



**FRIEDRICH-SCHILLER-
UNIVERSITÄT
JENA**

Heterogeneous Data to Knowledge Graphs Matching

Dissertation
zur Erlangung des akademischen Grades
Doktor-Ingenieur (Dr.-Ing.)

vorgelegt dem Rat der Fakultät für Mathematik und Informatik
der Friedrich-Schiller-Universität Jena

von

Nora Youssef Fahmy Abdelmageed
geboren am 25.10.1990 in Kairo, Ägypten



**FRIEDRICH-SCHILLER-
UNIVERSITÄT
JENA**

Heterogeneous Data to Knowledge Graphs Matching

**Dissertation
submitted for the degree of
Doktor-Ingenieur (Dr.-Ing.)**

presented to the Council of the Faculty of Mathematics and Computer Science of
Friedrich Schiller University Jena

submitted by
Nora Youssef Fahmy Abdelmageed
born 25.10.1990 in Cairo, Egypt

Gutachter:

1. **Prof. Dr. Birgitta König-Ries**
Friedrich-Schiller-Universität Jena, 07743 Jena, Germany
2. **Prof. Dr. Paul Groth**
Universiteit van Amsterdam, 1012 WX Amsterdam, Netherlands
3. **Dr. Ernesto Jiménez-Ruiz**
City, University of London, Northampton Square, London EC1V 0HB,
United Kingdom

Tag der öffentlichen Verteidigung: 29th June 2023

Reviewers:

1. **Prof. Dr. Birgitta König-Ries**
Friedrich Schiller University Jena, 07743 Jena, Germany
2. **Prof. Dr. Paul Groth**
University of Amsterdam, 1012 WX Amsterdam, Netherlands
3. **Dr. Ernesto Jiménez-Ruiz**
City, University of London, Northampton Square, London EC1V 0HB,
United Kingdom

Disputation Date: 29th June 2023

Dedication

To those who truly admire crafting things.

Zusammenfassung

Viele Anwendungen sind auf die Existenz wiederverwendbarer Daten angewiesen. Die FAIR-Prinzipien identifizieren detaillierte Beschreibungen von Daten und Metadaten als Kernbestandteile für das Erreichen von Wiederverwendbarkeit. Die Erstellung beschreibender Daten erfordert jedoch bislang einen massiven manuellen Aufwand. Eine Möglichkeit, die Wiederverwendbarkeit von Daten zu gewährleisten, besteht prinzipiell darin, sie in Wissensgraphen einzubinden. Die semantische Grundlage dieser Graphen liefert die notwendige Beschreibung für die Wiederverwendung. Gleichzeitig erhöht die direkte Einbindung von Daten in wissenschaftliche Wissensgraphen, wie sie etwa im Rahmen des Open Research Knowledge Graph Vorhabens entwickelt werden, Datensätze und die aus ihnen abgeleiteten Erkenntnisse gemeinsam zugänglich zu machen. Dies stellt einen wesentlichen Schritt zur Weiterentwicklung reproduzierbarer Wissenschaft dar. In dieser Arbeit konzentrieren wir uns auf die Biodiversitätsforschung als Beispieldomäne, um unseren Ansatz zu entwickeln und zu evaluieren. Biodiversität ist die Gesamtheit des Lebens auf der Erde, die evolutionäre, ökologische, biologische und soziale Formen umfasst. Das Verständnis der Biodiversität und der ihr zugrundeliegenden Mechanismen ist unerlässlich, um diese lebenswichtige Grundlage des menschlichen Wohergehens zu erhalten. Es ist unerlässlich, den aktuellen Zustand der Biodiversität und ihre Veränderung im Laufe der Zeit zu überwachen und ihre Kräfte zu verstehen, die das Leben in all seiner Vielfalt und seinem Reichtum antreiben und erhalten. Dieser Bedarf hat dazu geführt, dass zahlreiche Arbeiten auf diesem Gebiet veröffentlicht wurden. Beispielsweise wurde eine große Menge tabellarischer Daten (Datensätze), Textdaten (Veröffentlichungen) und Metadaten (z. B. Datensatzbeschreibung) generiert. Es handelt sich also um eine datenreiche Domäne mit einem außergewöhnlich hohen Bedarf an Datenwiederverwendung. Die Verwaltung und Integration dieser heterogenen Daten der Biodiversitätsforschung bleibt eine große Herausforderung. Unser zentrales Forschungsproblem besteht darin, die Wiederverwendbarkeit tabellarischer Daten zu ermöglichen, was ein Aspekt der FAIR-Datenprinzipien ist. Die gewünschten Beschreibungen, die eine erfolgreiche Datenwiederverwendung ermöglichen, werden als Wissensgraph dargestellt. Heutzutage erfordert die Erstellung solcher Beschreibungen jedoch einen erheblichen manuellen Aufwand. Im Bereich Biodiversität sucht ein Forschungsteam nach allen Datensätzen, die für seine Forschungsfrage relevant sind. Dies geschieht über Recherchen in Datenrepositorien, Literaturrecherchen und persönliche Verbindungen. Zum einen werden die gefundenen Publikationen dann gelesen, um wesentliche Referenzen zu finden. Andererseits werden aus ihnen Metadaten zu Datensätzen extrahiert und bei Datenrepositories die Informationen heruntergeladen. All diese Informationen werden dann manuell zusammengetragen, sodass dieser Prozess mehrere Monate dauern kann. Die Bereitstellung gut beschriebener Daten in einem Wissensgraph würde den erforderlichen Aufwand also drastisch reduzieren. Darüber hinaus ermöglicht diese Transformation die Abfrage des ursprünglichen Datensatzes und seiner sekundären Beschreibungen mithilfe einer strukturierten Abfragesprache wie SPARQL. In dieser Arbeit zielen wir darauf ab, die automatische Integration von Informationen aus verschiedenen Datenquellen innerhalb der Biodiversitätsdomäne zu ermöglichen, indem wir Technologien des semantischen Webs und Techniken des maschinellen Lernens kombinieren, um bestehende Wissensgra-

phen zu erweitern und anzureichern. Wir schließen tabellarische Datensätze ein, indem wir ihre Komponenten Zelle, Spalte und Spaltenpaar mit ihren Gegenstücken aus einem Knowledge Graph (KG) abgleichen. Darüber hinaus reichern wir den resultierenden KG aus tabellarischen Daten an, indem wir Hilfsinformationen wie vorhandene Metadaten und die zugehörigen Publikationen nutzen. Um dieses Ziel zu erreichen, führen wir verschiedene Ansätze ein, um jede Datenquelle in Wissensgraphen umzuwandeln. Die Beiträge dieser Arbeit sind: 1) Wir schlagen ein Framework, JenTab, vor, um tabellarische Daten mit Wissensgraphen abzugleichen. Wir haben JenTab im Rahmen der Semantic Web Challenge on Tabular Data to Knowledge Graph Matching (SemTab)-Challenge entwickelt und liefern weitere Studien und Analysen in dieser Arbeit. 2) Wir erstellen einen Benchmark für tabellarische Biodiversitätsdaten, Biodivtab, tabellarischen Datenbenchmark ‘BiodivTab’, um JenTab und die bestehenden State-of-the-Art-Systeme zu evaluieren. BiodivTab ist ein biodiversitätsspezifischer Benchmark, der aus 50 Tabellen mit manuellen Anmerkungen und Datenerweiterung besteht. 3) Wir entwerfen ein konzeptionelles Modell ‘BiodivOnto’, um die am häufigsten verwendeten Konzepte und die Beziehungen im Biodiversitätsbereich zu bestimmen. 4) Wir entwickeln ein Modell ‘BiodivBERT’ um Entitäten und Relationen aus unstrukturiertem Text zu extrahieren. 5) Wir konstruieren zwei Korpora ‘BiodivNERE’, um BiodivBERT auf zwei Downstream-Aufgaben, nämlich der Erkennung benannter Entitäten und Beziehungsextraktion zu evaluieren. Schließlich, 6) entwickeln wir einen Ansatz ‘Meta2KG’ um halbstrukturierte Daten, Metadaten, in Wissensgraphen umzuwandeln.

Abstract

Many applications rely on the existence of reusable data. The FAIR (Findability, Accessibility, Interoperability, and Reusability) principles identify detailed descriptions of data and metadata as the core ingredients for achieving reusability. However, creating descriptive data requires massive manual effort. One way to ensure that data is reusable is by integrating it into Knowledge Graphs (KGs). The semantic foundation of these graphs provides the necessary description for reuse. In the Open Research KG, they propose to model artifacts of scientific endeavors, including publications and their key messages. Datasets supporting these publications are essential carriers of scientific knowledge and should be included in KGs. We focus on biodiversity research as an example domain to develop and evaluate our approach. Biodiversity is the assortment of life on earth covering evolutionary, ecological, biological, and social forms. Understanding such a domain and its mechanisms is essential to preserving this vital foundation of human well-being. It is imperative to monitor the current state of biodiversity and its change over time and to understand its forces driving and preserving life in all its variety and richness. This need has resulted in numerous works being published in this field. For example, a large amount of tabular data (datasets), textual data (publications), and metadata (e.g., dataset description) have been generated. So, it is a data-rich domain with an exceptionally high need for data reuse. Managing and integrating these heterogeneous data of biodiversity research remains a big challenge. Our core research problem is how to enable the reusability of tabular data, which is one aspect of the FAIR data principles. The desired descriptions that enable successful data reuse are represented as a KG. However, today, creating such descriptions requires considerable manual effort. In the biodiversity domain, a research team searches for all datasets relevant to their research question. This happens via searches in data repositories, literature, and personal connections. On the one hand, the publications found are then read to find essential references. On the other hand, metadata about datasets are extracted from them, and the information is downloaded in the case of data repositories. All of this information is then manually collated. This process could take several months. Thus, providing well-described data in a KG would drastically reduce the required effort. In addition, this transformation enables querying the original dataset and its secondary descriptions using structured query language like SPARQL. In this thesis, we aim to enable the automatic integration of information from various data sources within the biodiversity domain by combining semantic web technologies and machine learning techniques to extend and enrich existing KGs. We include tabular datasets by matching their components: cell, column, and column-pair to their counterparts from a KG. In addition, we enrich the resultant KG from tabular data by leveraging auxiliary information like existing metadata and the associated publications. Working towards this goal, we introduce various approaches to transform each data source into KGs. The contributions of this thesis are: 1) We propose a framework ‘JenTab’ to match tabular data to KG. We developed JenTab in the scope of the Semantic Web Challenge on Tabular Data to Knowledge Graph Matching (SemTab) challenge, and we provide further studies and analysis in this thesis. JenTab is a top system that tackles the tasks of transforming tabular data into KGs. 2) We construct a tabular data benchmark ‘BiodivTab’ to evalu-

ate JenTab and the existing state-of-the-art systems. BiodivTab is a biodiversity-specific benchmark that consists of 50 tables, we constructed it using manual annotations and data augmentation. 3) We design a conceptual model ‘BiodivOnto’ to determine the most commonly used concepts and their relations in the biodiversity domain. 4) We develop a model ‘BiodivBERT’ to extract entities and relations from unstructured text. 5) We construct two corpora ‘BiodivNERE’ to evaluate BiodivBERT on two downstream tasks: Named Entity Recognition (NER), and Relation Extraction (RE). Finally, 6) We develop an approach ‘Meta2KG’ to transform semi-structured data, and metadata into KGs.

Acknowledgments

Working on a Ph.D. is a long journey with load of ups and downs. I was lucky to have had help from many people along the way, and I want to thank all of them for believing in me and for their continuous support.

Since family comes first, I would like to thank my parents, husband, and sisters for all their prayers and support. My mother's continuous brace and unconditional care kept me on the right path. My husband joined the ride mid-way, accompanying me with his love and support. It was especially needed during hard times, and I would not have been able to finish my Ph.D. without him.

My sincere gratitude to Prof. Dr. Birgitta König-Ries for her continuous guidance, efforts, and support. She gave me the freedom to explore many things my own way. She was always by my side and encouraged me throughout the whole journey.

I would also like to thank all FUSION members, either those who directly worked with me or those who helped and advised me. First of all, many thanks Sirko Schindler for his efforts and countless discussions during the development of JenTab and for his constant support until the very last moment of working on the Ph.D. Second, I would like to thank Alsayed Algergawy for his efforts in developing BiodivOnto and his general support, starting with helping me find settle at Jena! Moreover, thanks to Felicitas Löffler, Leila Feddoul, Sheeba Samuel, Jan Martin Keil, and Frank Löffler for all their efforts and help. I would also like to thank our biodiversity experts Cornelia Fürstenau, Jitendra Gaikwad, and Andreas Ostrowski, who participated in discussion rounds for the BiodivTab and BiodivNERE projects.

Many thanks to the Computer Vision group. My gratefulness goes to Prof. Dr. Joachim Denzler, who taught me skills I will never forget. I would like to thank Björn Barz for his feedback and support during and after the BiodivBERT project. Also, many thanks to Oliver Mothes, Sven Sickert, and Maha Shadaydeh for their help at the early stage of my Ph.D.

My gratitude goes to Christina Lohr and Luise Modersohn, the JULIE lab, for their time, discussions, and help during the BiodivBERT project.

Special thanks to the SemTab challenge organizers, Jiaoyan Chen, Vasilis Efthymiou, Ernesto Jiménez-Ruiz, Vincenzo Cutrona, Madelon Hulsebos, and Oktie Hassanzadeh.

I would like to thank Muhammad Abbady, Björn Barz, Sven Thiel, Tarek Al Mustafa, Andreas Ostrowski, Vamsi Krishna, and Franziska Zander for helping me proofread this thesis.

Last but not least, many thanks to Prof. Dr. Paul Groth and Dr. Ernesto Jiménez-Ruiz, who agreed to serve as external reviewers. I appreciate them taking the time to assess this thesis despite their numerous responsibilities.

Ehrenwörtliche Erklärung

Hiermit erkläre ich,

- dass mir die Promotionsordnung der Fakultät bekannt ist,
- dass ich die Dissertation selbst angefertigt habe, keine Textabschnitte oder Ergebnisse eines Dritten oder eigenen Prüfungsarbeiten ohne Kennzeichnung übernommen und alle von mir benutzten Hilfsmittel, persönliche Mitteilungen und Quellen in meiner Arbeit angegeben habe,
- dass ich die Hilfe eines Promotionsberaters nicht in Anspruch genommen habe und dass Dritte weder unmittelbar noch mittelbar geldwerte Leistungen von mir für Arbeiten erhalten haben, die im Zusammenhang mit dem Inhalt der vorgelegten Dissertation stehen,
- dass ich die Dissertation noch nicht als Prüfungsarbeit für eine staatliche oder andere wissenschaftliche Prüfung eingereicht habe.

Bei der Auswahl und Auswertung des Materials sowie bei der Herstellung des Manuskripts haben mich folgende Personen unterstützt:

- Prof. Dr. Birgitta König-Ries

Ich habe die gleiche, eine in wesentlichen Teilen ähnliche bzw. eine andere Abhandlung bereits bei einer anderen Hochschule als Dissertation eingereicht: Ja / Nein.

[Nora Youssef Fahmy Abdelmageed] Jena, 1. August 2023

Contents

Contents	xii
I Preliminary	1
1 Introduction	3
1.1 Motivation	5
1.2 Contributions	5
1.3 Thesis Structure	7
1.4 Publications	9
2 Problem Statement	13
2.1 Research Questions	14
2.2 Research Objectives	14
2.3 Requirements	15
2.3.1 Tabular Data Interpretation	15
2.3.2 Textual Data Interpretation	16
2.3.3 Metadata Interpretation	16
2.4 Research Methodology	17
3 Background	19
3.1 Definitions	19
3.1.1 Knowledge Graph	19
3.1.2 Tabular Data	20
3.1.3 Textual data	21
3.1.4 Metadata	21
3.2 Annotation Tasks	21
3.2.1 Semantic Table Interpretation (STI)	21
3.2.2 Textual Data Understanding	23
3.3 Summary	24
4 Related Work	25
4.1 Tabular Data Interpretation	25
4.1.1 Approaches	25
4.1.2 Benchmarks	31
4.2 Textual Data Interpretation	33
4.2.1 Named Entity Recognition (NER)	33
4.2.2 Relation Extraction (RE)	34
4.2.3 BERT-based Models	35
4.3 Metadata Interpretation	35
4.4 Summary	37

5	Solution Overview	39
5.1	Individual Components	40
5.2	Orchestration	40
II	Tabular Data Interpretation	43
6	JenTab Toolkit	45
6.1	Architecture	46
6.2	Preprocessing	47
6.3	Offline Resources	48
6.4	Disambiguation Contexts	49
6.5	CFS Pattern	50
6.5.1	Create	50
6.5.2	Filter	52
6.5.3	Select	53
6.6	Default Pipeline	54
6.7	Other Pipelines	55
6.8	Evaluation	56
6.8.1	Benchmarks	56
6.8.2	Datatype Prediction Assessment	57
6.8.3	Generic Lookup Coverage	57
6.8.4	Audit Results	58
6.8.5	Accuracy Scores	59
6.8.6	Runtime	64
6.9	Summary	65
7	BiodivTab: Table Annotation Benchmark	69
7.1	Construction Pipeline	70
7.1.1	Data Collection	70
7.1.2	Annotation Process	71
7.1.3	Data Augmentation	72
7.1.4	General Semantic Types	73
7.1.5	Assembly and Release	73
7.1.6	Ground Truth Extension	74
7.2	Evaluation	74
7.2.1	BiodivTab Insights	74
7.2.2	Availability and Licensing	76
7.3	Summary	76
III	Textual Data Interpretation	79
8	BiodivOnto: Biodiversity Data Model	81
8.1	Existing Ontologies	82
8.2	Methodology	83
8.2.1	Data Acquisition	83
8.2.2	Term Extraction	83
8.2.3	Term Filtration	84
8.2.4	Concepts and Relations Determination	85
8.3	Early Version	87
8.4	BiodivOnto Evolution	88

8.4.1	Biodiversity Questions	88
8.4.2	The Final BiodivOnto	90
8.5	Summary	91
9	BiodivBERT	93
9.1	Approach	94
9.2	Pre-training	94
9.2.1	Pre-training Data	95
9.2.2	Pre-training Task	96
9.3	Fine-Tuning	97
9.3.1	Named Entity Recognition	98
9.3.2	Relation Extraction	98
9.4	Evaluation	99
9.5	Summary	101
10	BiodivNERE Corpora	103
10.1	Resources Reuse	104
10.2	BiodivNER Construction Pipeline	105
10.2.1	Annotation Guidelines	105
10.2.2	Data Preparation	106
10.2.3	Trial and Pilot Phase	107
10.2.4	Annotation Process	107
10.2.5	Reconciliation	107
10.3	BiodivRE Construction Pipeline	109
10.3.1	Initial Construction	109
10.3.2	Random Sampling	109
10.3.3	Round Robin Sampling	109
10.3.4	Annotation Process	110
10.4	Evaluation	110
10.4.1	BiodivNER Insights	111
10.4.2	BiodivRE Insights	112
10.4.3	Availability and Licensing	114
10.5	Summary	114
IV	Metadata Interpretation	117
11	Meta2KG Framework	119
11.1	Approach	120
11.1.1	Data Acquisition	120
11.1.2	Preprocessing	121
11.1.3	Ontology Development	122
11.1.4	Embeddings Generation	123
11.1.5	Match	125
11.1.6	Validate & Populate	125
11.1.7	Release	125
11.2	Evaluation	125
11.2.1	Matching Results	127
11.2.2	Resultant Knowledge Graph	127
11.3	Summary	127

V The Final	129
12 Evaluation	131
12.1 Achievements	133
12.2 Retrospective; Limitations of the Solutions	140
12.3 Summary	142
13 Conclusions and Future Work	145
13.1 Available Materials	147
13.2 Future Directions	147
List of Figures	151
List of Tables	153
Listings	155
Bibliography	157
Appendices	177
A Certificates and Awards	179

Part I

Preliminary

Chapter 1

Introduction

Many applications rely on the existence of reusable data. One way to ensure the reusability of data is by integrating it into a Knowledge Graph (KG). KGs have become popular as a means to represent domain knowledge. Auer et al. [1] propose them as a way to bring scholarly communication to the 21st century. In the Open Research KG, they propose to model artifacts of scientific endeavors, including publications and their key messages. Datasets supporting these publications are essential carriers of scientific knowledge and should be included in KGs. A substantial side effect of this inclusion is that it supports FAIRness of data. The FAIR (Findability, Accessibility, Interoperability, and Reusability) principles [2] identify rich descriptions as the primary prerequisite for reusability. Since KGs make the semantics of the data explicit, they provide these rich descriptions. It is not trivial; adding datasets to KGs by a manual mapping is prohibitively expensive. Thus, the automatic transformation of datasets into KGs is an open demand. In this work, we will address this problem and focus on one example domain, biodiversity research.

Biodiversity is the assortment of life on earth covering evolutionary, ecological, biological, and social forms. Understanding such domain and its mechanisms are crucial to preserving this vital foundation of human well-being. It is imperative to monitor the current state of biodiversity and its change over time and to understand its forces driving and preserving life in all its variety and richness. This need has resulted in numerous works being published in this field. With this, a large amount of tabular data (datasets), textual data (publications), and metadata (e.g., dataset description) have been generated. The management and integration of these heterogeneous data of the biodiversity research remains a big challenge [3]. Thus, we will develop and test the proposed approach using datasets from biodiversity research since it is a data-rich domain with an exceptionally high need for data reuse, e.g., by KGs.

In this thesis, we aim to enable the automatic integration of information from various data sources within the biodiversity domain by combining semantic web technologies and machine learning techniques to extend and enrich existing KGs. We include tabular datasets by matching their components: cell, column, and column-pair to their counterparts from a KG. In addition, and to enrich the resultant KG from tabular data, we leverage auxiliary information like existing metadata and the associated publications. By this means, we discover more hidden concepts and relations among these secondary data; we have a fine-grained and complete final KG that describes the given datasets. We explain our data sources: tabular, textual, and metadata, in addition to the annotation tasks in Chapter 3.

This chapter motivates our work in Section 1.1. Section 1.2 gives an overview of our contributions. We outline the structure of this thesis in Section 1.3. Finally, Section 1.4 lists our publications that have been published as parts of the work presented in this thesis.

	B	C	D	E	F	G	H
1	Study_Name	Site_Name	Observational	Latitude	Longitude	Altitude	Country
6	Szlavec2006	Oregon Ridge	NA	39.486	-76.6901	NA	NA
7	Szlavec2006	Liberty Reservoir South	NA	39.395	-76.8749	NA	NA
17	Pechoro-Ilychskiy	For_4	Observation	62	58	NA	Russia
18	Pechoro-Ilychskiy	For_5	Observation	62	58	NA	Russia
19	Orenburg_Reg_1	Plantations	Observation	52.2	56.2	NA	Russia
20	Orenburg_Reg_1	Tselina	Observation	52.2	56.2	NA	Russia
25	norgrove	7	Observation	3.85	11.45	NA	Cameroon
26	norgrove	8	Observation	3.85	11.45	NA	Cameroon
27	moos	DairyT1CT_2	Observation	53.769	10.52511	NA	Germany
59	Komi_Reg	Knyazh_1	Experimental	62.28	50.68	NA	Russia

(a)

Abstract

Earthworms are an important soil taxon as ecosystem engineers, providing a variety of crucial ecosystem functions and services. Little is known about their diversity and distribution at large spatial scales, despite the availability of considerable amounts of local-scale data. Earthworm diversity data, obtained from the primary literature or provided directly by authors, were collated with information on site locations, including coordinates, habitat cover, and soil properties. Datasets were required, at a minimum, to include abundance or biomass of earthworms at a site. Where possible, site-level species lists were included, as well as the abundance and biomass of individual species and ecological groups. This global dataset contains 10,840 sites, with 184 species, from 60 countries and all continents except Antarctica.

(b)

Data_Provider_Contact_Details	<ul style="list-style-type: none"> * Name: Helen R. P. Phillips * Email: helen.phillips@idiv.de * Affiliation: iDiv * Country: Germany
Role *	<ul style="list-style-type: none"> Data_Creator: true Data_Manager: true Data_Owner: true Primary_Contact: true
Research_Group *	<ul style="list-style-type: none"> * Research_Group: Experimental Interaction Ecology Other-Specify:
Dataset *	<ul style="list-style-type: none"> * Project_or_Workshop_or_Thesis_Title: sWorm Workshop * Dataset_Title: Global distribution of earthworm diversity * Short_Abstract: Dataset for the analysis presented in the manuscript "Global distribution of earthworm diversity" * Keywords: earthworms, species richness, abundance, biomass, global, soil * Data_Access_Policy: Open (CC BY 4.0) Extensive_Description_Of_The_Dataset Data_Origin: Data Compilation Other-Specify: Status_Of_The_Data_Collection: Completed

(c)

Figure 1.1: Entities of interest per data source. (a) Tabular - Structured data, (b) Text - Unstructured Data, (c) Metadata - Semi-structured data.

1.1 Motivation

As a basis for our work, we held several meetings with biodiversity scientists. We found that biodiversity synthesis work is done today as follows: the research team searches for all datasets relevant to their research question. This happens via searches in data repositories, literature searches, and personal connections. The publications found are then read to find essential references. Metadata about datasets is extracted from them, and the information is downloaded in the case of data repositories. All of this information is then manually collated. This serves as a basis to decide on which data is usable for the study at hand, which conversions and error corrections are necessary, and how the data can be integrated. This process can take several months [4]. Providing well-described data in KGs would drastically reduce the required effort.

Various solutions aim at domain-specific KG construction exist. In biodiversity domain, it is imperative to monitor its current state and change over time and to understand the forces driving it to preserve life in all its variety and richness. Page [5] shows guidelines for a biodiversity KG creation. However, the resultant KG is coarse-grained. For example, the author proposes linking a whole dataset to a publication and an author. A more fine-grained solution, a rule-based framework [6] constructs a Biodiversity KG from publication text. It covers both named entity recognition and relation extraction tasks. The authors use different types of taggers to capture a wide range of information inside the document. We discuss in detail the related work in Chapter 4. At this point, there is a broad interest in building scientific KGs evidenced by the existing approaches. However, none of them deal efficiently with tabular datasets, publication text, and metadata altogether. A motivational example is given by Figure 1.1 to show how rich each data source is. We analyzed the provided data by [7], thus, we created a snippet for a) tabular, b) textual data, and c) metadata. We highlight possible candidates to be linked into a KG for each data type. Such example shows that each data source contributes different pieces of information to the target KG. In this thesis, we will develop and test the proposed approach using datasets from the biodiversity research since it is a data-rich domain with an exceptionally high need for data reuse., e.g., by KGs.

1.2 Contributions

This thesis aims to match raw data to an existing KG (in the case of tables) and create the target KG from scratch (the case of textual data and metadata). Raw data is any form of data without any semantic annotations. In our case, it could be either primary (tables) or secondary (text or metadata) data. Our contributions are divided into separate modules, each of which handles the desired data source and tackles its unique challenges. In the following, we list our contributions and map them into the individual chapters in this thesis. We explain the orchestration of these contributions (modules and benchmarks) in Chapter 5.

1. Tabular data understanding framework

We developed a complete framework matching the individual table components to their counterparts from the target KG. The input for the such framework is raw tables. Then, it will map each table's cell, column, and column-pair to the KG entity, semantic type (class), and property. Such output could be mapped easily to triples and RDF files. This framework is the module that we created for tabular data understanding. This contribution is presented in Chapter 6.

2. A biodiversity-specific tabular data benchmark

We studied the available state-of-the-art benchmarks that are commonly used for evaluating the tabular data understanding framework. To the best of our knowledge, none of them are derived from the biodiversity research field. Thus, we constructed a benchmark for semantic table annotations. It is based on manually annotated real tables and data augmentation. We support annotations from two KGs, Wikidata and DBpedia. The dataset provides the basis for domain-specific evaluation of the framework. In addition, the benchmark has been made available for public use in 2021 and 2022. This contribution is detailed in Chapter 7.

3. Assessment of the domain-specific tabular data benchmark

We compare our biodiversity-specific benchmark for tabular data annotation against the most common state-of-the-art datasets. In addition, we highlight its unique characteristics and report the best-performing systems that annotate tabular data to KGs using this benchmark. By this means, we feature the importance of such a benchmark to the community. This contribution is explained in Chapter 7.

4. Tabular data understanding framework evaluation

We continuously developed and tested the tabular data to KG framework in the scope of SemTab challenge. The challenge started in 2019 and provided a unified framework for the systems that tackle the tabular data interpretation tasks. We evaluated our framework using general domain benchmarks since 2020 for three consecutive years. Most of these datasets are Automatically Generated (AG). Our framework has been among the best-performing systems during the years of development (shown in Chapter 6). In addition, we evaluated such a framework on our own domain-specific tabular data benchmark for two years using different target KGs. This contribution is shown in Chapter 12.

5. BiodivOnto data model and ontology

We aim to capture the most common and important concepts and relations that are used in the biodiversity domain. These classes and relations represent what we will extract from unstructured text. Thus, we developed a data-driven approach that extracts those entities and relations of interest. In addition, we conducted several interviews with biodiversity experts to verify our outcome. We applied all the input from our experts. BiodivOnto features our data model for the following three contributions. This contribution is presented in Chapter 8.

6. Biodiversity-specific benchmarks for downstream tasks

We studied the existing benchmark that we could use to evaluate our textual data understanding module. To the best of our knowledge, there are many available but limited to species classification. Thus, we constructed two benchmarks for Named Entity Recognition (NER) and Relation Extraction (RE) that captures a broader range of entities and relations of interest as outlined in the BiodivOnto data model. This contribution is presented in Chapter 10.

7. Assessment of the benchmarks against the existing state of the art

We compare the developed benchmarks for downstream tasks (NER and RE) against the state-of-the-art datasets. In addition, we feature their unique properties and outline their insights. This contribution is detailed in Chapter 10.

8. Textual data understanding module

We aim to construct domain-specific embeddings to extract domain entities and relations. Thus, we pre-trained a BERT-based model using large domain-specific

corpora. We constructed these corpora based on biodiversity publications (abstracts and full text). This contribution is detailed in Chapter 9.

9. Textual data understanding module evaluation

We evaluate our textual data understanding framework on the constructed domain-specific benchmarks. Thus, we fine-tune our framework on two downstream tasks: NER and RE. The scores show an improvement compared to the state-of-the-art models. In addition, the need for such a domain-specific framework. This contribution is outlined in Chapter 9.

10. Biodiversity Metadata Ontology (BMO) data model and ontology

We manually crafted an ontology representing the common vocabulary in metadata files. This terminology is used to describe the datasets from the biodiversity domain. Similar to BiodivOnto, we aim to capture the concepts and relations we extract from the semi-structured data source (metadata). This contribution is explained in Chapter 11.

11. Semi-structured data understanding module

We developed a framework that captures both entities and relations from the semi-structured data, e.g., dataset descriptions in metadata. This framework is based on unsupervised learning techniques and embeddings. This framework's output is an RDF file that represents the automatically generated KG from metadata. This contribution is presented in Chapter 11.

12. Semi-structured understanding module evaluation

We use two embeddings' sources to demonstrate the performance of the framework. We present the quality of the automatically generated KG, and the matching scores of the semi-structured data understanding. In addition, we discuss the obstacles we faced during such automatic transformation. This contribution is shown in Chapter 11.

1.3 Thesis Structure

We organized the thesis into five parts: 1) Preliminary, it includes chapters from Chapter 1 - Chapter 5. 2) Tabular Data Interpretation, it contains Chapter 6 and Chapter 7. 3) Textual Data Interpretation, it includes chapters from Chapter 8 to Chapter 10. 4) Metadata Interpretation, it has Chapter 11. Finally, 5) The Final, which has Chapter 12 and Chapter 13. Parts 2-4 represent our core contributions to this thesis. In their chapters, we include the developed methodology, the evaluation strategy, and the reported scores. The complete list of this thesis' chapters is organized as follows:

- Chapter 1, this chapter, presents the motivation for the overall work. We briefly define the main pillars, e.g., data sources of interest. In addition, we outline our contributions and list our publications that represent parts of this dissertation.
- Chapter 2 explains our main problem statement, research questions. In addition, we formulate the concrete requirements that we fulfill in this thesis. Moreover, we describe our research methodology.
- Chapter 3 gives an overview of the necessary background for this thesis. For example, we define our primary (tabular data) and auxiliary (unstructured and semi-structured) data sources that we transform into KGs. In addition, we explain the

common tasks we tackle in various parts of this thesis. For example, Semantic Table Interpretation (STI) tasks, Named Entity Recognition (NER), and Relation Extraction (RE).

- Chapter 4 outlines the current state of the art in the context of our three pillars. At first, for tabular data understanding, we discuss the tabular data to knowledge graph matching systems. Second, in textual data interpretation, we summarize the downstream tasks we use to extract entities and their relations. In addition, the best-performing models tackling both tasks. Finally, we highlight the most relevant state of the art that transforms metadata to KGs in semi-structured data understanding.
- Chapter 5 orchestrates our developed individual modules and benchmarks. It sketches our data sources (three pillars), the corresponding developed framework, and the used benchmark for evaluation. It gives the big picture of the entire thesis. It also points to the possible integration of the fundamental components into one framework.
- Chapter 6 demonstrates our contribution to develop a framework that matches tabular data components to KGs. This chapter is also relevant for the tabular data understanding module. We explain our developed approach to tackle the STI tasks. We include the framework evaluation during its continuous development using general and domain-specific benchmarks.
- Chapter 7 explains our methodology to construct a domain-specific tabular data benchmark for STI tasks. We include the statistics of such benchmark and the comparison versus the existing benchmarks in the same chapter.
- Chapter 8 represents our data model of the biodiversity domain. We outline our data-driven approach to reach these classes and relations. In addition, we include our continuous improvements for this ontology.
- Chapter 9 represents our developed framework for textual data interpretation. We explain our pre-training and fine-tuning for the machine-learning-based technique we developed. In addition, we include the evaluation scores of this framework compared to other state-of-the-art approaches using various benchmarks.
- Chapter 10 shows our manually crafted benchmarks to evaluate the textual data understanding framework. The used classes and relations for annotation are from Chapter 8. We also include a state of the art comparison for these benchmarks.
- Chapter 11 explains our developed data model and framework for semi-structured data interpretation. We include evaluation scores for the matching algorithm that transform metadata to a KG. In addition, we discuss the common obstacles that face such an automatic transformation process.
- Chapter 12 shows the summary of the developed benchmarks and frameworks. In addition, we discuss how the developed modules fulfill the requirements.
- Chapter 13 concludes the entire research work and discusses its current state. In addition, it depicts possible future directions to continue this research.

