level.

Scientific Concept Extraction: Figure 5.6 depicts the ten *domain-specific* classifier results for extracting each of the four scientific concepts. It can be observed that *Agr*, *Med*, *Bio*, and *Ast* classifiers are the best in extracting their respective Process, Method, Material, and Data concepts.

5.5.2 Domain-Independent Classifier with Active Learning

The results of the active learning experiment over the full dataset plotted over the 25 iterations are depicted in Figure 5.7, showing that MNLP clearly outperforms the random baseline. While using only 52% of the training data, the best result of the *domain-independent* classifier trained with all training data is surpassed with an F1 score of 65.5% (± 1.0). The random baseline achieves an F1 score of only 62.5% (± 2.6) with the same proportion of training data. When 76% of the data are sampled by MNLP, the best active learning per-

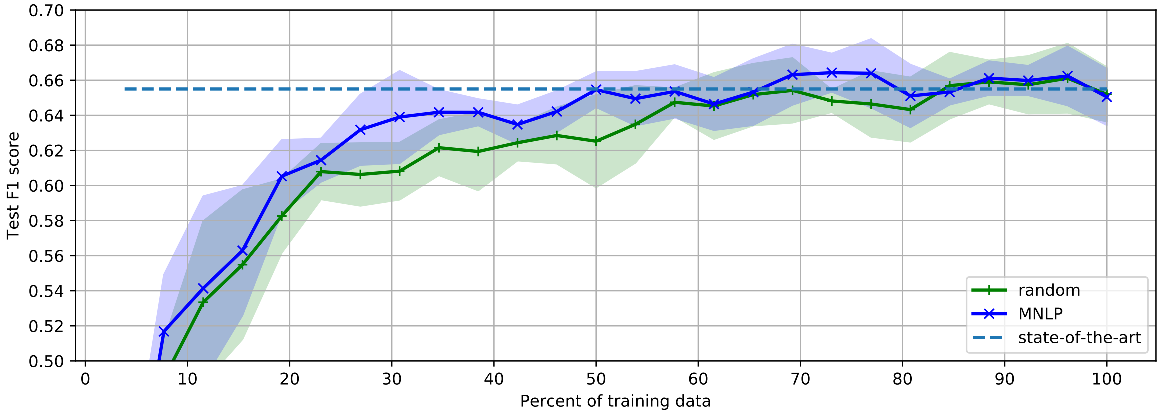


Figure 5.7: Progress of active learning with MNLP and random sampling strategy; the areas represent the standard deviation (std) of the F1 score across 5 folds for MNLP and random sampling strategy, respectively.

Table 5.5: Performance (F1 in percent) of active learning with MNLP and random sampling strategy for the fraction of training data when the performance with entire training dataset is achieved; for SciERC and ScienceIE-17 results are reported across 5 random restarts.

	Training Data	F1 (MNLP)	F1 (random)	F1 (full data)
STM (our corpus)	52 %	65.5 (± 1.0)	62.5 (± 2.6)	65.5 (± 1.3)
SciERC [178]	62 %	65.3 (± 1.5)	62.3 (± 1.5)	65.6 (± 1.0)
ScienceIE17 [9]	38 %	43.9 (± 1.2)	42.2 (± 1.8)	43.8 (± 1.0)

formance across all steps is achieved with an F1 score of 69.0% on the validation set, having the best F1 of 66.4% (± 2.0) on the test set. Thus, 76% of our annotated sentences suffice to train an optimal performing model.

Analysing the distribution of sentences in the training data sampled by MNLP, shows (*Math*, *CS*) as the most preferred domains and (*Eng*, *MS*) the least preferred ones. Nonetheless, all domains are represented, that is a non-uniformly mix of sentences sampled by MNLP yields the most generic model with less training data. In contrast, the random sampling strategy uniformly samples sentences from all domains.

Furthermore, we show in Table 5.5 the proportion of training data for MNLP when the performance using the entire training dataset is achieved for related SciERC [178] and ScienceIE-17 [9] datasets. The results indicate that also for related datasets on scientific texts, MNLP can significantly reduce the amount of labelled training data.

Table 5.6: Inter-annotator agreement (κ) and the number of concept phrases (#) per domain; F1 and std of domain-specific classifiers on their domains; F1 and std of domain-independent and active learning classifier (AL-trained) on each domain; the right side depicts correlation coefficients (R) of each row with κ and the number of concept phrases.

	Agr	*Ast*	*Bio*	*Che*	*CS*	*ES*	*Eng*	*MS*	*Mat*	*Med*	R κ	R #
κ	0.6	0.57	0.75	0.77	0.85	0.81	0.79	0.9	0.58	0.94	1.00	-0.02
# concept phrases (#)	741	791	649	483	553	698	741	574	297	600	-0.02	1.00
domain-specific (F1)	0.58	0.61	0.71	0.54	0.49	0.46	0.64	0.61	0.31	0.55	0.20	0.70
domain-independent (F1)	0.68	0.66	0.71	0.64	0.65	0.63	0.71	0.69	0.48	0.61	0.28	0.76
AL-trained (F1)	0.65	0.67	0.74	0.65	0.62	0.63	0.72	0.69	0.50	0.60	0.23	0.68
domain-specific (std)	0.06	0.06	0.09	0.08	0.05	0.06	0.04	0.11	0.06	0.07	0.29	0.28
domain-independent (std)	0.04	0.04	0.11	0.08	0.07	0.05	0.03	0.04	0.06	0.03	-0.11	-0.05
AL-trained (std)	0.04	0.04	0.09	0.08	0.07	0.04	0.07	0.05	0.15	0.02	-0.41	-0.72

5.5.3 Correlations between Inter-Annotator Agreement and Performance

In this section, we analyse the correlations (Pearson's R) of inter-coder agreement κ and the number of annotated concepts per domain (#) on (1) the performance F1 and (2) variance resp. standard deviation (std) of the classifiers across five-fold cross validation.

Table 5.6 summarises the results of our correlation analysis. The active learning classifier (AL-trained) has been trained with 52% training data sampled by MNLP since it is the point at which the performance of the full data trained model is surpassed (see Table 5.5). For the domain-specific, domain-independent, and AL-trained classifier we observe a strong correlation between F1 and number of concepts per domain (R 0.70, 0.76, 0.68), and a weak correlation between κ and F1 (R 0.20, 0.28, 0.23). Thus, we surmise that the number of annotated concepts in a particular domain has more influence on the performance than the inter-annotator agreement.

The correlation values for the variance are different between the classifier types. For the domain-specific classifier the correlation between κ and std, and the number of concepts per domain and std are slightly positive (R 0.29, 0.28), i.e. the higher the agreement and the size of the domain, the higher the variance of the domain-specific classifier. For the domain-independent classifier, there is no correlation (R 0.11, -0.05) and for the AL-trained classifier, the correlations become negative (R -0.41, -0.72), i.e. higher agreement and more annotated concepts per domain lead to less variance for the AL-trained classifier. In summary, we hypothesise that more diverse training data across multiple domains lead to better performance and lower variance because the classifier can generalise better.

5.6 Summary

This chapter has addressed the research questions **RQ2** (adaptable to new domains with few labelled data) and **RQ3** (domain-independent extraction) for the task of scientific concept extraction. To investigate **RQ3**, we have introduced the novel task of *domain-independent concept extraction* from scientific texts. With a systematic annotation procedure involving domain experts, we have identified four general core concepts that are relevant across ten domains from Science, Technology, and Medicine (STM). Using this concept set, we have annotated a corpus of abstracts from these domains. We have verified the adequacy of the concepts by evaluating the inter-annotator agreement for our corpus. The results indicate that the identification of the *generic* concepts in a corpus of ten different scholarly domains is feasible by non-experts with moderate agreement (0.52 κ) and after consultation of domain experts with substantial agreement (0.76 κ).

Furthermore, we evaluated a state-of-the-art system [19] on our annotated corpus, which achieved a fairly high F1 score (65.5% overall). The domain-independent system noticeably outperforms the domain-specific systems, which indicates that the model can generalise well across domains. We also observed a strong correlation between the number of annotated concepts per domain and classifier performance, and only a weak correlation between inter-annotator agreement per domain and the performance. It is assumed that more annotated data positively influence the performance in the respective domain.

Finally, to address **RQ2**, we have proposed active learning for our novel task. We have shown that only approximately five annotated abstracts per domain are sufficient as training data to build a performant model. Our active learning results for SciERC [178] and ScienceIE17 [9] datasets were similar. Thus, active learning can significantly save annotation costs and enables fast adaptation to new domains.

◊◊◊

This chapter has addressed the task of domain-independent scientific concept extraction, which is the first essential step in the KG population pipeline (see Section 2.4.2.2). In the next Chapter 6, we extend our domain-independent information extraction approach with the task of coreference resolution and populate an RKG.

6 Coreference Resolution for Knowledge Graph Population

This chapter extends the domain-independent extraction approach introduced in the previous Chapter 5 with the task of coreference resolution that is a further essential step in the KG population pipeline (see Section 2.4.2.2). Consequently, this chapter also addresses **RQ2** and **RQ3** but for the task of coreference resolution, namely:

> **RQ2:** *How can we modify machine learning methods for information extraction from scientific texts to be adaptable to new domains with few labelled data?*
>
> **RQ3:** *How can we automatically extract information from research papers from multiple scientific domains in a domain-independent manner?*

Furthermore, this chapter addresses **RQ4** and describes how an RKG can be populated using the domain-independent extraction approach, namely:

> **RQ4:** *How can we automatically populate an RKG that covers multiple scientific domains?*

The remainder of the chapter is organised as follows: Section 6.1 motivates the task of coreference resolution while Section 6.2 summarises related work for this task. Section 6.3 describes the annotation procedure and the characteristics of our annotated corpus, and our proposed approaches for coreference resolution, KG population, and KG evaluation. The experimental setup and results are reported in Section 6.4 and Section 6.5, while Section 6.6 summarises this chapter.

6.1 Introduction

Coreference resolution is the task of identifying mentions in a text which refer to the same entity or concept (see also Section 2.4.2.2). It is an essential step for automatic text understanding and facilitates down-stream tasks such as text summarisation, question answering, or KG population. As depicted in an excerpt text from the Astronomy domain in Figure 6.1, coreference resolution can recognise that the mentions “The moon Enceladus” and “the moon” both refer to the same entity. This enables to extract the fact $(Cassini\_spacecraft, flew\_by, moon\_Enceladus)$.

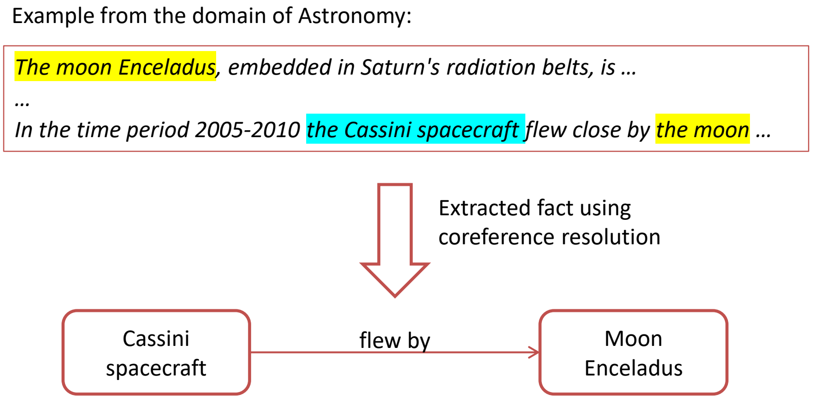

Example from the domain of Astronomy:
The moon Enceladus, embedded in Saturn's radiation belts, is ...
...
In the time period 2005-2010 the Cassini spacecraft flew close by the moon ...
Extracted fact using coreference resolution
Cassini spacecraft
flew by
Moon Enceladus

Figure 6.1: An excerpt from the abstract of a research paper in the STM corpus (see Chapter 5) from the domain of Astronomy and the extracted fact with the help of coreference resolution.

Current methods for coreference resolution based on deep learning achieve quite impressive results (e.g. an F1 score of 79.6% for the OntoNotes 5.0 dataset [136]) in the general domain, that is non-academic data from phone conversations, news, magazines, etc. But results of previous work indicate [58, 146, 201, 248] that general coreference resolution systems perform poorly on scientific text. This is presumably caused by the specific terminology and phrasing used in a scientific domain. Some other studies state that the annotation of scientific text is costly since it demands certain expertise in the respective domain [9, 92]. Most corpora for research papers cover only a single domain (e.g. biomedicine [58], artificial intelligence [178]) and are thus limited to these domains. As a result, the annotated corpora are relatively small and overall, only a few domains are covered. Datasets for the general domain are usually much larger, but they have not been exploited yet by approaches for coreference resolution in research papers.

As described in Section 2.4.2.2, coreference resolution is also one of the main steps in the KG population pipeline [179, 226]. However, to date it is not clear, to which extent (a) coreference resolution can help to reduce the number of scientific concepts in the populated KG, and (b) how coreference resolution influences the quality of the populated KG. Besides, a KG covering multiple scientific domains has not been populated yet.

In this chapter, we address the task of coreference resolution in research papers and subsequent RKG population. Our contributions can be summarised as follows:

1. First, we annotate a corpus for coreference resolution that consists of 110 abstracts from ten domains from Science, Technology, and Medicine (STM). The systematic annotation resulted in a substantial inter-coder agreement (0.68 κ). We provide and compare baseline results for this dataset by evaluating five different state-of-the-art approaches. Our experimental results confirm that state-of-the-art coreference approaches do not perform well on research papers.

2. Consequently, we propose sequential transfer learning for coreference resolution in research papers. This approach utilises our corpus by fine-tuning a model that is pre-trained on a large corpus from the general domain [222]. Experimental results show that our approach significantly outperforms the best state-of-the-art baseline (F1 score of 61.4%, i.e. +11.0% absolute improvement).
3. We investigate the impact of coreference resolution on automatic RKG population. To evaluate the quality of various RKG population strategies, we (i) compile a gold-standard RKG from our annotated corpus that contains scientific concepts referenced by mentions from text, and (ii) present a procedure to evaluate the clustering results of mentions.
4. We release (i) an automatically populated RKG from 55,485 abstracts of the ten STM domains and (ii) a gold RKG (Test-STM-KG) from the annotated STM corpus. Experimental results show that coreference resolution has only a small impact on the number of concepts in a populated RKG, but helps to improve the quality of the RKG significantly: the population with coreference resolution yields an F1 score of 63.5% evaluated against the gold-standard RKG (+21.8% absolute improvement).
5. We have released the data corpora and the source code to facilitate further research: `https://github.com/arthurbra/stm-coref`

6.2 Related Work

In the following, we provide an overview of approaches for coreference resolution and available datasets from research papers.

6.2.1 Approaches for Coreference Resolution

For a given document, the task of coreference resolution is (a) to extract mentions of scientific concepts, and (b) to cluster those mentions that refer to the same concept (see also Section 2.4.2.2). Recent approaches mostly rely on supervised learning and can be categorised into three groups [199]:

1. Mention-pair models [200, 265] are binary classifiers that determine whether two mentions are coreferent or not.
2. Entity-mention models [51, 231] determine whether a mention is coreferent to a preceding *cluster*. A cluster has more expressive features compared to a mention in mention-pair models.
3. Ranking-based models [66, 170, 187] simultaneously rank all candidate antecedents (i.e. preceding mention candidates). This enables the model to identify the most probable antecedent.

Lee et al. [170, 171] propose an end-to-end neural coreference resolution model. It is a ranking-based model that jointly recognises mentions and clusters. Therefore, the model considers all spans in the text as possible mentions and learns distributions over possible antecedents for each mention. For computational efficiency, candidate spans and antecedents are pruned during training and inference. Joshi et al. [137] enhance Lee et al.'s model with BERT-based word embeddings [71], while Ma et al. [182] improve the model with better attention mechanisms and loss functions.

Furthermore, several approaches propose multi-task learning, such that related tasks may benefit from knowledge in other tasks to achieve better prediction accuracy: Luan et al. [178, 287] train a model on three tasks (coreference resolution, entity, and relation extraction) using one dataset of research papers. Sanh et al. [246] introduce a multi-task model that is trained on four tasks (mention detection, coreference resolution, entity, and relation extraction) using two different datasets in the general domain.

Results of some previous studies [58, 146, 201, 248] revealed that general coreference systems do not work well in the biomedical domain due to the lack of domain knowledge. For instance, on Colorado Richly Annotated Full Text (CRAFT) corpus [58] a coreference resolution system for the news domain achieves only 14.0% F1 (-32.0%).

To the best of our knowledge, a transfer learning approach from the general to the scientific domain has not been proposed for coreference resolution yet.

6.2.2 Corpora for Coreference Resolution in Research Papers

For the general domain, multiple datasets exist for coreference resolution, e.g. Message Understanding Conference (MUC-7) [256], Automatic Content Extraction (ACE05) [72], or OntoNotes 5.0 [222]. The OntoNotes 5.0 dataset [222] is the largest one and is used in many benchmark experiments for coreference resolution systems [137, 170, 182].

Various annotated datasets for coreference resolution exist also for research papers (see also Section 3.2.3.1): The CRAFT corpus [58] covers 97 papers from biomedicine. The corpus of Schäfer et al. [248] contains 266 papers from computational linguistics and language technology. Chaimongkol et al. [42] annotated a corpus of 284 papers from four subdisciplines in computer science. The SciERC corpus [178] comprises 500 abstracts from the artificial intelligence domain and features annotations for scientific concepts and relations. It was used to generate an artificial intelligence (AI) knowledge graph [70]. Furthermore, as described in Section 3.2.3.1, several datasets exist for scientific concept extraction and relation extraction that cover various scientific domains.

To the best of our knowledge, a corpus for coreference resolution that comprises a broad range of scientific domains is not available yet.

6.3 Coreference Resolution in Research Papers

As the discussion of related work reveals, existing corpora for coreference resolution in scientific papers normally cover only a single domain, and coreference resolution approaches do not perform well on scholarly texts. To address these issues, we systematically annotate a corpus with coreferences in abstracts from ten different science domains. Furthermore, current approaches for coreference resolution in research papers do not exploit existing annotated datasets from the general domain, which are usually much larger than in the scientific domain. We propose a sequential transfer learning approach that takes advantage from large, annotated datasets. Finally, to the best of our knowledge, the impact of (a) coreference resolution and (b) cross-domain collapsing of mentions to scientific concepts on KG population with multiple science domains has not been investigated yet. Consequently, we present an evaluation procedure for the clustering aspect in the KG population pipeline.

In the sequel, we describe our annotated corpus, our transfer learning approach for coreference resolution, and an evaluation procedure for clustering in KG population.

6.3.1 Corpus for Coreference Resolution in 10 STM Domains

In this section, we describe the corpus, which we used as the basis for the annotation, our annotation process, and the characteristics of the resulting corpus.

STM Corpus: We extend our STM corpus introduced in Chapter 5 with coreference annotations. In particular, we (1) annotate coreference links between existing scientific concept mentions in abstracts using the BRAT annotation tool [271], and (2) annotate further mentions, i.e. pronouns and noun phrases consisting of multiple consecutive mentions.

Annotation Process: Other studies have shown that non-expert annotations are viable for the scientific domain [43, 85, 248, 279], and they are less costly than domain-expert annotations. Therefore, we also annotate the corpus with non-domain experts, i.e. by two students in computer science. Furthermore, we follow mostly the annotation procedure of the STM corpus (see Section 5.3.2), which consists of the following three phases:

1. *Pre-Annotation:* This phase aims at developing annotation guidelines through trial annotations. We adapted the comprehensive annotation guidelines of the OntoNotes 5.0 dataset [223], which were developed for the general domain, to research papers. In particular, we provide briefer and simpler descriptions with examples from the scientific domain. Within three iterations, both annotators labelled independently 10, 9 and 7 abstracts (i.e. 26 abstracts), respectively. After each iteration, the annotators discussed the outcome and refined the annotation guidelines.

Table 6.1: Per-domain and overall inter-annotator agreement (Cohen's κ and MUC) for coreference resolution annotation in our STM corpus.

	Mat	*Med*	*Ast*	*CS*	*Bio*	*Agr*	*ES*	*Eng*	*Che*	*MS*	*Overall*
κ	0.84	0.80	0.78	0.72	0.70	0.66	0.61	0.58	0.56	0.52	0.68
MUC	0.83	0.69	0.78	0.73	0.70	0.72	0.61	0.66	0.56	0.63	0.69

2. *Independent Annotation:* After the annotation guidelines were finalised, both annotators independently re-annotated the previously annotated abstracts and 24 additional abstracts. The final inter-coder agreement was measured on the 50 abstracts (5 per domain) using Cohen's κ [57, 155] and MUC [284]. As shown in Table 6.1, we achieve a substantial agreement with 0.68 κ and 0.69 MUC.

3. *Consolidation:* Finally, the remaining 60 abstracts were annotated by one annotator and the annotation results of this author were used as the gold standard corpus.

Corpus Characterstics: Table 6.2 shows the characteristics of the resulting corpus broken down per concept type, while they are listed per domain in Table 6.3. The original corpus has in total 6,127 mentions. 2,577 mentions were annotated as coreferent resulting in 908 coreference clusters. Thus, each coreference cluster contains on average 2.84 mentions, while Method clusters contain the most (3.4 mentions) and Data clusters the least (2.3 mentions). Furthermore, 705 mentions were annotated additionally (referred to as NONE) since they represent pronouns (422 mentions) or noun phrases consisting of multiple consecutive original mentions (283 mentions) such as *'... [[A], [B], and [C] [treatments]]... [These treatments]...'*. Fifty clusters (5%) contain mentions with different concept types (referred to as MIXED) due to disagreements between the annotators of the original concept mentions, and the annotators of coreferences. For instance, non-coreferent mentions were annotated as coreferent, or coreferent mentions have different concept types. Finally, 138 clusters (15%) do not have a concept type (NONE) since they form clusters which are not coreferent with the original concept mentions.

6.3.2 Transfer Learning for Coreference Resolution

We suggest sequential transfer learning [238] for coreference resolution in research papers (see Section 4.2.2 for an overview for transfer learning). Thus, we fine-tune a model pretrained on a large (source) dataset to our (target) dataset. As the source dataset, we use the English portion of the OntoNotes 5.0 dataset [222], since it is a broad corpus that consists of 3,493 documents with telephone conversations, magazine and news articles, Web data, broadcast conversations, and the New Testament. Furthermore, our annotation guidelines were adapted from OntoNotes 5.0.

Table 6.2: Characteristics of the annotated STM corpus with 110 abstracts per concept type in terms of number of scientific concept mentions, number of coreferent mentions, number of coreference clusters and singleton clusters, and the number of overall clusters. MIXED denotes clusters consisting of mentions with different concept types, NONE denotes coreference mentions and clusters without a scientific concept mention.

	Data	Material	Method	Process	MIXED	NONE	Total
# mentions	1,658	2,099	258	2,112	0	0	6,127
# coreferent mentions	351	910	101	510	0	705	2,577
# coreference clusters	153	339	30	198	50	138	908
# singleton clusters	1,307	1,189	157	1,602	0	0	4,255
# overall clusters	1,460	1,528	187	1,800	50	138	5,163

Table 6.3: Characteristics of the STM corpus per domain (11 abstracts per domain).

	Agr	Ast	Bio	Che	CS	ES	Eng	MS	Mat	Med	Total
# mentions	741	791	649	553	483	698	741	574	297	600	6,127
# coreferent mentions	276	365	275	282	181	241	318	256	124	259	2,577
# coreference clusters	106	120	98	90	67	93	117	87	48	82	908
# singleton clusters	520	549	443	384	339	525	503	371	210	411	4,255
# clusters	626	669	541	474	406	618	620	458	258	493	5,163

For the model, we utilise *BERT for Coreference Resolution (BFCR)* [137] with *SpanBERT* [136] word embeddings. This model achieves state-of-the-art results on the OntoNotes dataset [136]. Another advantage is the availability of the pre-trained model and the source code. The BFCR model improves Lee et al.'s approach [171] by replacing the LSTM encoder with the SpanBERT transformer-encoder. SpanBERT [136] has different training objectives than BERT [71] to better represent spans of text.

6.3.3 Cross-Domain Research Knowledge Graph Population

Figure 6.2 illustrates the collapsing of five clusters from two example abstracts to four scientific concepts. In the following, this process is described formally. Let $d \in \mathbb{D}$ be an abstract, $M(d) = \{m_1, ..., m_h\}$ the mentions of scientific concepts in d, and $c_d(m_i) \subseteq M(d)$ the corresponding coreference cluster for mention m_i in d. If mention m_s is not coreferent with other mentions in d, then $c_d(m_s) = \{m_s\}$ is a singleton cluster. The set of all clusters is denoted by $\mathbb{C}$. An equivalence relation *collapsable* $\subseteq \mathbb{C} \times \mathbb{C}$ defines if two clusters can be collapsed, i.e. if the clusters refer to the same scientific concept. To create the set of all concepts $\mathbb{E}$, we build the quotient set for the set of clusters $\mathbb{C}$ with respect to the relation

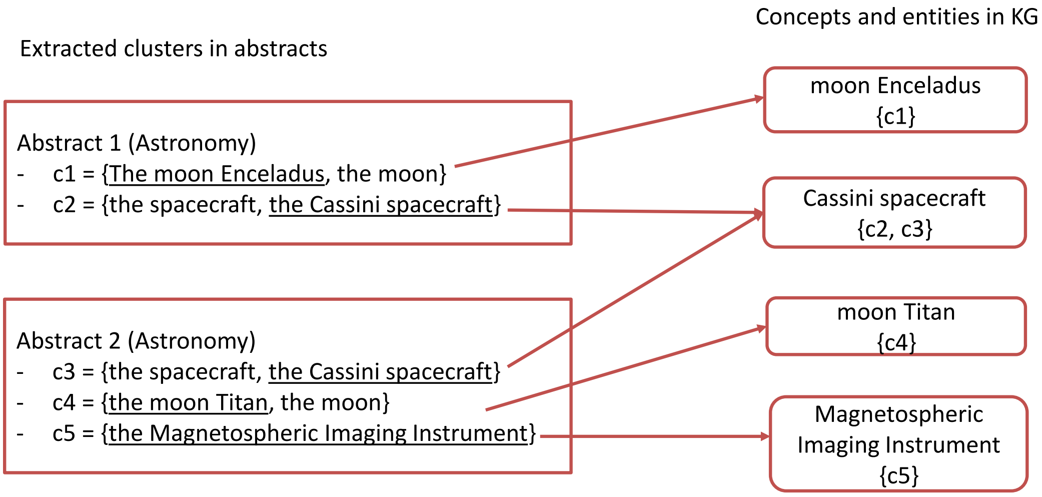

Extracted clusters in abstracts
Concepts and entities in KG
Abstract 1 (Astronomy)
- c1 = {The moon Enceladus, the moon}
- c2 = {the spacecraft, the Cassini spacecraft}
Abstract 2 (Astronomy)
- c3 = {the spacecraft, the Cassini spacecraft}
- c4 = {the moon Titan, the moon}
- c5 = {the Magnetospheric Imaging Instrument}
moon Enceladus {c1}
Cassini spacecraft {c2, c3}
moon Titan {c4}
Magnetospheric Imaging Instrument {c5}

Figure 6.2: The process of collapsing clusters c1-c5 of two example abstracts to four scientific concepts in the KG.

collapsable:

$$\mathbb{C} = \{c_d(m) \mid d \in \mathbb{D}, m \in M(d)\} \tag{6.1}$$

$$[c] = \{x \in C \mid \mathit{collapsable}(c, x)\} \tag{6.2}$$

$$\mathbb{E} = \{[c] \mid c \in \mathbb{C}\} \tag{6.3}$$

Now, we can construct the KG: For each paper $d \in \mathbb{D}$ and for each scientific concept $e \in \mathbb{E}$, we create a node in the KG. The scientific concept type of e is the most frequent concept type of all mentions in e. Then, for each mention $m \in M(d)$, we create a “mentions” link between the paper and the corresponding scientific concept $[m] \in \mathbb{E}$.