1.4 Publications

Parts of this dissertation have been published in conferences and journals as follows:

- Nora Abdelmageed, Towards Transforming Tabular Datasets into Knowledge Graphs. The Semantic Web: ESWC 2020 Satellite Events. pp. 217-228, vol. 12124, https://doi.org/10.1007/978-3-030-62327-2_37, 2020.
This paper gives an abstract overview of the entire thesis. We report on the first phase of this work, i.e., our meetings with the biodiversity experts and requirement analysis. In addition, we report on the early-stage work results. (Corresponds to Chapter 1 and Chapter 2).
- Nora Abdelmageed, Sirko Schindler: JenTab: Matching Tabular Data to Knowledge Graphs. Proceedings of the Semantic Web Challenge on Tabular Data to Knowledge Graph Matching (SemTab 2020) co-located with the 19th International Semantic Web Conference (ISWC 2020). pp. 40-49, vol. 2775, <http://ceur-ws.org/Vol-2775/paper4.pdf>, 2020.
This paper presents our first attempt to develop the tabular data understanding framework. It describes its main building blocks of it in detail. This paper shows our first participation in SemTab challenge. (Corresponds to Chapter 6).
- Nora Abdelmageed, Sirko Schindler: JenTab: A Toolkit for Semantic Table Annotations. Proceedings of the 2nd International Workshop on Knowledge Graph Construction co-located with the 18th Extended Semantic Web Conference (ESWC 2021). vol. 2873, <http://ceur-ws.org/Vol-2873/paper5.pdf>, 2021.
This paper extends the above publication. We included various settings to solve the required task. We investigated the accuracy of each of them and analyzed the processing time. (Corresponds to Chapter 6).
- Nora Abdelmageed, Sirko Schindler: JenTab Meets SemTab 2021's New Challenges. Proceedings of the Semantic Web Challenge on Tabular Data to Knowledge Graph Matching co-located with the 20th International Semantic Web Conference (ISWC 2021). pp. 42-53, vol. 3103, <http://ceur-ws.org/Vol-2775/paper4.pdf>, 2021.
This paper extends the JenTab framework. It includes various pipelines that we develop based on the given dataset characteristics. (Corresponds to Chapter 6).
- Nora Abdelmageed, Sirko Schindler: JenTab: Do CTA solutions affect the entire scores? Proceedings of the Semantic Web Challenge on Tabular Data to Knowledge Graph Matching co-located with the 21st International Semantic Web Conference (ISWC 2022). pp. 72-79, vol. 3320, <https://ceur-ws.org/Vol-3320/paper8.pdf>, 2022.
This paper represents our last contribution to the SemTab challenge. It includes a new pipeline that tries to solve the required tasks based on the given header information. (Corresponds to Chapter 6).
- Nora Abdelmageed, Sirko Schindler, Birgitta König-Ries: BiodivTab: A Table Annotation Benchmark based on Biodiversity Research Data. Proceedings of the Semantic Web Challenge on Tabular Data to Knowledge Graph Matching co-located with the 20th International Semantic Web Conference (ISWC). pp. 13-18, vol. 3103, <http://ceur-ws.org/Vol-3103/paper1.pdf>, 2021.
This paper introduces our first version of the domain-specific benchmark for tabular data annotation (BiodivTab). We manually annotated biodiversity datasets with entities and classes from Wikidata. BiodivTab is made available for the public and used during SemTab 2021. (Corresponds to Chapter 7).

- Nora Abdelmageed, Sirko Schindler, Birgitta König-Ries: BiodivTab: Semantic Table Annotation Benchmark Construction, Analysis, and New Additions. Proceedings of the 17th International Workshop on Ontology Matching co-located with the 21st International Semantic Web Conference (ISWC 2022). 2022.
This paper gives the details of our pipeline to construct the benchmark. In addition, it shows BiodivTab’s statistics and insights. It also explains our methodology for creating a new ground truth from DBpedia. BiodivTab (DBpedia) is made available for the public and used during SemTab 2022. (Corresponds to Chapter 7).
- Nora Abdelmageed, Alsayed Algergawy, Sheeba Samuel, Birgitta König-Ries: BiodivOnto: Towards a Core Ontology for Biodiversity. The Semantic Web: ESWC 2021 Satellite Events. pp. 3-8, vol. 12739, https://doi.org/10.1007/978-3-030-80418-3_1, 2021.
This paper gives an overview of our pipeline to construct the BiodivOnto data model. Such ontology, in this paper, contains seven core concepts and their relations. (Corresponds to Chapter 8).
- Nora Abdelmageed, Alsayed Algergawy, Sheeba Samuel, Birgitta König-Ries: A Data-driven Approach for Core Biodiversity Ontology Development. Proceedings of the Joint Ontology Workshops 2021 Episode VII: The Bolzano Summer of Knowledge co-located with the 12th International Conference on Formal Ontology in Information Systems (FOIS 2021) and the 12th International Conference on Biomedical Ontologies (ICBO 2021). vol.2969, <http://ceur-ws.org/Vol-2969/paper5-s4biodiv.pdf>, 2021.
This paper extends the one above. It explains our data-driven approach in detail. It includes more statistics about the collected data and discusses the open issues about the suggested methodology. (Corresponds to Chapter 8).
- Nora Abdelmageed, Felicitas Löffler, Leila Feddoul, Alsayed Algergawy, Sheeba Samuel, Jitendra Gaikwad, Anahita Kazem, Birgitta König-Ries: BiodivNERE: Gold standard corpora for named entity recognition and relation extraction in the biodiversity domain. Biodiversity Data Journal, ISSN. 1314-2836, vol: 10 <https://doi.org/10.3897/BDJ.10.e89481>, 2022.
This paper presents our manually crafted benchmarks for both NER and RE tasks. It describes our pipeline to construct both corpora and shows their statistics compared to the existing benchmarks. (Corresponds to Chapter 10).
- Nora Abdelmageed, Felicitas Löffler, Birgitta König-Ries: BiodivBERT: a Pre-Trained Language Model for the Biodiversity Domain SWAT4HCLS 2023: The 14th International Conference on Semantic Web Applications and Tools for Health Care and Life Sciences, pp. 62–71, vol: 3415, <https://ceur-ws.org/Vol-3415/paper-7.pdf>, 2023.
This paper introduces our domain-specific BERT-based model as a textual data understanding framework. It describes our collected pre-training data, task, and fine-tuning of BiodivBERT. In addition, it points to the ontology population as a possible application of the model. (Corresponds to Chapter 9).
- Nora Abdelmageed, Birgitta König-Ries: Meta2KG: Transforming Metadata to Knowledge Graphs. Proceedings of the 17th International Workshop on Ontology Matching co-located with the 21st International Semantic Web Conference (ISWC 2022). pp. 226-228, vol. 3324, https://ceur-ws.org/Vol-3324/om2022_poster3.pdf, 2022.
This paper briefly describes our developed unsupervised technique to transform the metadata files into a KG. (Corresponds to Chapter 11).

- Nora Abdelmageed, Birgitta König-Ries: Meta2KG: An Embeddings-based Approach for Transforming Metadata to Knowledge Graphs. Proceedings of the Fourth International Workshop on Knowledge Graph Construction co-located with the 20th Extended Semantic Web Conference (ESWC 2023). 2023.

This paper describes our developed unsupervised technique to transform the metadata files into a KG with detailed evaluation and additional baseline approaches. (Corresponds to Chapter 11).

Other Publications The following list includes other relevant papers that do not map directly to chapters but either give further context to the thesis or are used in evaluating specific parts of our contributions:

- Felicitas Löffler, Nora Abdelmageed, Samira Babalou, Pawandeep Kaur, Birgitta König-Ries: Tag Me If You Can! Semantic Annotation of Biodiversity Metadata with the QEMP Corpus and the BiodivTagger. Proceedings of The 12th Language Resources and Evaluation Conference, LREC, pp. 4557-4564, <https://aclanthology.org/2020.lrec-1.560/>, 2020.

Our contribution to this paper is the construction of the QEMP corpus. We used this corpus in evaluating our developed textual data interpretation framework (BiodivBERT).

- Vincenzo Cutrona, Jiaoyan Chen, Vasilis Efthymiou, Oktie Hassanzadeh, Ernesto Jiménez-Ruiz, Juan Sequeda, Kavitha Srinivas, Nora Abdelmageed, Madelon Hulsebos, Daniela Oliveira, Catia Pesquita: Results of SemTab 2021. Proceedings of the Semantic Web Challenge on Tabular Data to Knowledge Graph Matching co-located with the 20th International Semantic Web Conference (ISWC 2021). pp. 1-12, vol. 3103, <http://ceur-ws.org/Vol-3103/paper0.pdf>, 2021.

Our contribution to this paper is the use of BiodivTab for the first time during SemTab 2021, where Wikidata is the target KG. This paper gives a more profound context about the challenge, evaluation mechanism, and lessons learned from this edition of the challenge.

- Nora Abdelmageed, Vincenzo Cutrona, Jiaoyan Chen, Vasilis Efthymiou, Oktie Hassanzadeh, Ernesto Jiménez-Ruiz, Juan Sequeda, Kavitha Srinivas, Madelon Hulsebos, Daniela Oliveira, Catia Pesquita: Results of SemTab 2022. Proceedings of the Semantic Web Challenge on Tabular Data to Knowledge Graph Matching co-located with the 21st International Semantic Web Conference (ISWC 2022). pp. 1-13, vol. 3320, <http://ceur-ws.org/Vol-3320/paper0.pdf>, 2022.

Our contribution to this paper is the use of BiodivTab for the second time during SemTab 2021, where DBpedia is the target KG. Similar to the paper above, this paper concludes the 2022 edition of the challenge.

Chapter 2

Problem Statement

Our core research problem is how to enable the reusability of tabular data which is one aspect of the FAIR data principles [2]. A good, ideal machine-readable description of the data is essential for successful data reuse. The desired descriptions are represented as a Knowledge Graph (KG). However, today, creating such descriptions requires considerable manual effort. As our conducted meetings with the biodiversity experts showed, a biodiversity research team searches for all datasets relevant to their research question. This happens via searches in data repositories, literature, and personal connections. On the one hand, the publications found are then read to find essential references. On the other hand, metadata about datasets are extracted from them, and the information is downloaded in the case of data repositories. All of this information is then manually collated. This serves as a basis to decide on which data is usable for the study at hand, which conversions and error corrections are necessary, and how the data can be integrated. This process can take several months [4]. Thus, providing well-described data in a KG would drastically reduce the required effort. In addition, we enable querying the original dataset and its secondary descriptions using a structured query language like SPARQL. We believe that the automatic construction of such a fine-grained KG will be possible by leveraging auxiliary information besides the dataset itself. The crucial sources of additional information are, in our case, metadata and publications.

The main goal of this research is to enhance the reusability of datasets by the automatic transformation of raw data either in structured (tabular data), unstructured (textual data), or semi-structured (metadata) into a KG. In order to achieve this ultimate goal, we focus on three research areas: The first research dimension is the **Tabular data interpretation (TabI)**. In this field, we match the individual table components (cell, column, column-pair) to their counterparts from existing KGs. The second dimension of this research is the **Textual data interpretation (TexI)**. In this research phase, we extract both entities and relations of interest from unstructured text. In our context, such textual data is given by either a dataset-associated abstract or a full publication. The last area of research in this work is the **Metadata interpretation (MI)**. This RA is similar to TexI in terms of the task objectives. We extract both entities and their relations but, the input data is semi-structured, as represented in metadata files. The tasks we try to solve per research area provide the basis for constructing the KG. For example, each of them helps us identify the triple’s subject, predicate, and object. We explain our input format (data sources), the matching tasks for tabular data, and textual and metadata understanding format in Chapter 3.

In this chapter, in the context of these three research areas, we identify research questions and objectives in Section 2.1 and Section 2.2 respectively. We translate these objectives into concrete requirements in Section 2.3. Finally, we explain our adopted research methodology for this thesis in Section 2.4.

2.1 Research Questions

Our research focuses on how to automate the transformation of tabular datasets along with their auxiliary data (publication text and metadata) into a KG. We divide this general research problem into three fine-grained research questions, each of which maps to the research area as described at the beginning of this chapter:

How can we use tabular datasets for KG construction? (RQ1) (related to TabI)

How can we benefit from the information in the associated publications to enrich the constructed KG? (RQ2) (related to TexI)

How can we leverage the existing metadata to enrich the constructed KG? (RQ3) (related to MI)

2.2 Research Objectives

Our overall contribution will enable the automatic integration of tabular datasets with their secondary data (publications and metadata) into KGs, thereby considerably increasing FAIRness, particularly reusability. Our domain of interest is the biodiversity domain since it is a data-rich field that needs automatic mechanisms to make the best use of such untapped wealth. The following objectives will reach our ultimate objective:

Objective 1 Develop methods that take a tabular dataset as input and automatically create a KG out of it. These methods will determine the meaning of individual columns, and their data type and relationships across columns. Such tools are useful to increase tabular data understanding even without the subsequent transformation into a KG. This objective answers the first research question RQ1.

Objective 2 Construct a domain-specific tabular data benchmark that would be used to evaluate the methods implemented in Objective 1. This objective also contributes to RQ1 to enforce the assessment of the tabular data understanding framework on the biodiversity domain and not only on the general domain.

Objective 3 Design and create a conceptual model using semantic web technologies that describe the essential concepts and relations of the domain. This objective contributes to the second RQ2, where it targets textual data transformation into a KG. Such a task would need predefined categories or entity types. This objective helps us to determine those classes using a systematic approach.

Objective 4 Develop a textual data transformer to leverage potentially available auxiliary information in publications describing the dataset based on the data model created in Objective 3. This objective also contributes to the second RQ2 with a technique that can detect the entities and relations of interest from text.

Objective 5 Construct domain-specific benchmarks for textual data downstream tasks. This objective plays a vital role to the second RQ2 as well. To ensure the effectiveness of the developed framework in Objective 4 using the biodiversity domain.

Objective 6 Design and create a conceptual model using semantic web technologies that represents the most common dataset descriptions in the associated metadata files. This objective and the next two solve the third research question RQ3. Similar to the previous three objectives that tackle textual data interpretation, these three transform the metadata files into a KG with the same order.

Objective 7 Develop methods that transform the semi-structured data (metadata) into a KG based on the data model created in Objective 6.

Objective 8 Evaluate the developed methods that are presented in Objective 7 using domain-specific benchmarks.

2.3 Requirements

To answer our research questions for Section 2.1, and to achieve the goals of our work we discussed in Section 2.2, we formulate our functional and non-functional requirements of the three research areas as follows.

2.3.1 Tabular Data Interpretation

Since we have RQ1 that targets the transformation of tabular data into KGs, we require a framework that can solve such a question. In the following, we discuss the requirements of the framework and the domain-specific benchmark, emphasizing the effectiveness of the biodiversity domain.

R1.1 The tabular data understanding framework should be able to match the individual table components to their counterparts of a target KG (Objective 1).

R1.2 The framework should have extensible and configurable components. For example, it should provide an easy way to change the target KG and enable various settings for solving (Objective 1).

R1.3 The framework should be scalable. It can process large-scale tables in a reasonable time (Objective 1).

R1.4 The framework should have a reasonable amount of dependencies and provides an easy way to configure on a local development environment (Objective 1).

R1.5 The framework should provide an analysis of the current processing input data. For example, how many tables are in progress, successfully completed, failed to complete, and the errors that were returned (Objective 1).

R1.6 The framework should store the results and have the feature to extract them in a specific format (Objective 1).

R1.7 The framework should have trusted accuracy scores and demonstrate its effectiveness in the biodiversity domain (Objective 1 and Objective 2).

R1.8 The domain-specific benchmark should reflect real-world challenges (Objective 2).

R1.9 The benchmark should have annotations that convey human-level knowledge. (Objective 2).

R1.10 The benchmark should be diverse, and capture various aspects from the biodiversity domain (Objective 2)

2.3.2 Textual Data Interpretation

Our previously discussed RQ2 would be covered by a domain-specific model that can extract the essential information from text to construct a KG or enrich an existing one. We need a domain-specific model since estimating the general domain methods on domain-specific tasks is hard [8]. In the following, we discuss the requirements of these concepts and relations, model, and evaluation corpora.

- R2.1** The fundamental information that this model detects should reflect the important concepts and relations of the domain (Objective 3).
- R2.2** The model should be trained on domain-specific textual data (Objective 4).
- R2.3** The model should be able to detect the most important entities and relations of the biodiversity domain (Objective 4).
- R2.4** The model should be easy to use (Objective 4).
- R2.5** The model should demonstrate the effectiveness of the extracted information (Objective 4 and Objective 5).
- R2.6** The evaluation corpora should be diverse to reflect various aspects from the biodiversity research field (Objective 5).
- R2.7** The evaluation corpora should be trusted and gold-standard level to ensure high quality of annotations (Objective 5).
- R2.8** The evaluation corpora should be aligned with the schema in R2.1 (Objective 5).
- R2.9** The evaluation corpora should support machine learning format (Objective 5).
- R2.10** The evaluation corpora should demonstrate a good balance for classes and relations between the training and testing data folds (Objective 5).

2.3.3 Metadata Interpretation

Our research question RQ3 would be answered by building a technique that transforms a given semi-structured file (metadata) into a KG. In the following, we discuss the requirements of such a technique.

- R3.1** The technique requires an underlying model that should reflect the most common vocabulary used in biodiversity repositories metadata (Objective 6).
- R3.2** The technique should be able to capture the most important entities and relations of the biodiversity metadata (Objective 7).
- R3.3** The technique should effectively auto-populate the underlying data model (Objective 7).
- R3.4** The technique should produce a machine-readable format output, e.g., RDF (Objective 7).
- R3.5** The technique should show the effectiveness of the extracted information from the given descriptions on a ground truth data (Objective 8).

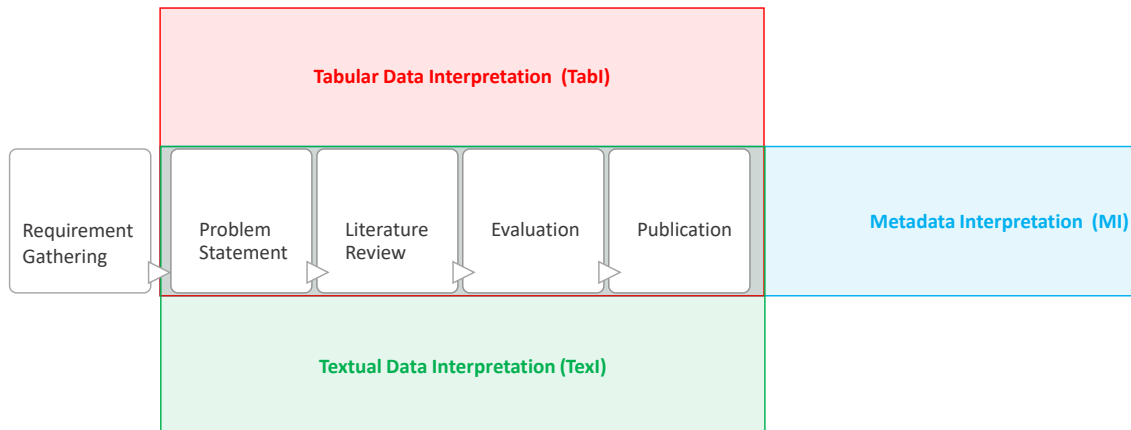


Figure 2.1: Abstract view of our research methodology

2.4 Research Methodology

The research methodology followed in this thesis is an iterative process for each research area as depicted in Figure 2.1 to tackle our 8-objectives problem statement. In the first step of our pipeline, we conducted several meetings with domain experts from the biodiversity field for requirement gathering as described in Chapter 1. Based on their requirements, we came up with three main stages for our project: Firstly, we aim to build a KG from the tabular dataset itself as a standalone data source. Secondly, we do further extensions to the resultant KG using related publications of the tabular data, either by the use of abstracts or the full texts. Finally, we will add information gained from metadata or any auxiliary semi-structured data. For each of these stages, we will perform a complete development cycle from analyzing the literature and concept development to implementation, evaluation, and publication. In the following, we explain each stage that maps to an individual research area, thus, how we achieved our objectives as listed in Section 2.2.

To understand the tabular data, we reviewed the literature in terms of the annotation tasks, existing approaches, and the common challenges in this research field. In addition, we reviewed the existing benchmarks that are used for evaluating these approaches. We give a detailed overview of this research area, TabI, in Section 4.1. In this context, we designed and implemented JenTab toolkit for the tabular data understanding module (Objective 1). In JenTab, we solve three tasks of the Semantic Table Interpretation (STI): 1) Cell Entity Annotation (CEA), Column Type Annotation (CTA), Column Property Annotation (CPA). We give the details of these tasks in Chapter 3. We developed and tested it using various large scale benchmarks for STI since 2020 to 2022 [9, 10, 11, 12]. We explain the evolution of JenTab in Chapter 6. After reviewing the existing benchmarks for STI, our findings showed that no benchmark is derived from a real biodiversity research field. To bring JenTab closer to our domain of interest, we constructed BiodivTab [13, 14] for tabular data annotation. BiodivTab is a biodiversity-specific benchmark that is based on real data and data augmentation. It consists of 50 tables that are annotated tables using Wikidata and DBpedia (Objective 2). We explain the details of BiodivTab construction, its insights, and the common domain challenges in Chapter 7.

To interpret the textual data in publication abstracts or full text, we investigated the widely used vocabulary in the biodiversity domain. This terminology act as a basis for our domain-specific classes and relations. To come up with this data model, we developed a data-driven approach that resulted in our conceptual model BiodivOnto [15, 16] (Objective 3). During the model formulation, we held several meetings, calls, and discussions with the biodiversity experts to verify our derived data model. We outline our data-driven

approach, conceptual model, and its further extensions in Chapter 8. The next step is to build a framework that automatically extracts our vocabulary from textual data. We studied the related work concerning both Named Entity Recognition (NER) and Relation Extraction (RE). The former detects entities of interest while the latter extracts their relations. During this stage of the work, we developed BiodivBERT as a BERT-based model. It is pre-trained on domain-specific data (Objective 4). We explain the developed approach, pre-training, and fine-tuning task in Chapter 9. To evaluate BiodivBERT in the biodiversity domain and due to the lack of available domain-specific benchmarks, we constructed BiodivNERE, two corpora for both NER and RE [17] (Objective 6). Both benchmarks contain manually annotated statements from biodiversity abstracts and metadata that use concepts and relations from BiodivOnto. We give a detailed overview of this research area, TextI, in terms of the approaches and existing benchmarks in Section 4.2.

We investigated related approaches to understand the metadata, MI. We found inspiring methods, especially from the scholarly communication field. We give the overview of these approaches in Section 4.3. In that sense, we manually developed the BMO ontology that describes the shared vocabulary in the metadata files (Objective 6). Then, we designed and implemented Meta2KG [18, 19], an unsupervised learning approach matching fields from the semi-structured data to the BMO ontology. Then, we evaluated our approach on a manually curated set of metadata files and constructed a unified KG out of them (Objective 8). We explain the BMO ontology, our approach, and the evaluation results in Chapter 11.

Chapter 3

Background

In this chapter, we explain prior and required background for this thesis. On the one hand, we start with an overview of Knowledge Graphs (KGs) and our input data sources. We define the primary data (tabular data), textual data obtained from publications abstracts or full text, and metadata. We use ‘tabular data’, ‘table’, and ‘dataset’ interchangeably. These terms represent the main unit to be annotated from a KG (table). We explain two dimensions that describe tables: Inner-relationship and orientation dimension. Our described data sources are raw data; they lack the semantic layer. A table consists of a set of cells, each of which has plain text. The same case applies to textual and metadata; an input piece of text represents a sentence without any semantic annotation. On the other hand, we define annotation tasks for both tabular and textual data. Such tasks aim to extract subjects, predicates, and objects helping us generate the target KG. We demonstrate an example for each task that describes the change in the given raw data till reaching the semantic annotation.

3.1 Definitions

We define the building blocks for this work in the current section. Our ultimate goal is to construct a KG from heterogeneous data sources. In the following, we define what a KG, primary data source, and auxiliary data are.

3.1.1 Knowledge Graph

A knowledge graph is a graph-based model built to accumulate and convey real-world knowledge; it contains a set of nodes and edges representing entities of interest, and their relations [20, 21]. Auer et al. [1] propose them to bring scholarly communication to the 21st century. However, there is no exact definition of a KG. We adopt the inclusive definition from [20, 21] in Definition 3.1.1. A more formal definition of a KG is given by Definition 3.1.2. An individual triple is shown in Figure 3.1 where the nodes S and O represent subject and object, respectively. They refer to entities or classes in a KG. Such nodes are interlinked by relation or property P. Ideally, S, O, and P refer to real-world entities. For example, S P O could be mapped to ‘Cairo’ ‘is the capital of’ ‘Egypt’.

Definition 3.1.1 *A knowledge graph is a graph of data intended to accumulate and convey knowledge of the real world, whose nodes represent entities of interest and whose edges represent relations between these entities.*

Definition 3.1.2 *A knowledge graph is a subset of the cross product $N \times E \times N$, where N is a set of nodes (entities or classes), and E is a set of edges (relations). Each member of this set is referred to as a triple.*

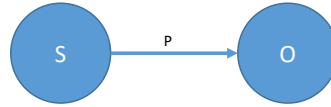


Figure 3.1: A triple representation.


Country or territory	Average Internet connection speed (2020) ^[41]	Smartphone usage (2016)	Use of renewable electricity
Hong Kong	21.8 Mbit/s	87% ^[42]	0.3%
Singapore	47.5 Mbit/s	100% ^[43]	3.3%
South Korea	59.6 Mbit/s	89%	2.1%
Taiwan	28.9 Mbit/s	78% ^[44]	4.4%

(a)

Perceived vowel	i	e	a	o	u
Vowel produced					
i	15		1		
e	1	1			
a			79	5	
o			4	15	3
u				2	2

(b)

Charles Bridge



Coordinates 50°05'11"N 14°24'43"E

Carries Pedestrian only

Crosses Vltava River

Locale Prague

Official name Karluv most

Other name(s) Stone Bridge (Kamenný most), Prague Bridge (Pražský most)

Characteristics

Design Stone

Material Bohemian sandstone

Total length 515.8 metres (1,692 ft)^[1]

Width 9.5 metres (31 ft)^[1]

Longest span 13.4 metres (44 ft)^[1]

No. of spans 16

Clearance below 13 metres (43 ft)^[1]

(c)

Figure 3.2: Inner-relationship examples. (a) Relational, (b) Matrix, and (c) Entity

3.1.2 Tabular Data

Tabular data such as CSV files are a common way to publish data and represent a precious resource [22, 23, 24]. Traditionally, they capture a lot of knowledge using free text entries. In this thesis, tables are the primary and most important data source we match into a KG. We define this data unit in Definition 3.1.3.

Definition 3.1.3 *A table is a two-dimensional arrangement of data with n rows and m columns. It enables a compact visualization for reading. A cell is the basic element of a table where $T_{ij}(0 \leq i \leq n, 0 \leq j \leq m)$ indicates the cell from row i and column j of the table T .*

Tables are highly heterogeneous in terms of structure, content, and purpose. Therefore, before interpreting a table, it is important to identify its type, so that potential specificity can be taken into account in the Semantic Table Interpretation (STI) process [4]. A table could be just layout or encapsulates a certain amount of information [25]. The former is used for visualization (layout table). However, the latter expresses a topic or thing (genuine table). We express a genuine table in two dimensions [26, 27]: 1) Inner-relationship dimension; a table could be Relational (Figure 3.2(a))¹, Matrix (Figure 3.2(b))² table, or Entity (Figure 3.2(c))³, 2) Orientation dimension considers the direction of relationships inside a table, it could be horizontal, vertical, or matrix. Entities are described row-wise in horizontal tables (Figure 3.2(a)). In the case of vertical tables, entities are described by a column as shown in Figure 3.2(c). Matrix tables cannot be interpreted row by row or column by column but rather cell by cell while simultaneously considering both horizontal and vertical headers as given by Figure 3.2(b). In our work, we interpret relational horizontal tables and transform them into a KG.

¹https://en.wikipedia.org/wiki/Four_Asian_Tigers#Technology

²https://en.wikipedia.org/wiki/Whistled_language#Lack_of_comprehension

³https://en.wikipedia.org/wiki/Charles_Bridge

3.1.3 Textual data

Text that describes tabular datasets is the first auxiliary source of information. In our context, these textual data are either the abstract or full text of associated publications. We interpret such text to identify the essential named entities and their relations. We expect that either the abstract or the full text contains a different set of named entities and relations than those we discovered in the tabular data. Thus, we enrich the constructed KG from individual tables. We define the textual data in publication (document) in Definition 3.1.4. Afterwards, our task is to extract a set of named entities (E) and their relations (R) from the given text.

Definition 3.1.4 *A document D that represents a formal publication consists of several sections S . Each section s consists of textual blocks called paragraphs P . A paragraph p contains a set of sentences Sen . Each sentence se might have a named entity or entities; we refer to the set of entities as E . Such E are interlinked with a relation(s) R .*

3.1.4 Metadata

Metadata is the second source of auxiliary data that we interpret to extract further information to enrich the resultant KG. It is usually given by an XML or JSON file that contains semi-structured information. Metadata is the primary data source in a dataset retrieval system [28]. We adapt the definition of the metadata [28] in Definition 3.1.5 where the Findability is an aspect of the FAIR (Findability, Accessibility, Interoperability, and Reusability) data principles [2]. There are notable efforts that emphasize the richness of the metadata in Life Sciences, i.g., for classification purposes [29, 28].

Definition 3.1.5 *Metadata is used to describe data in a way that enables their Findability. It includes information about the who, when, where, how, and why of data collection. It supports various applications like search and knowledge deviation.*

3.2 Annotation Tasks

In the following, we explain the core tasks we solve for each data source. Our goal is to extract the building blocks that form a triple: subject, predicate, and object. The extraction of these items differs from tabular data to textual data. The former is solved by Semantic Table Interpretation (STI). The latter is tackled by textual data understanding, i.e., a combination of Named Entity Recognition (NER) and Relation Extraction (RE). We define each category of tasks in the following.

3.2.1 Semantic Table Interpretation (STI)

We analyzed the state of the art to investigate how we could transform tabular data in, e.g., CSV format to a KG. Figure 3.3 summarizes our findings. It gives an overview of the five tasks of STI. First, **Cell to Instance** aims at linking a table cell value to a KG entity. In the shown case, ‘Egypt’ would be linked to <http://www.wikidata.org/entity/Q79> if the target KG is Wikidata or <https://dbpedia.org/page/Egypt> if the target KG is DBpedia. Second, **Column to Type** maps the entire column to a semantic type. In the example, it annotates the column to <https://www.wikidata.org/entity/Q6256> if Wikidata is the target KG, or <https://dbpedia.org/ontology/Location> or <https://dbpedia.org/ontology/Place> if DBpedia is the target KG. Third, **Property Detection** links a column pair (subject-object) with a semantic property from the target KG. Country and capital columns would be linked through <http://www.wikidata.org/entity/P1376> from Wikidata and <https://dbpedia.org/ontology/capital> from DBpedia. Fourth, **Row to Instance** maps the

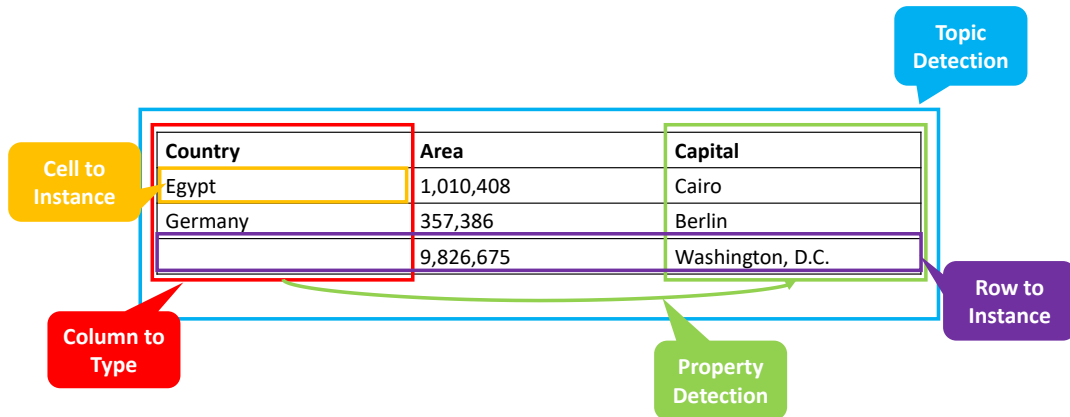


Figure 3.3: A summary of Semantic Table Interpretation (STI) tasks.

entire row to a KG entity. Its output is different from the first task since the subject column is absent in this example. In this case, row to instance would be able to detect that the entire row refers to the ‘United States of America’ <http://www.wikidata.org/entity/Q30> or https://dbpedia.org/resource/United_States from Wikidata and DBpedia respectively. Finally, **Topic Detection** classifies the entire table to a topic. Wikipedia article is a perfect source of the expected output from this task. In the given example, <https://en.wikipedia.org/wiki/Country> would be the solution.

In this thesis, we use the following notations to abbreviate the complete link of the annotated data: In Wikidata, we use `wd` and `wdt` for entities or types and properties, respectively. For example, `(wd:Q79, Egypt)` and `(wd:Q6256, Country)`, and for properties, `(wdt:P1379, Capital of)`. In DBpedia, we use `dbr` and `dbo`. The former represents the DBpedia resource, e.g., `(dbr:Egypt)`. The latter denotes the type from DBpedia ontology, e.g., `(dbo:country)`.

The Semantic Web Challenge on Tabular Data to Knowledge Graph Matching (SemTab)⁴ challenge defines three core STI tasks. It has co-hosted with the ISWC conference and the Ontology Matching (OM) workshop from 2019 to 2022. It provides a unified framework to evaluate the current STI systems. So far, SemTab, uses Wikidata, DBpedia, and Schema.org as target KGs to perform the matching tasks. Each year, from its start to 2022, it provides a series of large-scale datasets. It asks participants to solve three STI tasks for given targets without any prior knowledge about the ground truth data for each task. After the conclusion of the challenge, the organizers publish the hidden ground truth.

In this thesis, we solve SemTab’s tasks. We participated in the challenge starting from 2020 until 2022. We solved various tasks given different KGs. We developed and tested the core contributions (see Chapter 6 and Chapter 7) for tabular data understanding in this thesis in the scope of the SemTab challenge. The three STI tasks are commonly used in this thesis. Given a data table and a target KG, Cell Entity Annotation (CEA) links a cell to an entity within the KG (Figure 3.4a). Column Type Annotation (CTA) assigns a semantic type (e.g., a class) to a column (Figure 3.4b). Finally, Column Property Annotation (CPA) annotates a suitable semantic relation (predicate) from a KG to column pairs (Figure 3.4c). We give a detailed overview of the semantic table interpretation systems and the most common benchmarks in Chapter 4.

⁴<https://www.cs.ox.ac.uk/isg/challenges/sem-tab/>

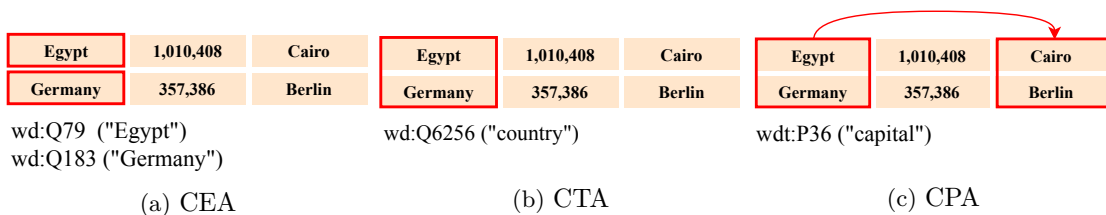


Figure 3.4: SemTab tasks summary.

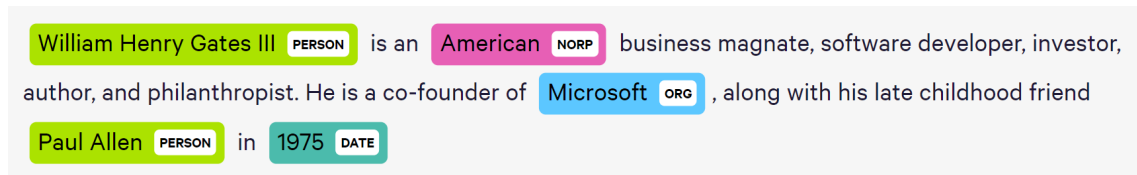


Figure 3.5: A NER example.

3.2.2 Textual Data Understanding

We identify two essential tasks for interpreting textual data in extracting and identifying structured information from unstructured text or raw sentences: Named Entity Recognition (NER) helps detect both subject and object entities from the text. Relation Extraction (RE) detects the relation between such a subject and an object. After successfully executing both tasks, we can construct a triple, the atomic unit of the desired KG. In the following, we define both tasks and give illustrative examples for each of them. However, we give a detailed overview of them in Chapter 4.

Named Entity Recognition (NER) is an essential task for most of the Natural Language Processing (NLP) processes. It allows for acquiring structured knowledge from unstructured text. We adopt the definition of it from [30, 31] in Definition 3.2.1. Entity types, classes, or tags are either derived from general domain, e.g., spaCy [32] or domain-specific types like those from ontologies [29]. Figure 3.5 shows an example of NER. We created such an example using the online visualization tool spaCy⁵ with the first sentences of Bill Gates Wikipedia article⁶.

Definition 3.2.1 *NER is the task to identify mentions of rigid designators from text belonging to predefined semantic types such as a person, location, organization, etc.*

Relation Extraction (RE) is the second core task for the information extraction processes. Similar to NER, it allows us to acquire structured knowledge from unstructured text. We identify the RE task based on these two surveys [33, 34] as in Definition 3.2.2. From the definition, RE requires the output of NER. Figure 3.6 depicts an example of RE given two identified named entities from the NER example above.

Definition 3.2.2 *RE is the task of detecting or identifying the semantic relation between entity pairs.*

⁵<https://demos.explosion.ai/displacy-ent>, 04/11/2022

⁶https://en.wikipedia.org/wiki/Bill_Gates

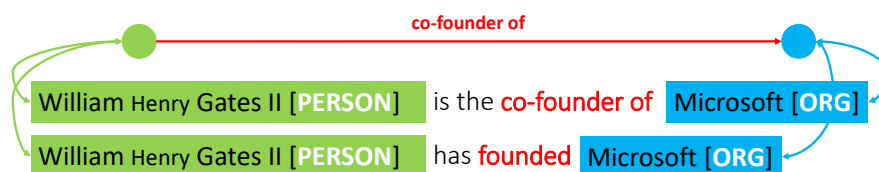


Figure 3.6: An RE example.

3.3 Summary

In this chapter, we provided the required prior knowledge for this thesis. We defined the three data sources: 1) tabular, 2) textual, and 3) metadata. In addition, we introduced the required annotation tasks that we solve for each data source to create KGs. For tabular data, we introduced five Semantic Table Interpretation (STI) tasks. In this thesis, we focus on three of them: 1) Cell Entity Annotation (CEA), 2) Column Type Annotation (CTA), and Column Property Annotation (CPA). For textual data, we explained Named Entity Recognition (NER) and Relation Extraction (RE). These tasks aim to extract entities, concepts, and their relations from a given data source and subsequently, construct a KG as a final outcome.

Chapter 4

Related Work

Building a Knowledge Graph (KG) from heterogeneous data sources touches several research areas. Since our ultimate goal is to construct a domain-specific KG from tabular data and their associated publications, abstracts and metadata, we identify three research areas that are related to our work (see Chapter 2). Thus, we divide this chapter as follows: 1) Tabular data interpretation (TabI) has Semantic Table Interpretation (STI) tasks to understand the table semantically and enable KG construction on top of it. We give an overview of the current STI approaches and the most common benchmarks used to evaluate these approaches. 2) Textual data interpretation (TexI) understands associated publications text for a given table. We discuss the essential tasks to interpret such data, and also the benchmarks that are widely used for STI-tasks. 3) Metadata interpretation (MI) covers the recent techniques that transform metadata files into a KG.

In this chapter, we discuss the approaches and datasets concerning TabI in Section 4.1. Regarding TexI, we give an overview of the related tasks and benchmarks in Section 4.2. Concerning MI, we point out similar work in other domains in Section 4.3. We summarize this chapter and related work in Section 4.4.

4.1 Tabular Data Interpretation

The recent STI contributions (systems or benchmarks) are developed and tested in the Semantic Web Challenge on Tabular Data to Knowledge Graph Matching (SemTab) challenge [22, 23, 24, 35] from 2019 to 2022 as illustrated in the survey by Liu et al [27]. Such a challenge presents a unified framework to evaluate STI systems and provides several large-scale benchmarks for such evaluation. In this section, we cover both approaches and benchmarks for Semantic Table Interpretation (STI) tasks.

4.1.1 Approaches

In the following, we give an overview of the existing approaches that solve STI tasks. We categorize them into three main categories: Heuristic methods, feature-engineering techniques, and deep-learning models. Table 4.1 summarizes these recent approaches that tackle the STI tasks in terms of supported STI tasks and target KG.

4.1.1.1 Heuristic Techniques

The category of heuristic techniques contains a wide range of approaches. Usually, a typical approach relies on candidate annotation generation via one or more lookup API, an iterative disambiguation process, and a final selection strategy. STI tasks are carried out using heuristic techniques, mostly are string similarity and majority voting, followed

Table 4.1: Summary of the current state-of-the-art STI approaches.

Category	Approach	CEA	CTA	CPA	R2I	KG	Released
Heuristic	MTab	Yes	Yes	Yes		Wikidata	2019-2021
	DAGOBAB	Yes	Yes	Yes		Wikidata, DBpedia	2020-2021
	bbw	Yes	Yes	Yes		Wikidata	2020
	CSV2KG	Yes	Yes	Yes		DBpedia	2019
	Tabularisi	Yes	Yes	Yes		DBpedia	2019
	MantisTable	Yes	Yes	Yes		DBpedia, Wikidata	2019-2021
	ADOG	Yes	Yes	Yes		DBpedia	2019
	LinkingPark	Yes	Yes	Yes		Wikidata	2020
	SSL	Yes	Yes	Yes		Wikidata	2020
	Magic	Yes	Yes	Yes		DBpedia, Wikidata	2021
AMALGAM	Yes	Yes			Wikidata	2020	
Feature-Engineering	Limaye	Yes	Yes	Yes		YAGO	2010
	Neumaier et.al		Yes			DBpedia	2016
Deep Learning	Efthymiou et. al	Yes			Yes	Wikidata	2017
	DAGOBAB-DL	Yes	Yes			DBpedia	2019
	ColNet		Yes			DBpedia	2019
	Chen et. al		Yes			DBpedia	2019
	Sherlock		Yes			DBpedia	2019
	Turl	Yes	Yes	Yes		DBpedia	2021

by TF-IDF and probabilistic frameworks. In the following, we walk through heuristic approaches used by the state of the art.

MTab [36, 37, 38] solves the three STI tasks jointly. The authors explicitly mention clear assumptions on what kind of table they solve, for example, the target KG is complete and correct, and input tables are independent, which means there is no sharing of information between input tables. MTab depends on a joint probability distribution for the three tasks. The authors introduce a pipeline of seven steps. The lookup part is built on top of multiple lookup services. It is started with DBpedia lookup and endpoint in 2019. In 2020, the authors identified weaknesses from the previous year; they changed their lookup module to support both fuzzy entity and statement search. They supported Wikidata as a target KG, including the entity dump and all history revisions in a given time interval to generate as precise candidates as possible. MTab is computationally expensive, and the author has to disable various parts to be able to release it to the public in 2021¹. MTab is SemTab first place winner in 2019 and 2020 in the Accuracy Track, and they had the first place prize in the Usability Track in 2021.

DAGOBAB [39, 40] constructs an annotation workflow to solve the three STI tasks. In the first step, the authors perform an entity lookup from Wikidata dump where such dump is stored in the Hadoop distributed file system, using Wikidata toolkit² and PySpark³ framework for candidate annotation retrieval using Levenshtein distance for similarity measurement. In the second step, they follow a candidate scoring technique. A confidence score is calculated as a mixture of a context score using Levenshtein distance and SMAPE for literal values and numbers, respectively, and a weighted similarity score for literals. Then, they continue with the CPA annotations. The authors retrieve all entity candidates and select the most co-occurred property, a straightforward majority voting technique. Afterwards, the authors solve the CEA task by selecting a final candidate with maximum

¹<https://mtab.app/mtabes/docs>

²<https://github.com/Wikidata/Wikidata-Toolkit>

³<https://spark.apache.org/docs/latest/api/python/>

CPA score support. Finally, they come up with CTA solutions. The same procedure for selecting the CEA, CTA candidate is selected by a majority vote of CEA in the same column. In addition, the authors take into account three levels of hierarchical types where they prefer the direct types in most cases. DAGOBAB required 250 machines to run its 2020 configurations, although the authors managed to reduce it to 30 machines in 2021. Moreover, it is the third place winner at SemTab 2020 and a first place winner at SemTab 2021.

bbw [41] solves the three STI tasks. *bbw* consists of a seven-step pipeline; we summarize its steps into two core ideas. First, for candidate generation, the authors use a locally deployed SearX⁴ as a meta-lookup which enables search over more than 80 engines. The authors do not use any Wikidata dump. This module collects the lookup results and ranks them automatically. Second, the authors rely on the Wikidata SPARQL endpoint for contextual matching using two features: entity and property labels. Then, the authors pick the best matches using edit distance. For the CTA solutions, the authors select the type through `wdt:P31 (instanceOf)` without any further exploration of multi-hop hierarchical types, achieving the best score on CTA with 98% F1-score during the challenge. *bbw* won the third rank of SemTab 2020.

CSV2KG [42] addresses the three tasks of STI. The authors follow an iterative process with the following steps: (i) get entity matchings using lookup services; the authors adopt several services on the DBpedia; DBpedia lookup service, DBpedia Spotlight, and DBpedia resource. Each service returns a ranked list of candidates. On this large pool of candidates, they apply a disambiguation step by selecting the candidate with an `rdfs:label` has the lowest edit distance with the cell value. (ii) infer the column types and relations; the majority vote is the used strategy of the final selection. The authors reported some inaccuracies on the DBpedia level, e.g., Barack Obama and not Donald Trump is the president of USA⁵. (iii) refine cell mappings with the inferred column types and relations. (iv) refine subject cells using the remaining cells of the row. Finally, (v) re-calculate the column type with all the corrected annotations. CSV2KG is the second-place winner at SemTab 2019.

Tabularisi [43] solves the three STI tasks. The authors use lookup services to generate CEA candidates. For each candidate, an adapted TF-IDF⁶ score is calculated. An entity candidate is represented by a binary feature vector in which each feature is an indicator (1 if present, else 0) of a property used to describe the entity (e.g., `instanceOf`). Different features have various expressiveness. They are thus weighted by the TD-IDF technique. Specifically, the ‘Term Frequency’ of a feature is the number of cells whose first entity candidate has this feature, and the ‘Document Frequency’ is the total occurrences of this feature in all entity candidates of all cells. The score of an entity candidate is a weighted combination of its TD-IDF score, the Levenshtein distance between cell value and candidate label, and a distance measure between cell value and the URL tokens is used to determine the final annotation. The CTA solutions are obtained by a top-down, brute-force search in the KG class hierarchy tree. Finally, the CPA solutions are determined using a row-by-row majority voting technique.

MantisTable [44] tackles the three set of the STI tasks. The authors introduce a pipeline that starts with classifying each column into three types: Named entity, Literal, and

⁴<https://github.com/searx/searx>

⁵Checked on Late Feb 2020

⁶<https://en.wikipedia.org/wiki/Tf-idf>

Subject. The candidate generation of this approach is based on SPARQL queries which extract all candidates containing the cell mentions. Then, the system handles the CEA annotations by row-wise compatibility analysis and CPA by majority voting. For the CTA task, the authors list all candidate types in addition to their number of occurrences in the table (row coverage). After threshold filtering, the rest of the type candidates are transformed into a graph according to the ontology hierarchy. The type scores then are updated with the distance to the root, the highest score representing the most accurate and specific annotation at the end, and then picked as a solution.

MantisTable SE [45, 46] the authors optimized the system by updating the scoring function, accessing the LamAPI⁷ API (instead of using a SPARQL endpoint) and adding a final disambiguation step. LamAPI is considered an efficient way to retrieve all necessary data for the three tasks independently from the target KG.

MantisTable V [47] is an even more optimized version that still relies on LamAPI and applies complex string similarity functions for the entity generation task.

ADOG [48] solves the three STI tasks. It considers scores combined with string similarities, frequencies of properties, and the normalized Elasticsearch score from each match from DBpedia for the CEA task. The system weighs these scores with TF-IDF score for types. To be able to compute the Levenshtein distance and TF-IDF, the authors use ArangoDB⁸ to load DBpedia dump and index its components

LinkingPark [49] tackles the three STI tasks. Their framework consists of three main modules: i) entity linker to generate CEA. ii) type inference to get CTA. Finally, iii) property linker to obtain the possible relations. The entity linker contains two sub-modules for entity generation and disambiguation. The authors use regular mention through Access-MediaWiki API search, 1-edit distance typo via a spelling mistake corrector, and other mentions that are used to build a fine-grained Elasticsearch index as a cascaded pipeline for entity generation. For entity disambiguation, the authors follow a coarse-to-fine-grained disambiguation. They rely on an iterative classification algorithm to conduct an approximate inference of an entity. Afterwards, they would able to characterize both types' consistency (coarse) and select the discriminative value. The type inference retrieves the types of the generated CEA candidates. Then it uses the majority vote to select a final CTA candidate. To break the ties, the authors use a minimum average level score that is defined and based on the underlying ontology structure of the target KG. In the property linker, the author support object properties matching by the entity property linker sub-module and literals matching using the either perfect or fuzzy matching technique. For the object properties, the author obtains the relations row-by-row and finally, selects the most co-occured one as a final selection. The same procedure is applied for the lexical property match but with an error tolerance to allow fuzzy matching. LinkingPark is the second-place winner at SemTab 2020.

SSL [50] generates a Wikidata subgraph over a table using a four-stage pipeline to solve the three STI tasks. At first, it leverages advanced SPARQL queries for the three tasks. Then, it selects the subject with the highest probability value to update the resultant graph. Afterwards, the authors apply a crawling process through Google search engine to suggest better words for not found subjects and repeat the first two steps. Thus, the authors overcome the problem of spelling mistakes. Finally, for literal properties matching, the authors tolerate $\pm 1.5\%$ of the numerical values.

⁷<https://bitbucket.org/disco-unimib/lamapi>

⁸<https://www.arangodb.com/>

Magic [51] addresses the three STI tasks. It adopts the approach of generating comparison matrices, namely INK embeddings, to speed up computational efficiency. INK embeddings are representations of attributes and values for an entity or table context for a cell mention. The complete comparison matrix is generated by fusing multiple candidates. The system annotates CEA by measuring the compatibility between INK embeddings from KG and the input table. For annotations, the authors focus on the key column. They do the lookup using public endpoints of the target KG for each cell in the key column, then leverage its neighborhood to find the candidates for surrounding cells in the same row (they avoid performing the lookup on the whole table due to the limitation of public API usage). Misspellings might be challenging for Magic, and detecting synonyms of attributes as well. Nonetheless, INK embeddings improve computational efficiency and provide a way to implement column integration.

AMALGAM [52] covers both CEA and CTA only using a three-step pipeline. The authors rely on Wikidata lookup services with a focus on spelling mistakes handling. They use the row context where all entities belong to the same thing as the only annotation context. This gives a reason why AMALGAM is efficient and less computationally expensive. Results show a maximum score of 92% for both tasks.

4.1.1.2 Feature-Engineering Techniques

This category relies on the hand-crafted feature vectors that represent a table. For example, the authors of this category extract statistical and lexical features. Such as the distribution of numerical values, the occurrence of cell mentions, and textual similarity among table rows and columns. Then, they use such features with machine learning models like SVM, Random Forest, and KNN algorithms. Since these are all supervised learning techniques, a labeled dataset is required for the training. The amount and quality of training data, and consequently the quality of input features, significantly impact the model performance. We notice that those techniques suites the CTA task more than the others, since columns are more subject to statistical features than other STI tasks.

Limaye [53] introduce one of the earliest work on STI that tackles the the three tasks. The approach collects the TF-IDF cosine similarity between cell mention and entity label and the compatibility between the cell type and the column type to execute the CEA task. CTA task depends on TF-IDF cosine similarity between column header and entity label in the corresponding KG. The CPA annotation depends on the compatibility between the relation and column pairs. All these features are weighted through a machine learning framework.

Neumaier et.al [54] focus on CTA labeling for numerical columns. Their work is not limited to labeling a unique prediction but expands the scope of labeling to the surrounding information. For example, instead of labeling ‘height’, this system will label it as ‘the height of an athlete playing basketball in the NBA’. To achieve this, the authors proposed three steps pipeline: i) build Background Knowledge Graph (BKG) based on DBpedia where its nodes consist of typical numerical values, annotated with context information. For example, grouped by properties and their shared domain (subject) pairs. Such BKG contains the hierarchical structure and is divided into multiple multi-level groups to provide context. ii) to search for mappings using k-nearest neighbours (kNN) for making predictions. Finally, iii) to aggregate the results at different levels of the BKG to find the most likely context in terms of properties and types. The authors also explore the system’s performance at different hierarchy levels for the BKG built on DBpedia and

Open Data. They pointed out that DBpedia still has limitations in terms of data coverage and freshness when compared with other open datasets. For example, the Austrian Open Data Portal has tables generated by weather stations every 15 minutes. Comparatively, DBpedia typically has numeric values only for ‘current’ or ‘latest’.

4.1.1.3 Deep Learning Techniques

Deep Learning has achieved many successes in various domains thanks to the availability of massive amount of data and powerful computing resources. It has attracted the attention of the STI community over the past few years. KG embedding techniques represent one direction beyond this work’s scope. However, in the following, we show some examples that leverage deep learning as a direct NLP model in the scope of STI.

Efthymiou et. al [55] provide different methods for entity linking, normal cell to an entity annotation (CEA), as well as mapping the entire row to instance (R2I). Such a system assumes that a column’s correct CEA candidates should be semantically close. From this assumption, a weighted correlation subgraph in which node represents a CEA candidate is constructed. The edges are weighted by the cosine similarity between two related nodes. The best candidates are the ones whose accumulated weights over all incoming and outgoing edges are the highest. In addition, a hybrid system, as a combination of a correlation subgraph method and an ontology matching system, is also introduced, which achieves a significant improvement in the final results.

DAGOBDAH-DL [56] Solves the three STI tasks The author assumes that entities in the same column are close in the embedding space. Candidates are first retrieved using a lookup based on regular expressions and the Levenshtein distance. Then, the author converts the retrieved entities into vector space. The selection of the candidate is made on several steps starting from the clustering formulation (rows coverage-wise), The author applies a K-means clustering that is performed using TransE’s pre-trained embedding to cluster the entity candidates. Then they give each cluster a score, pick the one with the highest score, and rank its candidates. The final ambiguity is resolved via a confidence score based on the row context of the candidates.

ColNet [57] tackles only the CTA task. The authors use a Convolutional Neural Network (CNN) trained by classes contained within a KG. The predicted annotations are combined with the results of a traditional KG. The final annotation is selected using a score that selects the lookup solutions with high confidence and otherwise resorts to the CNN predictions. Results have shown that CNN prediction outperforms the lookup service for a larger knowledge gap. In addition, the authors explored the idea of ensembling the predictions from ColNet and those from the lookup service. Such an ensemble strategy gave the best scores since it benefits from both techniques. Afterwards, the authors extended the entity representation by considering other cells in the same row (row context). Thus they create a property feature vector, a.k.a, Property to Vector (P2Vec). Such P2Vec is an additional signal to the neural network, which yields better results [58].

Sherlock [59] Learns the CTA task using 1588 features extracted from a single column of a given relational table. The features are divided into four categories: i) character-wise statistics (e.g., frequency of the character ‘c’). ii) column statistics (e.g., mean, std of numerical values). iii) word embedding, and iv) paragraph embedding. Except for the column statistics feature, other features are compressed into a fixed embedding size

Table 4.2: Annotation method, data sources for existing STI benchmarks and their corresponding targets and release year. Entries for SemTab are aggregated over all rounds each.

Method	Dataset	Data Source	Target Annotation	Released
Automatic	SemTab 2019	Wikidata, Wikipedia	DBpedia	2019
	SemTab 2020	Wikidata, Wikipedia	Wikidata	2020
	SemTab 2021	Wikidata, Wikipedia	Wikidata, DBpedia	2021
	GitTables	GitHub	DBpedia, Schema.org	2021
Manual	Limaye	Wikipedia, HTML Tables	YAGO, DBpedia	2010
	T2Dv2	WebTables	DBpedia	2016
	2T	WebTables, Other	Wikidata	2020

using a subnetwork. A two-fully connected layer network is trained on both the embedding features and column statistics feature to predict a column type annotation among 75 types inherited from T2Dv2 [60] dataset. The evaluation shows a limited result on various column types, including Dates and Industry. However, it is less sensitive to purely numerical values or values appearing in multiple classes.

Turl [61] pioneers the application of pre-trained language models such as BERT in the STI tasks. It provides a universal contextualized representation for each table element (e.g., caption, header, content cells), which can be fine-tuned and applied in various downstream tasks such as CEA, CTA, and CPA. The model employs a Transformer-based encoder [62] to capture the information from table elements. For this goal, the input table is first serialized into a sequence of caption tokens, title tokens, header tokens, and row-by-row cells. A cell consists of its content (mention) and a candidate entity representing it in a KG. The sequence of tokens is then converted into embedding using a word embedding for textual tokens and KG embedding for entity tokens. To reduce the redundancy in the fully connected attention learning and better draw the inter and intra-column, inter-and intra-row, and column-row interaction, the conventional attention layer is masked by a so-called visibility matrix which allows only a portion of table elements to participate in the modeling of a specific element. For example, cells in the same row or the same column can interact with each other. Apart from the BERT’s Masked Language Model objective, TURL introduces an additional Masked Entity Recovery objective to reinforce the learning of factual knowledge embedded in the table and represented by KG entities. The model is pre-trained on 570K unlabeled Wikipedia tables.

4.1.2 Benchmarks

In the following, we show the current and the most used benchmarks as STI tasks. Table 4.2 describes the existing benchmarks in terms of their original data source, target KG, and released year. Table 4.3 demonstrates the statistics of the current benchmarks. It shows their number of tables, average rows, columns, and cells, and their coverage for STI tasks.

Limaye [53] is one of the earliest gold standard used for STI tasks. It aims to annotate web tables using the YAGO KG. The dataset is divided into four subsets according to the data source, the labeling method, and application scenarios. Three subsets are manually labeled, while the fourth one is automatically generated. Altogether, constructs the final benchmark with 428 annotated tables. Annotation errors were reported for the automatically labeled subset [63], which were corrected by Bhagavatula et al. [64] in 2015. Later on, in 2017, Efthymiou et al. [55] adapted the disambiguation links to the DBpedia KG.

Table 4.3: Summary of existing STI benchmarks. *ST19 - ST21* (SemTab editions). *ST21-R1_W* and *ST21-R1_D* use Wikidata (W) and DBpedia (D) as targets. *ST21-H2*, and *H3* are HardTables for Round 2 and 3 during SemTab. *ST21-Bio* is BioTables at SemTab Round 2. *ST21-Git* is the published version of GitTables during SemTab Round 3.

Dataset	Tables	Avg. Rows (\pm Std Dev.)	Avg. Cols (\pm Std Dev.)	Avg. Cells (\pm Std Dev.)	CEA	CTA	CPA
ST19-R1	64	142 \pm 139	5 \pm 2	696 \pm 715	8,418	120	116
ST19-R2	11,924	25 \pm 52	5 \pm 3	124 \pm 281	463,796	14,780	6,762
ST19-R3	2,161	71 \pm 58	5 \pm 1	313 \pm 262	406,827	5,752	7,575
ST19-R4	817	63 \pm 52	4 \pm 1	268 \pm 223	107,352	1,732	2,747
ST20-R1	34,294	7 \pm 4	5 \pm 1	36 \pm 20	985,110	34,294	135,774
ST20-R2	12,173	7 \pm 7	5 \pm 1	36 \pm 18	283,446	26,726	43,753
ST20-R3	62,614	7 \pm 5	4 \pm 1	23 \pm 18	768,324	97,585	166,633
ST20-R4	22,390	109 \pm 11,120	4 \pm 1	342 \pm 33,362	1,662,164	32,461	56,475
ST21-R1_W	180	1,080 \pm 2,798	5 \pm 2	4125 \pm 10947	663,655	539	NA
ST21-R1_D	180	1,080 \pm 2,798	4 \pm 2	3,952 \pm 10,129	636,185	535	NA
ST21-H2	1,750	17 \pm 8	3 \pm 1	55 \pm 32	47,439	2,190	3,835
ST21-Bio	110	2,448 \pm 193	6 \pm 1	14,605 \pm 2,338	1,391,324	656	546
ST21-H3	7,207	8 \pm 5	2 \pm 1	20 \pm 15	58,948	7,206	10,694
ST21-Git	1,101	58 \pm 95	16 \pm 12	690 \pm 1,159	NA	2,516	NA
ST21-Git	1,101	58 \pm 95	16 \pm 12	690 \pm 1,159	NA	720	NA
2T	180	1,080 \pm 2,798	5 \pm 2	4125 \pm 10947	663,655	539	NA
T2Dv2	779	85 \pm 270	5 \pm 3	359 \pm 882	NA	237	NA
Limaye	428	24 \pm 22	2 \pm 1	51 \pm 50	NA	84	NA

T2Dv2 [60] is the recent edition of T2D [65] gold standard where annotation errors are fixed. It is widely used by STI systems like [53] and others. Together with Limaye, and before 2019, to the best of our knowledge, were the only benchmarks that are used by STI systems. T2Dv2 covers the tasks of row-to-instance (R2I), attribute-to-property that maps to our definition of CPA, and table-to-class for 779 tables that are derived from WebTables [66] where the target KG is DBpedia. In addition, T2Dv2 provides extensive metadata, such as the context of the table and whether the table has a header.

SemTab2019 [67] is the first benchmark that is introduced by SemTab challenge. It is an automatically generated dataset from Wikidata and Wikipedia with DBpedia as a target KG. Such a benchmark consists of four folds representing the rounds of the challenge. In total, it consists of 15k tables annotated for the three STI tasks. The authors introduced a data generator that consists of three steps: 1) Profiling, where it outputs a list of classes with their number of instances. Then, the number of instances that have a value for the property, the datatype for datatype properties, and the range class for object properties. Such kind of information are used in 2) Raw table generation by using the SPARQL endpoint of the target KG. Finally, 3) Table refinement, where the authors add artificial noise like spelling mistakes to cell values.

SemTab2020 [68] is the used name in this section to refer to the automatically generated dataset by the SemTab challenge during its second edition in 2020. It has four folds, each of which is released during each round of the challenge. In total, it consists of more than 31k tables that are annotated from Wikidata. The common data issues in such benchmarks are misspellings and ambiguity among table rows. SemTab2020 also focuses on testing the ability of the matching algorithm to scale due to its high number of tables. The first two editions of SemTab benchmarks cover the three tasks of STI.

ToughTables (2T) [69] is a set of 180 tables that are annotated from Wikidata. The first use of it is during SemTab 2020 fourth round. It focuses on the ambiguity among entity mentions in a way that makes it hard to disambiguate by a human expert. The authors did not rely on the automatic generation of the dataset only but also provided manual curation of such annotation to avoid false positives while evaluating a matching algorithm. It contains real tables that reflect a knowledge gap between the target KG and an input table. Misspellings are frequent and intense, which is useful for testing the weight of lexical features an algorithm could use. In addition, a large number of rows is used to evaluate the system’s performance. Such kind of unique features distinguish 2T from SemTab2020’s automatically generated benchmarks and make it hard to solve by a matching technique.

SemTab2021 [70] is the benchmark that was introduced by SemTab’s third edition in 2021. It has three folds that represent the rounds of the challenge. In the first fold, the authors reused the 2T dataset and created another version that is annotated from DBpedia (see Table 4.3 ST21_R1_W, ST21_R1_D respectively). The other folds consist of so-called, HardTables, which focus on the misspellings and ambiguities like the first two editions of SemTab benchmarks. In addition, a special fold during the third round is also presented. This subset is derived from the biomedical domain. Its unique characteristic is that it contain some columns with too long descriptions from Wikidata. Those characteristics are for testing matching algorithm performance.

GitTables [71] is a subset of [72] that is used during SemTab 2021 third round. It is a collection of 1,101 tables crawled from GitHub. They are annotated from DBpedia and schema.org for only CTA task. However, we analyzed the provided CTA targets; we found that the first column has only a valid CTA type while the rest are either object or data properties. We mean by valid CTA type as an actual semantic class; thus, it follows the definition of CTA task. GitTables has a sparse table structure, e.g., it might have a target asking for a CTA of an empty column.

4.2 Textual Data Interpretation

In this section, we cover two core tasks for text interpretation. At first, Named Entity Recognition (NER) to extract proper entities from the text; those would act as KG nodes. Second, Relation Extraction (RE) to detect and classify the relations among these entities that would be the named edges of the desired KG. In addition, we highlight a few examples of the domain-specific BERT-based models that typically solve such tasks.

4.2.1 Named Entity Recognition (NER)

To the best of our knowledge, there are two surveys for NER task. On the one hand, the classical survey by Nandau et al. [31] in 2007. On the other hand, the deep learning-based techniques by Li et al. [30] in 2020. Both cover all aspects for NER tasks. Based on their classification, we can categorize NER approaches into 1) feature-engineering methods and 2) deep learning-based techniques. In the first category, authors manually craft the feature vector to detect the entities of interest. Such approaches are time-consuming and have limited capabilities of generalization.

In the second category, for example, BioNER [73] uses deep neural networks to solve the NER task. In such a family of techniques, authors rely on the network architecture to learn the feature vector in an end-to-end manner. The cutting-edge BERT [74] models

outperform the state-of-the-art approaches in such a task. In the below section, we give an overview of these domain-specific BERT-based models.

There is a wide range of evaluation benchmarks for NER task. Most of them are from the general domain. For example, CoNLL03-05 [75, 76] is a four tags annotated benchmark based on Reuters news. In addition, OntoNotes [77] is an annotated corpus collected from magazines, news, web with 18 classes. However, in our work, we are interested in biodiversity-specific benchmarks. We give an overview of them in the following.

COPIOUS [78] is a gold standard corpus covering a wide range of biodiversity entities. It has 668 documents downloaded from the Biodiversity Heritage Library (BHL) with over 26K sentences and more than 28K entities. Only two annotators manually annotated the corpus with five categories of entities as follows: taxon names, geographical locations, habitats, temporal expressions, and person names. The proposed gold standard has been developed to support the development of NER and RE using two different machine-learning techniques. However, NER corpus is the only available one.

Species-800 [79] is based on 800 PubMed abstracts, such that each 100 is from one of eight categories: bacteriology, botany, entomology, medicine, mycology, protistology, virology, and zoology. Similar to COPIOUS, the Species-800 corpus is annotated with taxon entities and normalized to the NCBI Taxonomy database⁹.

LINNAEUS [80] is a set of 100 full-text document from the PubMed Central Open Access (PMC OA) document set that was randomly selected and annotated for species mentions. The corpus was only annotated for species (except for the cases where genus names were incorrectly used when referring to species). As in the case with COPIOUS and Species-800, all mentions of species terms were manually annotated and normalized to the NCBI taxonomy IDs of the intended species, except for terms where the author did not refer to the species.

4.2.2 Relation Extraction (RE)

One of the most used techniques for RE is distant supervision that appeared in two well-known surveys [33, 34]. Distant supervision aims at reducing the manual effort of labeling training data by exploiting a third-party source to generate such data, e.g., KGs. However, the provided evaluation benchmarks are typical of the general domain, and adapting it to a domain-specific task, e.g., biodiversity, is challenging.

EU-ADR [81] has been annotated for drugs, disorders, genes, targets (Target: genes, proteins, and sequence variants of genes and proteins), and their inter-relationships. For each of the drug-disorder, drug-target, and target-disorder relations, three experts have annotated a set of 100 PubMed abstracts, three levels of relationship: positive and negative associations, speculative associations (more focus on binary relations - exist/ non-existence, hypothesis).

GAD [82] is a successor development of EU-ADR (same researchers). The authors used BeFree system, a text mining system, to extract relations between genes and diseases, drugs and diseases. In the evaluation of RE systems for gene-disease associations, the original GAD corpus does not contain locations where the entities exist. Afterwards, 'Location' was added with an NLP tool. Then curators manually added the relationship, only binary (true/false) relations. The original EU-ADR and GAD data are unavailable.

⁹<https://www.ncbi.nlm.nih.gov/taxonomy>

In this thesis, we used a published pre-processed version¹⁰ for evaluating our textual data interpreter.

BioRelEX [83] contains 2010 sentences from biomedical journals, 33 types of entities such as proteins, genes, chemicals, and processes. In addition, it contains various sub-entity types such as protein-complex and protein-domain. For relation, it includes only binary relation annotation (1- binding exists, 0 - not sure, -1 - binding does not exist), It considered nested entities as well. There is no information on how many annotators were involved. The data is available¹¹.

COPIOUS [78] is generated by using PASMED—a pattern-based system that can identify any binary relations between entities within a single sentence. It supports four relations: Taxon *occur* Habitat, Taxon *occur* Temporal Expression, Taxon *occur* Geographic Location, Taxon *seen by* Person. This data is not available.