Cross-Domain vs. In-Domain Collapsing: One commonly used approach to define the *collapsable* relation is to treat two clusters as equivalent, if and only if the ‘label’ of the clusters is the same. The label of a cluster is the longest mention in the cluster normalised by (a) lower-casing, (b) removing articles, possessives, and demonstratives, (c) resolving acronyms, and (d) lemmatisation using WordNet [82] to transform plural forms to singular. Other studies [70, 178] used a similar label function for KG population.

However, an RKG that comprises multiple scientific disciplines has not been populated yet. Thus, it is not clear whether it is feasible to collapse clusters across domains. Usually, terms within a scientific domain are unambiguous, but some terms can have different meanings across scientific disciplines (e.g. “neural network" in *CS* and *Med*). Thus, we investigate both cross-domain and in-domain collapsing strategies.

Table 6.4: Characteristics of the *Test-STM-KG*: number of concepts per concept type and per domain. MIX denotes the number of cross-domain concepts.

	Agr	Ast	Bio	CS	Che	ES	Eng	MS	Mat	Med	MIX	Total
DATA	5	18	3	20	4	9	28	13	37	8	9	154
MATERIAL	27	35	30	20	26	52	32	30	9	40	7	308
METHOD	1	1	1	21	6	2	4	10	3	8	7	64
PROCESS	17	12	21	34	13	33	20	25	15	38	8	236
Total	50	66	55	95	49	96	84	78	64	94	31	762

Knowledge Graph Population Approach: We populate an RKG with research papers from multiple scientific domains, i.e. 55,485 abstracts of Elsevier with CC-BY licence from the 10 investigated domains. First, we extract (a) concept mentions from the abstracts using the scientific concept extractor of the STM corpus (see Section 5.4), and (b) clusters within the abstracts with our transfer learning coreference model. Then, those mention clusters, which contain solely mentions recognised by the coreference resolution model and not by the scientific concept extraction model, are dropped, since the coreference resolution model does not recognise the concept type of the mentions. Finally, the remaining clusters serve for the population of the KG as described above.

6.3.4 Evaluation Procedure of Clustering in Knowledge Graph Population

One common approach to evaluate the quality of a populated KG is to annotate a (random) subset of statements by humans as true or false and to calculate precision and recall [70, 291]. To evaluate recall, small collections of ground-truth capturing *all* knowledge is necessary, that are usually difficult to obtain [291]. To the best of our knowledge, a common approach to evaluate the clustering aspect of the KG population pipeline does not exist yet. Thus, in the following, we present (1) an annotated test KG, and (2) metrics to evaluate clustering of mentions to concepts in KG population.

Test-STM-KG: To enable evaluation of KG population strategies, we compile a test KG, referred to as *Test-STM-KG*. For this purpose, we reuse the STEM-ECR corpus [61], in which 1,221 mentions of the STM corpus are linked to Wikipedia entities. First, we extract all annotated clusters of the STM corpus in which all mentions of the cluster uniquely refer to the same Wikipedia entity. Then, we collapse all clusters which refer to the same Wikipedia entity to concepts. Formally, the Test-STM-KG is a partition of mentions, where each part denotes a concept, i.e. a disjoint set of mentions. A mention is uniquely represented by the tuple (start offset, end offset, concept type, document id).

Table 6.4 shows the characteristics of the compiled Test-STM-KG. It consists of 920 clusters, of which 711 are singleton clusters. These clusters were collapsed to 762 concepts, of which 31 concepts are used across multiple domains (referred to as MIX).

Evaluation Procedure: To evaluate the clustering result of a KG population strategy, we use the metrics of coreference resolution. The three popular metrics for coreference resolution are MUC [284], B^3 [12], and $CEAFe_{\phi 4}$ [180]. Each of them represents different evaluation aspects (see Section 2.5.2.3 and Pradhan et al. [221] for more details). To calculate these metrics, we treat the gold concepts (i.e. a partition of mentions) of the Test-STM-KG as the ‘key’ and the predicted concepts as the ‘response’. We also report the *CoNLL P/R/F1* scores, that is the averages of MUC’s, B^3’s and $CEAFe_{\phi 4}$’s respective precision (P), recall (R) and F1 scores.

6.4 Experimental Setup

Here we describe our experimental setup for coreference resolution and KG population.

6.4.1 Automatic Coreference Resolution

We evaluate three different state-of-the-art architectures on our STM dataset:

(I) *BERT for Coreference Resolution (BFCR)* [137] with *SpanBERT* [136] word embeddings (referred to as *BFCR_ Span*).

(II) BFCR with *SciBERT* [19] word embeddings (referred to as *BFCR_ Sci*).

(III) *Scientific Information Extractor (SCIIE)* [178] with ELMo [217] word embeddings (referred to as *SCIIE*).

The three architectures are evaluated with the following six approaches (#1 - #6):

- *Pre-Trained Models:* We evaluate already pre-trained models on the test sets of the STM corpus, i.e. #1 *BFCR_ Span* trained on the English portion of the OntoNotes dataset [223], and #2 *SCIIE* trained on SciERC [178] from the AI domain.
- *Supervised Learning:* We train a model from scratch with the three architectures using the training data of the STM corpus and evaluate their performance with the test sets of STM: #3 *BFCR_ Span*, #4 *BFCR_ Sci*, and #5 *SCIIE*.
- *Transfer Learning:* This is our proposed approach #6. We fine-tune all parameters of a pre-trained model on the English portion of the OntoNotes dataset [136] with the training data of our STM corpus. For that, we use the *BFCR_ Span* architecture.

Evaluation: We use the metrics MUC [284], B^3 [12], $CEAFe_{\phi 4}$ [180], and $CoNLL$ [221] in compliance with other studies on coreference resolution [137, 170, 182]. To obtain robust results, we apply five-fold cross-validation, according to the data splits of the original STM corpus, and report averaged results. For each fold, the dataset is split into train/validation/test sets with 8/1/2 abstracts per domain, respectively, i.e. 80/10/20 abstracts. We reuse the original implementations and default hyperparameters of the above architectures. Hyperparameter-tuning of the best baseline approach #3 according to [137] confirmed that the default hyperparameters of *BFCR_ Span* perform best on our corpus.

6.4.2 Evaluation of Knowledge Graph Population Strategies

We compare four KG population strategies: (1) cross-domain and (2) in-domain collapsing with coreference resolution, as well as (3) cross-domain and (4) in-domain collapsing without coreference resolution. To evaluate cross-domain and in-domain collapsing, we take the gold clusters (i.e. mention clusters within the abstracts) of the Test-STM-KG and collapse them to concepts according to the respective strategy. When leaving out the coreference resolution step, we treat all mentions in the Test-STM-KG as singleton clusters and collapse them to concepts according to the respective strategy. Finally, we calculate the metrics as described in Section 6.3.4.

6.5 Results and Discussion

In this section, we discuss the experimental results for automatic coreference resolution and KG population.

6.5.1 Automatic Coreference Resolution

Table 6.5 shows the overall results of the six evaluated approaches and Table 6.6 the results per domain of the best baseline #3 and our approach #6. Our transfer learning approach #6 *BFCR_ Span* from OntoNotes (Onto) [222] to STM significantly outperforms the best baseline approach #3 with an overall CoNLL F1 of 61.4% (+10.0%) and a low standard deviation ±1.5 across the five folds.

The approaches #1 *BFCR_ Span* pre-trained on OntoNotes [222], and #2 *SCIIE* pre-trained on SciERC [178] achieve a CoNLL F1 score of 37.1% and 7.4%, respectively. These scores are quite low compared to the approaches #3 - #6 that use training data of the STM corpus. This indicates that models pre-trained on existing datasets do not generalise sufficiently well for coreference resolution in research papers. Models trained only on the STM corpus (i.e. #3 - #5) achieve better results. However, they have quite low recall scores indicating that the size of the training data might not be sufficient to enable the

Table 6.5: Performance of the baseline approaches (in percent) #1-#5 and our transfer learning approach #6 on the the STM corpus across five-fold cross validation.

		MUC			B^3			$CEAFe_{\phi 4}$			*CoNLL*		
	Training data	P	R	F1	P	R	F1	P	R	F1	P	R	F1
#1 BFCR_Span	OntoNotes	57.1	31.1	40.2	55.9	25.7	35.2	50.2	28.1	36.0	54.4	28.3	37.1
#2 SCIIE	SciERC	13.4	4.5	6.8	13.1	4.3	6.5	18.1	6.0	9.0	14.9	4.9	7.4
#3 BFCR_Span	STM	61.6	45.6	52.3	59.8	41.5	48.8	57.9	44.4	50.0	59.8	43.8	50.4
#4 BFCR_Sci	STM	61.9	40.2	48.6	59.7	36.1	44.9	61.7	36.9	46.0	61.1	37.7	46.5
#5 SCIIE	STM	60.3	45.2	51.6	57.6	41.7	48.3	56.6	43.6	49.1	58.1	43.5	49.7
#6 BFCR_Span	**Onto→STM**	**64.5**	**63.5**	**63.9**	**61.0**	**60.0**	**60.4**	**60.5**	**59.6**	**60.0**	**62.0**	**61.0**	**61.4**

Table 6.6: Per domain and overall CoNLL F1 results (in percent) of the best baseline #3 and our approach #6 on the STM corpus across five-fold cross validation.

	Training data	Agr	Ast	Bio	Che	CS	ES	Eng	MS	Mat	Med	Overall
#3 BFCR_Span	STM	48.0	50.5	52.2	49.0	59.1	39.6	52.8	47.6	42.5	51.0	50.4
#6 BFCR_Span	**Onto→STM**	**62.8**	**61.1**	**57.5**	**56.3**	**74.9**	**57.5**	**59.8**	**52.1**	**55.7**	**62.1**	**61.4**

model to generalise well. The approach SciBERT #4, although pre-trained on scientific texts, performs worse than SpanBERT #3. Presumably the reason is that SpanBERT has approximately 3 times more parameters than SciBERT. Our transfer learning approach #6 achieves the best results with quite balanced precision and recall scores.

Furthermore, to evaluate the effectiveness of our transfer learning approach, we compare the best baseline #3 and our transfer learning approach #6 also with the SciERC corpus [178]. The SciERC corpus comprises 500 abstracts from the AI domain. Since SciERC has around 5 times more training data than STM, we compare the approaches #3 and #6 also using only $\frac{1}{5}$th of the training data in SciERC while keeping the original validation and test sets. It can be seen in Table 6.7 that our transfer learning approach #6 improves slightly the baseline result using the whole training data with 60.1% F1 (+0.8%). When using only $\frac{1}{5}$th of the training data, our transfer learning approach noticeably outperforms the baseline with 54.2% F1 (+7.1%). Thus, our transfer learning approach can help significantly to improve the performance of coreference resolution in research papers with few labelled data.

6.5.2 Cross-Domain Research Knowledge Graph

In this subsection, we describe the characteristics of our populated RKG (referred to as STM-KG) and discuss the evaluation results of various KG population strategies.

Table 6.7: CoNLL scores (in percent) on the SciERC corpus [178] across 3 random restarts of the approaches: approach of Luan et al. [178], the best baseline approach (#3), and our transfer learning approach (#6). We report results using the whole and using only $\frac{1}{5}$th of the training data of SciERC [178] (referred to as $\frac{1}{5}$SciERC).

		Training data	P	R	F1
Luan et al. [178]		SciERC	52.0	44.9	48.2
#3	BFCR_Span	SciERC	63.3	55.7	59.3
#6	**BFCR_Span**	**OntoNotes→SciERC**	**63.9**	**57.1**	**60.1**
#3	BFCR_Span	$\frac{1}{5}$SciERC	63.1	39.1	47.1
#6	**BFCR_Span**	**OntoNotes→ $\frac{1}{5}$SciERC**	**52.8**	**56.7**	**54.2**

Table 6.8: Characteristics of the populated cross-domain and in-domain research STM-KGs per domain: (1) number of abstracts, number of extracted scientific concept mentions and coreferent mentions, (2) the number of scientific concepts for the KG with cross-domain collapsing, (3) in-domain collapsing, (4) cross-domain collapsing but without coreference resolution, and (5) in-domain collapsing but without coreference resolution. Reduction denotes the percentual reduction of mentions to scientific concepts and MIX the cross-domain concepts.

	Agr	Ast	Bio	CS	Che	ES	Eng	MS	Mat	Med	MIX	Total
# abstracts	7,731	15,053	11,109	1,216	1,234	2,352	3,049	2,258	665	10,818	-	55,485
# mentions	332,983	370,311	423,315	45,388	46,203	129,288	127,985	86,490	20,466	586,019	-	2,168,448
# coref. men.	108,579	120,942	143,292	17,674	14,059	40,974	42,654	25,820	8,510	203,884	-	726,388
cross-domain collapsing												
KG concepts	138,342	173,027	177,043	20,474	21,298	62,674	55,494	39,211	9,275	227,690	70,044	994,572
- Data	27,132	64,537	32,946	5,380	5,124	19,542	17,053	10,629	2,982	66,473	19,715	271,513
- Material	69,534	45,296	83,627	6,242	10,154	24,322	19,689	17,276	2,406	68,141	20,812	367,499
- Method	2,992	8,819	6,135	2,001	1,055	1,776	2,953	1,605	685	9,363	1,627	39,011
- Process	38,684	54,375	54,335	6,851	4,965	17,034	15,799	9,701	3,202	83,713	27,890	316,549
reduction	58%	53%	58%	55%	54%	52%	57%	55%	55%	61%	-	54%
in-domain collapsing												
KG concepts	180,135	197,605	229,201	30,736	32,191	81,584	78,417	55,358	14,567	278,686	-	1,178,480
reduction	46%	47%	46%	32%	30%	37%	39%	36%	29%	52%	-	46%
cross-domain collapsing without coreference resolution												
KG concepts	146,894	182,479	187,557	21,950	22,555	66,600	59,689	41,776	9,939	242,797	77,493	1,059,729
reduction	56%	51%	56%	52%	51%	48%	53%	52%	51%	59%	-	51%
in-domain collapsing without coreference resolution												
KG concepts	184,218	199,894	234,399	31,525	32,937	83,445	80,476	56,690	14,911	284,547	-	1,203,042
reduction	45%	46%	45%	31%	29%	35%	37%	34%	27%	51%	-	45%

6.5.2.1 Characteristics of the Research Knowledge Graph

Table 6.8 shows the characteristics of the populated cross-domain and in-domain STM-KGs per domain. The resulting STM-KGs with cross-domain and in-domain collapsing have more than 994,000 and 1.1 Mio. scientific concepts, respectively, obtained from 55,485 abstracts with more than 2,1 Mio. concept mentions and 726,000 coreferent mentions. *Ast* and *Bio* are the most represented domains, while *CS* and *Mat* are the most underrepresented.

Table 6.9: Performance of the collapsing strategies evaluated against the *Test-STM-KG*: in-domain and cross-domain collapsing with and without coreference resolution.

	#concepts	*MUC*			B^3			$CEAFe_{\phi4}$			*CoNLL*		
	in KG	P	R	F1	P	R	F	P	R	F1	P	R	F1
in-domain collapsing	859	**86.3**	70.6	77.7	**86.0**	69.0	76.6	84.1	23.1	36.2	**85.5**	54.2	63.5
- without coreferences	900	75.5	38.8	51.2	75.2	37.9	50.4	71.1	14.0	23.4	73.9	30.2	41.7
cross-domain collapsing	837	85.0	**73.0**	**78.5**	84.5	**72.1**	**77.8**	**84.7**	**24.6**	**38.1**	84.7	**56.6**	**64.8**
- without coreferences	876	73.5	41.0	52.6	72.2	15.5	25.5	72.2	15.5	25.5	73.0	32.4	43.5

6.5.2.2 Evaluation of Knowledge Graph Population Strategies

Next, we discuss the different KG population strategies. For each strategy, Table 6.8 reports the number of concepts in the populated KG and the percentage reduction of mentions to concepts, and in Table 6.9 the evaluation results of KGs against the Test-STM-KG.

Cross-Domain vs. In-Domain Collapsing: Cross-domain collapsing achieves a higher CoNLL F1 score of 64.8% than in-domain collapsing with a score of 63.5% (see Table 6.9). However, in-domain collapsing yields (as expected) a higher precision (CoNLL P 85.5%), since some terms have different meanings across domains (e.g. *Measure_(mathematics)* vs. *Measurement* in `https://en.wikipedia.org`). Furthermore, the Test-STM-KG has only 31 cross-domain concepts due to its small size. Thus, we expect that cross-domain collapsing would yield worse results on a larger test set.

Furthermore, as shown in Table 6.8, cross-domain collapsing yields less concepts than in-domain collapsing (more than 994,000 versus 1.1 Mio. concepts). We can also observe that only 70,044 (7%) of the concepts are used across multiple domains. This indicates that every scientific domain mostly uses its own terminology. However, the concepts used across domains can have different meanings. Thus, when precision is more important than recall in downstream tasks, in-domain collapsing should be the preferred choice.

Effect of Coreference Resolution: Coreference resolution has only a small impact on the number of resulting concepts in a populated STM-KG (see Table 6.8). However, as shown in Table 6.9, leaving out the coreference resolution step during KG population yields only low CoNLL F1 scores, i.e. 41.7% (-21.8) F1 and 43.5% (-21.3) F1. Thus, coreference resolution significantly improves the quality of a populated KG .

6.5.2.3 Qualitative Analysis

We also inspected the top-five frequent domain-specific concepts in the populated STM-KG. A list of these concepts can be found in our public repository and an excerpt is depicted in Table 6.10. As far as we can judge with our computer science background, we consider the

Table 6.10: Topmost domain-specific and cross-domain concepts in the populated STM-KG.

	Agr	Ast	Bio	CS	Che	ES	Eng	MS	Mat	Med	**MIX**
Process	pollinate	quantum chromodynamics	analytical ultracentrifugation	cyber attack	potent activity	new hydrological insight	thermal energy storage	spherical indentation	bisimulation	surgical treatment	increase
Method	braun-Blanquet procedure	standard model	SPSS 16	material point method	scanning tunneling microscopy	X-ray photoemission electron microscopy	contour method	magnetometry	spectral theory	intention-to-treat population	in vitro
Data	number	neutrino mass	mass spectrometry proteomics data	runtime performance	electrode potential	geological record	internal temperature	average crystallite size	finite sample	age and gender	number
Material	conservation status	black hole	PRIDE repository	interactive environment	spark ignition engine	volcano	near-wall region	polycrystalline sample	monoid	study group	model

extracted top frequent concepts to be reasonable and useful for the domains. For instance, in *Ast*, the method 'standard model' is frequently mentioned, while in *CS* the process 'cyber attack' appears most often. The frequency of the top concepts differs significantly between the domains: In *Med*, *Ast*, *Eng*, *ES*, and *Agr*, a top frequent concept is referenced 10.8, 10.2, 4.9, 3.8, and 3.1 times per 1000 abstracts, respectively. In *Che*, *MS*, *Mat*, *Bio*, and *CS*, a top frequent concept is referenced only by few abstracts (0.3, 0.4, 1.0, 1.4, and 2.3, respectively, per 1000 abstracts).

6.6 Summary

This chapter has addressed the research questions **RQ2** (few labelled data) and **RQ3** (domain-independent extraction) for the task of coreference resolution in research papers and also **RQ4** (RKG population) by populating an RKG. To investigate **RQ3**, we have annotated our STM corpus (see Chapter 5) comprising ten different domains from Science, Technology, and Medicine (STM) with coreference annotations and obtained a substantial inter-annotator agreement (0.68 κ). Baseline results on our corpus with current state-of-the-art approaches for coreference resolution confirmed that current approaches perform poorly on scientific text.

To address **RQ2**, we have proposed a sequential transfer learning approach that exploits annotated datasets from the general domain. Our experimental results demonstrated that the proposed approach noticeably outperforms the state-of-the-art baselines (F1 score of 61.4%, i.e. +11.0% absolute improvement). Thus, our transfer learning approach can help to reduce annotation costs for scientific papers while obtaining high-quality results at the same time.

Furthermore, we have explored **RQ4** (RKG population) in this chapter. First, we have investigated the impact of coreference resolution on KG population. For this purpose, we have compiled a gold KG from our annotated corpus and proposed an evaluation procedure for KG population strategies. We have demonstrated that coreference resolution has a small impact on the number of resulting concepts in the KG, but improved its quality significantly (F1 score of 63.5%, i.e. +21.8% absolute improvement). Moreover, collapsing mentions of scientific concepts across domains achieved a higher recall but a lower precision than collapsing mentions only within a single domain. Thus, collapsing of mentions within a domain should be preferred when precision is more important than recall in downstream tasks. Finally, we have generated an RKG (referred to as STM-KG) from 55,485 abstracts of the ten investigated domains. We have shown that every domain mainly uses its own terminology and that the populated RKG contains useful concepts.

◊◊◊

The objective of this chapter and the previous Chapter 5 was to populate a fine-grained RKG. In the next Chapter 7, we propose a novel approach for citation recommendation that can leverage such RKGs. Using that approach, we demonstrate the usefulness of our populated RKG.

7 Citation Recommendation via Knowledge Graphs

This chapter proposes a new approach for the task of citation recommendation using Research Knowledge Graphs (RKGs) and demonstrates the usefulness of our populated RKG introduced in the previous Chapter 6 for this task. Thus, this chapter addresses **RQ5**, namely:

> **RQ5:** *How can we exploit an automatically populated RKG to enhance the task of citation recommendation?*

This chapter is organised as follows: Section 7.1 motivates the usage of RKGs for the task of citation recommendation. Section 7.2 reviews existing approaches for citation recommendation, and Section 7.3 describes our proposed approach. The experimental setup and results are reported in Section 7.4 and 7.5, while Section 7.6 summarises this chapter.

7.1 Introduction

Citations are a core part of research articles as they enable the reader to position the novel contribution in the scientific context. Moreover, relating own contributions with relevant research via references can also improve visibility. In consequence, it is in the interest of authors to provide complete and high-quality citation links to existing research. However, this task becomes ever more complicated since the number of published research articles has been growing exponentially in the recent years [27, 188].

Consequently, the recommendation of suitable references for a piece of scientific writing is an important task to (a) improve the quality of future publications, (b) help authors and reviewers to point out additional relevant related work, and (c) discover interesting links to other areas of research. Färber and Jatowt [80] distinguish between *local citation recommendation* which aims to provide citations for a short passage of text, and *global citation recommendation* which uses the documents' full text or abstract as the input. Here, we focus on the task of global citation recommendation that is illustrated in Figure 7.1.

Current best-performing approaches for global citation recommendation [21, 56, 302] primarily leverage the articles' text and the citation network as information sources. In this

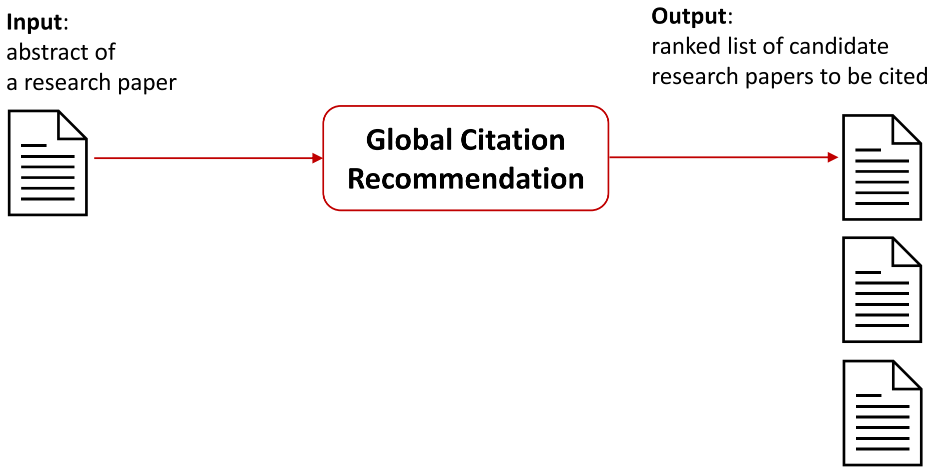


Figure 7.1: In the task of global citation recommendation, the system takes as input the abstract of a research paper and outputs a ranked list of candidate research papers to be cited.

chapter, we explore another source of information, that is the set of scientific concepts which are mentioned in the article. The assumptions are (1) that additionally to the article's text, these provide condensed evidence to the described problem statement, used methodology or evaluation metrics, and (2) that research papers which should cite one another usually share a similar set of concepts. Consequently, we investigate whether an RKG that interconnects papers based on the mentioned scientific concepts is instrumental in improving citation recommendation. For this purpose, we propose an approach which combines automatically extracted scientific concepts from the research articles with existing approaches for citation recommendation. The approach is evaluated on the STM-KG introduced in the previous Chapter 6 that has been automatically populated from papers of ten scientific domains. The experimental results demonstrate that our proposed approach consistently improves the state of the art with a MAP@50 (mean average precision of top-50 results) of 20.6% (+0.8% absolute improvement). To facilitate further research, we release all our corpora and source code: `https://github.com/arthurbra/citation-recommendation-kg`

7.2 Related Work

In the following, we outline recent approaches for global citation recommendation. For local citation recommendation, we refer to the survey of Färber and Jatowt [80].