4.2.3 BERT-based Models

First, we give an overview of the original BERT [74] model. It is a contextualized word representation that is based on a masked language model. BERT is pre-trained using bidirectional transformers [62]. Due to the nature of language modeling, where future words cannot be seen, previous language models were limited to a combination of two unidirectional language models (i.e., left-to-right and right-to-left). BERT uses a masked language model that predicts randomly masked words in a sequence and hence can be used for learning bidirectional representations. A standard procedure after pre-training is fine-tuning a BERT model with minimal architecture change. BERT obtains state-of-the-art performance on most NLP tasks. BERT is pre-trained on massive corpora that are derived from the general domain. However, applying it directly to domain-specific datasets showed limited performance and opened the demand for domain-specific BERT-based models. In the following, we give an overview of the domain-specific BERT-based models. BERT-based techniques usually solve the two tasks jointly. We present those domain-specific BERT-based models from the biomedical domain as follows. We compare them in terms of the pre-training data and downstream tasks they support. BioBERT [8] is a pioneer domain-specific BERT model in the biomedical domain. It is initialized by the original BERT's weights. BioBERT is pre-trained on biomedical corpora from PubMed and PubMed Central (PMC). In addition, BioBERT is fine-tuned on three downstream tasks for NER, RE, and Question Answering (QA). The results showed notable improvements in the obtained scores on the task-specific datasets. ClinicalBERT [84] is another example from the biomedical domain. It is pre-trained on around 2 million clinical notes, and unlike BioBERT, ClinicalBERT is fine-tuned for NER task only.

4.3 Metadata Interpretation

In this section, we give an overview of the recent work that translates various types of metadata into a KG. During our literature review, we found that scholarly metadata has witnessed various efforts transforming them into KG. Thus, we focus on such a domain.

¹⁰<https://github.com/dmis-lab/biobert>

¹¹<https://github.com/YerevaNN/BioRelEx/releases>

SCM-KG [85] integrate scholarly communication metadata into a KG that is called SCM-KG from two different sources DBLP¹² and Microsoft Academic Graph (MAG)¹³. Their motivating example is the disambiguation of person entities that represent authors. Such entities included a list of publication IDs as a disambiguation property. The authors claimed the completeness of SCM-KG since each data source covers different aspects. For example, DBLP has a complete listing for authors and publications. However, MAG has more keywords and abstracts. The authors introduced a pipeline that consists of 1) two manual steps concerning data acquisition and pre-processing. 2) three automatic steps, including ontology matching using rule-based techniques, similarity measurement, and instance linking. The authors deal with various data sources like CSVs, PDFs, and structured databases. Such heterogenous input may use different schemas. e.g., DBLP and MAG model the same concepts (e.g., affiliation) differently. Thus, the authors involved a mapping step to create their target unified graph through an ontology engineering phase. They used subsets from Dublin Core and FOAF ontology, and they created missing vocabulary themselves. They provided an entity linking step to their pipeline for ontology matching via a Jaccard similarity. They used the common title and the publication year, if provided, to match the instances to the ontology.

EVENTS [86] introduced a dataset for top-tier conferences in the computer science field, e.g., ISWC, ESWC, CVPR. It encapsulates scientific events in terms of historical data about the publications, submissions, start date, end date, location, and homepage for 25 top-prestigious event series (718 editions in total) in five computer science communities. The authors manually collected and analyzed the metadata (raw data) since 1990 of these conferences from different sources like DBLP, and ACM Digital Libraries¹⁴. Then, they applied a pre-processing data phase where they aimed to fill in the missing data, identify and correct incorrect data, and remove irrelevant information. Thus, four tasks are involved in this phase: data integration, data cleansing, data transformation, and event name unification. Then, the authors analyzed the collected metadata of the events in terms of, e.g., the h5 index, average acceptance rate, and the number of editions of each event. The primary use case of such work is the Question Answering (QA). The dataset is publicly available online in three formats (CSV, XML, and RDF).

EVENTSKG [87, 88] is the successor of the previous work. The authors released their dataset as a unified KG instead of individual RDF dumps, including data for more computer science communities. I.e., EVENTSKG is a knowledge graph that contains metadata of top-40 prestigious events series. Like EVENTS, the main goal of EVENTSKG is to facilitate the analysis of events metadata by enabling them to be queried using semantic query languages like SPARQL. This work relies on the Scientific Event Ontology (SEO) [89] as a data model. It is a reference ontology for modeling data about scientific events such as conferences, symposiums, and workshops. SEO reuses several well-designed ontologies, such as FOAF and Dublin Core. It defines its own vocabulary if missing from the existing ontologies. Two steps are included to enhance their previous pipeline. On the one hand, for the linked data generation, where the authors developed an RDefer, a Java tool to convert input data from CSV to linked data (RDF/XML syntax). The authors include two types of validations of the generated RDF, syntactic and semantic validation. The former validation is done through the RDF validation service¹⁵. The latter is applied via Protégé reasoners. On the other hand, the linked data enrichment (LDE) is included

¹²<https://dblp.org/>

¹³<https://www.microsoft.com/en-us/research/project/microsoft-academic-graph/>

¹⁴<https://dl.acm.org/>

¹⁵<https://www.w3.org/RDF/Validator/>

to infer the interlinking relationships between RDF triples using inference engines, i.e., reasoners. The input of LDE is the RDF triples produced by the Linked Data generation. The output is a set of consistent RDF triples, including the newly discovered relationships, where available.

Schröder et al. [90] File names are considered metadata for files that have a minimal context. Despite the unusual source to create a KG, the authors managed to create a Personal Knowledge Graph (PKG) from file names as the only data source used in a semi-automatic approach. A user that is defined as a knowledge engineer is responsible for creating the RDF triples. However, an active learning technique aids the knowledge engineer by suggesting entity types. The authors used rule-based techniques to extract terminologies of interest. They followed several steps to unify the extracted entities and populate the ontology. Then, they conducted taxonomic and non-taxonomic relations using language resources. The authors used Jaccard and Embedding-based similarities, for instance, matching, and type matching, respectively.

4.4 Summary

In this chapter, we explained three sub-areas related to our work. Figure 4.1 shows a summarization of these works for each research area. At first, we gave a detailed overview of the approaches that tackle the Semantic Table Interpretation (STI) tasks and the most common benchmarks used to evaluate such approaches. Based on the current state of the art, such STI approaches are resource hungry and require massive data to generate semantic mappings for a given table from a target KG. From the benchmark perspectives, the existing ones are AG and derived from the general domain. Second, we showed the core tasks for textual data understanding with their common approaches and benchmarks in the domain-specific and general domains. We can argue that the only available domain-specific resources are for the biomedical domain. Due to their limited performance, we cannot rely on the original general domain models for domain-specific tasks. Finally, we discussed the common recent approaches that transform metadata into a KG. Most of the introduced works are developed in the scope of the scholarly communication field. They are manual approaches with the aid of semi-automatic techniques for the entity linking step only.

We see these open areas that require a solution to fulfill our ultimate goal of having a domain-specific KG for the biodiversity domain: 1) An approach for STI that rely on minimal data sources for high usability and to suit real-world scenarios. 2) A semantic model that describes the core concepts and relations of the biodiversity domain 3) Biodiversity-specific models that can interpret textual data and metadata with high quality. 4) Biodiversity-specific evaluation benchmarks for these tools.

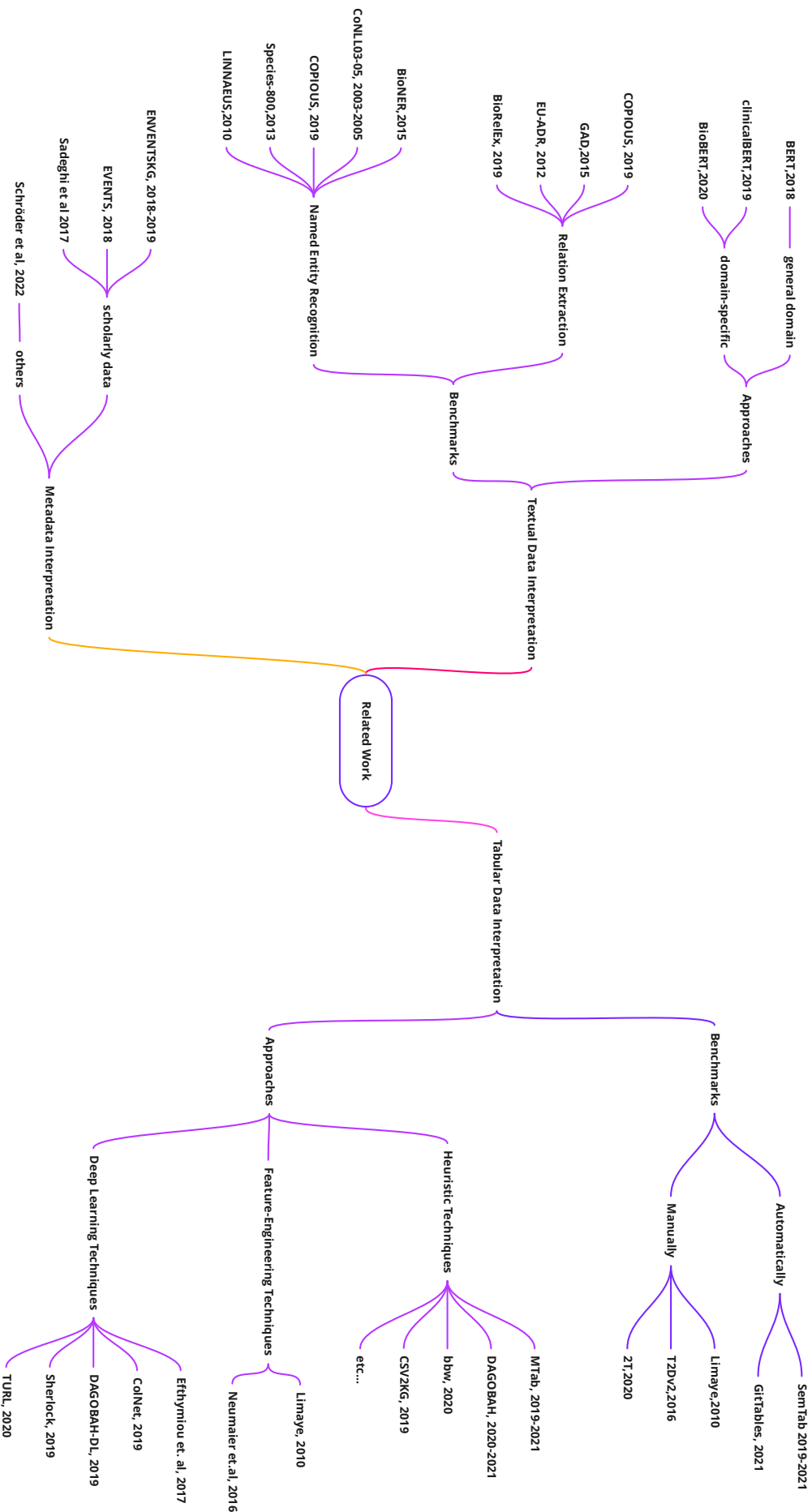


Figure 4.1: Related work of the three subareas overview.

Chapter 5

Solution Overview

Three research areas are involved in this work; Tabular data interpretation (TabI), Textual data interpretation (TexI), and Metadata interpretation (MI). We gave their definitions in Chapter 2. To facilitate the explanation of each contribution pillar in this thesis, Figure 5.1 shows the schema of each of them. This a figure gives an abstract overview of a research project: i) a **Data Model** that could be a conceptual model, ontology, or existing KG like Wikidata or DBpedia. ii) **Framework** that solves the required tasks for each module; given that, such framework *detects* both concepts and relations of the respective data model. Finally, iii) **Benchmark** is a collection of general or domain-specific benchmarks that exist or we develop. We use such benchmarks to assess our developed techniques. Thus, in the figure, our framework is *evaluated_by* benchmarks in the evaluation part. The domain-specific datasets we constructed have the *schema_of* the data model that is created at the first item to ensure a fair assessment.

Due to the heterogeneity of the data sources we deal with to construct a KG, the overall contribution that we present in this thesis is not a one-button solution. Each data source has its own set of unique characteristics and challenges. Thus, we developed individual components for each listed research area or data source. Such modules share the simple schema we discussed above. We give the details of each developed module in the following section. In addition, in this chapter, we explain how these modules orchestrate to define the ultimate goal of this work.

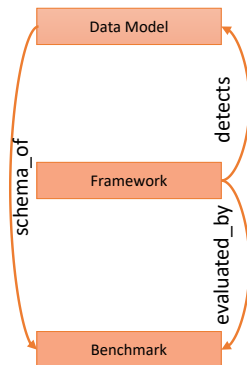


Figure 5.1: Schema of contribution pillar.

5.1 Individual Components

Since we have three research areas, we developed various contributions to answer our previous research questions (see Chapter 2). Following the schema we discussed earlier, our contributions are either: 1) a data model (conceptual model or an ontology), 2) a framework that solves specific tasks that detects the entities, classes, and relations, or 3) domain-specific benchmarks that are used to evaluate our developed framework. In the following, we list our contributions regarding the subject research area.

At first, concerning TabI, where this area should cover the ‘How can we use tabular datasets for KG construction? (RQ1)’. As shown in the current literature (see Chapter 4), the existing frameworks are either limited to solve only one task of the Semantic Table Interpretation (STI) tasks or require massive storage due to the dependencies on KG dumps. In addition, the respective benchmarks are either general domain or Automatically Generated (AG) datasets. Thus, on the one hand, we developed a framework, JenTab, that solves the complete set of the STI tasks. On the other hand, we constructed a biodiversity-specific benchmark, BiodivTab, that we used to evaluate JenTab. The former contribution is presented in Chapter 6. The latter is explained in Chapter 7 respectively.

Second, in the scope of TexI, such that this area covers the ‘How can we benefit from the information in the associated publications to enrich the constructed KG? (RQ2)’. We found limited contributions to the biodiversity domain in the current existing related work. Thus, we created a data model for the most common concepts and their relations in the domain. Such a conceptual model is defined as BiodivOnto. We give the details of its construction in Chapter 8. In addition, we developed a BERT-based model, BiodivBERT, that detects such concepts and relations from unstructured text. We explain the development phases of it in Chapter 9. BiodivBERT detects both concepts and relations that are in the BiodivOnto. Thus, such ontology acts as the schema of the BiodivBERT. To evaluate the BiodivBERT, and due to the limitations we explained in the related work regarding the limitations of the existing benchmarks, we constructed two corpora for both Named Entity Recognition (NER) and Relation Extraction (RE). We introduced both of them under one package, BiodivNERE. We give the details of the construction of such corpora in Chapter 10.

Finally, in the context of the last research area, MI, that covers the ‘How can we leverage the existing metadata to enrich the constructed KG? (RQ3)’. It is inspired by the work that is presented in the scope of the scholarly communication field (see Chapter 4). We constructed an ontology that captures the common concepts and their relations in the biodiversity metadata, BMO. In addition, we developed a framework, Meta2KG, that detects such concepts and relations using unsupervised learning techniques. Moreover, we manually labeled ground truth data, MetaGT, to evaluate our approach. We discuss the details behind the data model’s construction, the framework’s development, and the annotation process of the evaluation benchmark in Chapter 11.

5.2 Orchestration

Figure 5.2 depicts our contributions map and how the individual components interlink to formulate the big picture of this thesis. Horizontally, we demonstrate our modules that describe the desired data sources; tabular data, textual data, and metadata. Vertically, we use the default schema of a research project. Such schema consists of a data model (conceptual model, an ontology, or an existing KG) and its corresponding framework that detects the entities and their relations from the corresponding data source. Finally, the benchmark we used to evaluate such a technique. Such benchmark is either general domain or domain-specific that has the schema of the data model. In the figure, the ‘blue’ boxes

are the existing components that we directly use from the state of the art. However, the ‘green’ boxes illustrate our own contributions. Thus, our developed components vary from data models to benchmarks.

We define the first module by the contributions under the first TabI tabular data. Under this area, we developed JenTab as a framework that solves the STI tasks and evaluated by a set of general domain benchmarks; SemTab 2020-2022 [68, 70, 91], GitTables [71], and Tough Tables (2T) [69] dataset. In addition, we evaluate JenTab by the domain-specific benchmark, BiodivTab. Both contributions show our attempts to solve the first research question RQ1.

The second module shows our contributions concerning the second research area TextI, textual data. It contains our developed data model, BERT-based framework, and evaluation benchmarks. They correspond to BiodivOnto, BiodivBERT, and BiodivNERE. BiodivOnto is the schema of both BiodivBERT and BiodivNERE. In addition, we evaluate the BiodivBERT by BiodivNERE besides a collection of the existing benchmarks like Species-800 [79], COPIOUS [78], GAD [82], EU-ADR [81]. The three contributions demonstrate our trial to solve the second research question RQ2.

The final module defines our contributions to the third research area, MI, metadata. Similar to the second module, we contributed to the fundamental parts of the contribution schema. We developed Biodiversity Metadata Ontology (BMO), Meta2KG, and MetaGT that correspond to the data model, framework, and evaluation benchmark. Unlike the previous modules, we do not include general domain evaluation.

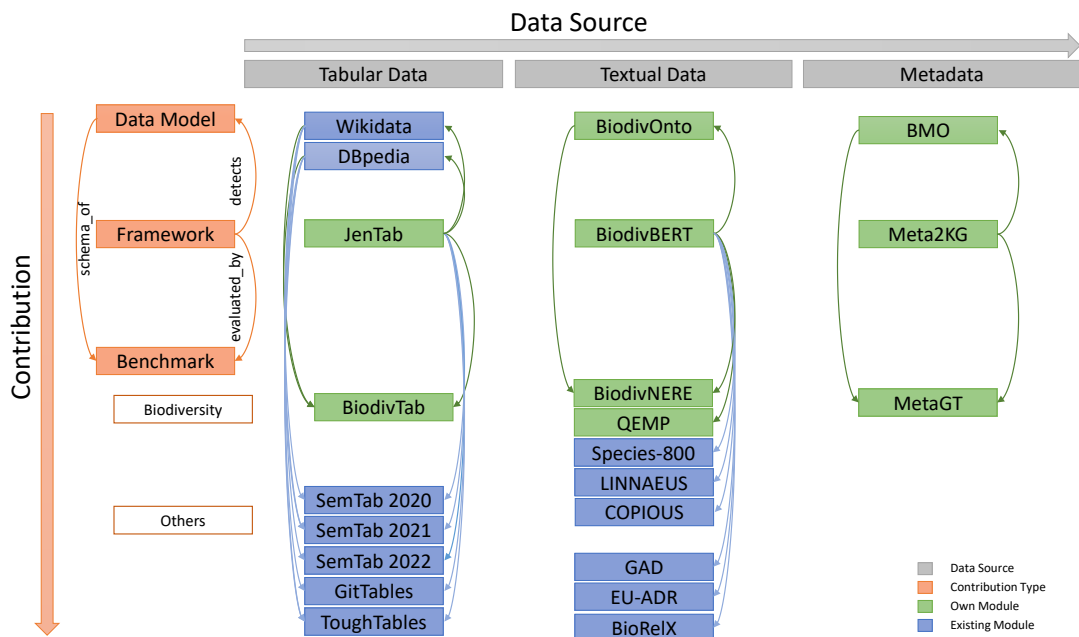


Figure 5.2: Contributions' map.

Part II

Tabular Data Interpretation

Chapter 6

JenTab Toolkit

Tabular data are a ubiquitous source of structured information. In our work, they are the primary data source we transform into Knowledge Graphs (KGs). The information contained within them is hardly accessible to automated processes. Cause issues range from misspellings and partial omissions to the ambiguity introduced using different naming schemes, languages, or abbreviations. The Semantic Web promises to overcome the ambiguities, but it requires annotating table elements like cells and columns with entities from existing KGs. However, automating this semantic annotation, especially for noisy tabular data, remains challenging.

The process of annotating a tabular dataset using a KG is called Semantic Table Interpretation (STI). We focus on three tasks throughout this chapter, a detailed discussion of STI is in Chapter 3. The objective is to map individual table elements to their counterparts from the KG as illustrated by a biodiversity example in Figure 6.1 (naming according to [22]): Cell Entity Annotation (CEA) matches cells to individuals, whereas Column Type Annotation (CTA) does the same for columns and classes. Furthermore, Column Property Annotation (CPA) captures the relationship between pairs of columns.

JenTab is a modular system to map table contents onto large KGs like Wikidata [92] and DBpedia [93]. This toolkit can annotate large corpora of tables. In our context, under the first research area **Tabular data interpretation (TabI)**, JenTab is the framework that matches tabular data to KGs. It follows a general pattern of Create, Filter and Select (CFS): First, for each annotation, initial candidates are generated using appropriate lookup techniques (Create). Subsequently, the available context is used in multiple iterations to narrow down these sets of candidates as much as possible (Filter). Finally, if multiple candidates remain, a solution is chosen among them (Select). We provide several modules for each of these steps. Different combinations (pipelines) allow for fine-tuning the annotation process by considering both the modules' performance characteristics and their impact on the generated solutions. Besides the default pipeline that implements the complete picture of the CFS pattern, since 2021, we have continuously developed various pipelines based on the benchmark characteristics we tried to tackle.

We developed and tested JenTab during the participation in the Semantic Web Challenge on Tabular Data to Knowledge Graph Matching (SemTab)¹ challenge 2020-2022. JenTab won the second place prize (Usability Track) by IBM Research² during ISWC 2021 [24]. In addition, JenTab was awarded the Artifacts Availability Badge by SemTab 2022. All experiments are based on the large corpora provided by SemTab [67, 68, 70, 69, 91] matching the content to Wikidata, DBpedia, schema.org. In addition, we evaluated JenTab in the biodiversity domain using the BiodivTab benchmark.

¹<http://www.cs.ox.ac.uk/isg/challenges/sem-tab/>

²<https://research.ibm.com/>

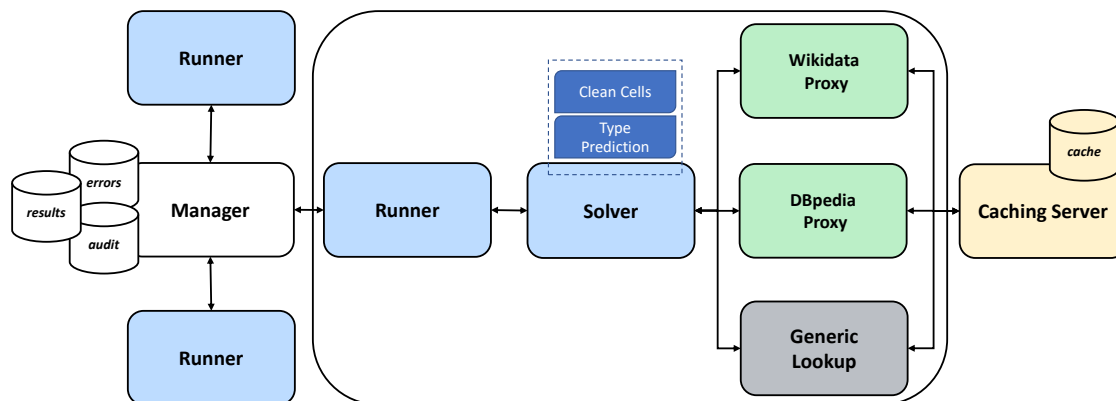


Figure 6.2: JenTab: Current system architecture (from [10]).

KG, e.g., Wikidata. On the other hand, Solver depends on Generic Lookup, which is our primary means of tackling spelling mistakes. Proxy services and Generic Lookup relies on a centralized ‘**Caching Server**’ for their results. It reduces the number of queries issued through the proxy services by caching responses already retrieved before.

The developed architecture has several advantages. First, caches for computationally expensive tasks or external dependencies increase the overall system performance. Furthermore, it reduces the pressure on downstream systems, especially when public third-party services are used. Second, when the target KG is to be substituted, all necessary changes, like adjusting SPARQL queries, are concentrated within just two locations: the corresponding lookup and endpoint services. Third, the distributed design allows for scaling well regarding the number of annotated tables. Any increase in the number of tables can be mitigated by adding new Runners to cope with the workload. Finally, the implementation allows reusable and self-encapsulated pieces of code. For example, Runner can deal with any other Solver implementation.

6.2 Preprocessing

Before executing the actual pipeline, each table is run through preprocessing steps. We group them in ‘Data Cleaning’ and ‘Datatype Prediction’. We give their details as follows:

Data Cleaning First, we apply a series of steps to fix data issues in a table.

1. We start with using *ftfy*⁵ to fix any encoding issues encountered.
2. We use a regular expression to split up terms that are missing spaces like in ‘*1stGlobal Opinion Leader’sSummit*’ into ‘*1st Global Opinion Leader’s Summit*’.
3. We remove certain special characters like parentheses.
4. We replace the explicitly mentioned unknown values to null. E.g., ‘NA’, ‘Unknown’, ‘Undetermined’ etc.
5. We capture the actual value of the classified date type cells. For example, ‘2010-11-23 November 23, 2010’ is translated to only ‘2010-11-23’.

⁵<https://github.com/LuminosoInsight/python-ftfy>

6. We apply an off-the-shelf spell checker, *autocorrect*⁶, to fix typos resulting in an ‘autocorrected value’ per cell.

The result of these steps is stored as a cell’s ‘clean value’. We use this cleaned value during the actual execution of JenTab’s pipeline. We deprecated the *autocorrect* step in 2021 due to its unstable behavior. Some values are corrected in unrealistic words, i.e., do not fit in the given context. For example ‘Rashmon’ that is expected to be ‘Rashomon’ (A produced Movie in 1950) is Fixed to ‘Fashion’. Instead, we relied on the Generic Lookup (we explain it in detail in Section 6.3). We apply such a series of steps to facilitate the CEA candidates generation by providing a cleaner version of the given cell values. In return, we expect a higher performance of candidates matching.

Datatype Prediction We determine the primitive datatype of each column. While the system distinguishes more datatypes, we aggregate the ones having a direct equivalent in Wikidata as the following four types:

1. **OBJECT** columns represent entities, those we solve CEA task for them since we check the given targets to solve such task.
2. **STRING** columns that were classified as **OBJECT** but not found among the given targets. They would be mapped to, e.g., entity labels or descriptions.
3. **DATE** columns represent date literals, which could be mapped later for a property like inception, start date, etc.
4. **NUMBER** columns represent numerical values that would be mapped to literal properties as well.

Besides locating the entities of interest that would be mapped to CEA, such classification helps us develop a datatype-specific property matching technique. For example, we solve the CPA task differently if the given column is **DATE** or **NUMBER**.

6.3 Offline Resources

In this section, we list the steps we conducted before the live run of JenTab. Usually, these steps are conducted to solve a specific issue we have encountered in datasets. For example, Generic Lookup and 2T Cleaning are meant to fix spelling mistakes but for different conditions. The Biodiversity Dictionaries are developed to tackle the problem of abbreviations in biodiversity datasets.

Generic Lookup Spelling mistakes and artificial noise are common challenges across STI benchmarks. A specialized lookup is our primary strategy for tackling this crucial issue. We created it ahead of time due to the resources required for comparing cell values against all labels (and aliases) within Wikidata or DBpedia. We extracted the unique values from all tables of a given dataset and matched those against the labels of the respective KG using an optimized Jaro-Winkler Similarity implementation based on [94] and a threshold for minimum similarity of 0.9. Such an approach will fetch and store all labels that have a similarity between 1 (exact matches) and 0.9. The result is a *sqlite*⁷ database where it has the highest priority for the lookup during the CEA-candidates generation.

⁶<https://pypi.org/project/autocorrect/>

⁷<https://www.sqlite.org/index.html>

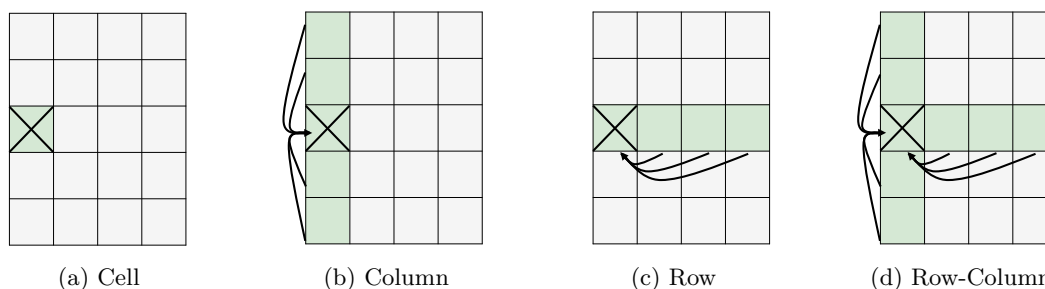


Figure 6.3: Possible contexts for resolving and disambiguating annotations for subject cells only. Arrows indicate information used (from [11]).

Biodiversity Dictionaries In 2021, we built biodiversity dictionaries for species and chemical elements or compounds to mitigate the impact of abbreviations. We constructed such dictionaries based on Wikidata. We queried Wikidata with all instances of the taxon (`?species wdt:P31 wd:Q16521`) and programmatically fetched the corresponding labels. From these labels, we created abbreviated variations by using one or two characters from the first word followed by the full second name. So, ‘Canna glauca’, e.g., was converted into both ‘C.glauca’ and ‘Ca.glauca’. These variants were subsequently used as aliases for the respective Wikidata entities. The corresponding lookup was used with the highest priority during our initial creation of CEA-candidates in the biodiversity setting.

2T Cleaning In 2022, we developed a cleaning module for 2T [69] dataset. It is known for the huge amount of artificially added misspellings to its tables. We have developed this strategy since our previously discussed Generic Lookup failed to catch these noisy elements. Thus, our core idea is to locate the cells that are correctly spelled and then replace all the artificial occurrences with the correct word. The first step is to find the correctly spelled words by querying those cells in Wikidata. Those with exact matches are considered correct words. The second step is to match the remaining values in the tables to the correctly identified values. We converted all the given cells into the embedding space using fasttext [95] to avoid the out-of-vocab (OOV) problem. Then, we applied cosine similarity among those vectors; we picked the correct target value if the similarity is $\geq 70\%$. We ran this step offline before the actual running of JenTab to solve the STI tasks. Thus, JenTab can work on relatively cleaner tables than those provided in the original 2T dataset.

6.4 Disambiguation Contexts

Tabular data offers different dimensions of context that can be used to either generate annotation candidates (Create-Phase) or remove highly improbable ones (Filter-Phase). Figure 6.3 illustrates those visually. The *Cell Context* is the most basic one, outlined in Figure 6.3a. Here, nothing but an individual cell’s content is available. We can then define a *Column Context* as shown in Figure 6.3b. It is based on the premise that all cells within a column represent the same characteristic of the corresponding tuples. For the annotation process, this can be exploited insofar that all cells of a column share the same class from the KG. Annotations for cells in **OBJECT**-columns further have a common class as required by the CTA task. Similarly, the assumption that each row refers to one tuple leads to the *Row Context* of Figure 6.3c. Annotation candidates for the subject cell, i.e., a cell holding the identifier for the respective tuple/row, have to be connected to their counterparts in all other cells within the same row. Finally, all contexts can be subsumed in the *Row-Column Context* as given by Figure 6.3d. It combines the last two assumptions representing the most exhaustive context.

6.5 CFS Pattern

We base our approach on collecting mostly independent building blocks for each task following the pattern of the CFS approach. The individual building blocks differ in what information they use and how accurate their results are. They further differ in their performance characteristics: On the one hand, this refers to the time needed to execute them. On the other hand, creation and filtration blocks vary in the number of candidates they output.

We maintain sets of candidates on different levels: For each cell, we maintain both the candidates for the respective CEA task and those induced by this cell for the corresponding CTA task. Each pair of cells in the same row has a set of candidates for the respective property connecting both cells contributing to the CPA task. The same is mirrored on the column level. Here, we keep the onset of candidates for the CTA task and CPA candidates for the combinations of columns.

Building blocks usually pertain to only one task as well as one CFS-stage: *Create* blocks generate candidates of possible solutions for the respective task. *Filter* blocks reduce given sets of candidates. Here, we take a rather conservative approach and only remove candidates that will not be a solution with a very high probability. Finally, *Select* blocks pick a solution for a given set of candidates. Figure 6.4 summarizes the developed building blocks which will be described in detail below. Unless noted otherwise, the building blocks only apply to columns of datatype `OBJECT` as classified in the preprocessing. Further to query the KG, we rely on the official SPARQL endpoint of Wikidata⁸ which imposes specific rate and execution time restrictions on our queries.

6.5.1 Create

In the following, we describe different building blocks that generate candidates for individual tasks.

CEA Label Lookup: The Label Lookup is the foundation of the pipeline. It does not depend on any prior information other than the cell’s content and their datatype. We apply a series of strategies to retrieve candidates based on the cells’ initial, clean, and autocorrected value. As each strategy succeeds in different cases, they are applied in sequence until candidates are retrieved by one. All but the Generic Strategy use the Wikidata Lookup Service⁹ to resolve from a given label to Wikidata entities.

- *Generic Strategy* compares the initial cell values with all Wikidata labels and aliases using the Jaro-Winkler distance [96]. We iteratively¹⁰ lower the selection threshold from 1 (exact matches) to 0.9 until a set of candidates is returned.
- *Full Cell Strategy* uses the initial cell values to query the lookup service.
- *All Tokens Strategy* splits a cleaned cell value into tokens removing all stopwords. The lookup service is then queried with all possible orderings of these tokens.
- *Selective Strategy* removes any addition in parenthesis and uses the remainder to query the lookup service.
- *Token Strategy* splits the cleaned cell value into tokens and queries for each token in isolation.

⁸<https://query.wikidata.org/>

⁹<https://www.wikidata.org/w/api.php>

¹⁰To speed up the process, we use a many-to-many implementation for calculation [94].