Bhagavatula et al. [21] propose a neural network-based document embedding model to retrieve candidate documents for a query document via similarity search [134] and a ranking model to rerank the top-k candidates. The document embedding model is trained via a triplet loss with the papers' abstract and title using a Siamese architecture. It learns a high cosine similarity between embeddings of papers citing each other. The reranker estimates the probability that a query document should cite a candidate document using the abstract, title, and optional metadata (e.g. author, venue) as features. Cohan et al. [56] propose a document embedding model named SPECTER (Scientific Paper Embeddings using Citationinformed TransformERs). It is trained with an approach similar to Bhagavatula et al. [21]. However, they use a BERT encoder [71] pre-initialised with SciBERT embeddings [19]. Furthermore, Cohan et al. [56] omit the reranking step and obtain the ranked results directly via the document embeddings' cosine similarity.

Graph-based approaches learn document embeddings via graph convolution networks on the citation graph [107, 149, 293]. However, they require the citation network also at inference time [56]. Other approaches [40, 132, 302] frame citation recommendation as a binary classification task: given a query and a candidate paper, the model learns to predict whether the query paper should cite the candidate paper. The models learn rich relationships between the contents of the two documents via various cross-document attention mechanisms. However, in contrast to the document embedding models [21, 56], such binary classification models cannot be used for retrieval, but only for reranking the top k results, since a query paper has to be compared with all other documents [45], that is inefficient.

To the best of our knowledge, approaches for citation recommendation that exploit KGs with scientific concepts have not been proposed yet.

7.3 Citation Recommendation using a Knowledge Graph

As the discussion of related work shows citation recommendation approaches have not exploited RKGs yet. To leverage RKGs, we propose an approach to combine document embeddings learned from textual content and the citation graph together with scientific concepts mentioned in the document.

Let $\mathcal{G} = (\mathbb{D}, \mathbb{E}, \mathbb{V})$ be a KG, $\mathbb{D}$ the set of documents, $\mathbb{E}$ the set of concepts, $\mathbb{V} \subseteq \mathbb{D} \times \mathbb{E}$ the set of links between papers and concepts, and $\mathbb{E}_d \subseteq \mathbb{E}$ the set of concepts mentioned in a paper $d \in \mathbb{D}$. Let $onehot(e_i) \in \mathbb{R}^{|\mathbb{E}|}$ be the one-hot vector for concept e_i in which the i-th component equals 1 and all remaining components are 0. Now, we construct the *concept vector* $\mathbf{c_d} \in \mathbb{R}^{|\mathbb{E}|}$ for a paper $d \in \mathbb{D}$ as follows:

$$\mathbf{c_d} = \sum_{e_i \in \mathbb{E}_d} onehot(e_i) \tag{7.1}$$

Furthermore, let $\mathbf{s_d}$ be a document embedding of a paper d obtained via an existing document embedding model (e.g. SPECTER [56]). The *vector representation* $\mathbf{d}$ of a paper d is the concatenation of the concept vector $\mathbf{c_d}$ and the document embedding $\mathbf{s_d}$:

$$\mathbf{d} = [\mathbf{c_d}, \mathbf{s_d}] \tag{7.2}$$

Table 7.1: Characteristics of the STM-KG (see Section 6.5.2) per domain in terms of number of abstracts, the number of citation links within the KG, and the number of scientific concepts in the cross-domain and in-domain KG. The number of concepts used across multiple domains are denoted as MIX. The domains are: Agriculture (Agr), Astronomy (Ast), Biology (Bio), Chemistry (Che), Computer Science (CS), Earth Science (ES), Engineering (Eng), Materials Science (MS), Mathematics (Mat), and Medicine (Med).

	Agr	Ast	Bio	CS	Che	ES	Eng	MS	Mat	Med	MIX	Total
# abstracts	7,731	15,053	11,109	1,216	1,234	2,352	3,049	2,258	665	10,818	-	55,485
# citations	1,670	1,853	1,347	171	151	477	677	375	65	2,116	-	15,395
cross-domain KG												
KG concepts	138,342	173,027	177,043	20,474	21,298	62,674	55,494	39,211	9,275	227,690	70,044	994,572
in-domain KG												
KG concepts	180,135	197,605	229,201	30,736	32,191	81,584	78,417	55,358	14,567	278,686	-	1,178,480

For a query paper $q \in D$ the task is to retrieve the top k results such that papers to be cited appear at the top of the list. We use cosine similarity for retrieval and ranking where $\mathbf{q}$ for the query paper q is constructed in the same way as $\mathbf{d}$ for a paper d:

$$rank(q, d) = \cos(\mathbf{q}, \mathbf{d}) = \frac{\mathbf{q}^{\intercal} \cdot \mathbf{d}}{||\mathbf{q}|| \cdot ||\mathbf{d}||} \tag{7.3}$$

7.4 Experimental Setup

In this section, we describe the experimental setup, i.e. the used benchmark dataset, baseline approaches, and the evaluation procedure.

Benchmark Dataset: Existing benchmark datasets for research paper citation recommendation (e.g. [21, 56, 177]) do not provide an RKG that interlinks papers with scientific concepts. Therefore, we use the STM-KG introduced in Chapter 6 as our benchmark dataset whose characteristics are depicted in Table 7.1. It has been populated from 55,485 abstracts in ten different STM domains and comes in two variants: (1) in-domain KG that shares scientific concepts only between papers of the same domain to avoid ambiguity of scientific terms (e.g. neural network in medicine vs. computer science), and (2) cross-domain KG that shares scientific concepts also between domains.

The KG contains 15,395 citation links within the KG in total, of which 2,200 citation links are across papers from different domains. For evaluation, analogous to related work [21, 56], we use only papers that cite at least four papers within the KG which results in 720 query documents and 4,069 citations links. In contrast to Cohan et al. [56], we pursue a realistic approach like Bhagavatula et al. [21], i.e. we retrieve top-k documents from *all* documents in the corpus instead of using predefined candidate sets of 30 documents (5 cited and 25 uncited papers) for each query document.

Baseline Approaches: We compare our approach with two simple (1 and 2) and three strong baselines (3, 4, and 5):

1. **Random:** We use randomly initialised document embeddings with dimension 200.
2. **Concept vector:** Only the concept vector is used for ranking (see Equation 7.1).
3. **GloVe:** Document embedding of a paper is the average of GloVe [212] word embeddings obtained from the abstract of the paper.
4. **SciBERT:** Document embedding is also the average of the contextual word embeddings obtained from the abstract of the paper via SciBERT [19] that is based on BERT [71] and has been pre-trained on scientific text. It has demonstrated superior performance in various downstream tasks on research papers [19].
5. **SPECTER:** Document embedding is obtained via SPECTER [56] from the title and the abstract. The SPECTER model has been trained on the textual content and the citation graph of research papers, and is the current state of the art.

To compute GloVe and SciBERT document embeddings, we use the *sentence transformers* library [235]. For SPECTER we use the implementation of Cohan et al. [56].

Evaluation: To evaluate the quality of the ranking results for the top k citation recommendations, we use *Mean Average Precision* (MAP@k) [16, 162] as in related work [56] (see also Section 2.5.2.4). The metric assumes that a user is interested in finding many relevant documents and is thus an appropriate evaluation metric for citation recommendation.

7.5 Results and Discussion

The boxplots in Figure 7.2 depict the distribution of cosine similarities of concept vectors between citing and non-citing papers. It can be seen that papers citing each other have on average a higher cosine similarity than papers not citing each other. This underlines our hypothesis that papers citing each other share a common set of scientific concepts.

Table 7.2 shows the results of the evaluated approaches. Using only the concept vectors for ranking outperforms the random baseline noticeably. When using only certain concept types (i.e. PROCESS, METHOD, MATERIAL, or DATA), we can observe that MATERIAL and PROCESS concept types contribute most to the results. However, using all concept types together yields the best results.

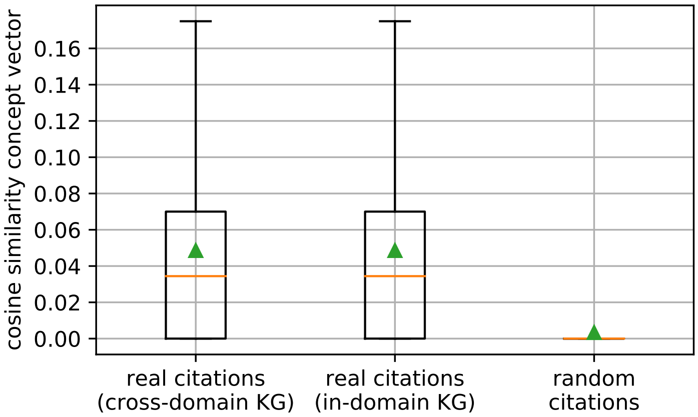


Figure 7.2: Boxplot for cosine similarities between concept vectors of papers citing each other (15,395 links) for cross-domain and in-domain KG, respectively, and papers citing random papers (15,395 links). The green triangles depict the mean values.

Table 7.2: Experimental results (in percent) for citation recommendation with random vectors, only the concept vector as well as document embeddings obtained from GloVe, SciBERT, and SPECTER with and without using the concept vector.

	MAP@10		MAP@20		MAP@50	
Random	0.0		0.0		0.0	
Concept vector (cross-domain KG)	7.5		8.0		8.5	
Concept vector (in-domain KG)	8.1		8.7		9.3	
- MATERIAL	3.7		4.1		4.4	
- PROCESS	3.6		3.9		4.2	
- DATA	1.9		2.1		2.2	
- METHOD	1.1		1.2		1.4	
GloVe	9.1		10.0		10.8	
GloVe + concept vector (cross-domain KG)	11.4	(+2.3)	12.5	(+2.5)	13.4	(+2.6)
GloVe + concept vector (in-domain KG)	11.3	(+2.2)	12.5	(+2.5)	13.5	(+2.7)
SciBERT	10.2		11.5		12.6	
SciBERT + concept vector (cross-domain KG)	12.1	(+1.9)	13.3	(+1.8)	14.4	(+1.8)
SciBERT + concept vector (in-domain KG)	11.9	(+1.7)	13.2	(+1.7)	14.4	(+1.8)
SPECTER	16.5		18.3		19.8	
SPECTER + concept vector (cross-domain KG)	16.9	(+0.4)	18.9	(+0.6)	20.5	(+0.7)
SPECTER + concept vector (in-domain KG)	**17.0**	(+0.5)	**19.0**	(+0.7)	**20.6**	(+0.8)

Baseline ranking approaches via document embeddings learned from the text (GloVe and SciBERT), or text and the citation graph (SPECTER) outperform the ranking only via concept vectors noticeably, while SPECTER performs best as expected. This indicates that concept vectors alone do not contain enough information for the task of citation recommendation. However, our proposed approach combining document embeddings and concept vectors consistently improves all baseline approaches. For SPECTER, the in-domain KG yields slightly better results than the cross-domain KG. However, in our error analysis we found out that concept vectors from the cross-domain KG provide more accurate rankings for cross-domain citations.

Our results indicate that the exploitation of an RKG as an additional source of information can improve the task of citation recommendation.

7.6 Summary

This chapter has addressed **RQ5** (exploit RKGs) by investigating whether an automatically populated RKG can enhance the task of citation recommendation. For this purpose, we have combined document embeddings learned from text and the citation graph together with concept vectors representing scientific concepts mentioned in a paper. The experimental results on our RKG introduced in Chapter 6 demonstrated that the concept vectors provide meaningful features for the task of citation recommendation. Thus, our approach could improve the state of the art with a MAP@50 (mean average precision of top-50 results) of 20.6% (+0.8% absolute improvement).

The successful application of our RKG for the task of citation recommendation supports our hypothesis introduced in Chapter 3 that automatically populated RKGs with high completeness but noisy data have the potential to support downstream information retrieval tasks such as finding related work or recommending research papers.

◊◊◊

This chapter has proposed an approach that can exploit an automatically populated RKG for an information retrieval task, namely citation recommendation. Thus, we could demonstrate the usefulness of the RKG for a downstream task. The next chapter concludes the thesis and outlines areas of future work.

8 Conclusions and Future Work

Current research infrastructures are not able to assist scientists appropriately in their core tasks since fundamental contents of research papers are not machine-interpretable. This thesis aims to extract relevant information from research papers with machine learning approaches for Research Knowledge Graphs (RKGs) that should structure and interlink scholarly knowledge.

The question of how to represent scholarly knowledge via KGs is an active area of research, and Section 1.2 outlined several challenges, i.e. the diversity and heterogeneity of scholarly knowledge, lack of labelled data and domain experts, and usefulness of automatically populated RKGs. Based on these challenges, we formulated five research questions **RQ1** to **RQ5** that are investigated in this thesis. In the following, Section 8.1 first provides results and answers with regard to the research questions, while Section 8.2 briefly summarises the contributions. Then, Section 8.3 outlines remaining limitations and sketches possible directions of future work.

8.1 Summary

As outlined in Section 1.2.1, scholarly knowledge is very heterogeneous and diverse, and its structured and interlinked representation yields conflicting requirements: On the one hand, we desire a comprehensive ontology and instance data with high correctness and completeness. On the other hand, comprehensive ontologies require a manual population of the instance data by domain experts, which is too time-consuming and costly and prevents high completeness; automatic approaches can only populate relatively simple ontologies with moderate accuracy and thus cannot ensure high correctness. To illuminate this problem, we have explored **RQ1**:

> **RQ1:** *What are the main requirements for scholarly knowledge representation to support various use cases in an RKG?*

In Chapter 3, we have conducted a requirements analysis for an ORKG [7] as an example for an RKG with regard to **RQ1**. Our analysis has focused on the elicitation of use cases, definition of quality requirements for the underlying KG to support these use cases, and construction strategies for an ORKG. First, we have presented literature-related use cases of researchers that should be supported by an ORKG and their specific requirements for the underlying ontology (i.e. granularity and domain-specificness) and instance data (i.e. completeness and correctness). The identified use cases are: *get research field overview* (#1), *find related work* (#2), *assess relevance of research papers* (#3), *extract relevant information from research papers* (#4), *get recommended articles* (#5), *obtain deep understanding of a research paper* (#6), and *reproduce results of a research paper* (#7). Then, based on this analysis, the identified use cases have been categorised into two groups:

1. The first group of use cases, i.e. *get research field overview* (#1), *extract relevant information from research papers* (#4), *obtain deep understanding of a research paper* (#6), and *reproduce results of a research paper* (#7), requires instance data with high correctness and rather fine-grained, domain-specific ontologies. However, moderate completeness of the instance data should be sufficient.
2. The second group of use cases, i.e. *find related work* (#2), *assess relevance of research papers* (#3), and *get recommended articles* (#5), requires high completeness of the instance data. However, the ontologies can be rather simple and domain-independent, and moderate correctness of the instance data should be adequate.

For each use case, we have presented strategies and described possible approaches for manual, semi-automatic, and fully-automatic construction of an ORKG. Manual and semi-automatic approaches are necessary for the first group of use cases since they require instance data with high correctness. Fully-automatic approaches are required for the second group of use cases since they require instance data of high completeness but can tolerate noisy data. Furthermore, we proposed a framework using lightweight ontologies (called templates) that can evolve by community curation. Moreover, we have outlined the interdependence of an ORKG with external systems, user interfaces, and APIs for third-party applications.

This thesis has focused on the second group of use cases and provided contributions for machine learning tasks that aim to support them. In Chapter 4, we addressed the task of *sequential sentence classification* that can assist the use case *assess relevance of research papers* (#3) since it enables identifying relevant sentences in research papers. Furthermore, we introduced the novel task of *domain-independent information extraction* that allows for the extraction of scientific concepts from research papers in a domain-independent manner. This can contribute to the use cases *find related work* (#2) and *get recommended articles* (#5) since we claim that a simple and domain-independent ontology and instance data with moderate correctness should be sufficient for them. Domain-independent information extraction consists of the two sub-tasks *scientific concept extraction* (see Chapter 5) and *coreference resolution* (see Chapter 6). To demonstrate the usefulness of our domain-independent information extraction approach, we proposed methods to populate an RKG spanning ten different scientific domains and presented a novel approach for the task of *citation recommendation* that can leverage this RKG (see Chapter 7). The task of citation recommendation can assist the use cases (#2) and (#5).

Next, we describe our results with regard to the research questions **RQ2-5** in the context of the aforementioned tasks. An RKG should cover multiple scientific domains, but the manual annotation of datasets for each scientific discipline is challenging and costly [9, 92]. Therefore, machine learning methods are required that can be adapted to new domains with few labelled data. This has been addressed by our next research question **RQ2**:

> **RQ2:** *How can we modify machine learning methods for information extraction from scientific texts to be adaptable to new domains with few labelled data?*

We have explored **RQ2** for three different tasks, namely *sequential sentence classification*, *scientific concept extraction*, and *coreference resolution*:

Sequential Sentence Classification: Previous work proposed different kinds of approaches for sequential sentence classification. For abstracts, deep learning is the preferred approach [54, 69, 100, 133, 257, 295], whereas for full papers, hand-crafted features and linear models have been suggested [6, 11, 85, 174]. We presented a unified deep learning approach that can be used to classify sentences in abstracts *and* full papers.

Furthermore, we have investigated transfer learning for sequential sentence classification since the community lacks studies for this task [15, 46, 106, 168, 169, 178, 207]. Transfer learning enables the combination of knowledge from multiple datasets to improve classification performance and thus reduces annotation costs. However, various studies showed that the success of transfer learning depends largely on the semantic relatedness of the tasks [196, 205, 238, 253]. We have presented a unified multi-task deep learning approach for sequential sentence classification and investigated the semantic relatedness of datasets from different scientific domains that cover either only abstracts or full papers. Our results demonstrated that classes of various dataset annotation schemes are semantically related, even though the datasets come from different domains, and cover either only abstracts or full papers. Thus, our approach enables us to combine datasets from different domains with varying structures to improve the overall prediction accuracy. Our cross-domain multi-task learning approach has outperformed the state of the art without any feature engineering on full paper datasets significantly while being competitive for datasets consisting of abstracts only.

Scientific Concept Extraction: Existing datasets for scientific concept extraction focus on at most three scientific domains and are annotated with rather domain-specific concept types [9, 53, 139, 178, 229]. We have introduced four generic scientific concept types, namely Process, Method, Material, and Data, and annotated a dataset comprising ten different scientific domains from Science, Technology, and Medicine (STM) using these concept types. We have demonstrated that a model trained with data from *all* domains noticeably outperforms the domain-specific models that were trained only with data from the respective domain. As for the results of transfer learning in sequential sentence classification (see above), the results confirm that the combination of training data from different scientific domains is also beneficial for scientific concept extraction.

Furthermore, we have proposed active learning in order to obtain an optimal selection of training instances, which to our knowledge, has been demonstrated for the first time on scholarly text. Our approach that combines active learning with a state-of-the-art deep learning system for scientific concept extraction achieves the same performance with only about half of the training data. Thus, our approach can significantly save annotation costs and enables a fast adaptation of machine leaning models to new domains.

Coreference Resolution: Datasets for coreference resolution in scientific text [42, 58, 178, 248] are smaller than datasets from the non-academic domain (e.g. [222]). However, current approaches for coreference resolution in research papers do not leverage these large datasets yet [137, 170, 171, 178, 182]. We have proposed a transfer learning approach in which a model is first trained with a large dataset from a non-academic domain and then fine-tuned with a (much smaller) dataset from a scientific domain. Our results showed that our approach significantly outperforms the state-of-the-art baselines so that it can help to reduce annotation costs while at the same time obtaining high-quality results.

In summary, we can conclude that a proper adaptation and selection of methods for (1) multi-task learning, (2) sequential transfer learning, and (3) active learning, as well as (4) the combination of training data from different scientific domains or even (5) exploiting training data from non-academic domains are effective strategies for information extraction from scientific texts. This enables machine learning models to adapt to new scientific domains with few additional labelled data.

As outlined above, an RKG should cover multiple scientific domains, but the manual annotation of datasets for each scientific domain is costly, time-consuming, and challenging. Therefore, our next research question **RQ3** has addressed information extraction from research papers in a domain-independent manner to avoid the manual annotation of datasets for each scientific discipline.

> **RQ3:** *How can we automatically extract information from research papers from multiple scientific domains in a domain-independent manner?*

Again, we have explored this research question for the three tasks of domain-independent *sequential sentence classification*, *scientific concept extraction*, and *coreference resolution.*

Domain-Independent Sequential Sentence Classification: Our proposed multi-task learning approach for sequential sentence classification enables us to exploit datasets from different scientific domains that cover either only abstracts or full papers. We have presented an approach to identify semantically related classes from different datasets semi-automatically. This allows for the support of manual comparison and inspection of different annotation schemes across domains and thus enables their consolidation. In contrast to prior work [175], our approach can avoid the re-annotation of datasets with different annotation schemes. Using our proposed approach, we have provided an analysis of four annotation schemes and presented a domain-independent model that allows for the classification of sentences in research papers with generic classes across disciplines. This model can support downstream applications such as academic search engines to structure research papers in a domain-independent manner.

Domain-Independent Scientific Concept Extraction: As stated above, existing datasets for scientific concept extraction cover at most three scientific domains that use rather domain-specific concept types. Moreover, these datasets were annotated either by domain experts [9, 68, 147, 175, 178, 229] (an approach that is costly) or non-experts [43, 85, 279] (an approach that is presumably cheaper). We have proposed a cost-efficient middle course: annotations by non-experts with scientific proficiency and consultation with domain experts. Using a systematic annotation procedure involving domain experts, we have annotated a corpus comprising ten STM domains and verified the adequacy of the concepts (i.e. Process, Method, Material, and Data) by evaluating the inter-annotator agreement. The results showed that the identification of the generic concepts in a corpus of ten different STM domains is feasible by non-experts with moderate agreement, and after consultation of domain experts with substantial agreement. Thus, the annotation of scientific datasets by non-experts annotators and involving domain experts for consultation is a promising strategy. Furthermore, as stated above, we have trained a state-of-the-art model using our annotated dataset that can extract scientific concepts from ten scientific domains with a fairly high F1 score.

Domain-Independent Coreference Resolution: Coreference resolution is an important complement for scientific concept extraction. However, existing corpora for coreference resolution in scientific texts are limited to a single domain [42, 58, 178, 248]. Furthermore, results of some previous studies [58, 146, 201, 248] revealed that general coreference systems do not work well in scientific domains due to the lack of domain knowledge. Therefore, we have extended our corpus for scientific concept extraction with coreference annotations. During a systematic annotation procedure, our non-domain experts achieved a substantial inter-annotator agreement. We have provided and compared baseline results for this dataset by evaluating five different state-of-the-art approaches. Our experimental results confirmed that state-of-the-art coreference approaches do not perform well on research papers. Therefore, we have proposed a sequential transfer learning approach that leverages an existing (large) dataset from non-academic domains (see above). The suggested approach noticeably outperformed five different state-of-the-art baselines on our annotated corpus.

In summary, we have proposed information extraction approaches to classify sentences in abstracts and full papers, and extract scientific concepts and coreferences from abstracts across several domains. Next, we have populated an RKG that spans multiple scientific domains. This has addressed our next research question **RQ4**:

> **RQ4:** *How can we automatically populate an RKG that covers multiple scientific domains?*

To the best of our knowledge, an RKG that spans multiple scientific domains has not been provided yet. Thus, it is not clear whether it is feasible to collapse coreference clusters of different papers across domains. Furthermore, the impact of coreference resolution on KG population has not been investigated yet. For this purpose, we have compiled a gold standard RKG from our annotated corpus that contains scientific concepts referenced by mentions from text, and presented a procedure to evaluate the clustering results of mentions. We have shown that coreference resolution has a small impact on the number of resulting concepts in the RKG, but improves its quality significantly. Moreover, we investigated two strategies of collapsing mentions of scientific concepts to entities in the RKG. The results demonstrated that collapsing mentions across domains achieved a higher recall but a lower precision than collapsing mentions only within a domain. Thus, collapsing of mentions within a domain should be preferred in downstream tasks when precision is more important than recall.

Consequently, we have generated an RKG from 55,485 abstracts of the ten investigated STM domains. We have shown that every domain mainly uses its own terminology and that the populated RKG contains useful concepts.

Finally, to demonstrate the usefulness of our populated RKG for the use cases *find related work* (#2) and *get recommended articles* (#3), approaches that can exploit such RKGs are required. For this purpose, we have explored the task of citation recommendation that can recommend researchers suitable related work based on a piece of scientific writing. This has been targeted by our last research question **RQ5**:

> **RQ5:** *How can we exploit an automatically populated RKG to enhance the task of citation recommendation?*

Current approaches for the task of citation recommendation primarily rely on the text of the papers and the citation network [21, 40, 56, 132, 302]. We have proposed to exploit an additional source of information, namely RKGs that interlink research papers based on mentioned scientific concepts. For this purpose, we have combined document embeddings learned from text and the citation graph together with concept vectors representing scientific concepts mentioned in a paper. The achieved experimental results were better than state-of-the-art baselines when utilising our populated RKG of ten STM domains. This indicates that the concept vectors provide meaningful features for the task of citation recommendation.

8.2 Summary of Contributions

Overall, the contributions of this thesis can be summarised as follows:

1. requirements analysis for an ORKG, including:
    - 1.1. a use case analysis of literature-related tasks of researchers;
    - 1.2. the definition of data quality requirements for the underlying ontologies and instance data;
    - 1.3. an elaboration of RKG construction strategies for the corresponding use cases.
    - 1.4. a new framework using lightweight ontologies (templates) that can evolve by community curation.
2. new methods for the task of sequential sentence classification, including:
    - 2.1. a unified deep learning approach that is applicable to datasets of different text types, i.e. abstract and full papers, without any feature engineering;
    - 2.2. a comprehensive empirical comparison of sequential transfer learning and multi-task learning approaches as well as an investigation of the transferability of various neural network layers;
    - 2.3. a new cross-domain multi-task learning approach that enables the utilisation of datasets from scientific domains with different structures and text types;
    - 2.4. a new semi-automatic approach for the identification and analysis of the semantic relatedness of sentence classes across different annotation schemes to support their consolidation;
    - 2.5. a new domain-independent sequential sentence classification approach that enables the classification of sentences in research papers in a domain-independent manner.
3. introduction of the novel task of domain-independent scientific concept extraction, including:
    - 3.1. an annotated corpus of 110 abstracts from ten different STM domains including:
        - i. scientific concepts of four generic scientific concept types;
        - ii. coreference links between the scientific concepts.

   3.2. demonstration of the usefulness of a new cost-effective annotation strategy by non-domain experts and consultation with domain experts;
   3.3. a new annotation strategy for scientific concepts that combines active learning with a state-of-the-art deep learning approach;
   3.4. a comprehensive comparison of the domain-specific classifiers and the domain-independent classifier for scientific concept extraction for ten STM domains;
   3.5. a new sequential transfer learning approach for coreference resolution that exploits large datasets from non-academic domains;
   3.6. a comprehensive evaluation of state-of-the-art approaches for coreference resolution on scientific text from ten STM domains.
4. population of a cross-domain RKG, including:
   4.1. an automatically populated RKG from over 55.000 abstracts that covers ten different STM domains;
   4.2. a novel approach is proposed that investigates:
      i. the impact of coreference resolution on the KG population;
      ii. KG population strategies (i.e. cross-domain and in-domain collapsing).
   4.3. a new compiled gold-standard RKG that enables to evaluate RKG population strategies;
   4.4. qualitative analysis of the extracted concepts in the populated RKG.
5. a novel approach for the task of global citation recommendation that exploits RKGs by combining existing state-of-the-art approaches for citation recommendation with concept vectors that represent scientific concepts of an RKG mentioned in a paper;
6. the source code and datasets of this thesis are made publicly available to facilitate further research.

8.3 Limitations and Future Work

The realisation of an RKG is an ambitious project that requires a lot of research and engineering effort. This thesis has made some important contributions towards RKGs in the context of cross-domain information extraction from research papers. However, there are still some remaining limitations and interesting areas of future research, which are described in the following.

Requirements Analysis: Our analysis aimed to give a holistic view of the requirements for an ORKG and outlined possible approaches for the construction of it. Next, following the Design Science Research (DSR) methodology [35, 121], the suggested approaches have to be refined, implemented, and evaluated in an iterative and incremental process. In this thesis, we focused on automatic KG population approaches that are necessary for specific use cases that require high completeness of the instance data (i.e. *find related work*, *get recommended articles*, *assess relevance*). Other use cases (e.g. *extract relevant information*, *get research field overview*) require a rather high correctness of the instance data, and thus semi-automatic KG population approaches are better suited for them. However, these use cases also require a high domain-specificness, so that it is necessary to annotate datasets and train machine learning models for each specific ontology. For instance, each defined template requires a separate machine learning model. One promising research direction is to employ few-shot learning, which enables training machine learning models with only a few samples (e.g. 16 to 32 samples). Current few-shot learning approaches leverage large pre-trained language models and achieve promising results [37, 95]. Moreover, the development of intuitive user interfaces with a high usability should also be addressed in the future to support manual curation of data in an ORKG. Furthermore, since ontologies and instance data evolve in an ORKG, solutions are required to adequately support this evolution process (e.g. editing, versioning, support to report inconsistencies, etc.).

Sequential Sentence Classification: Our unified multi-task learning approach for sequential sentence classification enabled us to combine training data from different scientific domains. However, there are several possible improvements to our approach. For instance, the architecture does not exploit the hierarchical structure of the text in full papers (e.g. sections and paragraphs). Furthermore, it can be extended with further tasks such as scientific concept extraction or entity linking to KGs utilising multi-task learning (e.g. [178, 246, 290]). The mentions of certain scientific concepts in a sentence could help the model to detect the sentence class more accurately and vice versa. Moreover, the domain-independent sentence classifier should be evaluated in an information retrieval scenario (e.g. [242]).

Domain-Independent Information Extraction: Our annotated corpus contains four generic concept types and covers ten STM domains. However, during our annotation study, we also identified further domain-independent candidate concept types such as Task, Object, and Result, which were almost always nested with the other scientific concepts. Therefore, a second layer using these concept types can be annotated in future work. Additionally, the concept types can be refined for specific domains (e.g. a participant in a clinical trial is treated as Material in our corpus), and the corpus should be extended with relations between scientific concepts, which would allow for the population of a richer RKG. Furthermore, our domain-independent information extraction approach requires the training of two models, one for scientific concept extraction and one for coreference resolution. Multi-task learning would allow for the training of a single model (e.g. [178, 246, 290]). Both tasks could also benefit from one another in a multi-task learning setting since concept extraction might help coreference resolution to detect the concepts more accurately and vice versa.

Cross-Domain Research Knowledge Graph: This thesis has evaluated our populated RKG only for one downstream task, namely for citation recommendation. In future work, the RKG should also be evaluated on further information retrieval tasks such as academic search engines [294] or graph-based research paper recommendation systems [18]. For this purpose, a much larger RKG should be populated and integrated with existing KGs such as the Microsoft Academic Knowledge Graph [78]. Furthermore, a large RKG might give us more insights into scientific language use and enable co-occurrence analysis between scientific concepts. Finally, the integration of graph completion tasks (see Section 2.4.2.2) is a promising research direction to remove irrelevant or incorrect data from a populated KG.

Citation Recommendation: Our approach for citation recommendation can leverage an RKG as an additional source of information and we evaluated the approach on our populated RKG from ten STM domains. In future work, the approach should also be evaluated with further RKGs. Furthermore, more powerful approaches that can learn document embeddings jointly from text, the citation graph, *and* the RKG may be explored (e.g. [56]).

◊◊◊

In summary, this thesis explored automatic methods for information extraction from scientific text for RKGs. The results suggest that automatic methods are appropriate for use cases, in which the extracted information is consumed by computers (e.g. retrieval and ranking for the use cases *find related work* and *get recommended articles*), or presented as suggestions for the user (e.g. in the use case *assess relevance*). However, for other use cases, where a user consumes the information (e.g. *get research field overview*), semi-automatic approaches are more appropriate and are thus also a promising direction of future research.

References

[1] Ahmed AbuRa'ed, Horacio Saggion, Alexander Shvets, and Àlex Bravo. "Automatic related work section generation: experiments in scientific document abstracting". In: *Scientometrics* 125.3 (2020), pp. 3159–3185. DOI: `10.1007/s11192-020-03630-2`. URL: `https://doi.org/10.1007/s11192-020-03630-2`.

[2] "American National standard for writing abstracts". In: *IEEE Transactions on Professional Communication* PC-20.4 (1977), pp. 252–254. DOI: `10.1109/TPC.1977.6591959`.

[3] Waleed Ammar et al. "Construction of the Literature Graph in Semantic Scholar". In: *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 3 (Industry Papers).* Ed. by Srinivas Bangalore, Jennifer Chu-Carroll, and Yunyao Li. Association for Computational Linguistics, 2018, pp. 84–91. DOI: `10.18653/v1/n18-3011`. URL: `https://doi.org/10.18653/v1/n18-3011`.

[4] Ron Artstein and Massimo Poesio. "Inter-Coder Agreement for Computational Linguistics". In: *Comput. Linguistics* 34.4 (2008), pp. 555–596. DOI: `10.1162/coli.07-034-R2`. URL: `https://doi.org/10.1162/coli.07-034-R2`.

[5] Amir Aryani and Jingbo Wang. "Research Graph: Building a Distributed Graph of Scholarly Works using Research Data Switchboard". In: *Open Repositories CONFERENCE.* June 2017. DOI: `10.4225/03/58c696655af8a`. URL: `https://figshare.com/articles/Research_Graph_Building_a_Distributed_Graph_of_Scholarly_Works_using_Research_Data_Switchboard/4742413`. published.

[6] Nasrin Asadi, Kambiz Badie, and Maryam Tayefeh Mahmoudi. "Automatic zone identification in scientific papers via fusion techniques". In: *Scientometrics* 119.2 (2019), pp. 845–862. DOI: `10.1007/s11192-019-03060-9`. URL: `https://doi.org/10.1007/s11192-019-03060-9`.

[7] Sören Auer. "Towards an Open Research Knowledge Graph". In: *Zenodo* (Jan. 2018). DOI: `10.5281/zenodo.1157185`.

[8] Sören Auer, Allard Oelen, Muhammad Haris, Markus Stocker, Jennifer D'Souza, Kheir Eddine Farfar, Lars Vogt, Manuel Prinz, Vitalis Wiens, and Mohamad Yaser Jaradeh. "Improving Access to Scientific Literature with Knowledge Graphs". In: *Bibliothek Forschung und Praxis* 44.3 (2020), pp. 516–529. DOI: `doi:10.1515/bfp-2020-2042`. URL: `https://doi.org/10.1515/bfp-2020-2042`.

[9] Isabelle Augenstein, Mrinal Das, Sebastian Riedel, Lakshmi Vikraman, and Andrew McCallum. "SemEval 2017 Task 10: ScienceIE - Extracting Keyphrases and Relations from Scientific Publications". In: *Proceedings of the 11th International Workshop on Semantic Evaluation, SemEval@ACL 2017, Vancouver, Canada, August 3-4, 2017*. Ed. by Steven Bethard, Marine Carpuat, Marianna Apidianaki, Saif M. Mohammad, Daniel M. Cer, and David Jurgens. Association for Computational Linguistics, 2017, pp. 546–555. DOI: 10.18653/v1/S17-2091. URL: https://doi.org/10.18653/v1/S17-2091.

[10] Lei Jimmy Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. "Layer Normalization". In: *CoRR* abs/1607.06450 (2016). arXiv: 1607.06450. URL: http://arxiv.org/abs/1607.06450.

[11] Kambiz Badie, Nasrin Asadi, and Maryam Tayefeh Mahmoudi. "Zone identification based on features with high semantic richness and combining results of separate classifiers". In: *J. Inf. Telecommun.* 2.4 (2018), pp. 411–427. DOI: 10.1080/24751839.2018.1460083. URL: https://doi.org/10.1080/24751839.2018.1460083.

[12] Amit Bagga and Breck Baldwin. "Algorithms for Scoring Coreference Chains". In: *In The First International Conference on Language Resources and Evaluation Workshop on Linguistics Coreference*. Vol. 1. 1998, pp. 563–566.

[13] Xiaomei Bai, Mengyang Wang, Ivan Lee, Zhuo Yang, Xiangjie Kong, and Feng Xia. "Scientific Paper Recommendation: A Survey". In: *IEEE Access* 7 (2019), pp. 9324–9339. DOI: 10.1109/ACCESS.2018.2890388. URL: https://doi.org/10.1109/ACCESS.2018.2890388.

[14] Krisztian Balog. "Entity-Oriented Search". In: vol. 39. The Information Retrieval Series. Springer, 2018. ISBN: 978-3-319-93933-9. DOI: 10.1007/978-3-319-93935-3. URL: https://eos-book.org.

[15] Soumya Banerjee, Debarshi Kumar Sanyal, Samiran Chattopadhyay, Plaban Kumar Bhowmick, and Partha Pratim Das. "Segmenting Scientific Abstracts into Discourse Categories: A Deep Learning-Based Approach for Sparse Labeled Data". In: *JCDL '20: Proceedings of the ACM/IEEE Joint Conference on Digital Libraries in 2020, Virtual Event, China, August 1-5, 2020*. Ed. by Ruhua Huang, Dan Wu, Gary Marchionini, Daqing He, Sally Jo Cunningham, and Preben Hansen. ACM, 2020, pp. 429–432. DOI: 10.1145/3383583.3398598. URL: https://doi.org/10.1145/3383583.3398598.

[16] Elias Bassani. *Rank_ eval: Blazing Fast Ranking Evaluation Metrics in Python.* https://github.com/AmenRa/rank_eval. 2021.

[17] Sean Bechhofer, Iain E. Buchan, David De Roure, Paolo Missier, John D. Ainsworth, Jiten Bhagat, Philip A. Couch, Don Cruickshank, Mark Delderfield, Ian Dunlop, Matthew Gamble, Danius T. Michaelides, Stuart Owen, David R. Newman, Shoaib Sufi, and Carole A. Goble. "Why linked data is not enough for scientists". In: *Future Gener. Comput. Syst.* 29.2 (2013), pp. 599–611. DOI: 10.1016/j.future.2011.08.004. URL: https://doi.org/10.1016/j.future.2011.08.004.

[18] Jöran Beel, Bela Gipp, Stefan Langer, and Corinna Breitinger. "Research-paper recommender systems: a literature survey". In: *Int. J. Digit. Libr.* 17.4 (2016), pp. 305–338. DOI: 10.1007/s00799-015-0156-0. URL: https://doi.org/10.1007/s00799-015-0156-0.

[19] Iz Beltagy, Kyle Lo, and Arman Cohan. “SciBERT: A Pretrained Language Model for Scientific Text”. In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*. Ed. by Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan. Association for Computational Linguistics, 2019, pp. 3613–3618. DOI: 10.18653/v1/D19-1371. URL: https://doi.org/10.18653/v1/D19-1371.

[20] Tim Berners-Lee and Mark Fischetti. *Weaving the Web: The Original Design and Ultimate Destiny of the World Wide Web by Its Inventor*. 1st. Harper San Francisco, 1999. ISBN: 0062515861.

[21] Chandra Bhagavatula, Sergey Feldman, Russell Power, and Waleed Ammar. “Content-Based Citation Recommendation”. In: *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers)*. Ed. by Marilyn A. Walker, Heng Ji, and Amanda Stent. Association for Computational Linguistics, 2018, pp. 238–251. DOI: 10.18653/v1/n18-1022. URL: https://doi.org/10.18653/v1/n18-1022.

[22] Christian Bizer. *Quality-Driven Information Filtering- In the Context of Web-Based Information Systems*. Saarbrücken, DEU: VDM Verlag, 2007. ISBN: 3836422328.

[23] Olivier Bodenreider. “The Unified Medical Language System (UMLS): integrating biomedical terminology”. In: *Nucleic Acids Res.* 32.Database-Issue (2004), pp. 267–270. DOI: 10.1093/nar/gkh061. URL: https://doi.org/10.1093/nar/gkh061.

[24] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomás Mikolov. “Enriching Word Vectors with Subword Information”. In: *Trans. Assoc. Comput. Linguistics* 5 (2017), pp. 135–146. URL: https://transacl.org/ojs/index.php/tacl/article/view/999.

[25] Kurt D. Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. “Freebase: a collaboratively created graph database for structuring human knowledge”. In: *Proceedings of the ACM SIGMOD International Conference on Management of Data, SIGMOD 2008, Vancouver, BC, Canada, June 10-12, 2008*. Ed. by Jason Tsong-Li Wang. ACM, 2008, pp. 1247–1250. DOI: 10.1145/1376616.1376746. URL: https://doi.org/10.1145/1376616.1376746.

[26] Grady Booch, James Rumbaugh, and Ivar Jacobson. *Unified Modeling Language User Guide, The (2nd Edition) (Addison-Wesley Object Technology Series)*. Addison-Wesley Professional, 2005. ISBN: 0321267974.

[27] Lutz Bornmann and Rüdiger Mutz. “Growth rates of modern science: A bibliometric analysis based on the number of publications and cited references”. In: *J. Assoc. Inf. Sci. Technol.* 66.11 (2015), pp. 2215–2222. DOI: 10.1002/asi.23329. URL: https://doi.org/10.1002/asi.23329.

[28] Arthur Brack, Jennifer D’Souza, Anett Hoppe, Sören Auer, and Ralph Ewerth. “Domain-Independent Extraction of Scientific Concepts from Research Articles”. In: *Advances in Information Retrieval - 42nd European Conference on IR Research, ECIR 2020, Lisbon, Portugal, April 14-17, 2020, Proceedings, Part I*. Ed. by Joemon M. Jose, Emine Yilmaz, João Magalhães, Pablo Castells, Nicola Ferro, Mário J. Silva, and Flávio Martins. Vol. 12035. Lecture Notes in Computer Science. Springer, 2020,

pp. 251–266. DOI: 10.1007/978-3-030-45439-5_17. URL: https://doi.org/10.1007/978-3-030-45439-5_17.

[29] Arthur Brack, Anett Hoppe, Pascal Buschermöhle, and Ralph Ewerth. "Cross-Domain Multi-Task Learning for Sequential Sentence Classification in Research Papers". In: *JCDL '22: Proceedings of the ACM/IEEE Joint Conference on Digital Libraries in 2022, Cologne, Germany, June 20-24, 2022 (accepted for publication).* ACM, 2022. DOI: 10.1145/3529372.3530922. URL: https://doi.org/10.1145/3529372.3530922.

[30] Arthur Brack, Anett Hoppe, and Ralph Ewerth. "Citation Recommendation for Research Papers via Knowledge Graphs". In: *Linking Theory and Practice of Digital Libraries - 25th International Conference on Theory and Practice of Digital Libraries, TPDL 2021, Virtual Event, September 13-17, 2021, Proceedings.* Ed. by Gerd Berget, Mark Michael Hall, Daniel Brenn, and Sanna Kumpulainen. Vol. 12866. Lecture Notes in Computer Science. Springer, 2021, pp. 165–174. DOI: 10.1007/978-3-030-86324-1_20. URL: https://doi.org/10.1007/978-3-030-86324-1_20.

[31] Arthur Brack, Anett Hoppe, Markus Stocker, Sören Auer, and Ralph Ewerth. "Requirements Analysis for an Open Research Knowledge Graph". In: *Digital Libraries for Open Knowledge - 24th International Conference on Theory and Practice of Digital Libraries, TPDL 2020, Lyon, France, August 25-27, 2020, Proceedings.* Ed. by Mark M. Hall, Tanja Mercun, Thomas Risse, and Fabien Duchateau. Vol. 12246. Lecture Notes in Computer Science. Springer, 2020, pp. 3–18. DOI: 10.1007/978-3-030-54956-5_1. URL: https://doi.org/10.1007/978-3-030-54956-5_1.

[32] Arthur Brack, Anett Hoppe, Markus Stocker, Sören Auer, and Ralph Ewerth. "Analysing the requirements for an Open Research Knowledge Graph: use cases, quality requirements, and construction strategies". In: *Int. J. Digit. Libr.* 23.1 (2022), pp. 33–55. DOI: 10.1007/s00799-021-00306-x. URL: https://doi.org/10.1007/s00799-021-00306-x.

[33] Arthur Brack, Daniel Uwe Müller, Anett Hoppe, and Ralph Ewerth. "Coreference Resolution in Research Papers from Multiple Domains". In: *Advances in Information Retrieval - 43rd European Conference on IR Research, ECIR 2021, Virtual Event, March 28 - April 1, 2021, Proceedings, Part I.* Ed. by Djoerd Hiemstra, Marie-Francine Moens, Josiane Mothe, Raffaele Perego, Martin Potthast, and Fabrizio Sebastiani. Vol. 12656. Lecture Notes in Computer Science. Springer, 2021, pp. 79–97. DOI: 10.1007/978-3-030-72113-8_6. URL: https://doi.org/10.1007/978-3-030-72113-8_6.

[34] Amy Brand, Liz Allen, Micah Altman, Marjorie M. K. Hlava, and Jo Scott. "Beyond authorship: attribution, contribution, collaboration, and credit". In: *Learn. Publ.* 28.2 (2015), pp. 151–155. DOI: 10.1087/20150211. URL: https://doi.org/10.1087/20150211.

[35] Richard Braun, Martin Benedict, Hannes Wendler, and Werner Esswein. "Proposal for Requirements Driven Design Science Research". In: *New Horizons in Design Science: Broadening the Research Agenda - 10th International Conference, DESRIST 2015, Dublin, Ireland, May 20-22, 2015, Proceedings.* Ed. by Brian Donnellan, Markus Helfert, Jim Kenneally, Debra E. VanderMeer, Marcus A. Rothenberger, and Robert Winter. Vol. 9073. Lecture Notes in Computer Science. Springer, 2015, pp. 135–151.

DOI: 10.1007/978-3-319-18714-3_9. URL: https://doi.org/10.1007/978-3-319-18714-3_9.

[36] Boyan Brodaric, Femke Reitsma, and Yi Qiang. "SKIing with DOLCE: toward an e-Science Knowledge Infrastructure". In: *Formal Ontology in Information Systems, Proceedings of the Fifth International Conference, FOIS 2008, Saarbrücken, Germany, October 31st - November 3rd, 2008*. Ed. by Carola Eschenbach and Michael Grüninger. Vol. 183. Frontiers in Artificial Intelligence and Applications. IOS Press, 2008, pp. 208–219. DOI: 10.3233/978-1-58603-923-3-208. URL: https://doi.org/10.3233/978-1-58603-923-3-208.

[37] Tom B. Brown et al. "Language Models are Few-Shot Learners". In: *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*. Ed. by Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin. 2020. URL: https://proceedings.neurips.cc/paper/2020/hash/1457c0d6bfcb4967418bfb8ac142f64a-Abstract.html.

[38] Adrian Burton, Amir Aryani, Hylke Koers, Paolo Manghi, Sandro La Bruzzo, Markus Stocker, Michael Diepenbroek, Uwe Schindler, and Martin Fenner. "The Scholix Framework for Interoperability in Data-Literature Information Exchange". In: *D Lib Mag.* 23.1/2 (2017). DOI: 10.1045/january2017-burton. URL: https://doi.org/10.1045/january2017-burton.

[39] CB Insights. *The Data Flywheel: How Enlightened Self-Interest Drives Data Network Effects.* https://www.cbinsights.com/research/team-blog/data-network-effects/. Accessed: 2020-11-10.

[40] Avi Caciularu, Arman Cohan, Iz Beltagy, Matthew E. Peters, Arie Cattan, and Ido Dagan. "CDLM: Cross-Document Language Modeling". In: *Findings of the Association for Computational Linguistics: EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 16-20 November, 2021*. Ed. by Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih. Association for Computational Linguistics, 2021, pp. 2648–2662. URL: https://aclanthology.org/2021.findings-emnlp.225.

[41] Andrew Carlson, Justin Betteridge, Bryan Kisiel, Burr Settles, Estevam R. Hruschka Jr., and Tom M. Mitchell. "Toward an Architecture for Never-Ending Language Learning". In: *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2010, Atlanta, Georgia, USA, July 11-15, 2010*. Ed. by Maria Fox and David Poole. AAAI Press, 2010. URL: http://www.aaai.org/ocs/index.php/AAAI/AAAI10/paper/view/1879.

[42] Panot Chaimongkol, Akiko Aizawa, and Yuka Tateisi. "Corpus for Coreference Resolution on Scientific Papers". In: *Proceedings of the Ninth International Conference on Language Resources and Evaluation, LREC 2014, Reykjavik, Iceland, May 26-31, 2014*. Ed. by Nicoletta Calzolari, Khalid Choukri, Thierry Declerck, Hrafn Loftsson, Bente Maegaard, Joseph Mariani, Asunción Moreno, Jan Odijk, and Stelios Piperidis. European Language Resources Association (ELRA), 2014, pp. 3187–3190. URL: http://www.lrec-conf.org/proceedings/lrec2014/summaries/286.html.

[43] America Chambers. "Statistical Models for Text Classification and Clustering: Applications and Analysis". PhD thesis. UNIVERSITY OF CALIFORNIA, IRVINE, 2013.

[44] Branden Chan, Stefan Schweter, and Timo Möller. "German's Next Language Model". In: *Proceedings of the 28th International Conference on Computational Linguistics, COLING 2020, Barcelona, Spain (Online), December 8-13, 2020*. Ed. by Donia Scott, Núria Bel, and Chengqing Zong. International Committee on Computational Linguistics, 2020, pp. 6788–6796. DOI: `10.18653/v1/2020.coling-main.598`. URL: `https://doi.org/10.18653/v1/2020.coling-main.598`.

[45] Wei-Cheng Chang, Felix X. Yu, Yin-Wen Chang, Yiming Yang, and Sanjiv Kumar. "Pre-training Tasks for Embedding-based Large-scale Retrieval". In: *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020. URL: `https://openreview.net/forum?id=rkg-mA4FDr`.

[46] Soravit Changpinyo, Hexiang Hu, and Fei Sha. "Multi-Task Learning for Sequence Tagging: An Empirical Study". In: *Proceedings of the 27th International Conference on Computational Linguistics, COLING 2018, Santa Fe, New Mexico, USA, August 20-26, 2018*. Ed. by Emily M. Bender, Leon Derczynski, and Pierre Isabelle. Association for Computational Linguistics, 2018, pp. 2965–2977. URL: `https://www.aclweb.org/anthology/C18-1251/`.

[47] Chuming Chen, Karen E. Ross, Sachin Gavali, Julie E. Cowart, and Cathy H. Wu. "COVID-19 Knowledge Graph from semantic integration of biomedical literature and databases". In: *Bioinform.* 37.23 (2021), pp. 4597–4598. DOI: `10.1093/bioinformatics/btab694`. URL: `https://doi.org/10.1093/bioinformatics/btab694`.

[48] Lin Chen, Lixin Duan, and Dong Xu. "Event Recognition in Videos by Learning from Heterogeneous Web Sources". In: *CVPR*. IEEE Computer Society, 2013, pp. 2666–2673.

[49] Kyunghyun Cho, Bart van Merrienboer, Çaglar Gülçehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. "Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation". In: *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*. Ed. by Alessandro Moschitti, Bo Pang, and Walter Daelemans. ACL, 2014, pp. 1724–1734. DOI: `10.3115/v1/d14-1179`. URL: `https://doi.org/10.3115/v1/d14-1179`.