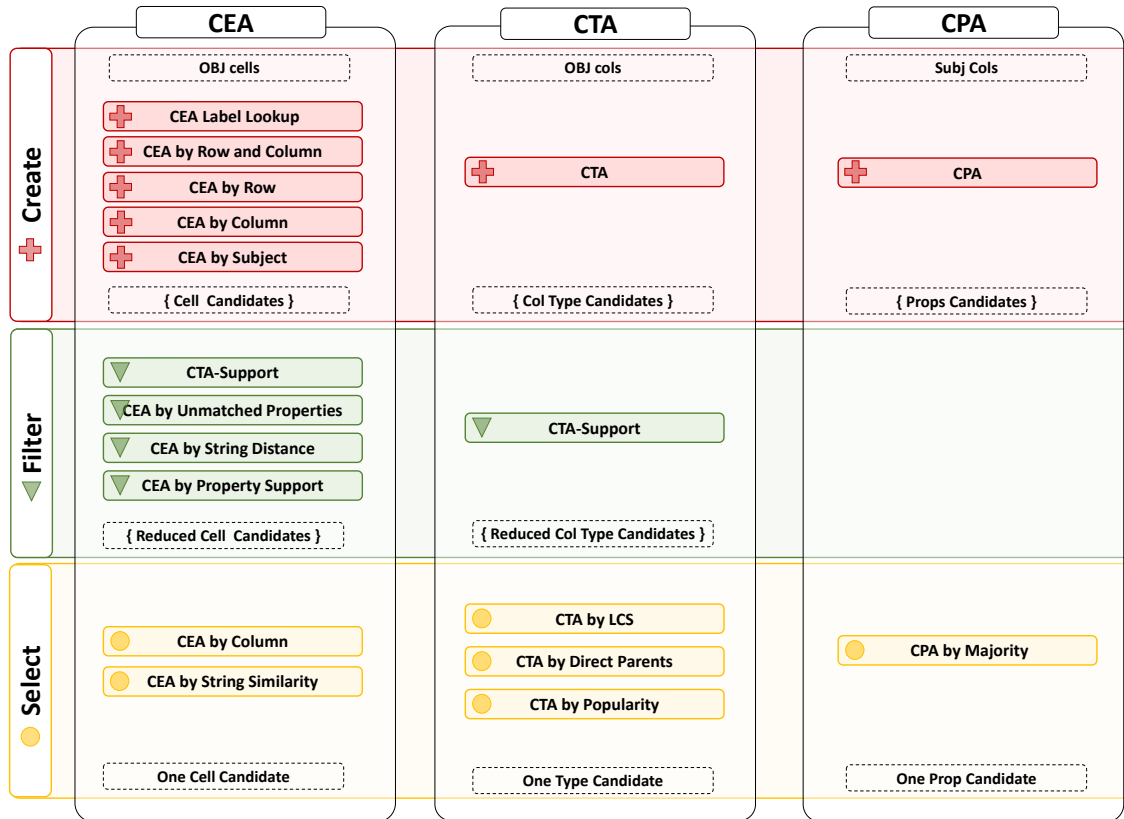


Figure 6.4: Pool of building blocks and their assignment to CFS-stage and task. *Create* is indicated in red with a plus icon, *Filter* is represented in green and a triangle sign and *Select* is shown in yellow with a circle symbol (from [9]).

- *Autocorrection Strategy* uses the autocorrected value from the preprocessing to query the lookup service.

CEA by row and column: This approach applies in cases where we could determine candidates for at least some cells of datatype OBJECT within a row but failed for the subject cell. Furthermore, for the subject column, existing candidates are required. If all conditions are met, we retrieve entities from the KG that are instances of a subject column candidate and have a connection to at least one of the other candidates in the same row. Subsequently, these entities are filtered such that the remaining ones have a connection to each object cell in the same row. Finally, we compute the string distances between the remaining labels and the initial cell value and discard all that exceed a certain threshold. Here, we use the Levenshtein distance [97] as implemented by edlib¹¹. Finally, we add the remaining entities as candidates to the subject cell in question.

CEA by row and **CEA by column:** These two approaches work in a similar fashion as *CEA by row and column*, but drop one of its preconditions respectively. In *CEA by row*, a candidate does not have to be an instance of the current column’s CTA-candidates. On the other hand, we apply *CEA by column*, when there are no other cells of datatype OBJECT in the same row, or those cells have no candidates either.

CEA by subject: This approach is the inverse to *CEA by row*. Assuming that the subject cell of a row has a set of candidates but any other cell of datatype OBJECT does not, it will fetch all entities from the Knowledge Base (KB) that are directly connected to a subject cell candidate. We filter the resulting entities again by their string-distance to the initial cell value before adding them as candidates.

¹¹<https://pypi.org/project/edlib/>

CTA: To find candidates for each column of datatype OBJECT, we first retrieve the types for each cell individually based on its current list of CEA-candidates. In DBpedia, we rely on `rdf:type` property to represent the types. In Wikidata, a type denotes a combination of *instanceOf* (P31) and *subclassOf* (P279) relations. The column’s type candidates are then given by the set of types appearing for at least one cell in this column. In addition, we configured three settings of CTA module to Wikidata specifically as follows:

- *P31* includes only direct parents using *instance of (P31)*. This strategy does not include any higher levels of classes.
- *2Hops* includes ‘P31’ with one additional parent (higher level) via *subclass of (P279)*.
- *Multi Hops* creates a more general tree of parents following *subclass of (P279)* relations.

CPA: Candidates for column pairs’ relations are generated by first retrieving candidates for the connections between cells of each row. We assume that there is a single subject column; thus, all other cells have to be connected in some way to the cell of that column.

First, we retrieve all properties for a subject cell’s candidates, including both literal and object properties. Second, we try to match individual properties to the values of other cells in the same row. If we have found a match, we add the respective property as a candidate for the corresponding cell pair. Object properties are rather easy to match. Here, we rely on previously generated CEA candidates of the respective target and merely compare those with the object properties retrieved.

On the other hand, literal properties require more care. For them, we only consider matches to a cell whose datatype has been determined as either DATE, STRING, or NUMBER in the preprocessing. If we can not establish an exact match for a cell’s value, we resort to fuzzy matching strategies depending on the corresponding datatype. For DATE-properties (RDF-type: `dateTime`), we try parsing the cell value using different date format patterns. If the parsing succeeds and both dates share the same day, month, and year, we consider this a match. We omit time and timezones for this comparison. In case of STRING-properties (RDF-type: `string` and `langString`), we extract words from the given value and the retrieved Wikidata label. Then, we count how many words are shared between the two string values. We consider a match if the overlap is above a certain threshold. For NUMBER-properties (RDF-type: `decimal`) we tolerate a 10% deviation to still be considered a match according to Equation 6.1 where *cell_value* is the table cell value and *property_value* is the retrieved property label.

$$(6.1) \quad Match = \begin{cases} true, & \text{if } |1 - \frac{cell_value}{property_value}| < 0.1 \\ false, & \text{otherwise} \end{cases}$$

Once candidates for each pair of cells are determined, we aggregate them to retrieve candidates on the column-level. This initial generation corresponds to the union of all candidates found on the row-level for cells within the respective columns.

6.5.2 Filter

Once we generated candidates for a particular task’s solution, we apply filter-functions to sort out highly improbable candidates. For create-functions depending on previously generated candidates, this can substantially reduce the queries required and overall running time.

CTA-support: This filter works separately on each column of datatype `OBJECT`. First, it calculates the support of all current CTA candidates concerning the cells of a column. A cell supports a given CTA candidate if any of its current candidates has the corresponding type¹². This filter neglects all cells that are either empty or have no CEA-candidates at the moment. Next, we remove all CTA-candidates from the respective column that do not have a support by at least 50% of the cells in this column. Finally, we remove all CEA-candidates from the corresponding cells, which have no types in the remaining CTA-candidates.

CEA by unmatched properties: After generating the properties for all cells on a row-level, some CEA-candidates will have no connection to any other cell in the same row. This filter removes these candidates, leaving only those that have a connection to at least one cell in their respective row. It applies to all cells of datatype `OBJECT`.

CEA by property support: This filter applies only to subject cells. We compute the support of a candidate as the number of cells in the same column it can be connected to¹³. We determine the maximum support for each of a cell's candidates and remove all those with lower support.

CEA by string distance: Some create-functions generate a relatively large number of CEA-candidates. This filter reduces that number by removing candidates whose label is too distant from the initial cell value. We rely on a normalized version of the Levenshtein distance [97], which uses the length of the involved strings as a normalizing factor. To keep any valid candidate, we resort to a rather conservative threshold, thus retain all candidates with a value of at least 0.5 for any of its labels.

6.5.3 Select

The target of all tasks; CEA, CTA, CPA is to select the most suitable solution per each task. Hence, at some point, we have to select a single entry from the remaining candidates. As some of the methods below cannot distinguish between the candidates in a certain situation, we apply them in sequence until we find a solution. If only one candidate is left after the previous filter steps, we pick it for an obvious reason.

CEA by string similarity: For the remaining candidates, we calculate the string distance to the original cell value using the Levenshtein distance [97]. If there are multiple candidates with the same distance, we break those ties using the 'popularity' of the respective candidates. We define popularity as the number of triples the respective candidate appears in them. The intuition is that if there was no other way to distinguish among the candidates in previous filter-steps, using the popularity results in the highest probability of selecting the correct candidate.

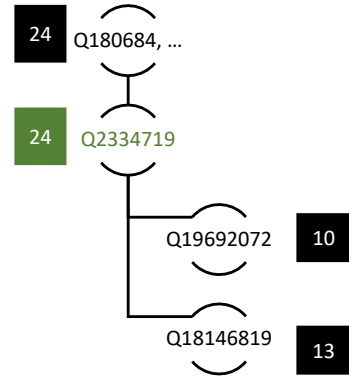
CEA by column: Sometimes filter-functions accidentally remove the correct solution from consideration. As those cases are quite rare, this affects only a small number of cells in a column. Further, the value of non-subject cells is often not unique throughout their column. In case no candidate is left for a cell, this function looks for occurrences of the same value in other cells of the same column. If we find a match, we apply their solution to the current cell.

CTA by Least Common Subsumer (LCS): The candidates for the CTA task do not stand in isolation but are part of a class hierarchy. Given an example of *Court Cases* in R3, cell types are shown in Figure 6.5a. We first expand this hierarchy for all remaining CTA-candidates, as shown in Figure 6.5b. Next, we compute the support for all candidates similar to the respective filtering-function. We remove all candidates with

¹²Kindly refer to the definition of 'type' in this context as given in the generation of CTA-candidates before.

¹³For non-subject cells, this value can only be 0 or 1, depending on whether they could be matched to the respective subject cell. This case is already covered in a different filter and thus excluded here.

Col0	P31
Spaziano . Florida	Q18146819, Q19692072
Smith v/ Maryland	Q18146819, Q19692072
SEC v. Texas Gulf Sumphur Co.	Q2334719
Reieer v. Thompso	Q18146819, Q19692072
Reed v. Pennsylvania Railroad Compan	Q18146819, Q2334719
Building Service Employees International Union Local 262 v/ Gazzam	Q18146819, Q19692072
Ramspeck v. Federal Trial Exainers Conference	Q18146819, Q2334719
Budk v. California	Q18146819, Q19692072
Cowma Dairy Company v. United States	Q18146819, Q2334719
Noswood v. Kirkpatrick	Q18146819, Q19692072
Mongomery Building & Construction Trades Council v. Ledbetter Erection Company	Q18146819, Q19692072
Southern Pacific Company v. Gileo	Q18146819, Q19692072
Colgate-Palmolive-Peft Company v. National Labor Relations Board	Q18146819, Q19692072
Unitee States v. United States Smelting Refining	Q18146819, Q19692072
Poizzi v. Cowles Magazies	Q18146819, Q19692072



(a) Cell values and corresponding types.

(b) Hierarchy of retrieved types.

Figure 6.5: Example for CTA selection by LCS, the selected type is highlighted in green (from [9]).

support less than the maximum. We choose the one with the longest path from the root node of the hierarchy as a solution from the remaining candidates.

CTA by Direct Parents: This function selects the CTA-solution by a majority vote. It will fetch the type for all remaining CEA-candidates of a column and then select the one appearing most often. In contrast to the previous definition of type, it only considers the direct connections of an entity, i.e. *instanceOf* (P31) and *subclassOf* (P279) but not their combination (P31/P279).

CTA by Popularity: In case the other methods failed to produce a result due to ties, this function will break those ties by using the candidates' popularity. Again, this is given by the number of corresponding triples the candidate appears in the entirety of the KB. As there is no semantic justification for this selection method, it is only used as a matter of last resort.

CPA by Majority Vote: We compute the support for a given CPA-candidate. Here, this refers to the number of remaining cell-candidate-pairs that use the respective property. We subsequently select the candidate with the highest support as a solution.

6.6 Default Pipeline

Figure 6.6 shows the details for the default configuration of our pipeline, `pipeline_full`. The current block order results from experimentation using the available input tables as a source. After each run, we scanned the results for tables lacking some mappings and investigated causes and possible solutions. We aggregate the individual building blocks into groups for the sake of brevity in the following description.

Group ① forms the core of our pipeline and is responsible for most of the mappings. Based on the *CEA Label Lookup*, it generates candidates for all three tasks and removes only the most unlikely candidates from further consideration.

Group ② represents a first attempt to find missing CEA-mappings based on the context of a row and a column. Be kindly reminded that all create blocks only work on cells that currently have no candidates attached. So both blocks here apply to cells with no mappings from Group ①. *CEA by Row and Column* is put before *CEA by Row*, as it relies on a narrower context and thus will provide more accurate results. However, it might fail, e.g., when the current CTA-candidates do not include the proper solution. In such cases, *CEA by Row* will loosen the requirements to compensate. If either of these attempts provided

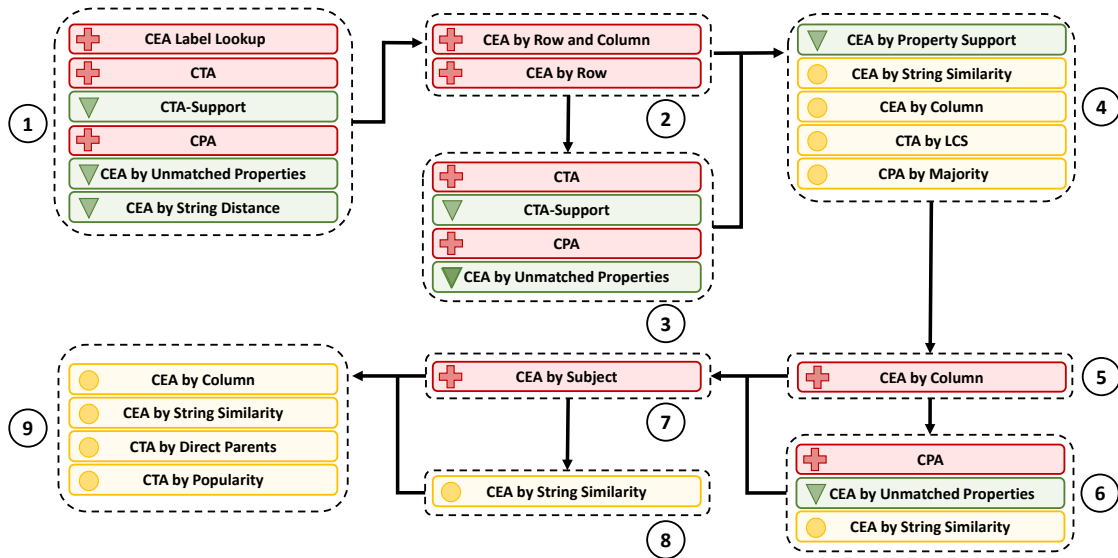


Figure 6.6: `pipeline_full` as an arrangement of CFS building blocks (from [9]).

new candidates, we re-execute the creation of CTA and CPA candidates afterwards in Group ③.

Group ④ is our first attempt at selecting solutions. After another filtering step on CEA-candidates using the row context, we continue to select high-confidence solutions. As hinted before, this might fail to produce proper mappings for a fraction of CEA-targets. Groups ⑤, and ⑦ try to fill in the gaps for at least some of them. If new candidates are found, groups ⑥ and ⑧ will filter and select from them.

Finally, Group ⑨ represents our means of last resort. Again, they only apply to targets for which we could not generate a solution before. Here, we assume that not only parts of the context are wrong but doubt every part of the context. The used blocks will reconsider all candidates discarded in the previous steps and attempt to find the best solution among them.

6.7 Other Pipelines

Since 2021 based on the original JenTab pipeline, `pipeline_full`, we derived multiple variations and evaluated them subsequently. This provided a better insight into the impact of individual components on the execution time and the quality of results. Most of these variations share the same initial phase to create candidates for CEA, CTA, and CPA if required.

`pipeline_essential` This pipeline reduces `pipeline_full` to its core components. In particular, each step runs only once, excluding any re-execution in the process. It became necessary initially as some tables proved too demanding when executed using `pipeline_full`.

`pipeline_no_cpa` Compared to `pipeline_full`, this pipeline removes any CPA-related components. In particular, filter operations involving relations among cells as well as selecting CPA-solutions are omitted in this pipeline. It was applied in tasks that featured only CEA and CTA targets and omitted any CPA ones.

pipeline_numeric This pipeline is specifically geared towards tables that feature a single object (the subject) column and one or more non-object (primarily numeric) column(s). After creating initial candidates for all tasks it filters them with the least frequent properties to determine the most likely CEA and thus indirectly CTA-candidates. The latter is then used while applying **CEA by Column** that adds new candidates to unmatched cells based on all instances of the identified types. After subsequent additional filter steps, the default selection process is used to determine the final solutions.

pipeline_conditional As **pipeline_numeric** only applies to a subset of tables, this pipeline uses a two-step approach: In a first step, **pipeline_numeric** is applied to all tables meeting the respective preconditions. If these conditions are not met, or the returned result covers less than 80% of targets, the table is again processed using **pipeline_full**.

pipeline_header Such pipeline is based on **pipeline_no_cpa** which contains all modules from the default configuration except the CPA creation, filtration, and selection parts. However, the new pipeline overrides the CTA with the headers candidates. Indeed, such method suits datasets that contain meaningful header.

6.8 Evaluation

We conducted a series of evaluations in the consecutive years: 2020, 2021, and 2022, and during the SemTab challenge on large-scale general domain datasets and using the biodiversity-specific dataset. In the following, we describe the evaluation corpora, pre-processing, and generic lookup strategy effectiveness. In addition, we show the load of the developed CEA creation strategies. Moreover, we demonstrate the accuracy scores of JenTab compared to the other systems tackling the same tasks. Finally, we discuss the runtime of JenTab on the evaluation benchmark and give an overview of its performance over the years of development.

6.8.1 Benchmarks

Before we evaluate JenTab in many various aspects, e.g., accuracy scores, we give an overview of the benchmarks we used to evaluate it.

- **SemTab 2020** is an Automatically Generated (AG) benchmark that contains over 130,000 tables from Wikidata and is divided into four rounds. The tasks are to annotate these tables from Wikidata [68].
- **HardTables** is an AG dataset with a first release during the second round of SemTab 2021 with 7,207 tables. The authors filtered it to include only the hardest cases in SemTab 2022 and released it again with 3,691 and 4,659 tables in the first and second rounds of that year’s challenge. The target KG concerning its tasks is Wikidata [91].
- **Tough Tables (2T)** is a dataset consisting of 180 tables annotated from both Wikidata and DBpedia [69]. It has been used during SemTab 2021 and 2022. This dataset is known for its very high level of entity ambiguity and its massive amount of artificially added noise to table cells.
- **BioTables** is a biomedical domain-related dataset for STI. It is also an AG dataset comprising 110 tables. It is released and used during SemTab 2021 [24]. This benchmark has a unique characteristic that features a high number of average columns, 2500.

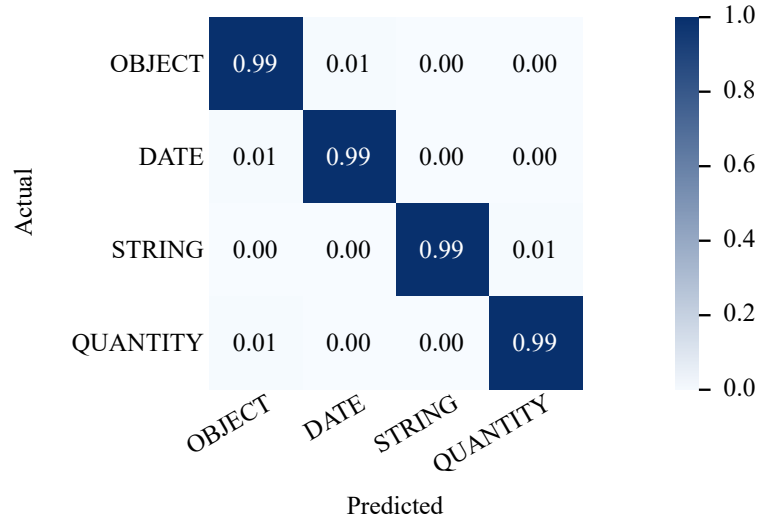


Figure 6.7: Confusion matrix for type prediction. A 4-label classification task (from [11]).

- **BiodivTab** is a real-world and manually annotated benchmark from biodiversity research. It consists of 50 tables. This benchmark is released and used for the first time during SemTab 2021, where the target KG is Wikidata. It is released and used again during the third round of SemTab 2022, where the target KG is changed DBpedia. This is our own contribution; we discuss its construction details and insights in Chapter 7.

6.8.2 Datatype Prediction Assessment

For evaluation, we start with assessing the preprocessing during the ‘Type Prediction’ step. This step is responsible for determining a column’s datatype; see Section 6.2. Figure 6.7 shows the confusion matrix of this step. It shows a 99% prediction accuracy score. We used the SemTab 2020 benchmark to create ground truth data for evaluating this step. We used such ground truth for CEA and CPA tasks to query Wikidata for their types; values represent the actual datatypes, and the predicted values are by our system results.

6.8.3 Generic Lookup Coverage

Spelling mistakes are a crucial problem we tackled using the ‘Generic Strategy’. We create this lookup before the run of the selected pipeline due to the resources required for comparing cell values against all labels (and aliases) within Wikidata or DBpedia. For this, we extract the unique values from all tables of a dataset and match those against the labels of the respective KG using an optimized Jaro-Winkler Similarity implementation based on [94] and a threshold for minimum similarity of 0.9. This lookup represents our primary and the only source to fix spelling mistakes since 2021. The effectiveness of lookup is illustrated in Table 6.1 over the three years. For SemTab 2020 benchmark, almost 99% of unique labels were covered in the first three rounds. However, this is reduced to $\sim 97\%$ in the last round.

In 2021, for most datasets, more than 89% of unique values could be matched up to a 99.76% success rate for R2’s HardTables. However, it is significantly lower in the case of the 2T dataset. This dataset has a massive amount of spelling mistakes that the selected threshold could not solve.

In 2022, for the synthetic dataset, HardTables, the matching percentage is high. It reached up to 99% in the first round. 2T dataset remains almost in the same range as the previous year. It reached around 89% in the second round, where DBpedia is the target

Table 6.1: Generic Strategy: Unique labels and ratio of resolved labels per round.

Year	Rounds	Dataset	Target	Unique Labels	Matched	Matched
2020	R1	SemTab20	Wikidata	252,329	249,806	99.0%
2020	R2	SemTab20	Wikidata	132,948	131,486	98.9%
2020	R3	SemTab20	Wikidata	361,313	357,700	99.0%
2020	R4	SemTab20	Wikidata	533,015	515,959	96.8%
2021	R1	2T	Wikidata	69,980	62,908	89.89%
2021	R1	2T	DBpedia	66,340	59,168	89.19%
2021	R2	HardTables	Wikidata	249,625	249,025	99.76%
2021	R3	HardTables	Wikidata	47,809	46,865	98.03%
2022	R1	HardTables	Wikidata	19,107	18,928	99%
2022	R2	HardTables & 2T	Wikidata	74,177	67,986	91.6%
2022	R2	2T	DBpedia	65,223	58,235	89.3%

KG. Such lower matching percentage guided us to develop a more sophisticated cleaning step before the actual run, as discussed in Section 6.3. We have not created a generic lookup for BiodivTab in 2021 and 2022 since our approach here relies on dictionaries for taxons and chemical elements - the prevalent type of cell values in this dataset.

6.8.4 Audit Results

We have implemented five strategies for creating initial CEA candidates, including Generic Lookup, Full Cell, Tokens, All tokens, Selective, and Autocorrect. We investigated if all of them are needed using various benchmarks.

CEA Creation Figure 6.8 shows how much each strategy is used for all general domain benchmarks. Generic Lookup proves its strength in solving spelling mistakes and, thus, is the most used strategy for all benchmarks. We recently disabled ‘Autocorrect’ and ‘All Tokens’ strategies due to their high computational requirements in the recent benchmarks (BioTables and HardTables). This underlines the need for various strategies to capture a wide range of useful information inside each cell. The shown distribution also reflects our chosen order of methods. For example, ‘Generic Strategy’ is our first priority, thus used most of the time. On the other hand, ‘Autocorrect’ has the lowest priority and is used as a means of last resort. We handle the BiodivTab benchmark differently, i.e., we build a dictionary for taxons and chemical compounds and use it as a customized lookup service. Figure 6.9 shows the distribution of the selected methods to create CEA candidates in both editions of BiodivTab, Wikidata and DBpedia.

CEA Selection Figure 6.10 demonstrates the use of each strategy. Our dominant select approach is ‘String Similarity’; it is used by 38% more than the ‘Column Similarity’ in all benchmarks except HardTables where the ‘String Similarity’ managed to solve all cases. For BiodivTab, Figure 6.11 depicts the methods distribution for the CEA selection strategies. Both versions, i.e., Wikidata and DBpedia, show that ‘String Similarity’ method managed to solve more than double the amount of the ‘Column Similarity’.

CTA Selection Figure 6.12 describes the distribution of CTA selection methods where the LCS method is the dominant strategy to select the final candidate. It shows continuous success in solving the task for all benchmarks, especially BioTables and HardTable. Both benchmarks did not need any backup solutions to select the CTA candidate at all. For BiodivTab, Figure 6.13 demonstrates the distribution of CTA selection strategies. The

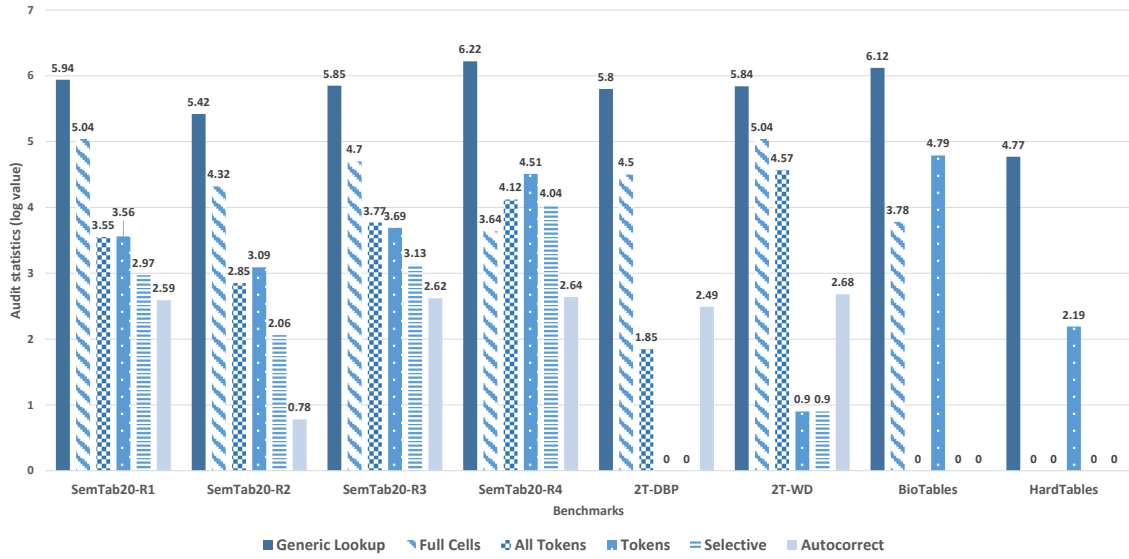


Figure 6.8: Audit statistics for CEA creation. y -axis is the \log scale of the solved cells.

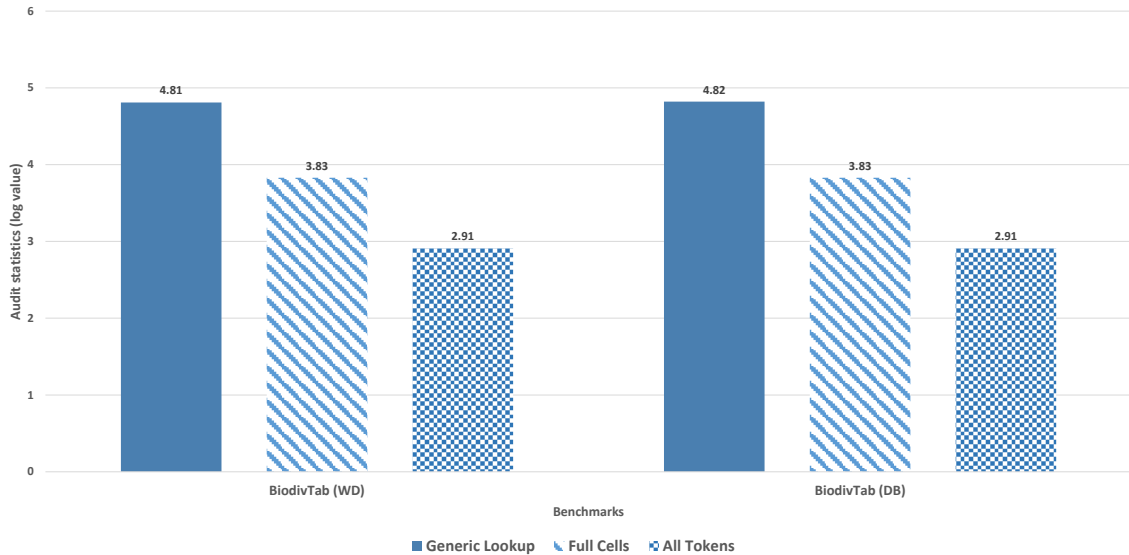


Figure 6.9: Audit statistics for CEA creation (**BiodivTab**). y -axis is the \log scale of the solved cells.

dominant method in the Wikidata version is the LCS where we activated the ‘2Hops’ mode. However, the primary method in the DBpedia version is Majority Vote. From the figure, we found that LCS successfully finds more solutions than the other, which yields less reliance on backup strategies or tiebreakers. The same exclusive execution concept in CEA selection is also applied in CTA selection methods. The dominant method, e.g., LCS or Majority Vote, is invoked more frequently due to its higher priority during execution. Other backup strategies try to solve the remaining columns if other methods fail to find a solution for them.

6.8.5 Accuracy Scores

In the following, we compare JenTab to the existing state-of-the-art systems that tackle the STI tasks to demonstrate its effectiveness¹⁴.

¹⁴We use the results of the SemTab challenge systems as is for comparison in this thesis.

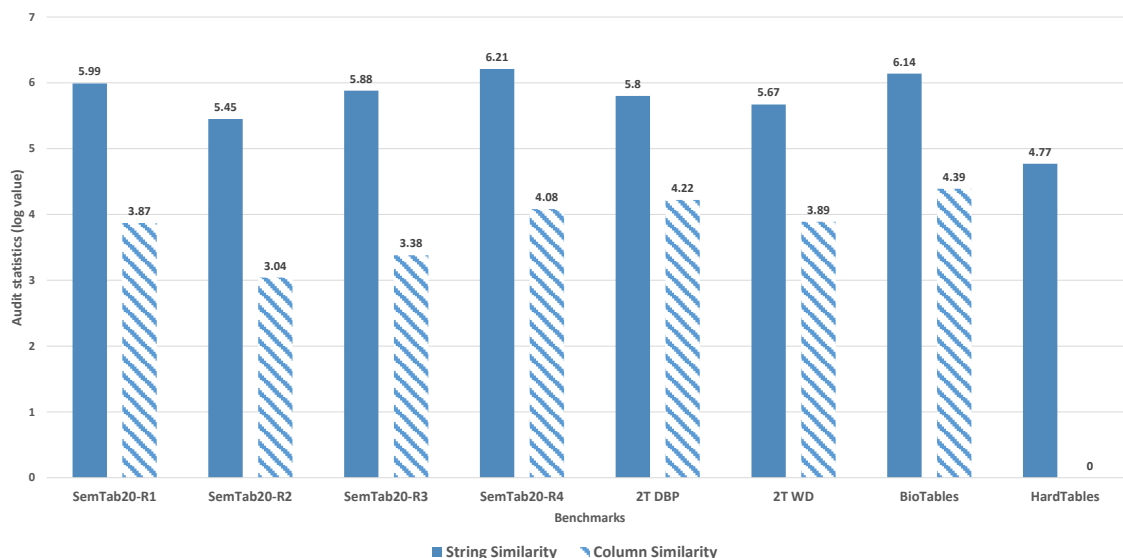


Figure 6.10: Audit statistics for CEA selection. y-axis is the \log scale of the solved cells.

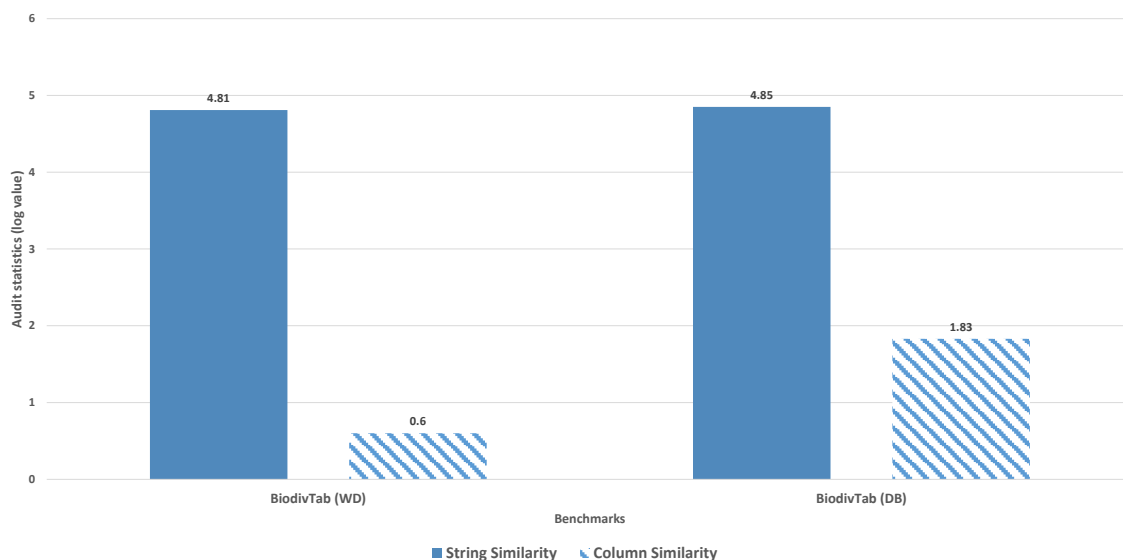


Figure 6.11: Audit statistics for CEA selection (**BiodivTab**). y-axis is the \log scale of the solved cells.

SemTab 2020 Table 6.2 shows the participants scores on such benchmark. This represents the first participation of JenTab to tackle the STI tasks. During the challenge, we did not have scores for CPA in the first round since we did not implement such a module at the time of participation. However, we continued development after the end of the challenge, and we report a complete list of scores here. In this benchmark, JenTab demonstrated high scores across all rounds, especially in the CPA task. JenTab is placed in the third rank given that benchmarks and without further complex dependencies or KG dump.

HardTables Table 6.3 shows the scores of both JenTab and the other participants on this benchmark in 2021 and 2022. JenTab kept its high performance and rank during 2021's edition of this benchmark. However, during the second round in 2022, it achieved lower scores. A reason could be that HardTables during this challenge's edition is way more complicated than the previous. This is due to a higher level of ambiguity has been introduced. All top participants, not only JenTab achieved lower scores than in their previous performance.