[50] Kyunghyun Cho, Bart van Merrienboer, Çaglar Gülçehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. "Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation". In: *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*. Ed. by Alessandro Moschitti, Bo Pang, and Walter Daelemans. ACL, 2014, pp. 1724–1734. DOI: `10.3115/v1/d14-1179`. URL: `https://doi.org/10.3115/v1/d14-1179`.

[51] Kevin Clark and Christopher D. Manning. “Entity-Centric Coreference Resolution with Model Stacking”. In: *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL 2015, July 26-31, 2015, Beijing, China, Volume 1: Long Papers.* The Association for Computer Linguistics, 2015, pp. 1405–1415. DOI: `10.3115/v1/p15-1136`. URL: `https://doi.org/10.3115/v1/p15-1136`.

[52] Alistair Cockburn. *Writing Effective Use Cases.* 1st. USA: Addison-Wesley Longman Publishing Co., Inc., 2000. ISBN: 0201702258.

[53] Arman Cohan, Waleed Ammar, Madeleine van Zuylen, and Field Cady. “Structural Scaffolds for Citation Intent Classification in Scientific Publications”. In: *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).* Ed. by Jill Burstein, Christy Doran, and Thamar Solorio. Association for Computational Linguistics, 2019, pp. 3586–3596. DOI: `10.18653/v1/n19-1361`. URL: `https://doi.org/10.18653/v1/n19-1361`.

[54] Arman Cohan, Iz Beltagy, Daniel King, Bhavana Dalvi, and Daniel S. Weld. “Pretrained Language Models for Sequential Sentence Classification”. In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019.* Ed. by Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan. Association for Computational Linguistics, 2019, pp. 3691–3697. DOI: `10.18653/v1/D19-1383`. URL: `https://doi.org/10.18653/v1/D19-1383`.

[55] Arman Cohan, Franck Dernoncourt, Doo Soon Kim, Trung Bui, Seokhwan Kim, Walter Chang, and Nazli Goharian. “A Discourse-Aware Attention Model for Abstractive Summarization of Long Documents”. In: *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers).* New Orleans, Louisiana: Association for Computational Linguistics, June 2018, pp. 615–621. DOI: `10.18653/v1/N18-2097`. URL: `https://www.aclweb.org/anthology/N18-2097`.

[56] Arman Cohan, Sergey Feldman, Iz Beltagy, Doug Downey, and Daniel S. Weld. “SPECTER: Document-level Representation Learning using Citation-informed Transformers”. In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020.* Ed. by Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel R. Tetreault. Association for Computational Linguistics, 2020, pp. 2270–2282. DOI: `10.18653/v1/2020.acl-main.207`. URL: `https://doi.org/10.18653/v1/2020.acl-main.207`.

[57] Jacob Cohen. “A Coefficient of Agreement for Nominal Scales”. In: *Educational and Psychological Measurement* 20.1 (1960), pp. 37–46. DOI: `10.1177/001316446002000104`. eprint: `https://doi.org/10.1177/001316446002000104`. URL: `https://doi.org/10.1177/001316446002000104`.

[58] K. Bretonnel Cohen, Arrick Lanfranchi, Miji Joo-young Choi, Michael Bada, William A. Baumgartner Jr., Natalya Panteleyeva, Karin Verspoor, Martha Palmer, and Lawrence E. Hunter. "Coreference annotation and resolution in the Colorado Richly Annotated Full Text (CRAFT) corpus of biomedical journal articles". In: *BMC Bioinform.* 18.1 (2017), 372:1–372:14. DOI: 10.1186/s12859-017-1775-9. URL: https://doi.org/10.1186/s12859-017-1775-9.

[59] The Gene Ontology Consortium. "The Gene Ontology Resource: 20 years and still GOing strong". In: *Nucleic Acids Res.* 47.Database-Issue (2019), pp. D330–D338. DOI: 10.1093/nar/gky1055. URL: https://doi.org/10.1093/nar/gky1055.

[60] Alexandru Constantin, Silvio Peroni, Steve Pettifer, David M. Shotton, and Fabio Vitali. "The Document Components Ontology (DoCO)". In: *Semantic Web* 7.2 (2016), pp. 167–181. DOI: 10.3233/SW-150177. URL: https://doi.org/10.3233/SW-150177.

[61] Jennifer D'Souza, Anett Hoppe, Arthur Brack, Mohamad Yaser Jaradeh, Sören Auer, and Ralph Ewerth. "The STEM-ECR Dataset: Grounding Scientific Entity References in STEM Scholarly Content to Authoritative Encyclopedic and Lexicographic Sources". In: *Proceedings of The 12th Language Resources and Evaluation Conference, LREC 2020, Marseille, France, May 11-16, 2020.* Ed. by Nicoletta Calzolari, Frédéric Béchet, Philippe Blache, Khalid Choukri, Christopher Cieri, Thierry Declerck, Sara Goggi, Hitoshi Isahara, Bente Maegaard, Joseph Mariani, Hélène Mazo, Asunción Moreno, Jan Odijk, and Stelios Piperidis. European Language Resources Association, 2020, pp. 2192–2203. URL: https://www.aclweb.org/anthology/2020.lrec-1.268/.

[62] Carmen Dayrell, Arnaldo Candido Jr., Gabriel Lima, Danilo Machado Jr., Ann A. Copestake, Valéria Delisandra Feltrim, Stella E. O. Tagnin, and Sandra M. Aluísio. "Rhetorical Move Detection in English Abstracts: Multi-label Sentence Classifiers and their Annotated Corpora". In: *Proceedings of the Eighth International Conference on Language Resources and Evaluation, LREC 2012, Istanbul, Turkey, May 23-25, 2012.* Ed. by Nicoletta Calzolari, Khalid Choukri, Thierry Declerck, Mehmet Ugur Dogan, Bente Maegaard, Joseph Mariani, Jan Odijk, and Stelios Piperidis. European Language Resources Association (ELRA), 2012, pp. 1604–1609. URL: http://www.lrec-conf.org/proceedings/lrec2012/summaries/734.html.

[63] Jay DeYoung, Iz Beltagy, Madeleine van Zuylen, Bailey Kuehl, and Lucy Lu Wang. "MS\ˆ2: Multi-Document Summarization of Medical Studies". In: *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021.* Ed. by Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih. Association for Computational Linguistics, 2021, pp. 7494–7513. URL: https://aclanthology.org/2021.emnlp-main.594.

[64] Auriol Degbelo. "A Snapshot of Ontology Evaluation Criteria and Strategies". In: *Proceedings of the 13th International Conference on Semantic Systems, SEMANTICS 2017, Amsterdam, The Netherlands, September 11-14, 2017.* Ed. by Rinke Hoekstra, Catherine Faron-Zucker, Tassilo Pellegrini, and Victor de Boer. ACM, 2017, pp. 1–8. DOI: 10.1145/3132218.3132219. URL: https://doi.org/10.1145/3132218.3132219.

[65] Kirill Degtyarenko, Paula de Matos, Marcus Ennis, Janna Hastings, Martin Zbinden, Alan McNaught, Rafael Alcántara, Michael Darsow, Mickaël Guedj, and Michael Ashburner. "ChEBI: a database and ontology for chemical entities of biological interest". In: vol. 36. Database-Issue. 2008, pp. 344–350. DOI: `10.1093/nar/gkm791`. URL: `https://doi.org/10.1093/nar/gkm791`.

[66] Pascal Denis and Jason Baldridge. "Specialized Models and Ranking for Coreference Resolution". In: *2008 Conference on Empirical Methods in Natural Language Processing, EMNLP 2008, Proceedings of the Conference, 25-27 October 2008, Honolulu, Hawaii, USA, A meeting of SIGDAT, a Special Interest Group of the ACL*. ACL, 2008, pp. 660–669. URL: `https://www.aclweb.org/anthology/D08-1069/`.

[67] Pascal Denis and Jason Baldridge. "Global joint models for coreference resolution and named entity classification". In: *Proces. del Leng. Natural* 42 (2009). URL: `http://journal.sepln.org/sepln/ojs/ojs/index.php/pln/article/view/2806/1305`.

[68] Franck Dernoncourt and Ji Young Lee. "PubMed 200k RCT: a Dataset for Sequential Sentence Classification in Medical Abstracts". In: *Proceedings of the Eighth International Joint Conference on Natural Language Processing, IJCNLP 2017, Taipei, Taiwan, November 27 - December 1, 2017, Volume 2: Short Papers*. Ed. by Greg Kondrak and Taro Watanabe. Asian Federation of Natural Language Processing, 2017, pp. 308–313. URL: `https://www.aclweb.org/anthology/I17-2052/`.

[69] Franck Dernoncourt, Ji Young Lee, and Peter Szolovits. "Neural Networks for Joint Sentence Classification in Medical Paper Abstracts". In: *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*. Ed. by Mirella Lapata, Phil Blunsom, and Alexander Koller. Association for Computational Linguistics, 2017, pp. 694–700. DOI: `10.18653/v1/e17-2110`. URL: `https://doi.org/10.18653/v1/e17-2110`.

[70] Danilo Dessì, Francesco Osborne, Diego Reforgiato Recupero, Davide Buscaldi, Enrico Motta, and Harald Sack. "AI-KG: An Automatically Generated Knowledge Graph of Artificial Intelligence". In: *The Semantic Web - ISWC 2020 - 19th International Semantic Web Conference, Athens, Greece, November 2-6, 2020, Proceedings, Part II*. Ed. by Jeff Z. Pan, Valentina A. M. Tamma, Claudia d'Amato, Krzysztof Janowicz, Bo Fu, Axel Polleres, Oshani Seneviratne, and Lalana Kagal. Vol. 12507. Lecture Notes in Computer Science. Springer, 2020, pp. 127–143. DOI: `10.1007/978-3-030-62466-8_9`. URL: `https://doi.org/10.1007/978-3-030-62466-8_9`.

[71] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding". In: *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)*. Ed. by Jill Burstein, Christy Doran, and Thamar Solorio. Association for Computational Linguistics, 2019, pp. 4171–4186. DOI: `10.18653/v1/n19-1423`. URL: `https://doi.org/10.18653/v1/n19-1423`.

[72] George R. Doddington, Alexis Mitchell, Mark A. Przybocki, Lance A. Ramshaw, Stephanie M. Strassel, and Ralph M. Weischedel. "The Automatic Content Extraction (ACE) Program - Tasks, Data, and Evaluation". In: *Proceedings of the Fourth International Conference on Language Resources and Evaluation, LREC 2004, May*

26-28, 2004, Lisbon, Portugal. European Language Resources Association, 2004. URL: `http://www.lrec-conf.org/proceedings/lrec2004/summaries/5.htm`.

[73] Martin Doerr, Athina Kritsotaki, Yannis Rousakis, Gerald Hiebel, and Maria Theodoridou. *Definition of the CRMsci: An Extension of CIDOC-CRM to support scientific observation.* Tech. rep. Version 1.2.8. FORTH, Feb. 2020. URL: `http://www.cidoc-crm.org/crmsci/ModelVersion/version-1.2.8`.

[74] Rezarta Islamaj Dogan, Robert Leaman, and Zhiyong Lu. "NCBI disease corpus: A resource for disease name recognition and concept normalization". In: *J. Biomed. Informatics* 47 (2014), pp. 1–10. DOI: `10.1016/j.jbi.2013.12.006`. URL: `https://doi.org/10.1016/j.jbi.2013.12.006`.

[75] Xin Dong, Evgeniy Gabrilovich, Geremy Heitz, Wilko Horn, Ni Lao, Kevin Murphy, Thomas Strohmann, Shaohua Sun, and Wei Zhang. "Knowledge vault: a web-scale approach to probabilistic knowledge fusion". In: *The 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '14, New York, NY, USA - August 24 - 27, 2014.* Ed. by Sofus A. Macskassy, Claudia Perlich, Jure Leskovec, Wei Wang, and Rayid Ghani. ACM, 2014, pp. 601–610. DOI: `10.1145/2623330.2623623`. URL: `https://doi.org/10.1145/2623330.2623623`.

[76] Aleksandra Edwards, José Camacho-Collados, Hélène de Ribaupierre, and Alun D. Preece. "Go Simple and Pre-Train on Domain-Specific Corpora: On the Role of Training Data for Text Classification". In: *Proceedings of the 28th International Conference on Computational Linguistics, COLING 2020, Barcelona, Spain (Online), December 8-13, 2020.* Ed. by Donia Scott, Núria Bel, and Chengqing Zong. International Committee on Computational Linguistics, 2020, pp. 5522–5529. DOI: `10.18653/v1/2020.coling-main.481`. URL: `https://doi.org/10.18653/v1/2020.coling-main.481`.

[77] Elsevier Labs. *Elsevier OA STM Corpus.* `https://github.com/elsevierlabs/OA-STM-Corpus`. Accessed: 2020-07-15. 2017.

[78] Michael Färber. "The Microsoft Academic Knowledge Graph: A Linked Data Source with 8 Billion Triples of Scholarly Data". In: *The Semantic Web - ISWC 2019 - 18th International Semantic Web Conference, Auckland, New Zealand, October 26-30, 2019, Proceedings, Part II.* Ed. by Chiara Ghidini, Olaf Hartig, Maria Maleshkova, Vojtech Svátek, Isabel F. Cruz, Aidan Hogan, Jie Song, Maxime Lefrançois, and Fabien Gandon. Vol. 11779. Lecture Notes in Computer Science. Springer, 2019, pp. 113–129. DOI: `10.1007/978-3-030-30796-7_8`. URL: `https://doi.org/10.1007/978-3-030-30796-7_8`.

[79] Michael Färber, Frederic Bartscherer, Carsten Menne, and Achim Rettinger. "Linked data quality of DBpedia, Freebase, OpenCyc, Wikidata, and YAGO". In: *Semantic Web* 9.1 (2018), pp. 77–129. DOI: `10.3233/SW-170275`. URL: `https://doi.org/10.3233/SW-170275`.

[80] Michael Färber and Adam Jatowt. "Citation recommendation: approaches and datasets". In: *Int. J. Digit. Libr.* 21.4 (2020), pp. 375–405. DOI: `10.1007/s00799-020-00288-2`. URL: `https://doi.org/10.1007/s00799-020-00288-2`.

[81] Said Fathalla, Sahar Vahdati, Sören Auer, and Christoph Lange. "Towards a Knowledge Graph Representing Research Findings by Semantifying Survey Articles". In: *Research and Advanced Technology for Digital Libraries - 21st International Conference on Theory and Practice of Digital Libraries, TPDL 2017, Thessaloniki, Greece,*

September 18-21, 2017, Proceedings. Ed. by Jaap Kamps, Giannis Tsakonas, Yannis Manolopoulos, Lazaros S. Iliadis, and Ioannis Karydis. Vol. 10450. Lecture Notes in Computer Science. Springer, 2017, pp. 315–327. DOI: 10.1007/978-3-319-67008-9_25. URL: https://doi.org/10.1007/978-3-319-67008-9_25.

[82] Christiane Fellbaum, ed. *WordNet: An Electronic Lexical Database*. Language, Speech, and Communication. Cambridge, MA: MIT Press, 1998. ISBN: 978-0-262-06197-1.

[83] David A. Ferrucci, Eric W. Brown, Jennifer Chu-Carroll, James Fan, David Gondek, Aditya Kalyanpur, Adam Lally, J. William Murdock, Eric Nyberg, John M. Prager, Nico Schlaefer, and Christopher A. Welty. “Building Watson: An Overview of the DeepQA Project”. In: *AI Mag.* 31.3 (2010), pp. 59–79. DOI: 10.1609/aimag.v31i3.2303. URL: https://doi.org/10.1609/aimag.v31i3.2303.

[84] A. Fink. *Conducting Research Literature Reviews: From the Internet to Paper*. SAGE Publications, 2014. ISBN: 9781452259499.

[85] Beatríz Fisas, Horacio Saggion, and Francesco Ronzano. “On the Discoursive Structure of Computer Graphics Research Papers”. In: *Proceedings of The 9th Linguistic Annotation Workshop, LAW@NAACL-HLT 2015, June 5, 2015, Denver, Colorado, USA*. Ed. by Adam Meyers, Ines Rehbein, and Heike Zinsmeister. The Association for Computer Linguistics, 2015, pp. 42–51. DOI: 10.3115/v1/w15-1605. URL: https://doi.org/10.3115/v1/w15-1605.

[86] J.L. Fleiss et al. “Measuring nominal scale agreement among many raters”. In: *Psychological Bulletin* 76.5 (1971), pp. 378–382.

[87] Achille Fokoue, Ian Horrocks, Bernardo Cuenca Grau, Zhe Wu, and Boris Motik. *OWL 2 Web Ontology Language Profiles (Second Edition)*. W3C Recommendation. https://www.w3.org/TR/2012/REC-owl2-profiles-20121211/. W3C, Dec. 2012.

[88] G. D. Forney. “The viterbi algorithm”. In: *Proceedings of the IEEE* 61.3 (1973), pp. 268–278. DOI: 10.1109/PROC.1973.9030.

[89] Markus Freitag and Yaser Al-Onaizan. “Beam Search Strategies for Neural Machine Translation”. In: *Proceedings of the First Workshop on Neural Machine Translation, NMT@ACL 2017, Vancouver, Canada, August 4, 2017*. Ed. by Thang Luong, Alexandra Birch, Graham Neubig, and Andrew M. Finch. Association for Computational Linguistics, 2017, pp. 56–60. DOI: 10.18653/v1/w17-3207. URL: https://doi.org/10.18653/v1/w17-3207.

[90] Annemarie Friedrich, Heike Adel, Federico Tomazic, Johannes Hingerl, Renou Benteau, Anika Marusczyk, and Lukas Lange. “The SOFC-Exp Corpus and Neural Approaches to Information Extraction in the Materials Science Domain”. In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*. Ed. by Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel R. Tetreault. Association for Computational Linguistics, 2020, pp. 1255–1268. DOI: 10.18653/v1/2020.acl-main.116. URL: https://doi.org/10.18653/v1/2020.acl-main.116.

[91] Michael Färber and David Lamprecht. “The Data Set Knowledge Graph: Creating a Linked Open Data Source for Data Sets”. In: *Quantitative Science Studies* (Nov. 2021), pp. 1–30. ISSN: 2641-3337. DOI: 10.1162/qss_a_00161. eprint: https://direct.mit.edu/qss/article-pdf/doi/10.1162/qss_a_00161/1971237/qss_a_00161.pdf. URL: https://doi.org/10.1162/qss_a_00161.

[92] Kata Gábor, Davide Buscaldi, Anne-Kathrin Schumann, Behrang QasemiZadeh, Haïfa Zargayouna, and Thierry Charnois. “SemEval-2018 Task 7: Semantic Relation Extraction and Classification in Scientific Papers”. In: *Proceedings of The 12th International Workshop on Semantic Evaluation, SemEval@NAACL-HLT 2018, New Orleans, Louisiana, USA, June 5-6, 2018.* Ed. by Marianna Apidianaki, Saif M. Mohammad, Jonathan May, Ekaterina Shutova, Steven Bethard, and Marine Carpuat. Association for Computational Linguistics, 2018, pp. 679–688. DOI: 10.18653/v1/s18-1111. URL: https://doi.org/10.18653/v1/s18-1111.

[93] Luis Antonio Galárraga, Christina Teflioudi, Katja Hose, and Fabian M. Suchanek. “AMIE: association rule mining under incomplete evidence in ontological knowledge bases”. In: *22nd International World Wide Web Conference, WWW ’13, Rio de Janeiro, Brazil, May 13-17, 2013.* Ed. by Daniel Schwabe, Virgílio A. F. Almeida, Hartmut Glaser, Ricardo Baeza-Yates, and Sue B. Moon. International World Wide Web Conferences Steering Committee / ACM, 2013, pp. 413–422. DOI: 10.1145/2488388.2488425. URL: https://doi.org/10.1145/2488388.2488425.

[94] Luis Galárraga, Simon Razniewski, Antoine Amarilli, and Fabian M. Suchanek. “Predicting Completeness in Knowledge Bases”. In: *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining, WSDM 2017, Cambridge, United Kingdom, February 6-10, 2017.* Ed. by Maarten de Rijke, Milad Shokouhi, Andrew Tomkins, and Min Zhang. ACM, 2017, pp. 375–383. DOI: 10.1145/3018661.3018739. URL: https://doi.org/10.1145/3018661.3018739.

[95] Tianyu Gao, Adam Fisch, and Danqi Chen. “Making Pre-trained Language Models Better Few-shot Learners”. In: *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021.* Ed. by Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli. Association for Computational Linguistics, 2021, pp. 3816–3830. DOI: 10.18653/v1/2021.acl-long.295. URL: https://doi.org/10.18653/v1/2021.acl-long.295.

[96] Matt Gardner, Joel Grus, Mark Neumann, Oyvind Tafjord, Pradeep Dasigi, Nelson F. Liu, Matthew Peters, Michael Schmitz, and Luke Zettlemoyer. “AllenNLP: A Deep Semantic Natural Language Processing Platform”. In: (July 2018), pp. 1–6. DOI: 10.18653/v1/W18-2501. URL: https://www.aclweb.org/anthology/W18-2501.

[97] “Gene Ontology Consortium: The Gene Ontology (GO) database and informatics resource”. In: *Nucleic Acids Res.* 32.Database-Issue (2004), pp. 258–261. DOI: 10.1093/nar/gkh036. URL: https://doi.org/10.1093/nar/gkh036.

[98] Sayar Ghosh Roy, Nikhil Pinnaparaju, Risubh Jain, Manish Gupta, and Vasudeva Varma. “Summaformers @ LaySumm 20, LongSumm 20”. In: *Proceedings of the First Workshop on Scholarly Document Processing.* Online: Association for Computational Linguistics, Nov. 2020, pp. 336–343. DOI: 10.18653/v1/2020.sdp-1.39. URL: https://www.aclweb.org/anthology/2020.sdp-1.39.

[99] Martin Glinz, Hans van Loenhoud, Stefan Staal, and Stan Bühne. *Handbook for the CPRE Foundation Level according to the IREB Standard (1.0.0). Education and Training for Certified Professional for Requirements Engineering (CPRE) Foundation Level.* Tech. rep. International Requirements Engineering Board, Nov. 2020. URL: ht

tps://www.ireb.org/content/downloads/5-cpre-foundation-level-handbook/cpre_foundationlevel_handbook_en_v1.0.pdf.

[100] Sérgio Gonçalves, Paulo Cortez, and Sérgio Moro. "A deep learning classifier for sentence classification in biomedical and computer science abstracts". In: *Neural Comput. Appl.* 32.11 (2020), pp. 6793–6807. DOI: 10.1007/s00521-019-04334-2. URL: https://doi.org/10.1007/s00521-019-04334-2.

[101] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning.* http://www.deeplearningbook.org. MIT Press, 2016.

[102] *Google Scholar.* https://scholar.google.com/. Accessed: 2019-09-12.

[103] Tudor Groza, Siegfried Handschuh, Knud Möller, and Stefan Decker. "SALT - Semantically Annotated LaTeX for Scientific Publications". In: *The Semantic Web: Research and Applications, 4th European Semantic Web Conference, ESWC 2007, Innsbruck, Austria, June 3-7, 2007, Proceedings.* Ed. by Enrico Franconi, Michael Kifer, and Wolfgang May. Vol. 4519. Lecture Notes in Computer Science. Springer, 2007, pp. 518–532. DOI: 10.1007/978-3-540-72667-8_37. URL: https://doi.org/10.1007/978-3-540-72667-8_37.

[104] Ramanathan Guha and Dan Brickley. *RDF Schema 1.1.* W3C Recommendation. https://www.w3.org/TR/2014/REC-rdf-schema-20140225/. W3C, Feb. 2014.

[105] Scikit Learn User Guide. *Metrics and scoring: quantifying the quality of predictions.* Accessed: 2021-08-10. 2021. URL: https://scikit-learn.org/stable/modules/model_evaluation.html.

[106] Komal Gupta, Ammaar Ahmad, Tirthankar Ghosal, and Asif Ekbal. "ContriSci: A BERT-Based Multitasking Deep Neural Architecture to Identify Contribution Statements from Research Papers". In: *Towards Open and Trustworthy Digital Societies - 23rd International Conference on Asia-Pacific Digital Libraries, ICADL 2021, Virtual Event, December 1-3, 2021, Proceedings.* Ed. by Hao-Ren Ke, Chei Sian Lee, and Kazunari Sugiyama. Vol. 13133. Lecture Notes in Computer Science. Springer, 2021, pp. 436–452. DOI: 10.1007/978-3-030-91669-5_34. URL: https://doi.org/10.1007/978-3-030-91669-5_34.

[107] William L. Hamilton, Zhitao Ying, and Jure Leskovec. "Inductive Representation Learning on Large Graphs". In: *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA.* Ed. by Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett. 2017, pp. 1024–1034. URL: https://proceedings.neurips.cc/paper/2017/hash/5dd9db5e033da9c6fb5ba83c7a7ebea9-Abstract.html.

[108] Steven Harris and Andy Seaborne. *SPARQL 1.1 Query Language.* W3C Recommendation. https://www.w3.org/TR/2013/REC-sparql11-query-20130321/. W3C, Mar. 2013.

[109] Zellig Harris. "Distributional structure". In: *Word* 10.23 (1954), pp. 146–162.

[110] Alexander Hars. "Structure of scientific knowledge". In: *From Publishing to Knowledge Networks: Reinventing Online Knowledge Infrastructures.* Berlin, Heidelberg: Springer Berlin Heidelberg, 2003, pp. 83–185. ISBN: 978-3-540-24737-1. DOI: 10.1007/978-3-540-24737-1_3. URL: https://doi.org/10.1007/978-3-540-24737-1_3.

[111] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. “Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification”. In: *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*. IEEE Computer Society, 2015, pp. 1026–1034. DOI: 10.1109/ICCV.2015.123. URL: https://doi.org/10.1109/ICCV.2015.123.

[112] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. “Deep Residual Learning for Image Recognition”. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*. IEEE Computer Society, 2016, pp. 770–778. DOI: 10.1109/CVPR.2016.90. URL: https://doi.org/10.1109/CVPR.2016.90.