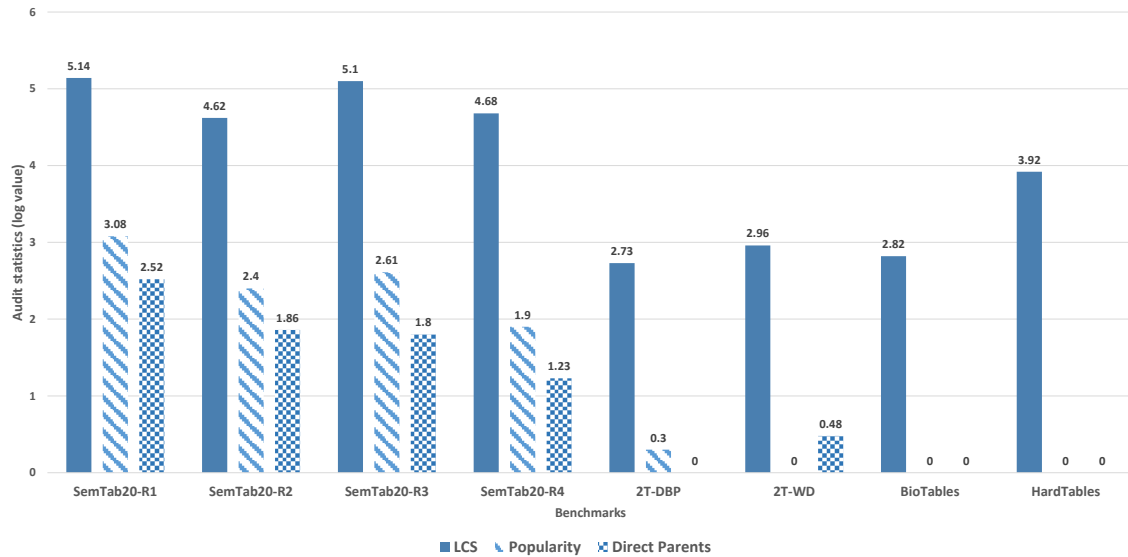


Figure 6.12: Audit statistics for CTA selection. ‘2Hops’ mode. y-axis is the *log* scale of the solved cells.

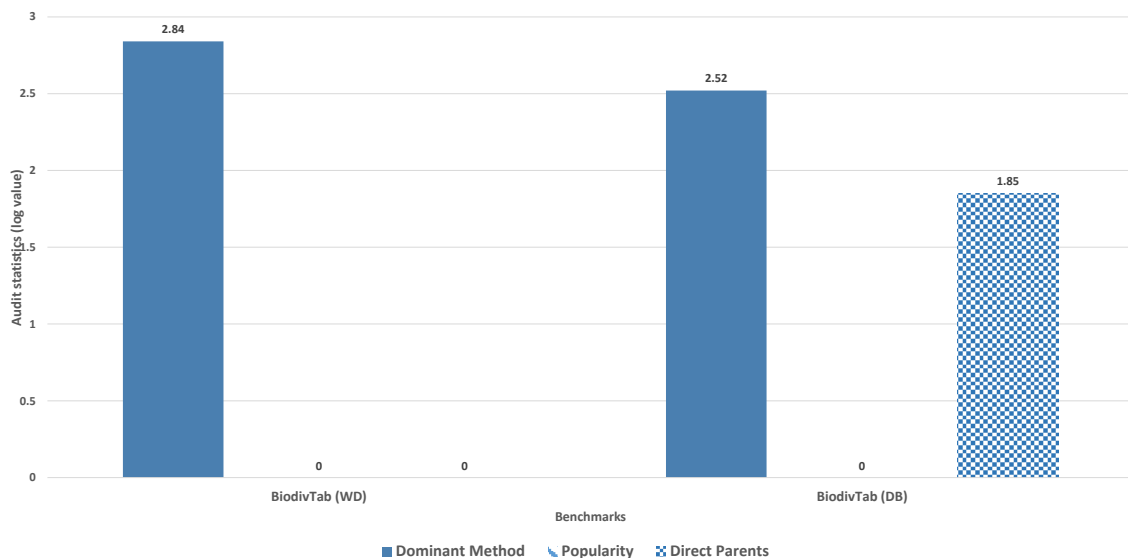


Figure 6.13: Audit statistics for CTA selection (**BiodivTab**). ‘2Hops’ mode. y-axis is the *log* scale of the solved cells.

Tough Tables (2T) Table 6.4 demonstrates our scores compared to the existing state-of-the-art on the 2T dataset. It appeared during the fourth round of SemTab 2020 for the first time, where Wikidata is the target KG. Its second appearance is in 2021. The authors updated the ground truth data to a recent Wikidata dump. In addition, they released a new version with DBpedia as an additional target KG. The scores by either JenTab or the existing systems are relatively lower than those on AG datasets. The major reason for these low scores are due to the high level of ambiguity that could confuse even a human [69] of the table cells. Another reason is the excessive amount of artificial spelling mistakes that are added to the cell values. In 2020, JenTab was still more sensitive to the latter issue than other systems. We continuously developed JenTab to tackle the problem of misspellings. For CEA, over the years, JenTab gained much higher scores on this dataset from 37% to slightly above 80%, given Wikidata as a target KG.

Table 6.2: SemTab 2020 top participants’ scores (**AG-datasets**). F1 - F1 Score, Pr - Precision, R - Recall, AF1, APr, and AR - Approximate version of F1 Score, Precision, and Recall respectively.

Round	System	CEA		CTA		CPA	
		F1	Pr	AF1	APr	F1	Pr
R1	MTab [37]	0.987	0.988	0.885	0.884	0.971	0.991
R1	LinkingPark [49]	0.987	0.988	0.926	0.926	0.967	0.978
R1	SSL [50]	0.936	0.936	0.861	0.860	0.943	0.943
R1	bbw [41]	NA	NA	NA	NA	NA	NA
R1	DAGOBAB [39]	0.922	0.944	0.834	0.854	0.914	0.962
R1	JenTab	0.968	0.969	0.962	0.965	0.984	0.988
R2	MTab [37]	0.995	0.995	0.984	0.984	0.997	0.997
R2	LinkingPark [49]	0.993	0.993	0.984	0.985	0.993	0.994
R2	SSL [50]	0.961	0.961	0.966	0.966	0.973	0.973
R3	bbw [41]	0.892	0.960	0.914	0.929	0.991	0.992
R2	DAGOBAB [39]	0.993	0.993	0.983	0.983	0.992	0.994
R2	JenTab	0.975	0.975	0.965	0.965	0.984	0.984
R3	MTab [37]	0.991	0.992	0.976	0.976	0.995	0.995
R3	LinkingPark [49]	0.986	0.986	0.978	0.979	0.985	0.988
R3	SSL [50]	0.906	0.906	0.913	0.913	0.815	0.815
R3	bbw [41]	0.954	0.974	0.960	0.966	0.949	0.957
R3	DAGOBAB [39]	0.985	0.985	0.974	0.974	0.993	0.994
R3	JenTab	0.967	0.967	0.955	0.959	0.981	0.987
R4	MTab [37]	0.993	0.993	0.981	0.982	0.997	0.997
R4	LinkingPark [49]	0.985	0.985	0.953	0.953	0.985	0.985
R4	SSL [50]	0.833	0.833	0.946	0.946	0.924	0.924
R4	bbw [41]	0.978	0.984	0.980	0.980	0.995	0.995
R4	DAGOBAB [39]	0.984	0.985	0.972	0.972	0.995	0.995
R4	JenTab	0.974	0.974	0.945	0.93	0.993	0.994

BioTables Table 6.5 shows the scores of JenTab and the existing state-of-the-art systems. JenTab showed consistent scores and ranked among the top systems. This benchmark showed new obstacles for all participants. Initially, we failed to run our `pipeline_full` due to several timeout errors. However, we developed our default pipeline and managed to execute it on this benchmark and reached competitive scores.

BiodivTab Table 6.6 shows scores of the top participants the SemTab challenge in 2021 and 2022. Scores have been published by the organizers of SemTab [24, 35]. JenTab, for its first attempt, achieved the best score of CEA task. However, the best CTA score is achieved by KEPLER [98]. We gained low scores of 60% and 10% in 2021 and 55% and 41% in 2022 for both CEA and CTA, respectively. In contrast, for the synthetic dataset, HardTables 2021, DAGOBAB achieved the maximum F1-score 97.4%, and 99% for CEA, and CTA respectively. Such low scores are due to the unique benchmark characteristics that differ from the traditional automatically generated and general-domain benchmark.

General Overview We give an overview of JenTab scores over the years on both AG (Figure 6.14), and 2T (Figure 6.15) datasets since 2020 to 2022. For the AG datasets, the average F1-score for the three tasks; CEA, CTA, and CPA are 0.91, 0.93, and 0.97 respectively. For the 2T dataset, the average F1-score for both CEA and CTA tasks is 0.54.

Table 6.3: JenTab & SOTA scores (**HardTables** 2021-2022). F1 - F1 Score, Pr - Precision, R - Recall, AF1, APr, and AR - Approximate version of F1 Score, Precision, and Recall respectively.

Year	Round	System	CEA		CTA		CPA	
			F1	Pr	AF1	APr	F1	Pr
2021	R2	MTab [38]	0.985	0.985	0.977	0.977	0.997	0.998
2021	R2	DAGOBAB [40]	0.975	0.975	0.976	0.976	0.996	0.996
2021	R2	Magic [51]	0.836	0.947	0.757	0.681	0.865	0.954
2021	R2	Kepler-aSI [98]	0.975	0.975	0.894	0.931	0.915	0.989
2021	R2	JenTab	0.966	0.967	0.914	0.917	0.996	0.997
2021	R3	MTab [38]	0.968	0.968	0.984	0.984	0.993	0.993
2021	R3	DAGOBAB [40]	0.974	0.974	0.990	0.990	0.991	0.995
2021	R3	Magic [51]	0.641	0.721	0.687	0.687	0.788	0.936
2021	R3	Kepler-aSI [98]	NA	NA	0.244	0.244	NA	NA
2021	R3	JenTab	0.940	0.940	0.942	0.942	0.992	0.992
2022	R1	KGCODE-Tab [99]	0.893	0.916	0.942	0.944	0.906	0.918
2022	R1	DAGOBAB [100]	0.954	0.955	0.975	0.975	0.984	0.99
2022	R1	s-elbat [101]	0.945	0.964	0.951	0.957	0.983	0.989
2022	R1	JenTab	0.945	0.946	0.938	0.940	0.975	0.986
2022	R2	KGCODE-Tab [99]	0.856	0.875	0.968	0.971	0.916	0.943
2022	R2	DAGOBAB [100]	0.904	0.905	0.96	0.96	0.931	0.97
2022	R2	s-elbat [101]	0.825	0.875	0.859	0.878	0.931	0.96
2022	R2	JenTab	0.751	0.758	0.836	0.881	0.872	0.921

Table 6.4: JenTab and existing systems scores (**2T** dataset). F1 - F1 Score, and Pr - Precision. AF1, and APr - Approximate version of F1 Score, Precision respectively.

Year	System	Wikidata				DBpedia			
		CEA		CTA		CEA		CTA	
		F1	Pr	AF1	APr	F1	Pr	AF1	APr
2020	MTab [38]	0.907	0.907	0.885	0.884	NA	NA	NA	NA
2020	LinkingPark [49]	0.810	0.811	0.926	0.926	NA	NA	NA	NA
2020	SSL [50]	0.198	0.198	0.861	0.860	NA	NA	NA	NA
2020	bbw [41]	0.863	0.927	NA	NA	NA	NA	NA	NA
2020	DAGOBAB [50]	0.412	0.749	0.834	0.854	NA	NA	NA	NA
2020	JenTab	0.374	0.541	0.574	0.626	NA	NA	NA	NA
2021	DAGOBAB [40]	0.923	0.923	0.832	0.832	0.945	0.946	0.422	0.424
2021	AMALGAM [52]	0.658	0.791	0.476	0.422	NA	NA	NA	NA
2021	Kepler-aSI [98]	0.194	0.760	0.464	0.481	0.110	0.644	0.027	0.133
2021	Magic [51]	NA	NA	NA	NA	0.184	0.506	0.159	0.628
2021	JenTab	0.457	0.520	0.697	0.697	0.607	0.669	0.460	0.468
2022	KGCODE-Tab [99]	0.905	0.913	0.543	0.546	0.827	0.830	0.480	0.484
2022	DAGOBAB [100]	0.945	0.946	0.409	0.409	NA	NA	NA	NA
2022	s-elbat [101]	0.937	0.938	0.366	0.366	0.789	0.808	0.373	0.375
2022	JenTab	0.802	0.807	0.346	0.356	0.572	0.792	0.234	0.290

Table 6.5: JenTab & SOTA scores (**BioTables** 2021). F1 - F1 Score, Pr - Precision. AF1, and APr - Approximate version of F1 Score, and Precision respectively.

Year	Round	System	CEA		CTA		CPA	
			F1	Pr	AF1	APr	F1	Pr
2021	R2	MTab [38]	0.985	0.985	0.977	0.977	0.997	0.998
2021	R2	DAGOBABH [40]	0.975	0.975	0.976	0.976	0.996	0.996
2021	R2	Magic [51]	0.836	0.947	0.757	0.681	0.865	0.954
2021	R2	Kepler-aSI [98]	0.975	0.975	0.894	0.931	0.915	0.989
2021	R2	JenTab	0.966	0.967	0.914	0.917	0.996	0.997

Table 6.6: JenTab & SOTA scores (**BiodivTab** 2021-2022) . F1 - F1 Score, Pr - Precision, AF1, and APr - Approximate version of F1 Score and, Precision respectively.

Year	Target	System	CEA		CTA	
			F1	Pr	AF1	AP1
2021	Wikidata	MTab [38]	0.522	0.527	0.123	0.282
2021	Wikidata	Magic [51]	0.142	0.192	0.10	0.253
2021	Wikidata	DAGOBABH [40]	0.496	0.497	0.381	0.382
2021	Wikidata	mantisTable [47]	0.264	0.785	0.061	0.076
2021	Wikidata	KEPLER [98]	NA	NA	0.593	0.595
2021	Wikidata	JenTab	0.602	0.611	0.107	0.107
2022	DBpedia	KGCODE-Tab [99]	0.910	0.910	0.870	0.870
2022	DBpedia	DAGOBABH [100]	NA	NA	0.620	0.620
2022	DBpedia	s-elbat [101]	NA	NA	0.06	0.06
2022	DBpedia	JenTab	0.550	0.610	0.420	0.412

This reflects the sensitivity of JenTab against the high level of artificial noise. However, in 2022, the CEA has significantly improved due to our sophisticated cleaning module for such a dataset.

6.8.6 Runtime

During JenTab’s years of development, we used normal laptops for execution. For example, two core i7 machines, one with 16GB RAM and the other is 8GB RAM. Starting from 2021, we hosted the central node (Manager) on a virtual machine with 256 cores and 64GB RAM. We set up three different experiments to test the effect of CTA selection strategies for the SemTab 2020 benchmark. Besides the low scores of the Multi-Hops, it is also computationally expensive. Table 6.7 shows the processing time for all four rounds with the number of used runners for each mode setting of the CTA task. Close inspection revealed that the execution time is largely dominated by the responses of Wikidata servers and thus beyond our control. The execution was time-scoped, i.e., an upper limit for the time per table was set. This allowed the system to converge faster compared to the initial implementation of JenTab in 2020, with, e.g., Round 4 showing a more than 50% reduction in time. Intermediate results are cached across rounds, saving time and lowering the number of requests to external services. Our modular approach allows us to scale the number of runners based on available resources and speed up the overall process.

We exclude the Multi-Hops from further usage and limit our setup to ‘P31’ and ‘2Hops’ only. Table 6.8 shows the runtime of JenTab during its participation in 2021 and 2022. This table shows the configuration that yields the best scores. For 2021, the setup of CTA

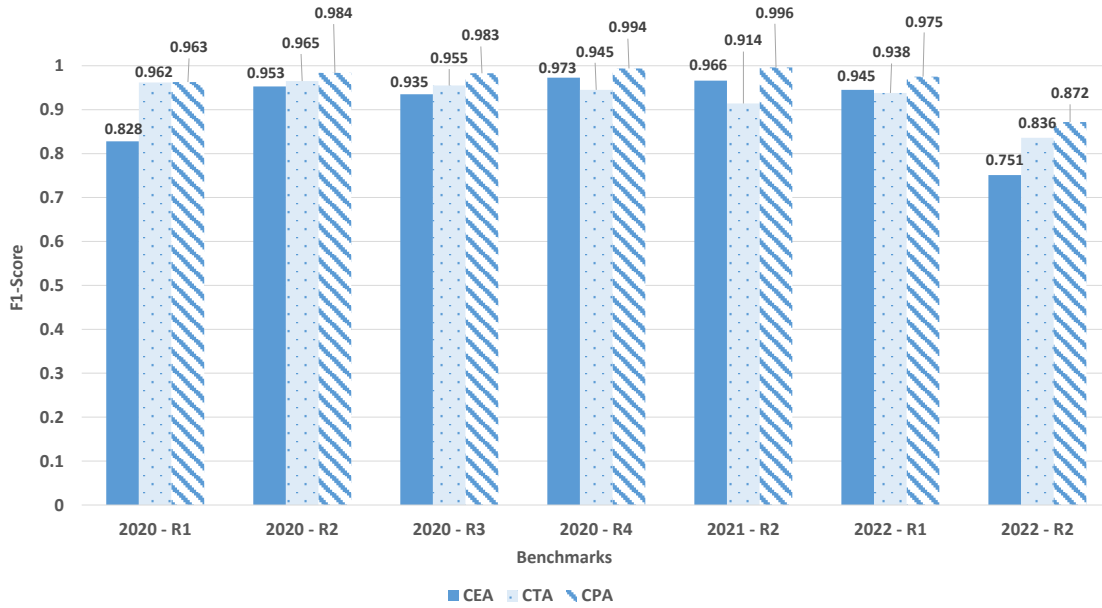


Figure 6.14: JenTab F1-scores Automatically Generated (AG) datasets [2020-2022].

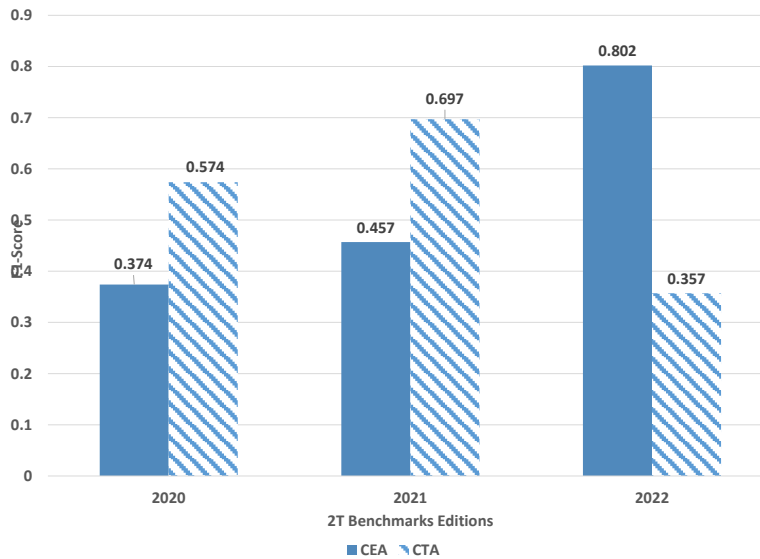


Figure 6.15: JenTab F1-scores Tough Tables (2T) datasets [2020-2022].

is 2Hops, but for 2022 benchmarks, we selected P31. The 2020 benchmark is the average of the corresponding setting's four rounds from the previous table. We summarize such a table to give an overview of JenTab's performance in Figure 6.16. From the figure, we point out the continuous enhancements that we developed to reduce the required time for the system across its years of development.

6.9 Summary

We introduced JenTab toolkit that tackles three tasks of the Semantic Table Interpretation (STI). It represents our contribution to the first research area Tabular data interpretation (TabI). JenTab is a distributed and modular system that follows the Create, Filter and Select (CFS) pattern. It is configured with various pipelines that we implemented based on the characteristics of the evaluation benchmarks. We developed and tested JenTab in the context of Semantic Web Challenge on Tabular Data to Knowledge Graph Matching

Table 6.7: Execution time for different setups (SemTab2020 benchmark).

Mode	R1		R2		R3		R4	
	Days	Runners	Days	Runners	Days	Runners	Days	Runners
P31	0.5	4	2.5	4	1.5	6	2	4
2Hops	1	4	1.2	4	2	4	1.1	8
Multi Hops	1	4	1.5	4	2.5	6	3.5	6

Table 6.8: Execution time for different setups (SemTab 2020-2022 benchmarks).

Mode	Year	Dataset	Target	Runner	Time
P31	2020	SemTab2020	Wikidata	4.5	1.5 Days
	2022	HardTables	Wikidata	3	11 Hours
	2022	2T	DBpedia	1	5 Hours
	2022	2T	Wikidata	1	3 Hours
	2022	BiodivTab	DBpedia	1	1 Hour
2 Hops	2020	SemTab2020	Wikidata	5	1.33 Days
	2021	HardTables	Wikidata	1	10.5 Hours
	2021	2T	DBpedia	1	1 Day
	2021	2T	Wikidata	2	11 Hours
	2021	BioTables	Wikidata	1	7 Hours
	2021	BiodivTab	Wikidata	1	1.5 Hours

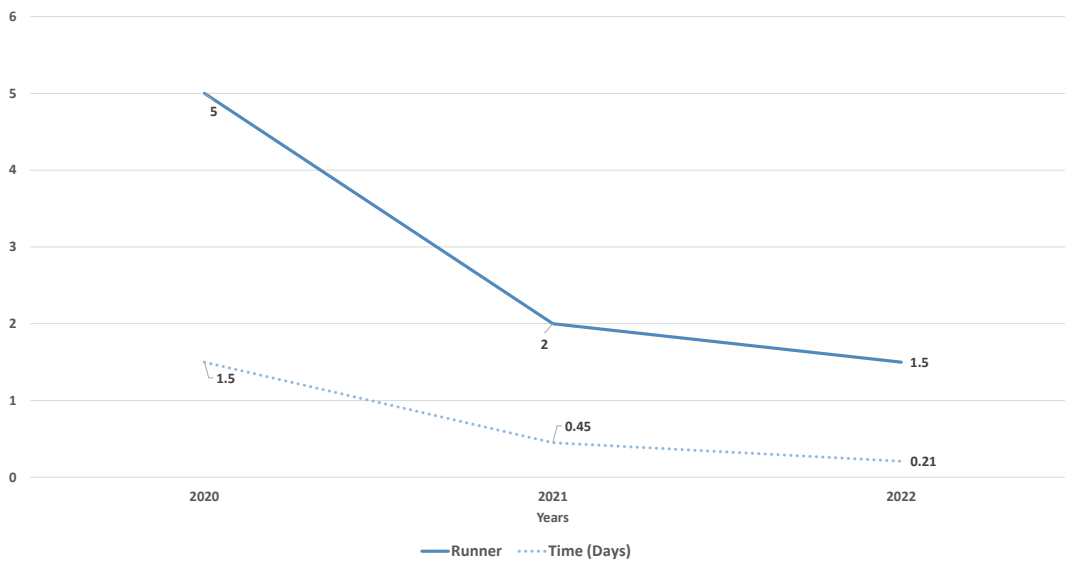


Figure 6.16: Runners and runtime of JenTab [2020-2022].

(SemTab) challenge from 2020 to 2022. Our evaluation demonstrated the effectiveness of JenTab on both Automatically Generated (AG) and domain-specific datasets for its three years of development. We also showed the enhancements that reflect the lower processing time over the years of development. JenTab is a top participant system that tackles STI tasks with minimal dependencies. We only use the live lookup and SPARQL endpoints of the target Knowledge Graph (KG). JenTab won the second prize in the Usability Track by IBM Research during ISWC 2021. We released the artifacts of this chapter publicly available under our GitHub¹⁵ repository. In addition, we made the code [102, 103, 104], pre-computed lookup [105, 106], and solution files [107, 108] available at Zenodo.

¹⁵<https://github.com/fusion-jena/JenTab>

Chapter 7

BiodivTab: Table Annotation Benchmark

Systems that tackle annotating tabular data semantically have gained increasing attention from the community in recent years. Semantic Table Interpretation (STI) tasks map individual table elements to their counterparts from a Knowledge Graph (KG) such as Wikidata [92], and DBpedia [93]. Here, individual cells and columns are assigned to KG entities and classes to disambiguate their meaning. The Semantic Web Challenge on Tabular Data to Knowledge Graph Matching (SemTab)¹ opened the call for semantic interpretation of tabular data inviting automated annotation systems. It established a common standard for evaluating those systems [22, 23, 24]. Most of its benchmarks are auto-generated with no particular domain focus [67, 68, 70, 72, 53, 60]. The 2T [69], introduced in the 2021 edition of the challenge, is the only exception involving manual curation but is still artificially derived from general domain data. Real-world and domain-specific datasets pose different challenges as witnessed, e.g., by evaluation campaigns in other domains like semantic web services evaluations [109]. STI-systems achieve high scores on the existing, synthetic benchmarks but often struggle on real-world datasets. Therefore, the development of STI systems has to be accompanied by suitable benchmarks to make them applicable in real-world scenarios. Such benchmark should reflect idiosyncrasies and challenges immanent in different domains.

In our work, we focus on the biodiversity domain. It is imperative to monitor the current state of biodiversity and its change over time and understand its driving forces to preserve life in all its varieties. The recent IPBES worldwide evaluation² predicts a dramatic decrease in biodiversity, causing an obvious decay in vital ecological functions. An expanding volume of heterogeneous data, especially tables, is produced and publicly shared in the biodiversity domain. Tapping into this wealth of information requires two main steps: On the one hand, individual datasets have to be fit for (re)use, which is a requirement that resulted in the FAIR principles [2]. On the other hand, complex analyses often require data of different sources, e.g., to examine the various interdependencies among processes in an ecosystem. The involved datasets need to be integrated which requires a certain degree of harmonization and mappings between them [3]. The semantic annotation of the respective datasets can substantially support both goals.

We constructed BiodivTab as a biodiversity-specific benchmark for STI-tasks. We made this benchmark available for public use and evaluating STI-systems in the scope of SemTab 2021-2022. BiodivTab is used to evaluate our developed framework, JenTab (see Chapter 6) for Semantic Table Interpretation (STI). In our scope, BiodivTab presents the second contribution under the first research area in this thesis **Tabular data interpret-**

¹<https://www.cs.ox.ac.uk/isg/challenges/sem-tab/>

²<https://ipbes.net/global-assessment>

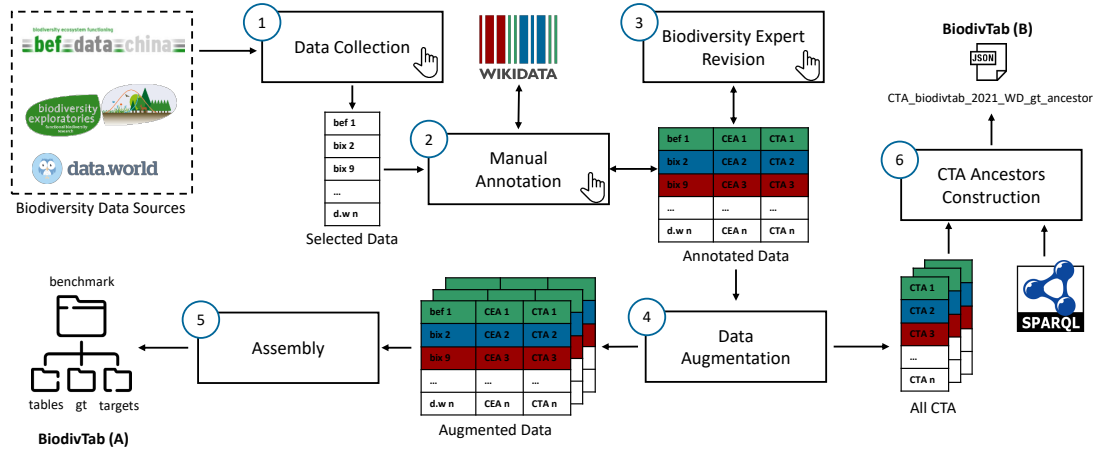


Figure 7.1: Steps of BiodivTab construction (from [14]).

ation (TabI). BiodivTab is awarded the first place prize (Applications Track) by IBM Research³ during ISWC 2021 [24].

In this chapter, we explain the six-stage construction pipeline in Section 7.1. Then, we show the characteristics of BiodivTab, and its availability and license Section 7.2. We summarize and conclude this chapter in Section 7.3⁴.

7.1 Construction Pipeline

In this section, we explain the creation of BiodivTab, and the data sources used. Moreover, we describe the manual annotation phase involving biodiversity experts, the data augmentation step, and the final assembly and release of the benchmark. Figure 7.1 summarizes the construction of BiodivTab, we detail in the following.

7.1.1 Data Collection

We decided on three data repositories that are well established for the ecological data: BExIS⁵, BEFChina⁶, and data.world⁷. We queried these portals using 20 keywords, e.g., abundance, and species, from our previous work [15]. Subsequently, we manually checked all of them regarding their suitability to the STI-tasks. We discarded datasets that contained a majority of, e.g., internal database ‘ID’ columns or numerical columns without any explanation or context. We consider those datasets are impossible to annotate automatically and of little benefit to the community. Consequently, we decided to include only datasets containing essential categorical information. We selected 6 out of 32 dataset from data.world, 4 out of 15 from BExIS, and 3 out of 25 from BEFChina. data.world provides the most suitable datasets for STI, thus, it contributes about half of the datasets in BiodivTab. Our analysis of the collected data shows that, in addition to common challenges, real-world datasets feature unique characteristics. We enumerate the encountered challenges in our sample of datasets. We summarize their prevalence in Table 7.1.

- *Nested Entities*: more than one proper entity in a single cell, e.g., a chemical compound is combined with a unit of measurements.

³<https://research.ibm.com/>

⁴This chapter is based on Abdelmageed et al [13, 14]. Thanks to Sirko Schindler who participated in the conceptualization.

⁵<https://www.bexis.uni-jena.de/>

⁶<https://data.botanik.uni-halle.de/bef-china/>

⁷<https://data.world/>

Table 7.1: Prevalence of challenges among the selected datasets (from [14]).

Dataset	Nested Entities	Acronyms	Typos	Numerical Data	Missing Values	Lack of Context	Synecdoche	Specimen Data
dataworld_1	○	●	○	●	●	●	○	●
dataworld_2	●	●	○	●	●	●	○	●
dataworld_4	●	●	●	●	●	●	●	●
dataworld_6	●	●	●	●	●	●	○	●
dataworld_10	○	●	●	●	●	○	○	●
dataworld_27	●	○	○	○	●	○	○	●
befchina_1	●	●	●	●	●	●	○	●
befchina_6	●	○	●	○	○	●	○	●
befchina_20	○	●	○	●	●	●	○	●
Bexis_24867	○	●	○	●	●	●	●	●
Bexis_25126	○	●	○	●	●	○	○	●
Bexis_25786	○	○	○	●	○	○	●	●
Bexis_27228	○	○	○	●	○	○	●	●

- *Acronyms*: Abbreviations of different sorts are common, e.g., ‘Canna glauca’, a particular kind of flower, is often referred to as ‘C.glauca’ or ‘Ca.glauce’.
- *Typos*: Data is predominantly collected manually by humans, so misspellings will occur, e.g., ‘Dead Leav’ is used for ‘Dead Leaves’.
- *Numerical Data*: Most of the collected datasets describe the specimen by various measurements in numerical form.
- *Missing Values*: Data collected can be sparse and may include gaps, e.g., a column ‘super kingdoms’ may consist of ‘unknown’ values for the most part.
- *Lack of Context*: The collected data may barely provide any informative context to facilitate semantic annotations. e.g., a column with a missing or severely misspelled header.
- *Synecdoche*: Scientists may use a general entity as a short form to a more particular one, e.g., ‘Kentucky’ is used instead of ‘Kentucky River’.
- *Specimen Data*: The collected datasets contain observations of particular specimens or groups, but do not pertain to the species as a whole.

7.1.2 Annotation Process

The annotation phase is the most time-consuming part of the benchmark creation. To ensure the quality of mappings, we manually annotated the selected tables with entities assembled from the live edition of Wikidata during September 2021, resulting in ground truth data for both CEA and CTA tasks. Concerning CEA, we have marked possible candidate columns, typically those with categorical values, to annotate their cells. For each cell value, we assembled possible matches via Wikidata’s built-in search. We manually selected the most suitable matches to disambiguate the cells semantically if we found multiple matches. If we could not have chosen only one annotation, we pick all possible ones and consider them true matches. Thus, the provided ground truth contains all proper candidates for a given cell value. Biodiversity experts revised around 1/3 of the annotations. This revealed an error rate of about 1%. Because of the low error rate, the

Table 7.2: Questionnaire: Which type would be correct for the given taxons? (from [14])

Taxon	Type (A)	Type (B)
Bacteria (wd:Q10876)	superkingdom (wd:Q19858692)	taxon (wd:Q16521)
Actinobacteria (wd:Q130914)	phylum (wd:Q38348)	taxon (wd:Q16521)
Actinobacteriales (wd:Q26262282)	class (wd:Q37517)	taxon (wd:Q16521)
Pseudonocardiales (wd:Q26265279)	order (wd:Q36602)	taxon (wd:Q16521)
Pseudonocardiales (wd:Q7255180)	family (wd:Q35409)	taxon (wd:Q16521)
Goodfellowiella (wd:Q26219639)	genus (wd:Q34740)	taxon (wd:Q16521)
Goodfellowiella coeruleoviolacea (wd:Q25859622)	species (wd:Q7432)	taxon (wd:Q16521)

effort of this step outweighs the benefits. Thus, we have decided to continue annotating the remainder without further revisions.

We followed the same procedure for CTA. For categorical columns, we looked for a common type among column cells, taking into consideration the header value, to decide on the semantic type from Wikidata. Most of these columns are identified by the value of (wdt:P31, instance of) as the perfect annotation. However, finding such perfect annotation for taxon-related columns is not that easy. Since all taxon-related fields are instance of `taxon`. We believed it might not be distinguishable enough. In the biodiversity domain, experts are keen on more fine-grained modeling. E.g., species, genus, and class would be different types in their opinion. We established a simple one-question questionnaire for our biodiversity experts to select the perfect semantic type for a given taxonomic term as shown in Table 7.2. The first column shows the cell values with the corresponding mapping entities. The question is to select either which type is the most accurate, A, or B. We derive Type A from (wdt:P105, taxon rank) and Type B from (wdt:P31, instance of) in Wikidata. Based on their answers, the most fine-grained classification is (Type A); however, they consider (Type B) as a correct type as well. Thus, we have selected the perfect types for taxons through (wdt:P105, taxon rank). For numerical columns, most of them are identified by the column headers.