[113] Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. “Deberta: decoding-Enhanced Bert with Disentangled Attention”. In: *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. URL: https://openreview.net/forum?id=XPZIaotutsD.

[114] Qi He, Bee-Chung Chen, and Deepak Agarwal. *Building The LinkedIn Knowledge Graph*. In: *LinkedIn Blog*. Accessed: 2021-06-28. 2016. URL: https://engineering.linkedin.com/blog/2016/10/building-the-linkedin-knowledge-graph.

[115] Alan R. Hevner, Salvatore T. March, Jinsoo Park, and Sudha Ram. “Design Science in Information Systems Research”. In: *MIS Q.* 28.1 (2004), pp. 75–105. URL: http://misq.org/design-science-in-information-systems-research.html.

[116] Sepp Hochreiter and Jürgen Schmidhuber. “Long Short-Term Memory”. In: *Neural Comput.* 9.8 (1997), pp. 1735–1780. DOI: 10.1162/neco.1997.9.8.1735. URL: https://doi.org/10.1162/neco.1997.9.8.1735.

[117] Aidan Hogan, Eva Blomqvist, Michael Cochez, Claudia d’Amato, Gerard de Melo, Claudio Gutiérrez, Sabrina Kirrane, José Emilio Labra Gayo, Roberto Navigli, Sebastian Neumaier, Axel-Cyrille Ngonga Ngomo, Axel Polleres, Sabbir M. Rashid, Anisa Rula, Lukas Schmelzeisen, Juan F. Sequeda, Steffen Staab, and Antoine Zimmermann. “Knowledge Graphs”. In: *ACM Comput. Surv.* 54.4 (2021), 71:1–71:37. DOI: 10.1145/3447772. URL: https://doi.org/10.1145/3447772.

[118] Gordana Ilic Holen. “Critical Reflections on Evaluation Practices in Coreference Resolution”. In: *Human Language Technologies: Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings, June 9-14, 2013, Westin Peachtree Plaza Hotel, Atlanta, Georgia, USA*. Ed. by Lucy Vanderwende, Hal Daumé III, and Katrin Kirchhoff. The Association for Computational Linguistics, 2013, pp. 1–7. URL: https://aclanthology.org/N13-2001/.

[119] Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. *spaCy: Industrial-strength Natural Language Processing in Python*. 2020. DOI: 10.5281/zenodo.1212303. URL: https://doi.org/10.5281/zenodo.1212303.

[120] Anett Hoppe, Jascha Hagen, Helge Holzmann, Günter Kniesel, and Ralph Ewerth. “An Analytics Tool for Exploring Scientific Software and Related Publications”. In: *Digital Libraries for Open Knowledge, 22nd International Conference on Theory and Practice of Digital Libraries, TPDL 2018, Porto, Portugal, September 10-13, 2018, Proceedings*. Ed. by Eva Méndez, Fabio Crestani, Cristina Ribeiro, Gabriel David, and João Correia Lopes. Vol. 11057. Lecture Notes in Computer Science. Springer,

2018, pp. 299–303. DOI: 10.1007/978-3-030-00066-0_27. URL: https://doi.org/10.1007/978-3-030-00066-0_27.

[121] I Horvath. "Comparison of three methodological approaches of design research". Undefined/Unknown. In: *Proceedings of the 16th International Conference on Engineering Design, ICED'07*. Ed. by S.n. null ; Conference date: 28-08-2007 Through 30-08-2007. Ecole Central Paris, 2007, pp. 1–11. ISBN: 1-904670-02-4.

[122] Yufang Hou, Charles Jochim, Martin Gleize, Francesca Bonin, and Debasis Ganguly. "Identification of Tasks, Datasets, Evaluation Metrics, and Numeric Scores for Scientific Leaderboards Construction". In: *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*. Ed. by Anna Korhonen, David R. Traum, and Lluís Màrquez. Association for Computational Linguistics, 2019, pp. 5203–5213. DOI: 10.18653/v1/p19-1513. URL: https://doi.org/10.18653/v1/p19-1513.

[123] Neil Houlsby, Ferenc Huszar, Zoubin Ghahramani, and Máté Lengyel. "Bayesian Active Learning for Classification and Preference Learning". In: *CoRR* abs/1112.5745 (2011). arXiv: 1112.5745. URL: http://arxiv.org/abs/1112.5745.

[124] Jeremy Howard and Sebastian Ruder. "Universal Language Model Fine-tuning for Text Classification". In: *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*. Ed. by Iryna Gurevych and Yusuke Miyao. Association for Computational Linguistics, 2018, pp. 328–339. DOI: 10.18653/v1/P18-1031. URL: https://www.aclweb.org/anthology/P18-1031/.

[125] John Ikerd. *Rethinking the Economics of Self-Interests*. In: *University of Missouri*. Accessed: 2021-04-28. 1999. URL: http://web.missouri.edu/ikerdj/papers/Rethinking.html.

[126] Neo4J Inc. *Neo4j Graph Database*. Accessed: 2021-09-20. 2021. URL: https://neo4j.com/product/neo4j-graph-database/.

[127] Ontotext Inc. *GraphDB*. Accessed: 2021-09-20. 2021. URL: https://graphdb.ontotext.com/.

[128] Sergey Ioffe and Christian Szegedy. "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift". In: *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*. Ed. by Francis R. Bach and David M. Blei. Vol. 37. JMLR Workshop and Conference Proceedings. JMLR.org, 2015, pp. 448–456. URL: http://proceedings.mlr.press/v37/ioffe15.html.

[129] Sarthak Jain, Madeleine van Zuylen, Hannaneh Hajishirzi, and Iz Beltagy. "SciREX: A Challenge Dataset for Document-Level Information Extraction". In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*. Ed. by Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel R. Tetreault. Association for Computational Linguistics, 2020, pp. 7506–7516. DOI: 10.18653/v1/2020.acl-main.670. URL: https://doi.org/10.18653/v1/2020.acl-main.670.

[130] Mohamad Yaser Jaradeh, Allard Oelen, Manuel Prinz, Markus Stocker, and Sören Auer. "Open Research Knowledge Graph: A System Walkthrough". In: *Digital Libraries for Open Knowledge - 23rd International Conference on Theory and Practice of Digital Libraries, TPDL 2019, Oslo, Norway, September 9-12, 2019, Proceedings.* Ed. by Antoine Doucet, Antoine Isaac, Koraljka Golub, Trond Aalberg, and Adam Jatowt. Vol. 11799. Lecture Notes in Computer Science. Springer, 2019, pp. 348–351. DOI: 10.1007/978-3-030-30760-8_31. URL: https://doi.org/10.1007/978-3-030-30760-8_31.

[131] Robin Jia, Cliff Wong, and Hoifung Poon. "Document-Level N-ary Relation Extraction with Multiscale Representation Learning". In: *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).* Ed. by Jill Burstein, Christy Doran, and Thamar Solorio. Association for Computational Linguistics, 2019, pp. 3693–3704. DOI: 10.18653/v1/n19-1370. URL: https://doi.org/10.18653/v1/n19-1370.

[132] Jyun-Yu Jiang, Mingyang Zhang, Cheng Li, Michael Bendersky, Nadav Golbandi, and Marc Najork. "Semantic Text Matching for Long-Form Documents". In: *The World Wide Web Conference, WWW 2019, San Francisco, CA, USA, May 13-17, 2019.* Ed. by Ling Liu, Ryen W. White, Amin Mantrach, Fabrizio Silvestri, Julian J. McAuley, Ricardo Baeza-Yates, and Leila Zia. ACM, 2019, pp. 795–806. DOI: 10.1145/3308558.3313707. URL: https://doi.org/10.1145/3308558.3313707.

[133] Di Jin and Peter Szolovits. "Hierarchical Neural Networks for Sequential Sentence Classification in Medical Scientific Abstracts". In: *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018.* Ed. by Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii. Association for Computational Linguistics, 2018, pp. 3100–3109. DOI: 10.18653/v1/d18-1349. URL: https://doi.org/10.18653/v1/d18-1349.

[134] Jeff Johnson, Matthijs Douze, and Hervé Jégou. "Billion-Scale Similarity Search with GPUs". In: *IEEE Trans. Big Data* 7.3 (2021), pp. 535–547. DOI: 10.1109/TBDATA.2019.2921572. URL: https://doi.org/10.1109/TBDATA.2019.2921572.

[135] Ian T. Jolliffe. "Principal Component Analysis". In: *International Encyclopedia of Statistical Science.* Ed. by Miodrag Lovric. Springer, 2011, pp. 1094–1096. DOI: 10.1007/978-3-642-04898-2_455. URL: https://doi.org/10.1007/978-3-642-04898-2_455.

[136] Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S. Weld, Luke Zettlemoyer, and Omer Levy. "SpanBERT: Improving Pre-training by Representing and Predicting Spans". In: *Trans. Assoc. Comput. Linguistics* 8 (2020), pp. 64–77. URL: https://transacl.org/ojs/index.php/tacl/article/view/1853.

[137] Mandar Joshi, Omer Levy, Luke Zettlemoyer, and Daniel S. Weld. "BERT for Coreference Resolution: Baselines and Analysis". In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019.* Ed. by Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan. Association for Computational Linguistics, 2019, pp. 5802–5807. DOI: 10.18653/v1/D19-1588. URL: https://doi.org/10.18653/v1/D19-1588.

[138] Dan Jurafsky and James H. Martin. *Speech and language processing: an introduction to natural language processing, computational linguistics, and speech recognition, 3rd Edition (draft).* 2021. URL: `https://web.stanford.edu/~jurafsky/slp3/`.

[139] David Jurgens, Srijan Kumar, Raine Hoover, Daniel A. McFarland, and Dan Jurafsky. "Measuring the Evolution of a Scientific Field through Citation Frames". In: *Trans. Assoc. Comput. Linguistics* 6 (2018), pp. 391–406. URL: `https://transacl.org/ojs/index.php/tacl/article/view/1266`.

[140] Salomon Kabongo, Jennifer D'Souza, and Sören Auer. "Automated Mining of Leaderboards for Empirical AI Research". In: *Towards Open and Trustworthy Digital Societies - 23rd International Conference on Asia-Pacific Digital Libraries, ICADL 2021, Virtual Event, December 1-3, 2021, Proceedings.* Ed. by Hao-Ren Ke, Chei Sian Lee, and Kazunari Sugiyama. Vol. 13133. Lecture Notes in Computer Science. Springer, 2021, pp. 453–470. DOI: `10.1007/978-3-030-91669-5_35`. URL: `https://doi.org/10.1007/978-3-030-91669-5_35`.

[141] Amar Viswanathan Kannan, Dmitriy Fradkin, Ioannis Akrotirianakis, Tugba Kulahcioglu, Arquimedes Canedo, Aditi Roy, Shih-Yuan Yu, Arnav V. Malawade, and Mohammad Abdullah Al Faruque. "Multimodal Knowledge Graph for Deep Learning Papers and Code". In: *CIKM '20: The 29th ACM International Conference on Information and Knowledge Management, Virtual Event, Ireland, October 19-23, 2020.* Ed. by Mathieu d'Aquin, Stefan Dietze, Claudia Hauff, Edward Curry, and Philippe Cudré-Mauroux. ACM, 2020, pp. 3417–3420. DOI: `10.1145/3340531.3417439`. URL: `https://doi.org/10.1145/3340531.3417439`.

[142] Marcin Kardas, Piotr Czapla, Pontus Stenetorp, Sebastian Ruder, Sebastian Riedel, Ross Taylor, and Robert Stojnic. "AxCell: Automatic Extraction of Results from Machine Learning Papers". In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020.* Ed. by Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu. Association for Computational Linguistics, 2020, pp. 8580–8594. DOI: `10.18653/v1/2020.emnlp-main.692`. URL: `https://doi.org/10.18653/v1/2020.emnlp-main.692`.

[143] Andrej Karpathy. *The Unreasonable Effectiveness of Recurrent Neural Networks.* In: *Andrej Karpathy blog.* Accessed: 2021-06-28. 2015. URL: `http://karpathy.github.io/2015/05/21/rnn-effectiveness/`.

[144] M. Kejriwal, C.A. Knoblock, and P. Szekely. *Knowledge Graphs: Fundamentals, Techniques, and Applications.* Adaptive Computation and Machine Learning series. MIT Press, 2021. ISBN: 9780262361880. URL: `https://mitpress.mit.edu/books/knowledge-graphs`.

[145] Gregg Kellogg, Pierre-Antoine Champin, and Dave Longley. *JSON-LD 1.1.* W3C Recommendation. `https://www.w3.org/TR/2020/REC-json-ld11-20200716/`. W3C, July 2020.

[146] Jin-Dong Kim, Ngan L. T. Nguyen, Yue Wang, Jun'ichi Tsujii, Toshihisa Takagi, and Akinori Yonezawa. "The Genia Event and Protein Coreference tasks of the BioNLP Shared Task 2011". In: *BMC Bioinform.* 13.S-11 (2012), S1. DOI: `10.1186/1471-2105-13-S11-S1`. URL: `https://doi.org/10.1186/1471-2105-13-S11-S1`.

[147] Su Kim, David Martínez, Lawrence Cavedon, and Lars Yencken. "Automatic classification of sentences to support Evidence Based Medicine". In: *BMC Bioinform.* 12.S-2 (2011), S5. DOI: `10.1186/1471-2105-12-S2-S5`. URL: `https://doi.org/10.1186/1471-2105-12-S2-S5`.

[148] Diederik P. Kingma and Jimmy Ba. "Adam: A Method for Stochastic Optimization". In: *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*. Ed. by Yoshua Bengio and Yann LeCun. 2015. URL: `http://arxiv.org/abs/1412.6980`.

[149] Thomas N. Kipf and Max Welling. "Semi-Supervised Classification with Graph Convolutional Networks". In: *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017. URL: `https://openreview.net/forum?id=SJU4ayYgl`.

[150] Tibor Kiss and Jan Strunk. "Unsupervised Multilingual Sentence Boundary Detection". In: *Comput. Linguistics* 32.4 (2006), pp. 485–525. DOI: `10.1162/coli.2006.32.4.485`. URL: `https://doi.org/10.1162/coli.2006.32.4.485`.

[151] Barbara Ann Kitchenham and Stuart Charters. *Guidelines for performing Systematic Literature Reviews in Software Engineering*. English. Tech. rep. EBSE 2007-001. Keele University and Durham University Joint Report, July 2007. URL: `https://www.elsevier.com/__data/promis_misc/525444systematicreviewsguide.pdf`.

[152] Iraklis A. Klampanos, Athanasios Davvetas, Antonis Koukourikos, and Vangelis Karkaletsis. "ANNETT-O: an ontology for describing artificial neural network evaluation, topology and training". In: *Int. J. Metadata Semant. Ontologies* 13.3 (2019), pp. 179–190. DOI: `10.1504/IJMSO.2019.099833`. URL: `https://doi.org/10.1504/IJMSO.2019.099833`.

[153] Holger Knublauch and Dimitris Kontokostas. *Shapes Constraint Language (SHACL)*. W3C Recommendation. https://www.w3.org/TR/2017/REC-shacl-20170720/. W3C, July 2017.

[154] Nikolaos Kolitsas, Octavian-Eugen Ganea, and Thomas Hofmann. "End-to-End Neural Entity Linking". In: *Proceedings of the 22nd Conference on Computational Natural Language Learning, CoNLL 2018, Brussels, Belgium, October 31 - November 1, 2018*. Ed. by Anna Korhonen and Ivan Titov. Association for Computational Linguistics, 2018, pp. 519–529. DOI: `10.18653/v1/k18-1050`. URL: `https://doi.org/10.18653/v1/k18-1050`.

[155] Mateusz Kopec and Maciej Ogrodniczuk. "Inter-annotator Agreement in Coreference Annotation of Polish". In: *Advanced Approaches to Intelligent Information and Database Systems*. Ed. by Janusz Sobecki, Veera Boonjing, and Suphamit Chittayasothorn. Cham: Springer International Publishing, 2014, pp. 149–158. ISBN: 978-3-319-05503-9. DOI: `10.1007/978-3-319-05503-9_15`.

[156] Jens Kringelum, Sonny Kim Kjærulff, Søren Brunak, Ole Lund, Tudor I. Oprea, and Olivier Taboureau. "ChemProt-3.0: a global chemical biology diseases mapping". In: *Database J. Biol. Databases Curation* 2016 (2016). DOI: `10.1093/database/bav123`. URL: `https://doi.org/10.1093/database/bav123`.

[157] Klaus Krippendorff. "Reliability in Content Analysis". In: *Human Communication Research* 30.3 (2004), pp. 411–433. DOI: `https://doi.org/10.1111/j.1468-2958.2004.tb00738.x`.

[158] Arun Krishnan. *Making search easier: How Amazon's Product Graph is helping customers find products more easily.* In: *Amazon Blog.* Accessed: 2021-06-28. 2018. URL: `https://www.aboutamazon.com/news/innovation-at-amazon/making-search-easier`.

[159] Markus Krötzsch, Pascal Hitzler, Bijan Parsia, Peter Patel-Schneider, and Sebastian Rudolph. *OWL 2 Web Ontology Language Primer (Second Edition).* W3C Recommendation. `https://www.w3.org/TR/2012/REC-owl2-primer-20121211/`. W3C, Dec. 2012.

[160] Taku Kudo. "Subword Regularization: Improving Neural Network Translation Models with Multiple Subword Candidates". In: *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers.* Ed. by Iryna Gurevych and Yusuke Miyao. Association for Computational Linguistics, 2018, pp. 66–75. DOI: `10.18653/v1/P18-1007`. URL: `https://aclanthology.org/P18-1007/`.

[161] Suchetha Nambanoor Kunnath, David Pride, Bikash Gyawali, and Petr Knoth. "Overview of the 2020 WOSP 3C Citation Context Classification Task". In: *Proceedings of the 8th International Workshop on Mining Scientific Publications.* Wuhan, China: Association for Computational Linguistics, May 2020, pp. 75–83. URL: `https://www.aclweb.org/anthology/2020.wosp-1.12`.

[162] "Mean Average Precision". In: *Encyclopedia of Database Systems.* Ed. by LING LIU and M. TAMER ÖZSU. Boston, MA: Springer US, 2009, pp. 1703–1703. ISBN: 978-0-387-39940-9. DOI: `10.1007/978-0-387-39940-9_3032`. URL: `https://doi.org/10.1007/978-0-387-39940-9_3032`.

[163] John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. "Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data". In: *Proceedings of the Eighteenth International Conference on Machine Learning (ICML 2001), Williams College, Williamstown, MA, USA, June 28 - July 1, 2001.* Ed. by Carla E. Brodley and Andrea Pohoreckyj Danyluk. Morgan Kaufmann, 2001, pp. 282–289.

[164] J. Richard Landis and Gary G. Koch. "The Measurement of Observer Agreement for Categorical Data". In: *Biometrics* 33.1 (1977).

[165] Christoph Lange. "Ontologies and languages for representing mathematical knowledge on the Semantic Web". In: *Semantic Web* 4.2 (2013), pp. 119–158. DOI: `10.3233/SW-2012-0059`. URL: `https://doi.org/10.3233/SW-2012-0059`.

[166] Ni Lao and William W. Cohen. "Relational retrieval using a combination of path-constrained random walks". In: *Mach. Learn.* 81.1 (2010), pp. 53–67. DOI: `10.1007/s10994-010-5205-8`. URL: `https://doi.org/10.1007/s10994-010-5205-8`.

[167] Anne Lauscher, Goran Glavas, and Kai Eckert. "ArguminSci: A Tool for Analyzing Argumentation and Rhetorical Aspects in Scientific Writing". In: *Proceedings of the 5th Workshop on Argument Mining, ArgMining@EMNLP 2018, Brussels, Belgium, November 1, 2018.* Ed. by Noam Slonim and Ranit Aharonov. Association for Computational Linguistics, 2018, pp. 22–28. DOI: `10.18653/v1/w18-5203`. URL: `https://doi.org/10.18653/v1/w18-5203`.

[168] Anne Lauscher, Goran Glavas, Simone Paolo Ponzetto, and Kai Eckert. "Investigating the Role of Argumentation in the Rhetorical Analysis of Scientific Publications with Neural Multi-Task Learning Models". In: *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*. Ed. by Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii. Association for Computational Linguistics, 2018, pp. 3326–3338. DOI: 10.18653/v1/d18-1370. URL: https://doi.org/10.18653/v1/d18-1370.

[169] Ji Young Lee, Franck Dernoncourt, and Peter Szolovits. "Transfer Learning for Named-Entity Recognition with Neural Networks". In: *Proceedings of the Eleventh International Conference on Language Resources and Evaluation, LREC 2018, Miyazaki, Japan, May 7-12, 2018*. Ed. by Nicoletta Calzolari, Khalid Choukri, Christopher Cieri, Thierry Declerck, Sara Goggi, Kôiti Hasida, Hitoshi Isahara, Bente Maegaard, Joseph Mariani, Hélène Mazo, Asunción Moreno, Jan Odijk, Stelios Piperidis, and Takenobu Tokunaga. European Language Resources Association (ELRA), 2018. URL: http://www.lrec-conf.org/proceedings/lrec2018/summaries/878.html.

[170] Kenton Lee, Luheng He, Mike Lewis, and Luke Zettlemoyer. "End-to-end Neural Coreference Resolution". In: *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017*. Ed. by Martha Palmer, Rebecca Hwa, and Sebastian Riedel. Association for Computational Linguistics, 2017, pp. 188–197. DOI: 10.18653/v1/d17-1018. URL: https://doi.org/10.18653/v1/d17-1018.

[171] Kenton Lee, Luheng He, and Luke Zettlemoyer. "Higher-Order Coreference Resolution with Coarse-to-Fine Inference". In: *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 2 (Short Papers)*. Ed. by Marilyn A. Walker, Heng Ji, and Amanda Stent. Association for Computational Linguistics, 2018, pp. 687–692. DOI: 10.18653/v1/n18-2108. URL: https://doi.org/10.18653/v1/n18-2108.

[172] Jens Lehmann, Robert Isele, Max Jakob, Anja Jentzsch, Dimitris Kontokostas, Pablo N. Mendes, Sebastian Hellmann, Mohamed Morsey, Patrick van Kleef, Sören Auer, and Christian Bizer. "DBpedia - A large-scale, multilingual knowledge base extracted from Wikipedia". In: *Semantic Web* 6.2 (2015), pp. 167–195. DOI: 10.3233/SW-140134. URL: https://doi.org/10.3233/SW-140134.

[173] Jiao Li, Yueping Sun, Robin J. Johnson, Daniela Sciaky, Chih-Hsuan Wei, Robert Leaman, Allan Peter Davis, Carolyn J. Mattingly, Thomas C. Wiegers, and Zhiyong Lu. "BioCreative V CDR task corpus: a resource for chemical disease relation extraction". In: *Database J. Biol. Databases Curation* 2016 (2016). DOI: 10.1093/database/baw068. URL: https://doi.org/10.1093/database/baw068.

[174] Maria Liakata, Shyamasree Saha, Simon Dobnik, Colin R. Batchelor, and Dietrich Rebholz-Schuhmann. "Automatic recognition of conceptualization zones in scientific articles and two life science applications". In: *Bioinform.* 28.7 (2012), pp. 991–1000. DOI: 10.1093/bioinformatics/bts071. URL: https://doi.org/10.1093/bioinformatics/bts071.

[175] Maria Liakata, Simone Teufel, Advaith Siddharthan, and Colin R. Batchelor. "Corpora for the Conceptualisation and Zoning of Scientific Papers". In: *Proceedings of the International Conference on Language Resources and Evaluation, LREC 2010, 17-23 May 2010, Valletta, Malta.* Ed. by Nicoletta Calzolari, Khalid Choukri, Bente Maegaard, Joseph Mariani, Jan Odijk, Stelios Piperidis, Mike Rosner, and Daniel Tapias. European Language Resources Association, 2010. URL: `http://www.lrec-conf.org/proceedings/lrec2010/summaries/644.html`.

[176] Andrew Lih. *The Wikipedia revolution : how a bunch of nobodies created the world's greatest encyclopedia.* London New York: Aurum Hyperion, 2009. ISBN: 978-1-84513-473-0.

[177] Kyle Lo, Lucy Lu Wang, Mark Neumann, Rodney Kinney, and Daniel S. Weld. "S2ORC: The Semantic Scholar Open Research Corpus". In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020.* Ed. by Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel R. Tetreault. Association for Computational Linguistics, 2020, pp. 4969–4983. DOI: `10.18653/v1/2020.acl-main.447`. URL: `https://doi.org/10.18653/v1/2020.acl-main.447`.

[178] Yi Luan, Luheng He, Mari Ostendorf, and Hannaneh Hajishirzi. "Multi-Task Identification of Entities, Relations, and Coreference for Scientific Knowledge Graph Construction". In: *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018.* Ed. by Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii. Association for Computational Linguistics, 2018, pp. 3219–3232. DOI: `10.18653/v1/d18-1360`. URL: `https://doi.org/10.18653/v1/d18-1360`.

[179] Mohamed Lubani, Shahrul Azman Mohd. Noah, and Rohana Mahmud. "Ontology population: Approaches and design aspects". In: *J. Inf. Sci.* 45.4 (2019). DOI: `10.1177/0165551518801819`. URL: `https://doi.org/10.1177/0165551518801819`.

[180] Xiaoqiang Luo. "On Coreference Resolution Performance Metrics". In: *HLT/EMNLP 2005, Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference, 6-8 October 2005, Vancouver, British Columbia, Canada.* The Association for Computational Linguistics, 2005, pp. 25–32. URL: `https://www.aclweb.org/anthology/H05-1004/`.

[181] Xiaoqiang Luo and Sameer Pradhan. "Evaluation Metrics". In: *Anaphora Resolution - Algorithms, Resources, and Applications.* Ed. by Massimo Poesio, Roland Stuckardt, and Yannick Versley. Theory and Applications of Natural Language Processing. Springer, 2016, pp. 141–163. DOI: `10.1007/978-3-662-47909-4_5`. URL: `https://doi.org/10.1007/978-3-662-47909-4_5`.