We maintain separate ground truth files to ease manual inspection, revision, and quality assurance for each table. So, ‘befchina_1’, e.g., is annotated by two such files: ‘befchina_1_CEA’ and ‘befchina_1_CTA’. The structure of the ground truth files follows the format of SemTab challenge. In particular, the solution files for CEA use a format of *filename*, *column id*, *row id*, and *ground truth*, whereas the ones for CTA employ a structure of *filename*, *column id*, and *ground truth*.

7.1.3 Data Augmentation

We further used data augmentation to increase the number of tables in our benchmark and reduce the human effort needed. In our context, we introduced challenges to the existing datasets based on our findings during the data collection and analysis phase, thus we rely on real-world challenges that we added programmatically to increase the amount of the data. Table 7.3 shows our used data augmentation techniques per dataset and the number of variations derived from it. In the following, we list techniques used and how they relate to the collected data issues:

- *Merge and Separate Columns* we either by introduced new nested entities or splited them up into separate columns.
- *Add and Fix Typos* we added noise to categorical cell values and, on rare occasions, fixed them.
- *Disambiguate* we replaced concepts with more accurate ones, e.g., the state is replaced by the river it stands for.

Table 7.3: Data augmentation technique per dataset (from [14]).

Dataset	Merge Cols	Separate Cols	Add Typos	Fix Typos	Disambiguate	Abbreviate	Increase Gap	Alter Cols	No. Files
dataworld_1	x3	-	-	-	-	-	x3	x1	7
dataworld_2	x3	-	x1	-	-	-	x1	-	5
dataworld_4	x4	-	-	x1	x1	x2	x1	-	9
dataworld_6	-	x1	-	-	-	-	-	-	1
dataworld_10	-	-	-	-	-	-	x1	-	1
dataworld_27	x1	-	x 2	-	-	-	-	-	3
befchina_1	x2	-	-	-	-	-	x1	-	3
befchina_20	x4	-	-	-	-	x1	-	-	5
Bexis_24867	-	-	-	-	-	x1	x2	-	3
Total									37

- *Abbreviate* we introduced more abbreviations especially with taxon-related values.
- *Alter Columns* we removed one or more data columns. This results in less informative and sparse datasets.

We managed to create the most variations from data.world since its datasets contain more categorical data that can be mapped to KG entities. Our used data augmentation strategy allowed us to increase the number of tables to 50 with less manual effort of the annotation.

7.1.4 General Semantic Types

To enable approximation of CTA F1, Precision and Recall scores [23], we provide an ancestors ground truth to our perfectly annotated types⁸. The corresponding file is structured in a key-value format with keys representing the perfect annotation and values listing parent classes. We refer to those parents as *okay* classes.

Initially, we collected all unique column types from manually assigned perfect annotations. These are used to initialize a dictionary. Afterwards, we ran a sequence of three SPARQL queries sent to the public endpoint to retrieve related classes for each of them. For the first level, we query for direct types via `wdt:P31`. We call them ‘E1’. For the second level, we query for further parent classes via `wdt:P279` of the previous E1, resulting in ‘E2’. For the third and last level, we repeat the last process using the entities in E2, yielding ‘E3’. The resultant hierarchy consists of one perfect annotation with three levels of classes that are considered okay annotations.

We marked the fine-grained taxonomy: kingdom, species, phylum, family, order, class, and genus as perfect annotations to follow the biodiversity experts’ recommendation. However, we have included (`wd:Q16521`, taxon) and (`wd:Q21871294`, living organism) as okay classes.

7.1.5 Assembly and Release

For publication, we anonymized the file names of tables to use unique identifiers using Python’s `uuid` functionalities. Subsequently, we aggregated the individual solutions of CEA and CTA-tasks into one file per task resulting in `CEA_biodivtab_2021_gt.csv` and `CTA_biodivtab_2021_gt.csv` respectively. We generated the corresponding ‘target-files’ by removing the ground truth columns from these solution files. We provided anonymized tables alongside the target files to evaluate a particular system. The ground truth

⁸https://raw.githubusercontent.com/fusion-jena/BiodivTab/main/benchmark/gt/CTA_biodivtab_2021_WD_gt_ancestor.json

files alongside the dictionary for related classes, CTA-ancestors, are subsequently used to evaluate the results. Such way this follows the general approach of SemTab hiding the ground truth of STI-tasks from participants during the challenge. BiodivTab is awarded the first prize of IBM Research⁹ at the third round of 2021’s SemTab challenge [24] for its new challenges in CEA and CTA tasks.

7.1.6 Ground Truth Extension

In 2022, we included annotations from DBpedia that are based on the Wikidata annotations in two ways: First, we exploited the link between Wikidata entities and corresponding Wikipedia pages. As there is a one-to-one correspondence between Wikipedia pages and DBpedia entities, we generated a Wikidata-DBpedia-mapping for them. Second, we extracted `owl:sameAs` mappings between Wikidata and DBpedia to complete our mapping from DBpedia itself. Despite these direct mappings appeared promising to begin with, they contain serious data quality issues. As of April 2022, *L-glutamic acid* (`wd:Q26995161`) is mapped to 1772 entities within the DBpedia graph using `owl:sameAs`. Thus, the resulting mappings were again manually verified to ensure the overall quality of the final DBpedia ground truth data. Generated types for CTA contained only instances/resources from DBpedia. During the manual verification, we further added classes from the DBpedia ontology as well. We attempted to replicate our approach from Wikidata using `rdf:type` and `rdfs:subClassOf` to retrieve the CTA-ancestors. However, some relations in the DBpedia ontology seemed unreasonable to us. For example, DBpedia at the time of writing contains a triple `dbr:Species rdf:type dbo:MilitaryUnit`. For these and other similar scenarios, we decided to not include an ancestor file for DBpedia.

7.2 Evaluation

In this section, we give a detailed overview of BiodivTab in terms of the size and content compared to existing benchmarks. In addition, we show the most and least frequent types of CTA. Finally, we demonstrate the application of our benchmark using the results of STI-systems during SemTab’s 2021 edition.

7.2.1 BiodivTab Insights

Table 7.4 summarizes the selected datasets in terms of their original and selected size, and the number of CEA and CTA mappings. For large datasets, e.g., *dataworld_4* and *dataworld_27*, we selected a subset of rows that retain the table characteristics. Most of the redundant species were dropped. Nevertheless, we kept the entire extent of BExIS datasets, including the redundant entries, to achieve a good balance between the large tables and those with the reasonable length for STI-systems. The column mappings show the characteristic of specimen data, those columns with only local measurements and with local database names that could not be matched to the KG. For example, only 4 out of 18 columns in *dataworld_1* could be matched to KG-entities. The same holds true for *befchina_1* where only 3 out of 16 columns have a proper annotation.

Figure 7.2 shows the domains distribution of the 83 unique semantic types in the CTA-solutions. Approximately two-thirds of these types belong to the biodiversity domain. The distinction into the biodiversity-related, general domain, and mixed types was made according to the definitions introduced in [15, 28]. General domain types include, e.g., visibility, scale, cost, and airport. Mixed domain types contain examples like river,

⁹<https://www.research.ibm.com/>

Table 7.4: Original and selected tables sizes, and entity and type mappings (from [14]).

Dataset	Original Size		Selected Size		Mappings	
	Rows	Cols	Rows	Cols	CTA	CEA
dataworld_1	332	18	100	18	4	210
dataworld_2	37	25	37	8	8	226
dataworld_4	42337	67	100	40	26	476
dataworld_6	271	6	100	6	4	103
dataworld_10	497	15	100	13	11	902
dataworld_27	95368	12	100	12	5	398
befchina_1	7553	16	145	16	3	294
befchina_6	26	4	26	4	2	53
befchina_20	787	45	99	43	28	304
Bexis_24867	151	13	151	13	9	159
Bexis_25126	4906	35	4906	14	6	9816
Bexis_25786	2001	39	2001	21	5	4017
Bexis_27228	1549	8	1549	8	3	4646
Total					114	21604
Avg.					8.8	1661.8

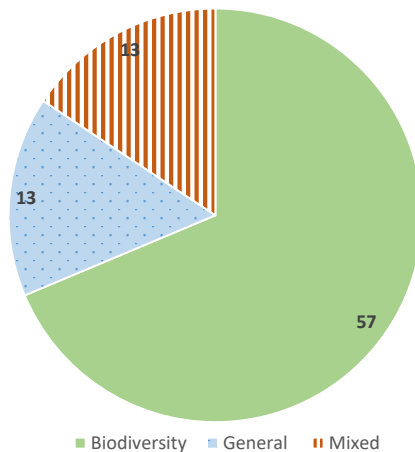


Figure 7.2: Domain distribution in BiodivTab benchmark (from [14]).

Table 7.5: Most and least frequent semantic types in BiodivTab (from [14]).

Most Frequent			Least Frequent		
Wikidata Id	Label	Freq.	Wikidata Id	Label	Freq.
wd:Q7432	Species	39	wd:Q8066	amino acid	1
wd:Q706	calcium	26	wd:Q11173	chemical compound	1
wd:Q577	year ¹⁰	19	wd:Q60026969	unit of concentration	1
wd:Q677	iron	16	wd:Q2463705	Special Protection Area	1
wd:Q731	manganese	16	wd:Q1061524	intensity	1

temperature, or sex of humans. Biodiversity-related types include taxon, chemical compounds, and soil type. In addition, Table 7.5 provides a list of most and least frequent semantic types in BiodivTab. Species (wd:Q7432) is the most frequent type, which reflects its importance in biodiversity research.

¹⁰Calendar year, wd:Q3186692, is equivalent to year, wd:Q577.

Table 7.6: Data sources for existing benchmarks and their corresponding targets. Entries for SemTab are aggregated over all rounds each (from [14]).

Dataset	Data Source	Target Annotation
SemTab 2019	Wikidata, Wikipedia	DBpedia
SemTab 2020	Wikidata, Wikipedia	Wikidata
SemTab 2021	Wikidata, Wikipedia	Wikidata, DBpedia
T2Dv2	WebTables	DBpedia
Limaye	Wikipedia	DBpedia
GitTables	GitHub	DBpedia, Schema.org
BiodivTab	BExIS, BEFChina, data.world	Wikidata, DBpedia

Table 7.6 shows both data sources and target KGs or resource for BiodivTab and existing benchmarks. The three editions of SemTab from 2019 to 2021 [22, 23, 24] used both Wikidata and Wikipedia [110] as table sources. However, the target KGs varies between using DBpedia, Wikidata, or both. T2Dv2 [60] and Limaye [53] use the WebTables [66] and Wikipedia as their data sources respectively while having annotations from DBpedia. GitTables [72] and the adapted version [71] for SemTab 2021 challenge, leverages GitHub as a table source and provide annotations from DBpedia and schema.org. Unlike all the previous benchmarks, BiodivTab uses domain-specific data portals, as table sources. It provides Wikidata and DBpedia annotations like SemTab 2021.

Table 7.7 shows a comparison between BiodivTab and existing benchmarks in terms of the average number of rows, columns, and cells. It also gives an overview of the targets for CEA, CTA, and CPA. BiodivTab is the smallest in terms of the number of tables. However, BiodivTab has the maximum average number of columns, and average number of rows except for SemTab 2021, Round 1, and BioTables in Round 2. This poses an additional challenge for STI systems. For CTA targets, BiodivTab is a middle point among the existing benchmarks.

7.2.2 Availability and Licensing

Resources should be easily accessible to allow replication and reuse. We follow the FAIR (Findable, Accessible, Interoperable, and Reusable) guidelines to publish our contributions [2]. We release our dataset [111] in such a way that researchers in the community can benefit from it. In addition, we release the code [112] that was used to augment the data, assemble, and reconcile the benchmark. Our dataset and code are released under the Creative Commons Attribution 4.0 International (CC BY 4.0) License, and Apache License 2.0 respectively. This should support replicability and subsequent extensions.

7.3 Summary

We introduced BiodivTab, the first biodiversity tabular benchmark for Semantic Table Interpretation tasks. It consists of a collection of 50 tables. We have created BiodivTab by manually annotating 13 tables from real-world biodiversity datasets and adding 37 more tables by augmenting them with noise based on challenges that are commonly observed in the domain. Annotations are based on Wikidata and DBpedia as target knowledge graphs. An evaluation during the SemTab challenge showed that current state-of-the-art systems still struggle with the posed domain-specific challenges. This highlights BiodivTab’s importance as a basis for further development in the field. BiodivTab itself and

Table 7.7: Comparison with existing benchmarks (from [14]). *ST19 - ST21* (SemTab editions). *_*W* and *_*D* use Wikidata and DBpedia as targets. *ST21-H2*, and *H3* are HardTables for Round 2 and 3 during SemTab. *ST21-Bio* is BioTables at SemTab Round 2. *ST21-Git* is the published version of GitTables during SemTab Round 3.

Dataset	Tables	Avg. Rows (\pm Std Dev.)	Avg. Cols (\pm Std Dev.)	Avg. Cells (\pm Std Dev.)	CEA	CTA	CPA
ST19-R1	64	142 \pm 139	5 \pm 2	696 \pm 715	8,418	120	116
ST19-R2	11,924	25 \pm 52	5 \pm 3	124 \pm 281	463,796	14,780	6,762
ST19-R3	2,161	71 \pm 58	5 \pm 1	313 \pm 262	406,827	5,752	7,575
ST19-R4	817	63 \pm 52	4 \pm 1	268 \pm 223	107,352	1,732	2,747
ST20-R1	34,294	7 \pm 4	5 \pm 1	36 \pm 20	985,110	34,294	135,774
ST20-R2	12,173	7 \pm 7	5 \pm 1	36 \pm 18	283,446	26,726	43,753
ST20-R3	62,614	7 \pm 5	4 \pm 1	23 \pm 18	768,324	97,585	166,633
ST20-R4	22,390	109 \pm 11,120	4 \pm 1	342 \pm 33,362	1,662,164	32,461	56,475
ST21-R1_ <i>W</i>	180	1,080 \pm 2,798	5 \pm 2	4125 \pm 10947	663,655	539	NA
ST21-R1_ <i>D</i>	180	1,080 \pm 2,798	4 \pm 2	3,952 \pm 10,129	636,185	535	NA
ST21-H2	1,750	17 \pm 8	3 \pm 1	55 \pm 32	47,439	2,190	3,835
ST21-Bio	110	2,448 \pm 193	6 \pm 1	14,605 \pm 2,338	1,391,324	656	546
ST21-H3	7,207	8 \pm 5	2 \pm 1	20 \pm 15	58,948	7,206	10,694
ST21-Git	1,101	58 \pm 95	16 \pm 12	690 \pm 1,159	NA	2,516	NA
ST21-Git	1,101	58 \pm 95	16 \pm 12	690 \pm 1,159	NA	720	NA
T2Dv2	779	85 \pm 270	5 \pm 3	359 \pm 882	NA	237	NA
Limaye	428	24 \pm 22	2 \pm 1	51 \pm 50	NA	84	NA
BiodivTab (<i>W</i>)	50	259 \pm 743	24 \pm 13	4,589 \pm 10,862	33,405	614	NA
BiodivTab (<i>D</i>)	50	259 \pm 743	24 \pm 13	4,589 \pm 10,862	33,405	569	NA

the code that is used to create it are available under our GitHub repository¹¹. In addition, we released the benchmark data at Zenodo [111, 112].

¹¹<https://github.com/fusion-jena/BiodivTab>

Part III

Textual Data Interpretation

Chapter 8

BiodivOnto: Biodiversity Data Model

Understanding biodiversity and the mechanisms underlying it is crucial to preserve this important foundation of human well-being. This demands the management and integration of biodiversity data [3]. A large amount of heterogeneous data is collected and generated in biodiversity research, which means integrating these heterogeneous data remains a big challenge. Semantic web in general and ontologies in particular play a vital role in coping with the integration and management of these heterogeneous data by allowing representing the relevant concepts and relations of a considered domain in a machine-readable format [113]. As a result, several domain-specific ontologies have been developed. For example, statistics on BioPortal¹ show that more than 1046 ontologies with more than 13 million concepts have been developed. Several domain ontologies like ENVO² and IOBC³ exist to model specific areas in the biodiversity domain [114]. However, there is a growing need to bridge the more refined biodiversity concepts and general concepts provided by the foundational ontologies. Foundational ontologies span many fields, modeling the basic concepts and relations that make up the world [115]. Core ontologies provide a precise definition of structural knowledge in a specific field that spans different application domains [116]. Hence, core ontologies provide a bridge between the foundational and subdomain ontologies. Several efforts have been made in different domains to represent the basic categories of the domain knowledge using core ontologies. Several approaches exist in the development of core ontologies, including manual and (semi)automatic ways.

We present the design of a core ontology, ‘BiodivOnto’, for the biodiversity domain. BiodivOnto represent our data model as a set of concepts and relations of interest. In our context, under the second research area **Textual data interpretation (TexI)**, such data model determines both classes and their relationships that a textual data interpreter would extract from unstructured text. To construct this model, we use a semi-automatic approach that includes the usage of fusion/merge strategy [117] for the core ontology development. We developed a four-phase pipeline with biodiversity experts and computer scientists involved at different stages. We collected and analyzed a set of heterogeneous biodiversity data sources, including tabular data, unstructured data, and metadata. To extract keywords from the collected data repositories, we used existing ontologies from BioPortal⁴ and AgroPortal⁵. We applied biodiversity experts’ recommendations to filter the keywords of interest. We generated the core concepts using automated approaches of

¹<https://biportal.bioontology.org/>, visited on 14.01.2023

²<https://biportal.bioontology.org/ontologies/ENVO>

³<https://biportal.bioontology.org/ontologies/IOBC>

⁴<https://biportal.bioontology.org/>

⁵<http://agroportal.lirmm.fr>

clustering. The relations between these core concepts are discussed and determined by the domain experts. We continuously evolved the BiodivOnto by leveraging more biodiversity experts' opinions as well as by intergarting other resources.

In this chapter, we give a brief overview on the existing ontologies showing their limitations in Section 8.1. We describe our data-driven approach in Section 8.2. We demonstrate the results and initial conceptual model of our method in Section 8.3. We show the evolutionary steps of BiodivOnto using existing resources in Section 8.4. We summarize and conclude this chapter in Section 8.5⁶.

8.1 Existing Ontologies

Biodiversity aims to study the totality and variability of organisms, their morphology and genetics, life history and habitats, and geographical ranges. It is strongly related to ecosystems' services, such as provision of water and food, and climate regulation. Therefore, it is critically important to understand and conserve it properly [3]. Core ontologies provide a precise definition of structural knowledge in a specific field that connects different application domains [118, 119, 116]. They are located between upper-level (foundation) and domain-specific ontologies, defining the core concepts of a specific field. They aim at linking general concepts of a top-level ontology to more domain-specific concepts from a sub-field.

There is a large number of available foundational ontologies [120], such as BFO [121], GFO [122], SUMO [123], PROTON [124] and, etc. At the same time, there is extensive work to formalize knowledge in the biodiversity domain, which results in many domain-specific ontologies. For example, there are 1046 ontologies in BioPortal among them 25 are titled core ontologies. The core ontology for biology and biomedicine (COB)⁷ and the ontology for core ecological entities (ECOCORE)⁸ are the only two relevant biodiversity core ontologies. The COB ontology has 73 concepts and 30 relations, while the ECOCORE ontology has more than 2400 concepts. The start of developing both ontologies was in 2020, which indicates a growing interest in developing such core ontologies. However, for both of them, detailed information on how these ontologies have been developed is missing.

A few core ontologies have been introduced in the biodiversity domain; however, several core ontologies developed in other related domains. The work introduced in [125] propose the design of a core ontology to deal with the different types of research activities performed in empirical research, encompassing (physical) sampling, sample preparation, and measurement. SemSur is a core ontology for the semantic representation of research findings [118]. The GeoCore ontology has been developed to be used as a core ontology for general use in the geology domain [119]. It makes use of the BFO ontology as an upper-level ontology.

According to [116], core ontologies should combine various features, such as axiomatization, modularity, extensibility, and reusability. Developing a core ontology following these features leads to an elegant way to achieve good interoperability in a complex domain, such as the biodiversity domain. There are different strategies to develop ontologies considering these features, such as fusion/merge and composition/integration strategies [117]. In this chapter, we use the fusion/merge strategy that builds an ontology by bringing together knowledge from source ontologies. Such ontology is our data model that consists of both concepts/classes and their relations that we use to interpret textual data.

⁶The early stage of the ontology is based on Abdelmageed et al [15, 16]. The author of this thesis contributed in every step to create the BiodivOnto. The evolution of the ontology is based on Abdelmageed et al [17]. Felicitas Löffler, Alsayed Algergawy, and Sheeba Samuel analyzed the Biodiversity Questions Corpus. The author of this thesis applied further analysis after the pervious setp and updated the ontology.

⁷<http://purl.obolibrary.org/obo/cob.owl>

⁸<http://purl.obolibrary.org/obo/ecocore.owl>



Figure 8.1: Proposed four-phase pipeline (from [16]).

8.2 Methodology

We implemented the proposed data-driven approach using the pipeline shown in Figure 8.1. It consists of four main phases: 1) Data Acquisition, we identified the data sources we use to develop the target ontology. 2) Term Extraction, we manually grasped biodiversity-related keywords from the collected data. 3) Term Filtration, we programmatically filtered the extracted keywords to represent our core concepts. Finally, 4) Concepts and Properties Determination, we decided on the final and selected concepts, and relations with the aid of our biodiversity domain experts. In the following, we describe these main steps of the proposed pipeline.

8.2.1 Data Acquisition

A first and crucial step is collecting and preparing a sufficient and relevant set of data sources from which we can extract core terms in the biodiversity domain. These data sources should be diverse, including structured data (tabular) and unstructured data (abstracts). To achieve this goal, we developed a crawling method, as shown in Figure 8.2. We have considered two important factors during this step: (i) *data resources*, from which data sources will be extracted from and (ii) *a set of keywords* that will be used to query these data resources. For the first point, we considered two well known data portals with very different characteristics (*BEFChina*⁹ and *data.world*¹⁰) to get tabular data. PubMed¹¹ with more than 35 Million abstracts is deemed to be the data resource for unstructured data. Once identified data sources, the next step is to collect a set of domain-specific keywords that will be used to query these data resources. To this end, we relaxed a version of the QEMP corpus' keywords [29], such as 'abundance', 'benthic', 'biomass', 'carbon', 'climate change', 'decomposition', 'earthworms', 'ecosystem' are selected. The selected set of keywords is used later as an input to the Semedico search engine [126] to get relevant publications from PubMed. Among them, 100 abstracts have been chosen, as shown in Figure 8.2 reflecting the biodiversity domain by applying an iterative manual process for revision and cleaning for the crawled data. The result of this phase is a data repository¹² which contains 100 abstracts, more than 50 tables, some datasets are given by multiple tables and, 50 metadata files. Our selected number of these data sources achieves a balance between biodiversity domain coverage and reasonable human labor time for annotation.

8.2.2 Term Extraction

Once relevant data sources have been collected, the next step is to process them to extract domain-specific terms. To this end, we manually annotated the collected data using GATE tool¹³ for document annotation. We have followed the annotation guidelines in [29] making the use of the same ontologies and adding more important ontologies and knowledge bases,

⁹<https://china.befdata.biow.uni-leipzig.de/>

¹⁰<https://data.world/>

¹¹<https://pubmed.ncbi.nlm.nih.gov/>

¹²<https://github.com/fusion-jena/BiodivOnto/tree/main/data>

¹³<https://gate.ac.uk/documentation.html>

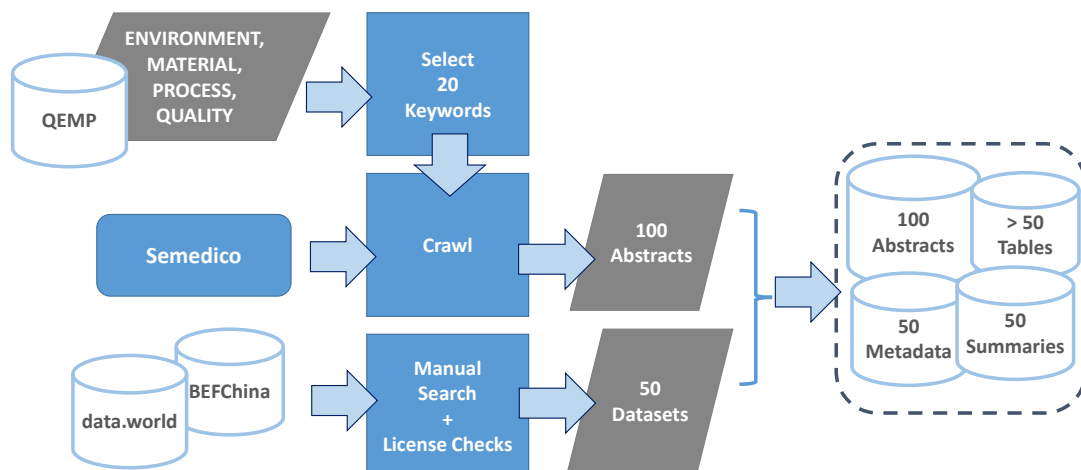


Figure 8.2: Crawling phase (from [15]).

like *IOBC*, *SWEET*¹⁴, *ECOCORE*, *ECSO*¹⁵, *CBO*¹⁶, *BCO*¹⁷, and the *Biodiversity A-Z* dictionary¹⁸ to cover wider ranges of domain-specific terms. We also used the BioPortal Annotator¹⁹ with the selected ontologies to fetch the possible annotations for a given term. The extraction and annotation process is not a simple task as it has several challenges to be addressed. On the one hand, some keywords are ambiguous; we could not decide to include them. We keep those keywords in a separate list as *Open Issues*. On the other hand, our main challenge is the handling of compound words. For example, *photosynthetic O2 production* is expanded into the following keyword list: [‘photosynthetic’, ‘O2’, ‘O2 production’, ‘photosynthetic O2 production’]. We have enriched the extracted list of terms using other existing resources: 1) annotated keywords in QEMP corpus, 2) keywords from AquaDiva²⁰ project, and 3) soil-related keywords [127]. These existing resources have 578, 222, and 410 keywords, respectively. Figure 8.3 shows that our project managed to capture the largest amount of related terms compared to the existing works.

8.2.3 Term Filtration

To get the final relevant terms, we discussed the *Open Issues* list with biodiversity experts. Based on their votes for each term, we decided on whether to include it or not. Some keywords are already filtered out manually at this stage. We applied an automatic filtration step for consistency, where we normalized keywords to be case insensitive and in a singular form. Furthermore, we manually revised the final list of keywords to exclude spelling mistakes. At the end of this step, we have 1107 unique keywords, which is 1.8x of QEMP corpus in size and covers a broader range of the biodiversity domain. Figure 8.4 illustrates the effect of this phase on the original keywords per each data source of our work, where the figure shows that the most significant number of unique keywords is collected using abstracts from PubMed using the Semedico search engine. However, Figure 8.4 shows that BEFChina has the least number of collected unique keywords. In addition, we have calculated the number of simple and complex keywords as shown in Figure 8.5. The used subset of AquaDiva project has only simple keywords, however, the soil-related keywords

¹⁴<https://bioportal.bioontology.org/ontologies/SWEET>

¹⁵<https://bioportal.bioontology.org/ontologies/ECISO>

¹⁶<https://bioportal.bioontology.org/ontologies/CBO>

¹⁷<https://bioportal.bioontology.org/ontologies/BCO>

¹⁸<https://www.biodiversitya-z.org/>

¹⁹<https://bioportal.bioontology.org/annotator>

²⁰<http://www.aquadiva.uni-jena.de/>

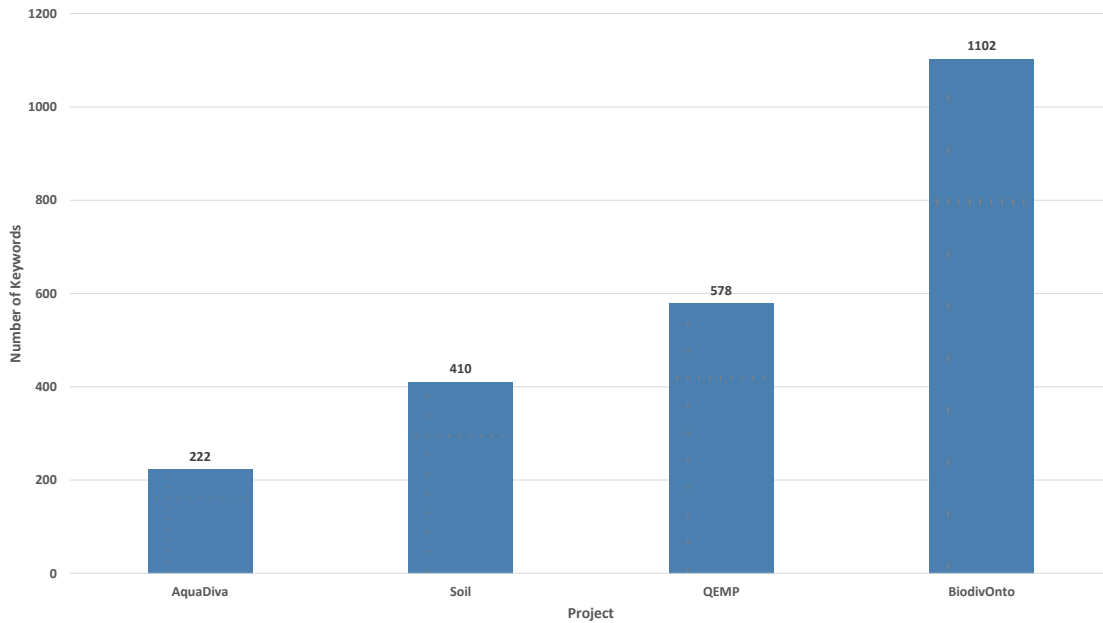


Figure 8.3: Extracted keywords vs. existing projects

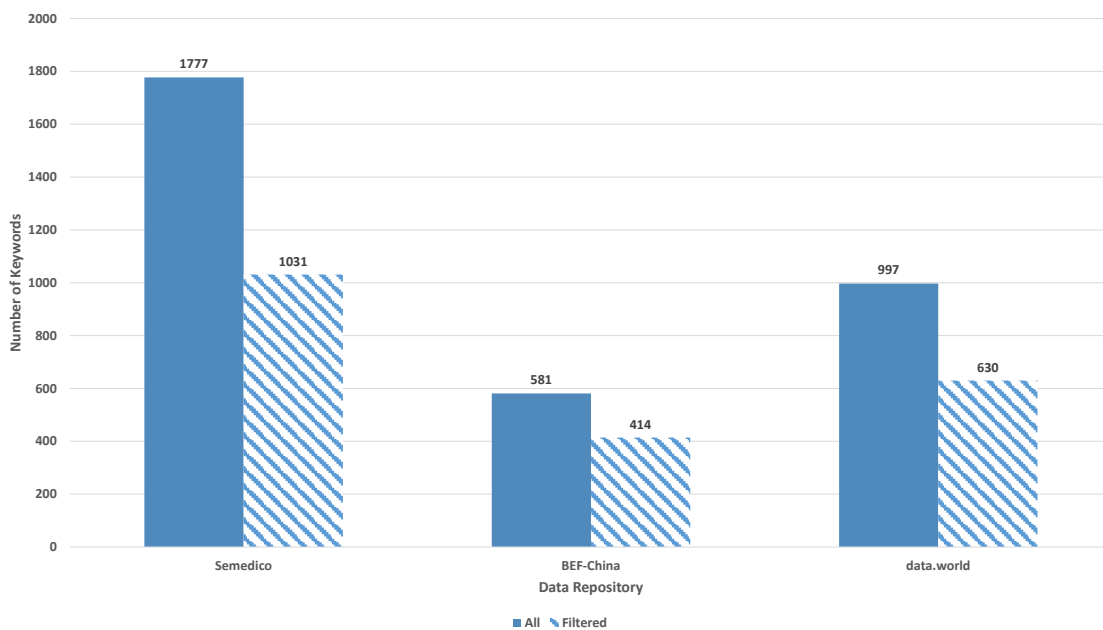


Figure 8.4: Filtration effect on the selected data sources (from [16]).

are only complex. QEMP and our work have a mixture of both, but our work achieves a better balance.

8.2.4 Concepts and Relations Determination

In this section, we cover how we have reached our initial set of core concepts and their interlinks.

8.2.4.1 Concepts Determination

Given the vast output list from the previous step, we automatically calculated the intersection among our work, QEMP, and AquaDiva lists. this yielded a narrowed list of keywords

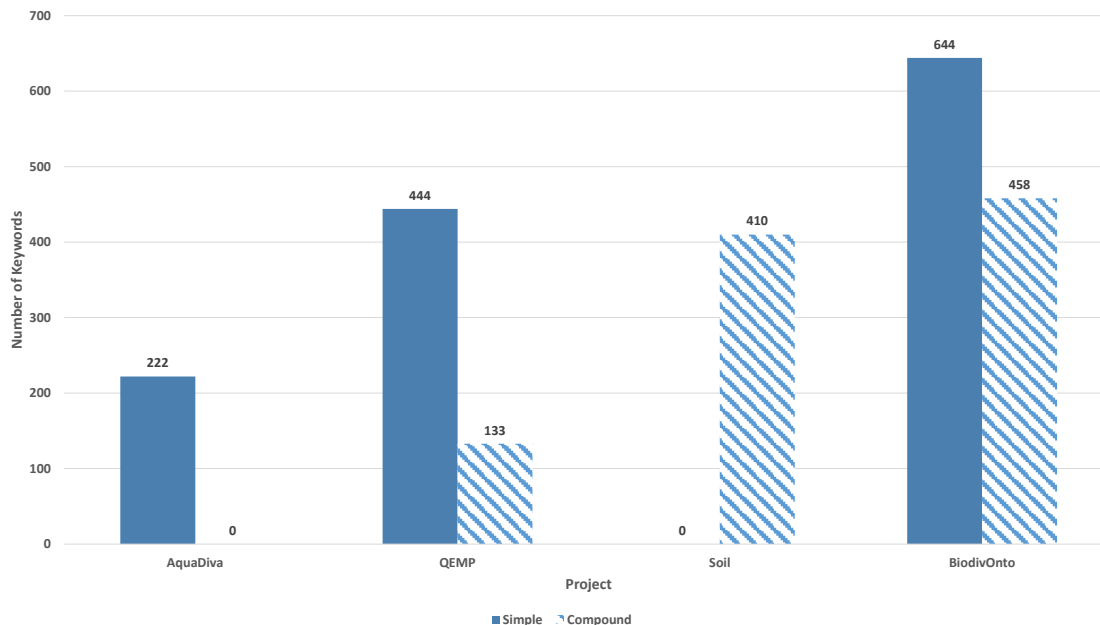


Figure 8.5: Simple vs. compound keywords in our work and compared to existing data sources (from [16]).

	depth	distance	grassland	habitat	size	temperature	nitrogen	abundance
depth	TRUE	F	F	F	TRUE	TRUE	F	TRUE
distance	F	TRUE	F	F	F	F	TRUE	F
grassland	F	F	TRUE	TRUE	F	F	F	F
habitat	F	F	TRUE	TRUE	F	F	F	F
size	TRUE	F	F	F	TRUE	TRUE	F	TRUE
temperature	TRUE	F	F	F	TRUE	TRUE	F	TRUE
nitrogen	F	TRUE	F	F	F	F	TRUE	F
abundance	TRUE	F	F	F	TRUE	TRUE	F	TRUE

Figure 8.6: A sample of seeds WordNet similarity, TRUE has a *threshold* ≥ 0.7 (from [16])

which we defined as *Seeds Candidates*²¹. For example, *carbon*, *climate*, *composition*, *forest*, *size* and etc. We considered those 30 terms, as they are the most critical and common keywords among various projects dealing with the biodiversity domain. Then, we applied a distance-based clustering technique to assign each of the remaining words to the closest seed. Word embeddings [128], [129], [130] are a good representation for words to capture their semantic meaning. For example, *grassland* is similar to *habitat* in the embedding space, so these pairs of words could be grouped in one cluster. Same case applies for *abundance* and *size*. Word embeddings are commonly used in applications that involve word-word similarity. Seeds and words are represented by 300D word embedding vectors using Word2Vec. Our selected metric is the cosine similarity, the default option to measure a distance between two vectors. Afterwards, we manually revised the created clusters multiple times. For each revision iteration, we checked how the remaining keywords are grouped, discusses the results with biodiversity experts, and modified the selected seeds by tending to more general concepts. In the last iteration, we performed the WordNet [131] similarity among the remaining seeds, clusters centroids, such that, if the similarity is 0.0, very unique seed, we picked it as a core concept. Figure 8.6 illustrates a sample of our seeds with WordNet similarity > 0.7 .