[182] Jie Ma, Jun Liu, Yufei Li, Xin Hu, Yudai Pan, Shen Sun, and Qika Lin. "Jointly Optimized Neural Coreference Resolution with Mutual Attention". In: *WSDM '20: The Thirteenth ACM International Conference on Web Search and Data Mining, Houston, TX, USA, February 3-7, 2020.* Ed. by James Caverlee, Xia (Ben) Hu, Mounia Lalmas, and Wei Wang. ACM, 2020, pp. 402–410. DOI: `10.1145/3336191.3371787`. URL: `https://doi.org/10.1145/3336191.3371787`.

[183] Xuezhe Ma and Eduard H. Hovy. “End-to-end Sequence Labeling via Bi-directional LSTM-CNNs-CRF”. In: *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers.* The Association for Computer Linguistics, 2016. DOI: 10.18653/v1/p16-1101. URL: https://doi.org/10.18653/v1/p16-1101.

[184] Paolo Manghi, Alessia Bardi, Claudio Atzori, Miriam Baglioni, Natalia Manola, Jochen Schirrwagen, and Pedro Principe. *The OpenAIRE Research Graph Data Model.* Version 1.3. Apr. 2019. DOI: 10.5281/zenodo.2643199. URL: https://doi.org/10.5281/zenodo.2643199.

[185] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. *Introduction to Information Retrieval.* Cambridge, UK: Cambridge University Press, 2008. ISBN: 978-0-521-86571-5.

[186] Christopher D. Manning and Hinrich Schütze. *Foundations of Statistical Natural Language Processing.* Cambridge, Massachusetts: The MIT Press, 1999. URL: http://nlp.stanford.edu/fsnlp/.

[187] Ana Marasovic, Leo Born, Juri Opitz, and Anette Frank. “A Mention-Ranking Model for Abstract Anaphora Resolution”. In: *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017.* Ed. by Martha Palmer, Rebecca Hwa, and Sebastian Riedel. Association for Computational Linguistics, 2017, pp. 221–232. DOI: 10.18653/v1/d17-1021. URL: https://doi.org/10.18653/v1/d17-1021.

[188] Niall McCarthy. *The Countries Leading The World In Scientific Publications.* In: *Statista.* Accessed: 2021-06-18. 2019. URL: https://www.statista.com/chart/20347/science-and-engineering-articles-published/.

[189] Chris McCormick. *Word2Vec Tutorial - The Skip-Gram Model.* Accessed: 2021-09-20. 2016. URL: https://mccormickml.com/2016/04/19/word2vec-tutorial-the-skip-gram-model/.

[190] Warren McCulloch and Walter Pitts. “A Logical Calculus of Ideas Immanent in Nervous Activity”. In: *Bulletin of Mathematical Biophysics* 5 (1943), pp. 115–133.

[191] Sepideh Mesbah, Kyriakos Fragkeskos, Christoph Lofi, Alessandro Bozzon, and Geert-Jan Houben. “Semantic Annotation of Data Processing Pipelines in Scientific Publications”. In: *The Semantic Web - 14th International Conference, ESWC 2017, Portorož, Slovenia, May 28 - June 1, 2017, Proceedings, Part I.* Ed. by Eva Blomqvist, Diana Maynard, Aldo Gangemi, Rinke Hoekstra, Pascal Hitzler, and Olaf Hartig. Vol. 10249. Lecture Notes in Computer Science. 2017, pp. 321–336. DOI: 10.1007/978-3-319-58068-5_20. URL: https://doi.org/10.1007/978-3-319-58068-5_20.

[192] *Microsoft Academic.* https://academic.microsoft.com/home. Accessed: 2019-09-12.

[193] Tomás Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. “Distributed Representations of Words and Phrases and their Compositionality”. In: *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States.* Ed. by Christopher J. C. Burges, Léon Bottou, Zoubin Ghahramani, and Kilian Q. Weinberger. 2013, pp. 3111–3119. URL:

https://proceedings.neurips.cc/paper/2013/hash/9aa42b31882ec039965f3c4923ce901b-Abstract.html.

[194] Nafise Sadat Moosavi and Michael Strube. "Which Coreference Evaluation Metric Do You Trust? A Proposal for a Link-based Entity Aware Metric". In: *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Berlin, Germany: Association for Computational Linguistics, Aug. 2016, pp. 632–642. DOI: 10.18653/v1/P16-1060. URL: https://aclanthology.org/P16-1060.

[195] Andrea Moro, Francesco Cecconi, and Roberto Navigli. "Multilingual Word Sense Disambiguation and Entity Linking for Everybody". In: *Proceedings of the ISWC 2014 Posters & Demonstrations Track a track within the 13th International Semantic Web Conference, ISWC 2014, Riva del Garda, Italy, October 21, 2014*. Ed. by Matthew Horridge, Marco Rospocher, and Jacco van Ossenbruggen. Vol. 1272. CEUR Workshop Proceedings. CEUR-WS.org, 2014, pp. 25–28. URL: http://ceur-ws.org/Vol-1272/paper_30.pdf.

[196] Lili Mou, Zhao Meng, Rui Yan, Ge Li, Yan Xu, Lu Zhang, and Zhi Jin. "How Transferable are Neural Networks in NLP Applications?" In: *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*. Ed. by Jian Su, Xavier Carreras, and Kevin Duh. The Association for Computational Linguistics, 2016, pp. 479–489. DOI: 10.18653/v1/d16-1046. URL: https://doi.org/10.18653/v1/d16-1046.

[197] Zara Nasar, Syed Waqar Jaffry, and Muhammad Kamran Malik. "Information extraction from scientific articles: a survey". In: *Scientometrics* 117.3 (2018), pp. 1931–1990. DOI: 10.1007/s11192-018-2921-5. URL: https://doi.org/10.1007/s11192-018-2921-5.

[198] Mariana L. Neves, Daniel Butzke, and Barbara Grune. "Evaluation of Scientific Elements for Text Similarity in Biomedical Publications". In: *Proceedings of the 6th Workshop on Argument Mining, ArgMining@ACL 2019, Florence, Italy, August 1, 2019*. Ed. by Benno Stein and Henning Wachsmuth. Association for Computational Linguistics, 2019, pp. 124–135. DOI: 10.18653/v1/w19-4515. URL: https://doi.org/10.18653/v1/w19-4515.

[199] Vincent Ng. "Machine Learning for Entity Coreference Resolution: A Retrospective Look at Two Decades of Research". In: *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA*. Ed. by Satinder P. Singh and Shaul Markovitch. AAAI Press, 2017, pp. 4877–4884. URL: http://aaai.org/ocs/index.php/AAAI/AAAI17/paper/view/14995.

[200] Vincent Ng and Claire Cardie. "Identifying Anaphoric and Non-Anaphoric Noun Phrases to Improve Coreference Resolution". In: *19th International Conference on Computational Linguistics, COLING 2002, Howard International House and Academia Sinica, Taipei, Taiwan, August 24 - September 1, 2002*. 2002. URL: https://www.aclweb.org/anthology/C02-1139/.

[201] Ngan L. T. Nguyen, Jin-Dong Kim, Makoto Miwa, Takuya Matsuzaki, and Junichi Tsujii. "Improving protein coreference resolution by simple semantic classification". In: *BMC Bioinform.* 13 (2012), p. 304. DOI: 10.1186/1471-2105-13-304. URL: https://doi.org/10.1186/1471-2105-13-304.

[202] Maximilian Nickel, Kevin Murphy, Volker Tresp, and Evgeniy Gabrilovich. "A Review of Relational Machine Learning for Knowledge Graphs". In: *Proc. IEEE* 104.1 (2016), pp. 11–33. DOI: 10.1109/JPROC.2015.2483592. URL: https://doi.org/10.1109/JPROC.2015.2483592.

[203] Allard Oelen, Mohamad Yaser Jaradeh, Markus Stocker, and Sören Auer. "Generate FAIR Literature Surveys with Scholarly Knowledge Graphs". In: *JCDL '20: Proceedings of the ACM/IEEE Joint Conference on Digital Libraries in 2020, Virtual Event, China, August 1-5, 2020.* Ed. by Ruhua Huang, Dan Wu, Gary Marchionini, Daqing He, Sally Jo Cunningham, and Preben Hansen. ACM, 2020, pp. 97–106. DOI: 10.1145/3383583.3398520. URL: https://doi.org/10.1145/3383583.3398520.

[204] Chitu Okoli. "A Guide to Conducting a Standalone Systematic Literature Review". In: *Commun. Assoc. Inf. Syst.* 37 (2015), p. 43. URL: http://aisel.aisnet.org/cais/vol37/iss1/43.

[205] Sinno Jialin Pan and Qiang Yang. "A Survey on Transfer Learning". In: *IEEE Trans. Knowl. Data Eng.* 22.10 (2010), pp. 1345–1359. DOI: 10.1109/TKDE.2009.191. URL: https://doi.org/10.1109/TKDE.2009.191.

[206] *Papers With Code.* https://paperswithcode.com/. Accessed: 2021-04-10.

[207] Seoyeon Park and Cornelia Caragea. "Scientific Keyphrase Identification and Classification by Pre-Trained Language Models Intermediate Task Transfer Learning". In: *Proceedings of the 28th International Conference on Computational Linguistics, COLING 2020, Barcelona, Spain (Online), December 8-13, 2020.* Ed. by Donia Scott, Núria Bel, and Chengqing Zong. International Committee on Computational Linguistics, 2020, pp. 5409–5419. DOI: 10.18653/v1/2020.coling-main.472. URL: https://doi.org/10.18653/v1/2020.coling-main.472.

[208] Linda Partridge. "Celebrating 350 years of Philosophical Transactions: life sciences papers". eng. In: *Philosophical transactions of the Royal Society of London. Series B, Biological sciences* 370.1666 (Apr. 2015), p. 20140380. ISSN: 1471-2970. DOI: 10.1098/rstb.2014.0380. URL: https://doi.org/10.1098/rstb.2014.0380.

[209] Razvan Pascanu, Tomás Mikolov, and Yoshua Bengio. "On the difficulty of training recurrent neural networks". In: *Proceedings of the 30th International Conference on Machine Learning, ICML 2013, Atlanta, GA, USA, 16-21 June 2013.* Vol. 28. JMLR Workshop and Conference Proceedings. JMLR.org, 2013, pp. 1310–1318. URL: http://proceedings.mlr.press/v28/pascanu13.html.

[210] Adam Paszke et al. "PyTorch: An Imperative Style, High-Performance Deep Learning Library". In: *Advances in Neural Information Processing Systems 32.* Ed. by H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett. Curran Associates, Inc., 2019, pp. 8024–8035. URL: http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf.

[211] Yifan Peng, Shankai Yan, and Zhiyong Lu. "Transfer Learning in Biomedical Natural Language Processing: An Evaluation of BERT and ELMo on Ten Benchmarking Datasets". In: *Proceedings of the 18th BioNLP Workshop and Shared Task, BioNLP@ACL 2019, Florence, Italy, August 1, 2019.* Ed. by Dina Demner-Fushman, Kevin Bretonnel Cohen, Sophia Ananiadou, and Junichi Tsujii. Association for Computational

Linguistics, 2019, pp. 58–65. DOI: 10.18653/v1/w19-5006. URL: https://doi.org/10.18653/v1/w19-5006.

[212] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. "Glove: Global Vectors for Word Representation". In: *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*. Ed. by Alessandro Moschitti, Bo Pang, and Walter Daelemans. ACL, 2014, pp. 1532–1543. DOI: 10.3115/v1/d14-1162. URL: https://doi.org/10.3115/v1/d14-1162.

[213] J.A. Perez-Ortiz and M.L. Forcada. "Part-of-speech tagging with recurrent neural networks". In: *IJCNN'01. International Joint Conference on Neural Networks. Proceedings (Cat. No.01CH37222)*. Vol. 3. 2001, 1588–1592 vol.3. DOI: 10.1109/IJCNN.2001.938396.

[214] Silvio Peroni and David M. Shotton. "FaBiO and CiTO: Ontologies for describing bibliographic resources and citations". In: *J. Web Semant.* 17 (2012), pp. 33–43. DOI: 10.1016/j.websem.2012.08.001. URL: https://doi.org/10.1016/j.websem.2012.08.001.

[215] Vayianos Pertsas and Panos Constantopoulos. "Scholarly Ontology: modelling scholarly practices". In: *Int. J. Digit. Libr.* 18.3 (2017), pp. 173–190. DOI: 10.1007/s00799-016-0169-3. URL: https://doi.org/10.1007/s00799-016-0169-3.

[216] Georgios Petasis, Vangelis Karkaletsis, Georgios Paliouras, Anastasia Krithara, and Elias Zavitsanos. "Ontology Population and Enrichment: State of the Art". In: *Knowledge-Driven Multimedia Information Extraction and Ontology Evolution - Bridging the Semantic Gap*. Ed. by Georgios Paliouras, Constantine D. Spyropoulos, and George Tsatsaronis. Vol. 6050. Lecture Notes in Computer Science. Springer, 2011, pp. 134–166. DOI: 10.1007/978-3-642-20795-2_6. URL: https://doi.org/10.1007/978-3-642-20795-2_6.

[217] Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. "Deep Contextualized Word Representations". In: *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers)*. Ed. by Marilyn A. Walker, Heng Ji, and Amanda Stent. Association for Computational Linguistics, 2018, pp. 2227–2237. DOI: 10.18653/v1/n18-1202. URL: https://doi.org/10.18653/v1/n18-1202.

[218] Joelle Pineau, Philippe Vincent-Lamarre, Koustuv Sinha, Vincent Larivière, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox, and Hugo Larochelle. "Improving Reproducibility in Machine Learning Research (A Report from the NeurIPS 2019 Reproducibility Program)". In: *CoRR* abs/2003.12206 (2020). arXiv: 2003.12206. URL: https://arxiv.org/abs/2003.12206.

[219] Leo L. Pipino, Yang W. Lee, and Richard Y. Wang. "Data Quality Assessment". In: *Commun. ACM* 45.4 (Apr. 2002), pp. 211–218. ISSN: 0001-0782. DOI: 10.1145/505248.506010. URL: https://doi.org/10.1145/505248.506010.

[220] Martin F. Porter. "An algorithm for suffix stripping". In: *Program* 14.3 (1980), pp. 130–137. DOI: 10.1108/eb046814. URL: https://doi.org/10.1108/eb046814.

[221] Sameer Pradhan, Xiaoqiang Luo, Marta Recasens, Eduard H. Hovy, Vincent Ng, and Michael Strube. "Scoring Coreference Partitions of Predicted Mentions: A Reference Implementation". In: *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, ACL 2014, June 22-27, 2014, Baltimore, MD, USA, Volume 2: Short Papers.* The Association for Computer Linguistics, 2014, pp. 30–35. DOI: 10.3115/v1/p14-2006. URL: https://doi.org/10.3115/v1/p14-2006.

[222] Sameer Pradhan, Alessandro Moschitti, Nianwen Xue, Hwee Tou Ng, Anders Björkelund, Olga Uryupina, Yuchen Zhang, and Zhi Zhong. "Towards Robust Linguistic Analysis using OntoNotes". In: *Proceedings of the Seventeenth Conference on Computational Natural Language Learning, CoNLL 2013, Sofia, Bulgaria, August 8-9, 2013.* Ed. by Julia Hockenmaier and Sebastian Riedel. ACL, 2013, pp. 143–152. URL: https://www.aclweb.org/anthology/W13-3516/.

[223] Sameer Pradhan, Alessandro Moschitti, Nianwen Xue, Olga Uryupina, and Yuchen Zhang. "CoNLL-2012 Shared Task: Modeling Multilingual Unrestricted Coreference in OntoNotes". In: *Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning - Proceedings of the Shared Task: Modeling Multilingual Unrestricted Coreference in OntoNotes, EMNLP-CoNLL 2012, July 13, 2012, Jeju Island, Korea.* Ed. by Sameer Pradhan, Alessandro Moschitti, and Nianwen Xue. ACL, 2012, pp. 1–40. URL: https://www.aclweb.org/anthology/W12-4501/.

[224] Eric Prud'hommeaux and Gavin Carothers. *RDF 1.1 Turtle.* W3C Recommendation. https://www.w3.org/TR/2014/REC-turtle-20140225/. W3C, Feb. 2014.

[225] Yada Pruksachatkun, Jason Phang, Haokun Liu, Phu Mon Htut, Xiaoyi Zhang, Richard Yuanzhe Pang, Clara Vania, Katharina Kann, and Samuel R. Bowman. "Intermediate-Task Transfer Learning with Pretrained Language Models: When and Why Does It Work?" In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020.* Ed. by Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel R. Tetreault. Association for Computational Linguistics, 2020, pp. 5231–5247. DOI: 10.18653/v1/2020.acl-main.467. URL: https://doi.org/10.18653/v1/2020.acl-main.467.

[226] Jay Pujara and Sameer Singh. "Mining Knowledge Graphs From Text". In: *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining, WSDM 2018, Marina Del Rey, CA, USA, February 5-9, 2018.* Ed. by Yi Chang, Chengxiang Zhai, Yan Liu, and Yoelle Maarek. ACM, 2018, pp. 789–790. DOI: 10.1145/3159652.3162011. URL: https://doi.org/10.1145/3159652.3162011.

[227] J. Pustejovsky and A. Stubbs. *Natural Language Annotation for Machine Learning.* A Guide to corpus-building for applications Bd. 9,S. 878. O'Reilly Media, Incorporated, 2012. ISBN: 9781449306663.

[228] Kader Pustu-Iren, Markus Mühling, Nikolaus Korfhage, Joanna Bars, Sabrina Bernhöft, Angelika Hörth, Bernd Freisleben, and Ralph Ewerth. "Investigating Correlations of Inter-coder Agreement and Machine Annotation Performance for Historical Video Data". In: *Digital Libraries for Open Knowledge - 23rd International Conference on Theory and Practice of Digital Libraries, TPDL 2019, Oslo, Norway, September 9-12, 2019, Proceedings.* Ed. by Antoine Doucet, Antoine Isaac, Koraljka Golub, Trond Aalberg, and Adam Jatowt. Vol. 11799. Lecture Notes in Computer Science.

Springer, 2019, pp. 107–114. DOI: 10.1007/978-3-030-30760-8_9. URL: https://doi.org/10.1007/978-3-030-30760-8_9.

[229] Behrang Q. Zadeh and Siegfried Handschuh. "The ACL RD-TEC: A Dataset for Benchmarking Terminology Extraction and Classification in Computational Linguistics". In: *Proceedings of the 4th International Workshop on Computational Terminology (Computerm)*. Dublin, Ireland: Association for Computational Linguistics and Dublin City University, Aug. 2014, pp. 52–63. DOI: 10.3115/v1/W14-4807. URL: https://www.aclweb.org/anthology/W14-4807.

[230] Behrang QasemiZadeh and Anne-Kathrin Schumann. "The ACL RD-TEC 2.0: A Language Resource for Evaluating Term Extraction and Entity Recognition Methods". In: *Proceedings of the Tenth International Conference on Language Resources and Evaluation LREC 2016, Portorož, Slovenia, May 23-28, 2016*. Ed. by Nicoletta Calzolari, Khalid Choukri, Thierry Declerck, Sara Goggi, Marko Grobelnik, Bente Maegaard, Joseph Mariani, Hélène Mazo, Asunción Moreno, Jan Odijk, and Stelios Piperidis. European Language Resources Association (ELRA), 2016. URL: http://www.lrec-conf.org/proceedings/lrec2016/summaries/681.html.

[231] Md. Altaf ur Rahman and Vincent Ng. "Supervised Models for Coreference Resolution". In: *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing, EMNLP 2009, 6-7 August 2009, Singapore, A meeting of SIGDAT, a Special Interest Group of the ACL*. ACL, 2009, pp. 968–977. URL: https://www.aclweb.org/anthology/D09-1101/.

[232] Yves Raimond and Guus Schreiber. *RDF 1.1 Primer*. W3C Note. https://www.w3.org/TR/2014/NOTE-rdf11-primer-20140624/. W3C, June 2014.

[233] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. "SQuAD: 100, 000+ Questions for Machine Comprehension of Text". In: *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*. Ed. by Jian Su, Xavier Carreras, and Kevin Duh. The Association for Computational Linguistics, 2016, pp. 2383–2392. DOI: 10.18653/v1/d16-1264. URL: https://doi.org/10.18653/v1/d16-1264.

[234] Nils Reimers and Iryna Gurevych. "Reporting Score Distributions Makes a Difference: Performance Study of LSTM-networks for Sequence Tagging". In: *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017*. Ed. by Martha Palmer, Rebecca Hwa, and Sebastian Riedel. Association for Computational Linguistics, 2017, pp. 338–348. DOI: 10.18653/v1/d17-1035. URL: https://doi.org/10.18653/v1/d17-1035.

[235] Nils Reimers and Iryna Gurevych. "Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks". In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*. Ed. by Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan. Association for Computational Linguistics, 2019, pp. 3980–3990. DOI: 10.18653/v1/D19-1410. URL: https://doi.org/10.18653/v1/D19-1410.

[236] Scott Richardson, M. C. Wilson, J. Nishikawa, and R. S. Hayward. "The well-built clinical question: a key to evidence-based decisions." English (US). In: *ACP journal club* 123.3 (Nov. 1995), A12–13. ISSN: 1539-8560.

[237] Peter J. Rousseeuw. "Silhouettes: A graphical aid to the interpretation and validation of cluster analysis". In: *Journal of Computational and Applied Mathematics* 20 (1987), pp. 53 –65. ISSN: 0377-0427. DOI: https://doi.org/10.1016/0377-0427(87)90125-7. URL: http://www.sciencedirect.com/science/article/pii/0377042787901257.

[238] Sebastian Ruder. "Neural Transfer Learning for Natural Language Processing". PhD thesis. National University of Ireland, Galway, 2019.

[239] Almudena Ruiz-Iniesta and Óscar Corcho. "A review of ontologies for describing scholarly and scientific documents". In: *Proceedings of the 4th Workshop on Semantic Publishing co-located with the 11th Extended Semantic Web Conference (ESWC 2014), Anissaras, Greece, May 25th, 2014*. Ed. by Alexander García Castro, Christoph Lange, Phillip W. Lord, and Robert Stevens. Vol. 1155. CEUR Workshop Proceedings. CEUR-WS.org, 2014. URL: http://ceur-ws.org/Vol-1155/paper-07.pdf.

[240] David E. Rumelhart, Geoffrey E. Hinton, and Ronald J. Williams. "Learning representations by back-propagating errors". In: *Nature* 323.6088 (Oct. 1986), pp. 533–536. ISSN: 1476-4687. DOI: 10.1038/323533a0. URL: https://doi.org/10.1038/323533a0.

[241] Chris Rupp. *Requirements-Engineering und -Management: das Handbuch für Anforderungen in jeder Situation.* Vol. 7. Hanser Verlag München, 2021. ISBN: 978-3-446-45587-0.

[242] Iqra Safder and Saeed-Ul Hassan. "Bibliometric-enhanced information retrieval: a novel deep feature engineering approach for algorithm searching from full-text publications". In: *Scientometrics* 119.1 (2019), pp. 257–277. DOI: 10.1007/s11192-019-03025-y. URL: https://doi.org/10.1007/s11192-019-03025-y.

[243] Iqra Safder, Saeed-Ul Hassan, Anna Visvizi, Thanapon Noraset, Raheel Nawaz, and Suppawong Tuarob. "Deep Learning-based Extraction of Algorithmic Metadata in Full-Text Scholarly Documents". In: *Inf. Process. Manag.* 57.6 (2020), p. 102269. DOI: 10.1016/j.ipm.2020.102269. URL: https://doi.org/10.1016/j.ipm.2020.102269.

[244] Angelo A. Salatino, Thiviyan Thanapalasingam, Andrea Mannocci, Aliaksandr Birukou, Francesco Osborne, and Enrico Motta. "The Computer Science Ontology: A Comprehensive Automatically-Generated Taxonomy of Research Areas". In: *Data Intell.* 2.3 (2020), pp. 379–416. DOI: 10.1162/dint_a_00055. URL: https://doi.org/10.1162/dint_a_00055.

[245] Erik F. Tjong Kim Sang and Fien De Meulder. "Introduction to the CoNLL-2003 Shared Task: Language-Independent Named Entity Recognition". In: *CoNLL*. ACL, 2003, pp. 142–147.

[246] Victor Sanh, Thomas Wolf, and Sebastian Ruder. "A Hierarchical Multi-Task Approach for Learning Embeddings from Semantic Tasks". In: *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*. AAAI Press, 2019, pp. 6949–6956. DOI: 10.1609/aaai.v33i01.33016949. URL: https://doi.org/10.1609/aaai.v33i01.33016949.

[247] Aysegul Say, Said Fathalla, Sahar Vahdati, Jens Lehmann, and Sören Auer. "Semantic Representation of Physics Research Data". In: *Proceedings of the 12th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management, IC3K 2020, Volume 2: KEOD, Budapest, Hungary, November 2-4, 2020*. Ed. by David Aveiro, Jan L. G. Dietz, and Joaquim Filipe. SCITEPRESS, 2020, pp. 64–75. DOI: 10.5220/0010111000640075. URL: https://doi.org/10.5220/0010111000640075.

[248] Ulrich Schäfer, Christian Spurk, and Jörg Steffen. "A Fully Coreference-annotated Corpus of Scholarly Papers from the ACL Anthology". In: *COLING 2012, 24th International Conference on Computational Linguistics, Proceedings of the Conference: Posters, 8-15 December 2012, Mumbai, India*. Ed. by Martin Kay and Christian Boitet. Indian Institute of Technology Bombay, 2012, pp. 1059–1070. URL: https://www.aclweb.org/anthology/C12-2103/.

[249] Guus Schreiber and Fabien Gandon. *RDF 1.1 XML Syntax*. W3C Recommendation. https://www.w3.org/TR/2014/REC-rdf-syntax-grammar-20140225/. W3C, Feb. 2014.

[250] Claudia Schulz, Steffen Eger, Johannes Daxenberger, Tobias Kahse, and Iryna Gurevych. "Multi-Task Learning for Argumentation Mining in Low-Resource Settings". In: *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 2 (Short Papers)*. Ed. by Marilyn A. Walker, Heng Ji, and Amanda Stent. Association for Computational Linguistics, 2018, pp. 35–41. DOI: 10.18653/v1/n18-2006. URL: https://doi.org/10.18653/v1/n18-2006.

[251] Mike Schuster and Kaisuke Nakajima. "Japanese and Korean voice search". In: *2012 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2012, Kyoto, Japan, March 25-30, 2012*. IEEE, 2012, pp. 5149–5152. DOI: 10.1109/ICASSP.2012.6289079. URL: https://doi.org/10.1109/ICASSP.2012.6289079.

[252] *Semantic Scholar*. https://www.semanticscholar.org/. Accessed: 2019-09-12.

[253] Tushar Semwal, Promod Yenigalla, Gaurav Mathur, and Shivashankar B. Nair. "A Practitioners' Guide to Transfer Learning for Text Classification using Convolutional Neural Networks". In: *Proceedings of the 2018 SIAM International Conference on Data Mining, SDM 2018, May 3-5, 2018, San Diego Marriott Mission Valley, San Diego, CA, USA*. Ed. by Martin Ester and Dino Pedreschi. SIAM, 2018, pp. 513–521. DOI: 10.1137/1.9781611975321.58. URL: https://doi.org/10.1137/1.9781611975321.58.

[254] Rico Sennrich, Barry Haddow, and Alexandra Birch. "Neural Machine Translation of Rare Words with Subword Units". In: *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*. The Association for Computer Linguistics, 2016. DOI: 10.18653/v1/p16-1162. URL: https://doi.org/10.18653/v1/p16-1162.

[255] Burr Settles. *Active Learning Literature Survey*. Computer Sciences Technical Report 1648. University of Wisconsin–Madison, 2009.

[256] *Seventh Message Understanding Conference: Proceedings of a Conference Held in Fairfax, Virginia, USA, MUC 1998, April 29 - May 1, 1998.* ACL, 1998. URL: `https://www.aclweb.org/anthology/volumes/M98-1/`.

[257] Xichen Shang, Qianli Ma, Zhenxi Lin, Jiangyue Yan, and Zipeng Chen. “A Span-based Dynamic Local Attention Model for Sequential Sentence Classification”. In: *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 2: Short Papers), Virtual Event, August 1-6, 2021.* Ed. by Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli. Association for Computational Linguistics, 2021, pp. 198–203. DOI: `10.18653/v1/2021.acl-short.26`. URL: `https://doi.org/10.18653/v1/2021.acl-short.26`.

[258] Yanyao Shen, Hyokun Yun, Zachary Chase Lipton, Yakov Kronrod, and Animashree Anandkumar. “Deep Active Learning for Named Entity Recognition”. In: *Proceedings of the 2nd Workshop on Representation Learning for NLP, Rep4NLP@ACL 2017, Vancouver, Canada, August 3, 2017.* Ed. by Phil Blunsom, Antoine Bordes, Kyunghyun Cho, Shay B. Cohen, Chris Dyer, Edward Grefenstette, Karl Moritz Hermann, Laura Rimell, Jason Weston, and Scott Yih. Association for Computational Linguistics, 2017, pp. 252–256. DOI: `10.18653/v1/w17-2630`. URL: `https://doi.org/10.18653/v1/w17-2630`.

[259] Aditya Siddhant and Zachary C. Lipton. “Deep Bayesian Active Learning for Natural Language Processing: Results of a Large-Scale Empirical Study”. In: *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018.* Ed. by Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun’ichi Tsujii. Association for Computational Linguistics, 2018, pp. 2904–2909. DOI: `10.18653/v1/d18-1318`. URL: `https://doi.org/10.18653/v1/d18-1318`.

[260] Mayank Singh, Barnopriyo Barua, Priyank Palod, Manvi Garg, Sidhartha Satapathy, Samuel Bushi, Kumar Ayush, Krishna Sai Rohith, Tulasi Gamidi, Pawan Goyal, and Animesh Mukherjee. “OCR++: A Robust Framework For Information Extraction from Scholarly Articles”. In: *COLING 2016, 26th International Conference on Computational Linguistics, Proceedings of the Conference: Technical Papers, December 11-16, 2016, Osaka, Japan.* Ed. by Nicoletta Calzolari, Yuji Matsumoto, and Rashmi Prasad. ACL, 2016, pp. 3390–3400. URL: `https://www.aclweb.org/anthology/C16-1320/`.

[261] Amit Singhal. *Introducing the Knowledge Graph: things, not strings.* In: *Google Blog.* Accessed: 2021-06-28. 2012. URL: `https://www.blog.google/products/search/introducing-knowledge-graph-things-not/`.

[262] Barry Smith, Michael Ashburner, Cornelius Rosse, Jonathan Bard, William Bug, Werner Ceusters, Louis J. Goldberg, Karen Eilbeck, Amelia Ireland, Christopher J. Mungall, Neocles Leontis, Philippe Rocca-Serra, Alan Ruttenberg, Susanna-Assunta Sansone, Richard H. Scheuermann, Nigam Shah, Patricia L. Whetzel, Suzanna Lewis, and The OBI Consortium. “The OBO Foundry: coordinated evolution of ontologies to support biomedical data integration”. In: *Nature Biotechnology* 25.11 (Nov. 2007), pp. 1251–1255. ISSN: 1546-1696. DOI: `10.1038/nbt1346`. URL: `https://doi.org/10.1038/nbt1346`.

[263] Rion Snow, Brendan O'Connor, Daniel Jurafsky, and Andrew Y. Ng. "Cheap and Fast - But is it Good? Evaluating Non-Expert Annotations for Natural Language Tasks". In: *2008 Conference on Empirical Methods in Natural Language Processing, EMNLP 2008, Proceedings of the Conference, 25-27 October 2008, Honolulu, Hawaii, USA, A meeting of SIGDAT, a Special Interest Group of the ACL*. ACL, 2008, pp. 254–263. URL: https://www.aclweb.org/anthology/D08-1027/.

[264] Larisa N Soldatova and Ross D King. "An ontology of scientific experiments". In: *Journal of The Royal Society Interface* 3.11 (2006), pp. 795–803. DOI: 10.1098/rsif.2006.0134. eprint: https://royalsocietypublishing.org/doi/pdf/10.1098/rsif.2006.0134. URL: https://royalsocietypublishing.org/doi/abs/10.1098/rsif.2006.0134.

[265] Wee Meng Soon, Hwee Tou Ng, and Chung Yong Lim. "A Machine Learning Approach to Coreference Resolution of Noun Phrases". In: *Comput. Linguistics* 27.4 (2001), pp. 521–544. DOI: 10.1162/089120101753342653. URL: https://doi.org/10.1162/089120101753342653.

[266] Alexander Spangher, Jonathan May, Sz-Rung Shiang, and Lingjia Deng. "Multitask Semi-Supervised Learning for Class-Imbalanced Discourse Classification". In: *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*. Ed. by Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wentau Yih. Association for Computational Linguistics, 2021, pp. 498–517. URL: https://aclanthology.org/2021.emnlp-main.40.

[267] Michael Sperberg-McQueen, Eve Maler, Jean Paoli, Tim Bray, and François Yergeau. *Extensible Markup Language (XML) 1.0 (Fifth Edition)*. W3C Recommendation. https://www.w3.org/TR/2008/REC-xml-20081126/. W3C, Nov. 2008.

[268] *Springer Nature SciGraph*. https://www.springernature.com/gp/researchers/scigraph. Accessed: 2020-02-12.

[269] UNESCO Institute for Statistics. *Total global spending on research and development (R&D) from 1996 to 2018*. In: *Statista*. Accessed: 2021-06-18. Mar. 2021. URL: https://www.statista.com/statistics/1105959/total-research-and-development-spending-worldwide-ppp-usd/.

[270] Connor Stead, Stephen Smith, Peter A. Busch, and Savanid Vatanasakdakul. "Emerald 110k: A Multidisciplinary Dataset for Abstract Sentence Classification". In: *Proceedings of the The 17th Annual Workshop of the Australasian Language Technology Association, ALTA 2019, Sydney, Australia, December 4-6, 2019*. Ed. by Meladel Mistica, Massimo Piccardi, and Andrew MacKinlay. Australasian Language Technology Association, 2019, pp. 120–125. URL: https://aclweb.org/anthology/papers/U/U19/U19-1016/.

[271] Pontus Stenetorp, Sampo Pyysalo, Goran Topic, Tomoko Ohta, Sophia Ananiadou, and Jun'ichi Tsujii. "brat: a Web-based Tool for NLP-Assisted Text Annotation". In: *EACL 2012, 13th Conference of the European Chapter of the Association for Computational Linguistics, Avignon, France, April 23-27, 2012*. Ed. by Walter Daelemans, Mirella Lapata, and Lluís Màrquez. The Association for Computer Linguistics, 2012, pp. 102–107. URL: https://www.aclweb.org/anthology/E12-2021/.

[272] Markus Stocker, Manuel Prinz, Fatemeh Rostami, and Tibor Kempf. "Towards Research Infrastructures that Curate Scientific Information: A Use Case in Life Sciences". In: *Data Integration in the Life Sciences - 13th International Conference, DILS 2018, Hannover, Germany, November 20-21, 2018, Proceedings*. Ed. by Sören Auer and Maria-Esther Vidal. Vol. 11371. Lecture Notes in Computer Science. Springer, 2018, pp. 61–74. DOI: 10.1007/978-3-030-06016-9_6. URL: https://doi.org/10.1007/978-3-030-06016-9_6.

[273] Xuefeng Su, Ru Li, and Xiaoli Li. "Multi-domain Transfer Learning for Text Classification". In: *Natural Language Processing and Chinese Computing - 9th CCF International Conference, NLPCC 2020, Zhengzhou, China, October 14-18, 2020, Proceedings, Part I*. Ed. by Xiaodan Zhu, Min Zhang, Yu Hong, and Ruifang He. Vol. 12430. Lecture Notes in Computer Science. Springer, 2020, pp. 457–469. DOI: 10.1007/978-3-030-60450-9_36. URL: https://doi.org/10.1007/978-3-030-60450-9_36.

[274] Fabian M. Suchanek, David Gross-Amblard, and Serge Abiteboul. "Watermarking for Ontologies". In: *The Semantic Web - ISWC 2011 - 10th International Semantic Web Conference, Bonn, Germany, October 23-27, 2011, Proceedings, Part I*. Ed. by Lora Aroyo, Chris Welty, Harith Alani, Jamie Taylor, Abraham Bernstein, Lalana Kagal, Natasha Fridman Noy, and Eva Blomqvist. Vol. 7031. Lecture Notes in Computer Science. Springer, 2011, pp. 697–713. DOI: 10.1007/978-3-642-25073-6_44. URL: https://doi.org/10.1007/978-3-642-25073-6_44.

[275] Fabian M. Suchanek, Gjergji Kasneci, and Gerhard Weikum. "Yago: a core of semantic knowledge". In: *Proceedings of the 16th International Conference on World Wide Web, WWW 2007, Banff, Alberta, Canada, May 8-12, 2007*. Ed. by Carey L. Williamson, Mary Ellen Zurko, Peter F. Patel-Schneider, and Prashant J. Shenoy. ACM, 2007, pp. 697–706. DOI: 10.1145/1242572.1242667. URL: https://doi.org/10.1145/1242572.1242667.

[276] John R. Talburt. "2 - Principles of Information Quality". In: *Entity Resolution and Information Quality*. Ed. by John R. Talburt. Boston: Morgan Kaufmann, 2011, pp. 39 –62. ISBN: 978-0-12-381972-7. DOI: https://doi.org/10.1016/B978-0-12-381972-7.00002-6. URL: http://www.sciencedirect.com/science/article/pii/B9780123819727000026.

[277] Simone Teufel. "Argumentative Zoning: Information Extraction from Scientific Text". PhD thesis. University of Edinburgh, 1999.

[278] Simone Teufel, Jean Carletta, and Marc Moens. "An annotation scheme for discourse-level argumentation in research articles". In: *EACL 1999, 9th Conference of the European Chapter of the Association for Computational Linguistics, June 8-12, 1999, University of Bergen, Bergen, Norway*. The Association for Computer Linguistics, 1999, pp. 110–117. URL: https://www.aclweb.org/anthology/E99-1015/.

[279] Simone Teufel, Advaith Siddharthan, and Colin R. Batchelor. "Towards Domain-Independent Argumentative Zoning: Evidence from Chemistry and Computational Linguistics". In: *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing, EMNLP 2009, 6-7 August 2009, Singapore, A meeting of SIGDAT, a Special Interest Group of the ACL*. ACL, 2009, pp. 1493–1502. URL: https://www.aclweb.org/anthology/D09-1155/.

[280] Jakob Uszkoreit. *Transformer: A Novel Neural Network Architecture for Language Understanding.* In: *Google AI Blog.* Accessed: 2021-08-04. 2017. URL: https://ai.googleblog.com/2017/08/transformer-novel-neural-network.html.

[281] Sahar Vahdati, Said Fathalla, Sören Auer, Christoph Lange, and Maria-Esther Vidal. "Semantic Representation of Scientific Publications". In: *Digital Libraries for Open Knowledge - 23rd International Conference on Theory and Practice of Digital Libraries, TPDL 2019, Oslo, Norway, September 9-12, 2019, Proceedings.* Ed. by Antoine Doucet, Antoine Isaac, Koraljka Golub, Trond Aalberg, and Adam Jatowt. Vol. 11799. Lecture Notes in Computer Science. Springer, 2019, pp. 375–379. DOI: 10.1007/978-3-030-30760-8_37. URL: https://doi.org/10.1007/978-3-030-30760-8_37.

[282] Pierre-Yves Vandenbussche, Ghislain Atemezing, María Poveda-Villalón, and Bernard Vatant. "Linked Open Vocabularies (LOV): A gateway to reusable semantic vocabularies on the Web". In: *Semantic Web* 8.3 (2017), pp. 437–452. DOI: 10.3233/SW-160213. URL: https://doi.org/10.3233/SW-160213.

[283] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. "Attention is All you Need". In: *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA.* Ed. by Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett. 2017, pp. 5998–6008. URL: https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html.

[284] Marc B. Vilain, John D. Burger, John S. Aberdeen, Dennis Connolly, and Lynette Hirschman. "A model-theoretic coreference scoring scheme". In: *Proceedings of the 6th Conference on Message Understanding, MUC 1995, Columbia, Maryland, USA, November 6-8, 1995.* ACL, 1995, pp. 45–52. DOI: 10.3115/1072399.1072405. URL: https://doi.org/10.3115/1072399.1072405.

[285] Denny Vrandecic and Markus Krötzsch. "Wikidata: a free collaborative knowledgebase". In: *Commun. ACM* 57.10 (2014), pp. 78–85. DOI: 10.1145/2629489. URL: https://doi.org/10.1145/2629489.

[286] Anita de Waard and Gerard Tel. "The ABCDE Format Enabling Semantic Conference Proceedings". In: *SemWiki2006, First Workshop on Semantic Wikis - From Wiki to Semantics, Proceedings, co-located with the ESWC2006, Budva, Montenegro, June 12, 2006.* Ed. by Max Völkel and Sebastian Schaffert. Vol. 206. CEUR Workshop Proceedings. CEUR-WS.org, 2006. URL: http://ceur-ws.org/Vol-206/paper8.pdf.

[287] David Wadden, Ulme Wennberg, Yi Luan, and Hannaneh Hajishirzi. "Entity, Relation, and Event Extraction with Contextualized Span Representations". In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019.* Ed. by Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan. Association for Computational Linguistics, 2019, pp. 5783–5788. DOI: 10.18653/v1/D19-1585. URL: https://doi.org/10.18653/v1/D19-1585.

[288] Lucy Lu Wang et al. “CORD-19: The Covid-19 Open Research Dataset”. In: *CoRR* abs/2004.10706 (2020). arXiv: 2004.10706. URL: https://arxiv.org/abs/2004.10706.

[289] Richard Y. Wang and Diane M. Strong. “Beyond Accuracy: What Data Quality Means to Data Consumers”. In: *J. Manag. Inf. Syst.* 12.4 (1996), pp. 5–33. URL: http://www.jmis-web.org/articles/1002.

[290] Zhepei Wei, Yantao Jia, Yuan Tian, Mohammad Javad Hosseini, Mark Steedman, and Yi Chang. “Joint Extraction of Entities and Relations with a Hierarchical Multi-task Tagging Model”. In: *CoRR* abs/1908.08672 (2019). arXiv: 1908.08672. URL: http://arxiv.org/abs/1908.08672.

[291] Gerhard Weikum, Xin Luna Dong, Simon Razniewski, and Fabian M. Suchanek. “Machine Knowledge: Creation and Curation of Comprehensive Knowledge Bases”. In: *Found. Trends Databases* 10.2-4 (2021), pp. 108–490. DOI: 10.1561/1900000064. URL: https://doi.org/10.1561/1900000064.

[292] Karl R. Weiss, Taghi M. Khoshgoftaar, and Dingding Wang. “A survey of transfer learning”. In: *J. Big Data* 3 (2016), p. 9. DOI: 10.1186/s40537-016-0043-6. URL: https://doi.org/10.1186/s40537-016-0043-6.

[293] Felix Wu, Amauri H. Souza Jr., Tianyi Zhang, Christopher Fifty, Tao Yu, and Kilian Q. Weinberger. “Simplifying Graph Convolutional Networks”. In: *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*. Ed. by Kamalika Chaudhuri and Ruslan Salakhutdinov. Vol. 97. Proceedings of Machine Learning Research. PMLR, 2019, pp. 6861–6871. URL: http://proceedings.mlr.press/v97/wu19e.html.

[294] Chenyan Xiong, Russell Power, and Jamie Callan. “Explicit Semantic Ranking for Academic Search via Knowledge Graph Embedding”. In: *Proceedings of the 26th International Conference on World Wide Web, WWW 2017, Perth, Australia, April 3-7, 2017*. Ed. by Rick Barrett, Rick Cummings, Eugene Agichtein, and Evgeniy Gabrilovich. ACM, 2017, pp. 1271–1279. DOI: 10.1145/3038912.3052558. URL: https://doi.org/10.1145/3038912.3052558.

[295] Kosuke Yamada, Tsutomu Hirao, Ryohei Sasano, Koichi Takeda, and Masaaki Nagata. “Sequential Span Classification with Neural Semi-Markov CRFs for Biomedical Abstracts”. In: *Findings of the Association for Computational Linguistics: EMNLP 2020*. Online: Association for Computational Linguistics, Nov. 2020, pp. 871–877. DOI: 10.18653/v1/2020.findings-emnlp.77. URL: https://www.aclweb.org/anthology/2020.findings-emnlp.77.

[296] Beyza Yaman, Michele Pasin, and Markus Freudenberg. “Interlinking SciGraph and DBpedia Datasets Using Link Discovery and Named Entity Recognition Techniques”. In: *2nd Conference on Language, Data and Knowledge, LDK 2019, May 20-23, 2019, Leipzig, Germany*. Ed. by Maria Eskevich, Gerard de Melo, Christian Fäth, John P. McCrae, Paul Buitelaar, Christian Chiarcos, Bettina Klimek, and Milan Dojchinovski. Vol. 70. OASICS. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2019, 15:1–15:8. DOI: 10.4230/OASIcs.LDK.2019.15. URL: https://doi.org/10.4230/OASIcs.LDK.2019.15.

[297] Zhilin Yang, Ruslan Salakhutdinov, and William W. Cohen. "Transfer Learning for Sequence Tagging with Hierarchical Recurrent Networks". In: *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017. URL: https://openreview.net/forum?id=ByxpMd9lx.

[298] Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alexander J. Smola, and Eduard H. Hovy. "Hierarchical Attention Networks for Document Classification". In: *NAACL HLT 2016, The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego California, USA, June 12-17, 2016*. Ed. by Kevin Knight, Ani Nenkova, and Owen Rambow. The Association for Computational Linguistics, 2016, pp. 1480–1489. DOI: 10.18653/v1/n16-1174. URL: https://doi.org/10.18653/v1/n16-1174.

[299] Amrapali Zaveri, Anisa Rula, Andrea Maurino, Ricardo Pietrobon, Jens Lehmann, and Sören Auer. "Quality assessment for Linked Data: A Survey". In: *Semantic Web* 7.1 (2016), pp. 63–93. DOI: 10.3233/SW-150175. URL: https://doi.org/10.3233/SW-150175.

[300] Ye Zhang, Matthew Lease, and Byron C. Wallace. "Active Discriminative Text Representation Learning". In: *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA*. Ed. by Satinder P. Singh and Shaul Markovitch. AAAI Press, 2017, pp. 3386–3392. URL: http://aaai.org/ocs/index.php/AAAI/AAAI17/paper/view/14174.

[301] Yu Zhang, Min Wang, Morteza Saberi, and Elizabeth Chang. "From Big Scholarly Data to Solution-Oriented Knowledge Repository". In: *Frontiers Big Data* 2 (2019), p. 38. DOI: 10.3389/fdata.2019.00038. URL: https://doi.org/10.3389/fdata.2019.00038.

[302] Xuhui Zhou, Nikolaos Pappas, and Noah A. Smith. "Multilevel Text Alignment with Cross-Document Attention". In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*. Ed. by Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu. Association for Computational Linguistics, 2020, pp. 5012–5025. DOI: 10.18653/v1/2020.emnlp-main.407. URL: https://doi.org/10.18653/v1/2020.emnlp-main.407.

[303] Yukun Zhu, Ryan Kiros, Richard S. Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. "Aligning Books and Movies: Towards Story-Like Visual Explanations by Watching Movies and Reading Books". In: *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*. IEEE Computer Society, 2015, pp. 19–27. DOI: 10.1109/ICCV.2015.11. URL: https://doi.org/10.1109/ICCV.2015.11.

Curriculum Vitae

Arthur Brack
Master of Engineering

Education

11/2018 – 12/2021	**PhD student at Leibniz University Hannover and TIB – Leibniz Information Centre for Science and Technology**
10/2007 – 12/2008	**Master of Engineering, FHDW, Hannover**
10/2003 – 09/2006	**Diplom-Informatiker (FH), FHDW, Hannover**
09/1990 – 06/2002	**Fachabitur, Hannover**

Professional Experience

Since 07/2019	**Lecturer for Data Analytics, FHDW, Hannover**
Since 01/2019	**Managing Director, SET GmbH, Hannover**
04/2007 – 12/2018	**Lead Software Engineer, SET GmbH, Hannover**
10/2009 – 09/2011	**Lecturer for student projects, FHDW, Hannover**
10/2006 – 03/2007	**Software Engineer, klickTel AG, Hannover**
07/2003 – 09/2003	**Programmer, ESE GmbH, Braunschweig**
08/2000 – 07/2001	**Programmer, caatoosee search technology GmbH, Hildesheim**

Research Activities

2021	**Reviewer, Data Mining and Knowledge Discovery (Springer Journal)**
09/2021	**Talk: Recent results and trends in research on Natural Language Processing, DOXNET e.V. conference, Baden-Baden**
2021	**Program chair member, S.I. "Scientific Knowledge Graphs and Research Impact Assessment" in Quantitative Science Studies (MIT Press journal)**
2021	**Program chair member, 24th International Conference on Business Information Systems (BIS) 2021, Hannover**
09/2019	**Talk: Information Extraction with Natural Language Processing, ORKG workshop at Semantics Conference, Karlsruhe**
09/2019	**Athens Natural Language Processing Summer School (AthNLP), Athens**
2019 – 2020	**Supervision of two bachelor theses and one master thesis**
04/2018	**Talk: Introduction into Deep Learning, Infomatech at FHDW, Hannover**

List of Publications

A. Brack, A. Hoppe, P. Buschermöhle, R. Ewerth:
Cross-Domain Multi-Task Learning for Sequential Sentence Classification in Research Papers,
In: ACM/IEEE Joint Conference on Digital Libraries (JCDL), Cologne, Germany, 2022

A. Brack, A. Hoppe, M. Stocker, S. Auer, R. Ewerth:
Analysing the Requirements for an Open Research Knowledge Graph: Use Cases, Quality Requirements and Construction Strategies,
In: International Journal on Digital Libraries (IJDL) 23,
Special Issue (Best papers from TPDL 2019 and TPDL 2020), 2022

A. Brack, A. Hoppe, R. Ewerth:
Citation Recommendation for Research Papers via Knowledge Graphs
In: International Conference on Theory and Practice of Digital Libraries (TPDL),
Virtual Event, 2021, 165-174

A. Brack, D. Müller, A. Hoppe, R. Ewerth:
Coreference Resolution in Research Papers from Multiple Domains,
In: European Conference on Information Retrieval (ECIR), Virtual Event, 2021, 79-97

A. Brack, A. Hoppe, M. Stocker, S. Auer, R. Ewerth:
Requirements Analysis for an Open Research Knowledge Graph,
In: International Conference on Theory and Practice of Digital Libraries (TPDL),
Lyon, France, 2020, 3-18

J. D'Souza, A. Hoppe, **A. Brack**, M. Y. Jaradeh, S. Auer, R. Ewerth:
The STEM-ECR Dataset: Grounding Scientific Entity References in STEM Scholarly Content to Authoritative Encyclopedic and Lexicographic Sources,
In: Proceedings of the 12th Conference on Language Resources and Evaluation (LREC),
Marseille, France, 2020, 2192–2203

A. Brack, J. D’Souza, A. Hoppe, S. Auer, R. Ewerth:
Domain-Independent Extraction of Scientific Concepts from Research Articles,
In: European Conference on Information Retrieval (ECIR),
Lisboa, Portugal, 2020, 251-266

T. Baum, F. Kortum, K. Schneider, **A. Brack**, J. Schauder:
Comparing Pre Commit Reviews and Post Commit Reviews Using Process Simulation,
In: Software Engineering 2017, 117-118

T. Baum, F. Kortum, K. Schneider, **A. Brack**, J. Schauder:
Comparing Pre Commit Reviews and Post Commit Reviews Using Process Simulation,
In: J. Softw. Evol. Process. 29, 2017

T. Baum, F. Kortum, K. Schneider, **A. Brack**, J. Schauder:
Comparing Pre Commit Reviews and Post Commit Reviews Using Process Simulation,
In: Proceedings of IEEE/ACM International Conference on Software and System Processes (ICSSP),
Austin, Texas, USA, 2016, 26-35, Best Paper Award