²¹<https://github.com/fusion-jena/BiodivOnto/blob/main/outcome/seeds.md>

Table 8.1: Core concepts in existing ontologies with examples [15].

Category	Ontology Modules	Terms sample inside category
Environment	ENVO, ECOCORE, ECSO, PATO	groundwater, garden
Organism	ENVO, ECOCORE, ECSO, BCO	mammal, insect
Phenomena	ENVO, PATO, BCO	decomposition, colonization
Quality	ENVO, PATO, CBO, ECSO	volume, age
Landscape	ENVO	grassland, forest
Trait	BCO	texture, structure
Ecosystem	ENVO, ECOCORE, ECSO, PATO	biome, habitat
Matter	ENVO, ECSO	carbon, H2O

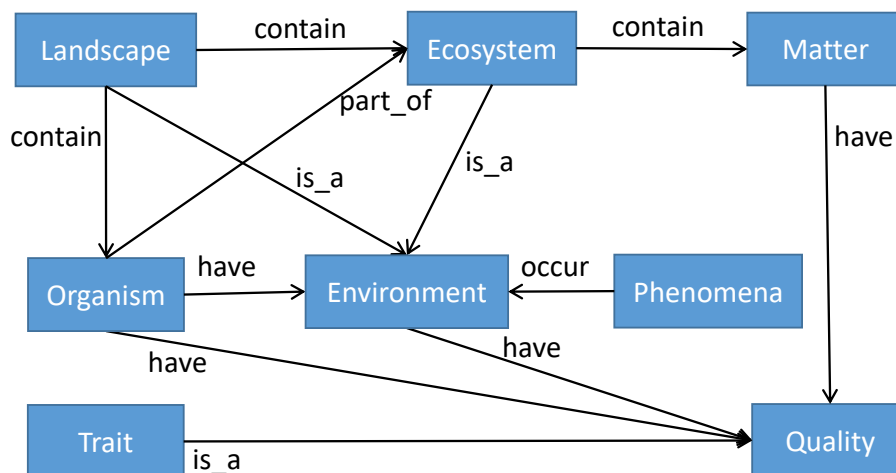


Figure 8.8: Core concepts and their relations (from [16]).

from each category. Table 8.1 demonstrates the results of this process. Figure 8.8 depicts the output in its two years of development. The next step is to combine (merge) the set of modules in each category to get a core ontology representing a certain category/core concept.

8.4 BiodivOnto Evolution

In this section, we explain the evolution of our developed conceptual model using an existing resource that we analyzed and integrated into the BiodivOnto in 2022.

8.4.1 Biodiversity Questions

The biodiversity question corpus consists of 169 questions provided by around 70 scholars of three biodiversity research-related projects [28]. Concerning the topics and granularity, the questions are very diverse and reflect different information needs. While some questions ask for facts such as ‘What butterfly species occur on calcareous grassland?’ others are more complex and aim to get an answer on associations, e.g., ‘How do autotrophic microorganisms influence carbon cycling in groundwater aquifers?’. The noun entities of these questions were manually labeled (including nested entities such as adjectives, e.g., autotrophic microorganisms). Nine biodiversity scholars grouped the labeled nouns and phrases into 13 proposed categories. Each annotator classified all 169 questions, which resulted in 592 total annotations. It turned out that seven categories (entity types) were

Table 8.2: Summary of the categories (entity types) used in NER annotation (from [17]).

Tag	Explanations	Examples
ORGANISM	all individual life forms such as microorganisms, plants, animals	mammal, insect, fungi, bacteria
PHENOMENA	occurring natural, biological, physical, or chemical processes including events	decomposition, colonization, climate change, deforestation
MATTER	chemical and biological compounds, and natural elements	carbon, H ₂ O, sediment, sand
ENVIRONMENT	Natural or man-made environments ORGANISM live in	groundwater, garden, aquarium, mountain
QUALITY	data parameters measured or observed, phenotypes and traits	volume, age, structure, morphology
LOCATION	geographic location (no coordinates)	China, United States

mentioned very often (at least 89 times per category): ORGANISM (e.g., plants and fungi), ENVIRONMENT (environments species live in), QUALITY (characteristics to be measured), MATERIAL (e.g., chemical compounds), PROCESS (re-occurring biological and physical processes), LOCATION (geographic location) and DATA TYPE (research results, e.g., lidar data). All annotations for which the inter-rater agreement was larger than 0.6 were exported to a final XML file. It represents a substantial agreement [133].

The identified relevant entity types from this question corpus were aligned with the detected categories of BiodivOnto in several discussion rounds. Table 8.2 shows the final categories we agreed on to use in the final conceptual model. The entity types were used to inspect the annotated questions again. This inspection consists of manually detecting the relations between the already annotated entities in each question. We omitted questions that do not pose any annotation of the final classes or provide only one class. We only considered questions that contain at least two annotations of the entity types. In total, 91 questions were utilized for the relation detection in the question corpus.

The main idea for the relation detection process was to come up with categorization for relations similar to the categories for noun entities. The detection process was conducted in several rounds. In the first pilot phase, three scholars analyzed only a few questions about the existence of relations. The initial instruction was to manually inspect the questions and to identify binary relations between the occurring entities. Scholars were also advised to inspect the given verbs (which mainly describe a relation) and to think about suitable categories for the relations. In a second round, the proposed relation categories were discussed. The outcome was used for the final detection round. The final agreed relation categories are:

- *influence* (an entity influences another entity, e.g., an ORGANISM influence PHENOMENA),
- *occur_in* (an entity occurs in another entity, e.g., PROCESS occur_in ENVIRONMENT),
- *of* (inverse relation of have: an entity of an entity or an entity has another entity, e.g., QUALITY of ORGANISM).

Complex questions with several entities were split into several relations. For example, the question ‘How do (autotrophic microorganisms)[ORGANISM] influence (car-

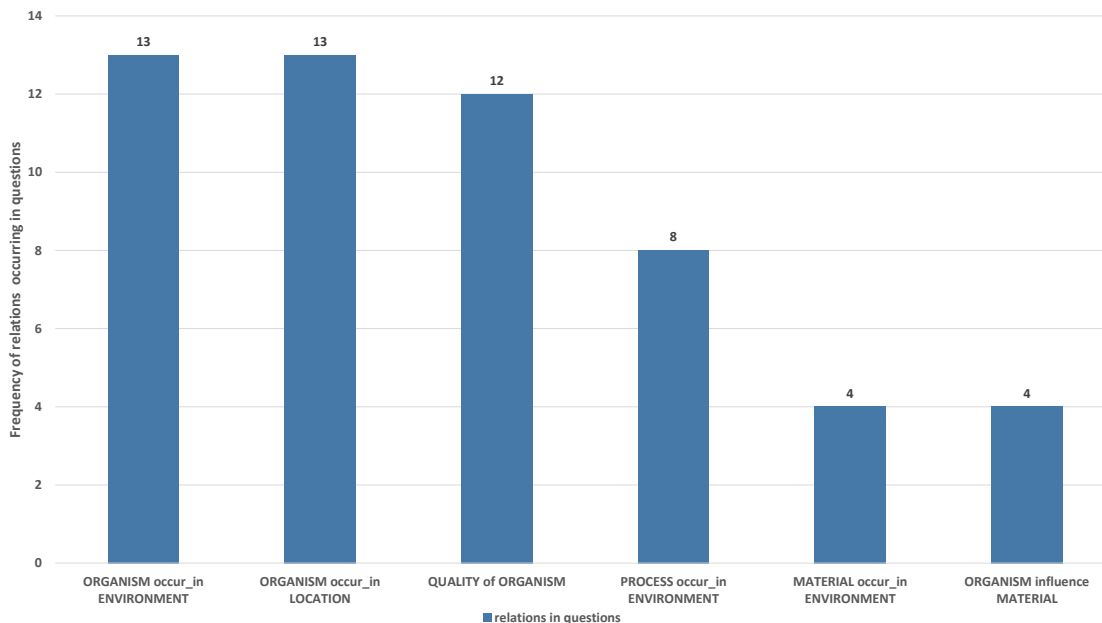


Figure 8.9: Occurrence frequency of relations in questions related to biodiversity research (from [17]).

bon cycling) (PHENOMENA) in (groundwater aquifers)[ENVIRONMENT]?’ This resulted in detecting two relations: *influence* (autotrophic microorganisms ORGANISM, carbon cycling PHENOMENA) and *occur_in* (carbon cycling PHENOMENA, groundwater aquifers ENVIRONMENT). Figure 8.9 presents the outcome of the relation detection of the question corpus. The most frequent relation patterns are ORGANISM *occur_in* ENVIRONMENT, and ORGANISM *occur_in* LOCATION, with 13 mentions each. This result served as input for the conceptual model, BiodivOnto.

8.4.2 The Final BiodivOnto

We integrated entity types and the detected relations from the biodiversity questions corpus into our developed conceptual data model. The proposed class names were discussed with two biodiversity experts. We finally agreed on: ORGANISM, PHENOMENA, ENVIRONMENT, LOCATION, QUALITY, and MATTER as the final naming. We use the final classes and relations to develop the textual data understanding framework (see Chapter 9). In addition, we also use them to construct and annotate two benchmarks to evaluate such a framework (see Chapter 10). Initially, BiodivOnto contains fine-grained as well, like ‘Ecosystem’ and ‘Landscape’, which are subclasses of the ENVIRONMENT class. To facilitate the annotation process, we decided to use the top-level classes only. In that sense, both ‘Ecosystem’, and ‘Landscape’ are substituted by ENVIRONMENT. The same applies to ‘Trait’ and QUALITY, where only the latter was used in annotation. LOCATION is added to BiodivOnto as a result of the analysis of the Biodiversity Questions corpus.

Regarding relations, BiodivOnto initially had the following:

- *have*: that appeared between ORGANISM-ENVIRONMENT, ORGANISM-QUALITY, ENVIRONMENT-QUALITY, and MATTER-QUALITY.
- *occur_in*: that appeared between PHENOMENA-ENVIRONMENT.

Similarly to what we did regarding concepts, we merged the outcome from the analysis of the Biodiversity Questions corpus; we included new relations as follows:

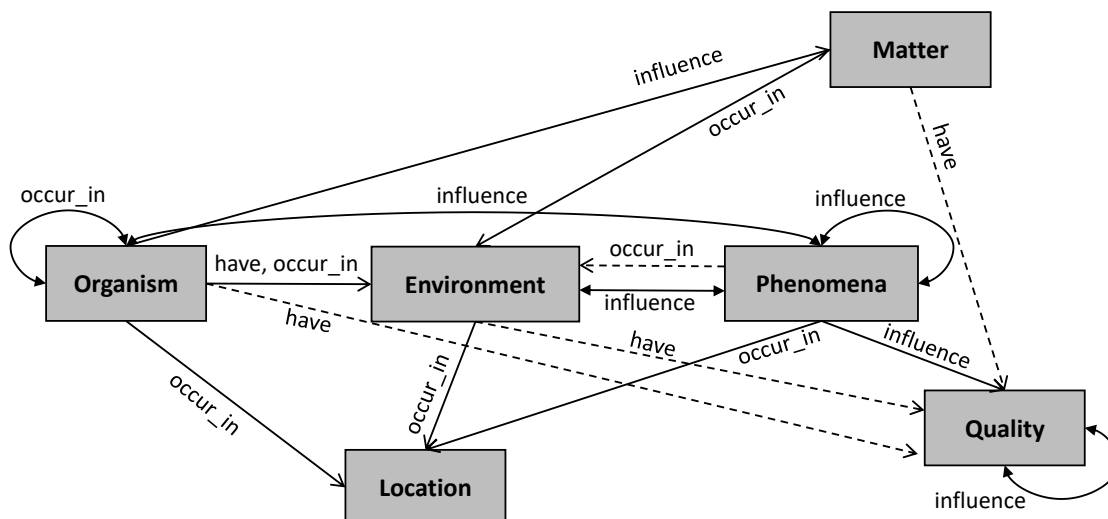


Figure 8.10: Final version of BiodivOnto (from [17]).

- *occur_in* links, in addition to the above, MATTER-ENVIRONMENT, ORGANISM-LOCATION, ORGANISM-ORGANISM, PHENOMENA-LOCATION, and ENVIRONMENT-LOCATION.
- *influence* relates ORGANISM-PHENOMENA, ORGANISM-MATTER, PHENOMENA-PHENOMENA, PHENOMENA-QUALITY, PHENOMENA-ENVIRONMENT, and QUALITY-QUALITY.

The early version of BiodivOnto included both *part_of* and *is_a* relations. We dropped them in the new ontology version since the most common relations in the Biodiversity Questions corpus lack them. We kept these relations that exist in both sources only.

Figure 8.10 illustrates the reconciled version of BiodivOnto after merging the results from the Biodiversity Questions corpus. It consists of 6 classes and 17 relations; this model is ready to be used for annotation.

8.5 Summary

We used a novel data-driven and semi-automatic approach involving domain experts and computer scientists to develop a core ontology; *BiodivOnto*. Our proposed method used the fusion/merge strategy by reusing existing ontologies and data from several data repositories in the biodiversity domain to guide it. It consists of four steps: data acquisition, term extraction, term filtration, and finally, concepts and relation determination. This approach is different from the traditional ones of manually developing ontologies. We reduce the manual effort of developing a core ontology using this semi-automatic data-driven approach. We also extracted the crucial concepts from the existing biodiversity domain ontologies to develop our core one. However, there are open questions regarding the quality of the developed core ontology. In the current state, we determined only the core concepts of BiodivOnto. The domain expert suggested the relations that interlink the core concepts of BiodivOnto. We need to determine how relations between these core concepts could be connected using an automatic approach. Those can be determined using the same approach as the core categories. For example, we could reuse the existing properties from the current ontologies to determine the relationship between the core concepts.

The involvement of domain experts is required for qualitative ontology development. In our methodology, a biodiversity domain expert has been involved in each stage of our pipeline. We included the other domain experts only after the core concepts creation and

for final evaluation and validation. We made ‘Quality’, and ‘Trait’ synonyms based on their opinion.

Each cluster has correctly captured the terms related to the core concept. However, there were non-relevant terms to the core concept were also included. As a result, a detailed and quantitative evaluation is required in addition to the domain expert evaluation. We also need to compare between data-driven engineering approach for ontology development and manual ontology development with the aid of domain experts. In our next phase, we need to combine the collected modules as a complete ontology. Currently, it is a conceptual data model with modules from existing ontologies put together.

Besides the method we presented in this chapter, we also demonstrated a continuous development of the conceptual model using an existing resource, the Biodiversity Questions corpus. After integrating such analysis results and the initially developed ontology, we reached the final model that is ready to be used in developing a textual data interpreter framework and constructing evaluation benchmarks. We describe the details of each of them in Chapter 9 and Chapter 10, respectively. We made the outcome and code from this chapter publicly available under our GitHub repository²⁶.

²⁶<https://github.com/fusion-jena/BiodivOnto>

Chapter 9

BiodivBERT

Information Extraction in Life Sciences is getting an increasing attention due to the constantly growing amount of data and text. Motivated by the predicted impending loss of biodiversity and the consequences of this loss for humanity [134], research in the biodiversity domain has recently witnessed an accelerated growth. For instance, the Biodiversity Heritage Library (BHL)¹ currently holds over 60 million² digitised pages of legacy biology text from the 15th – 21st centuries, representing a huge amount of textual content [78]. Moreover, Google Scholar returns more than 85,000 hits for a search using the term ‘biodiversity’ from 2021 til November 2022. Thus, text mining tools are an open demand in this field to leverage this untapped wealth. The recent progress of data mining techniques is applicable by the advancements of the deep learning models used in Natural Language Processing (NLP), however, directly applying such NLP techniques on biodiversity texts is not promising.

Modern word representation models such as Word2Vec [128], GloVe [129], ELMo [130] and BERT [74] are trained and tested on datasets containing general domain texts (e.g., Wikipedia). However, domain-specific, e.g., biodiversity, texts contain a considerable number of instances of domain-specific entity types. E.g., *Helianthus* (instance of species), calcareous grassland, or growth rate. Thus, it is difficult to estimate the performance of general-purpose models on domain-specific datasets. Techniques to improve the performance of cutting-edge approaches like BERT on domain-specific benchmarks have first been developed for the (bio-)medical domain with BioBERT [8], and clinicalBERT [84]. BioBERT is initialized with BERT weights and pre-trained on biomedical corpora that are based on PubMed³ and PubMed Central (PMC)⁴. It showed a significant improvement on three downstream tasks, namely Named Entity Recognition (NER), Relation Extraction (RE), and Question Answering (QA). To the best of our knowledge, there is no language model for the biodiversity domain that supports the extraction of named entities and their relations from textual data. That biodiversity-specific language model represents the textual data interpreter we aim to develop under our second research area **Textual data interpretation (TexI)**. After successfully fine-tuning this model on corpora that are annotated with classes and relations from BiodivOnto, it could extract named entities (subjects and objects) as an output from the NER task. In addition, it could detect relations among those named entities (predicate) as an output from the RE task. This means that BiodivBERT could construct a KG as a set of triples (subject, predicate, object) from unstructured text.

In this chapter, we introduce BiodivBERT, the first pre-trained language model for

¹<https://www.biodiversitylibrary.org/>

²Accessed on 14th January 2023

³<https://pubmed.ncbi.nlm.nih.gov/>

⁴<https://www.ncbi.nlm.nih.gov/pmc/>

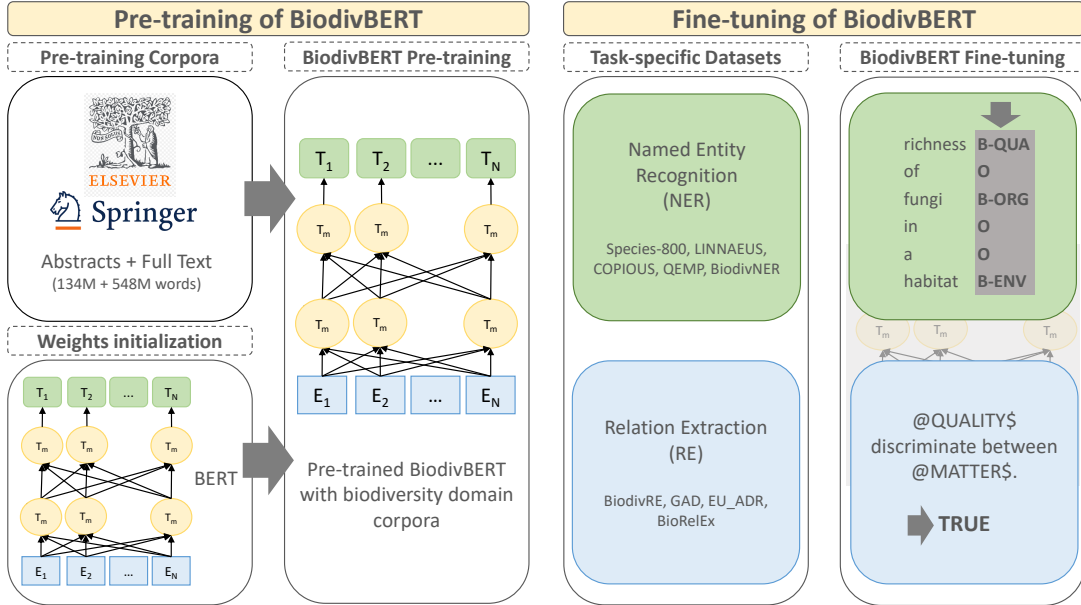


Figure 9.1: BiodivBERT pre-training and fine-tuning Overview (from [135]).

the biodiversity domain in Section 9.1. We discuss our two pre-training corpora (abstracts and abstracts + full text) that are based on a keyword search strategy from two main publishers in Life Sciences, namely, Springer and Elsevier in Section 9.2. In addition to the pre-training, we fine-tune BiodivBERT on two downstream tasks i.e., Named Entity Recognition (NER) and Relation Extraction (RE) using various state-of-the-art benchmarks in Section 9.3. We demonstrate the effectiveness of BiodivBERT in Section 9.4. We summarize and conclude this chapter in Section 9.5⁵.

9.1 Approach

We introduce BiodivBERT, which is a pre-trained language representation model for the biodiversity domain. The overall process of pre-training and fine-tuning BiodivBERT is shown by Figure 9.1. First, we initialize BiodivBERT with weights from BERT [74], which was pre-trained on general domain corpora (English Wikipedia and BooksCorpus). Then, BiodivBERT is pre-trained on our collected corpora from the biodiversity domain. The first corpus is based on abstracts (+Abs), while the other contains both abstracts and full text (+Abs+Full). To demonstrate the effectiveness of BiodivBERT in biodiversity text mining, we fine-tuned and evaluated it on two downstream tasks NER and RE using various task-specific datasets.

9.2 Pre-training

In this section, we explain our pre-training data sources, selection strategy, and data statistics. In addition, we discuss the pre-training task for BiodivBERT.

⁵This chapter is based on Abdelmageed et al [135]. Thanks to Felicitas Löffler who constructed the abstracts corpus for pre-training BiodivBERT. The author of this thesis applied further analysis beyond the scope of the published manuscript.

9.2.1 Pre-training Data

We discuss the construction of two pre-training corpora that are based on Abstracts (+Abs), and Full text (+Abs+Full). We explain our used keywords search strategy, workflow, and the resultant corpora statistics.

Keywords Search We discussed various options to crawl data for pre-training with three biodiversity experts. Our experts recommended to focus on sources that reflect recent research directions. Therefore, they suggested querying Elsevier⁶ and Springer⁷ rather than the BHL which contains more legacy data. In addition, these companies publish a diverse set of biodiversity-related journals. Moreover, they provide official APIs to crawl data. To crawl these massive reservoirs, our experts suggested 10 keywords that covered the domain as follows: ‘biodivers*’, ‘genetic diversity’, ‘*omic diversity’, ‘phylogenetic diversity’, ‘soil diversity’, ‘population diversity’, ‘species diversity’, ‘ecosystem diversity’, ‘functional diversity’, and ‘microbial diversity’ and recommended crawling data from the last three decades [1990-2020⁸]. ‘biodivers*’ and ‘*omic diversity’ are wild card representation that include multiple keywords like biodiversity, biodiverse, taxonomic diversity, and etc.

Corpora Construction Pipeline To show the effect of the pre-training data on the model performance through the downstream tasks, we created two pre-training corpora. One is based on abstracts (+Abs) only, while the other contains abstracts and full text of publications (+Abs+Full). Under the access rights provided by the selected publishers, we used abstracts and full texts of open-access papers and abstracts only for other publications. To construct the +Abs corpus, we used the pre-selected keywords and year range as input for both Elsevier and Springer’s provided full-text search APIs. For each of them, we retrieved the corresponding DOIs for each keyword in the given year’s range. We then applied a deduplication method to results. We made the collected DOIs of the pre-training text corpora publicly available [136]. We applied the same procedure for the second corpus, which is based on the full texts (+Abs+Full). Elsevier provided a straightforward API to obtain the parsed full text for a given article. However, for Springer’s full text, we downloaded the corresponding PDF file for each DOI, converted it to an XML format, and then extracted the text. We converted the downloaded PDFs to XML files using the GROBID service, and client [137]. We cleaned and merged the final text from both data sources. We applied shallow and deep cleaning steps for the collected data. For instance, we filtered the sentences to include unique ones. In addition, we used regular expressions to remove URLs and DOIs.

Corpora Statistics Figure 9.2 shows the available DOIs distribution of the selected keywords over the pre-defined year’s range, this reflects the contained publications in the Abstracts (+Abs) corpus. These results include both open and closed access publications; abstracts are always available for all kinds of manuscripts. Those numbers are decreased as shown in Figure 9.3 during the construction of the full text corpus due to the open access limitation by both publishers. From both figures we note that the top three keywords that yielded into the most publications from Springer are: ‘population diversity’, ‘species diversity’, and, ‘functional diversity’. ‘*omic diversity’ has the lowest number of publications in Springer. The top three keywords from Elsevier are: ‘biodivers*’, ‘genetic diversity’, and ‘species diversity’. ‘soil diversity’ and ‘ecosystem diversity’ keywords have less than 2000 publications from Elsevier only.

⁶<https://dev.elsevier.com/>

⁷<https://dev.springernature.com/>

⁸The starting year of this project.

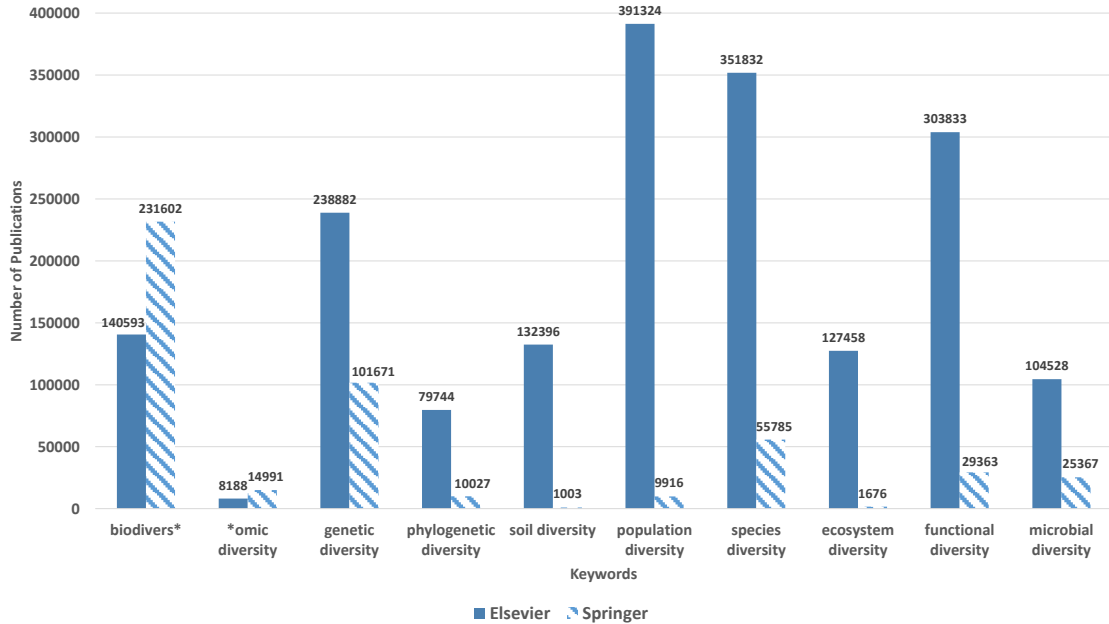


Figure 9.2: keywords results statistics used for abstracts corpus during years 1990-2000

Table 9.1: Final Pre-training Corpora Statistics (from [135]).

Corpus	Data Source	#Docs	#Sentences	#Words	Size
+Abs	Abstracts	1M Abstract	5M	134M	876 MB
+Abs+Full	Full Text	92K Article	25M	548M	3.81 GB

Table 9.1 gives an overview of our final corpora statistics. +Abs and +Abs+Full are around 1 GB, and 4 GB in size respectively. Our corpora include around 1M abstracts with 92K full publications. They contain 5M and 25M sentences, respectively. +Abs+Full is around 5 times of the +Abs in terms of the included sentences. All these numbers are derived after the cleaning step we applied in the construction pipeline. Thus, final included sentences in each corpus are unique.

We faced data loss as shown in Figure 9.4 due to several reasons: 1) not found (404) errors for some articles, because of either technical issues on the provider side or invalid DOIs. 2) Elsevier allows crawling only 6000 articles per keyword through its API. 3) GROBID failed to convert some PDFs from Springer. So, the shown numbers indicate the best we can use from both publishers under such circumstances. These issues reduced the available publication in case of Springer by 28%. They limited the final obtained full text in case of Elsevier by about half of the available; 51%.

9.2.2 Pre-training Task

We pre-trained BiodivBERT on our domain-specific corpora (+Abs), and (+Abs+Full). We initialized BiodivBERT with the `BERT_base_cased` weights for computation efficiency and to leverage the general domain learned weights from the Wikipedia and books corpora by the original model. For tokenization, we used the same BERT WordPiece [138] tokenizer which overcomes the issue of out-of-vocab (OOV). Similar to BioBERT [8], we used the cased vocabulary in our setting since it has higher performance on the downstream tasks and we used the original vocabulary of `BERT_base_cased` for the same reason and to be compatible with both BERT and BioBERT. We compare BiodivBERT to the state-of-the-art BERT-based models. In addition, we tested different combinations of pre-training corpora. We compare these settings in Table 9.2. We pre-trained BiodivBERT_{+Abs}, and

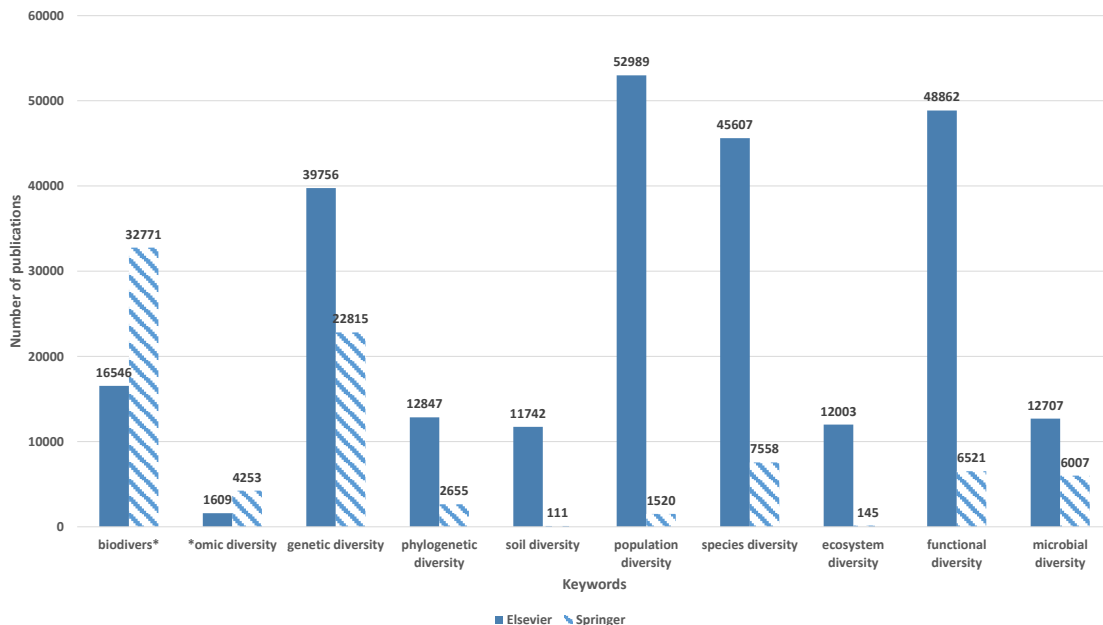


Figure 9.3: Open access keywords results statistics used for Full Text corpus during years 1990-2000

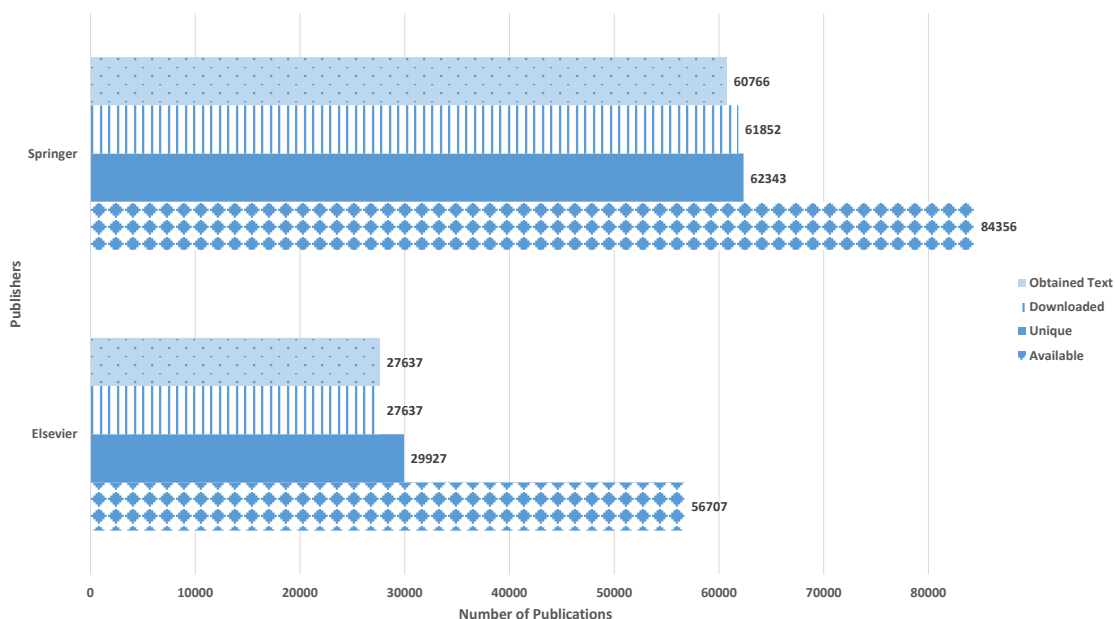


Figure 9.4: Data loss during full Text corpus construction (from [135]).

BiodivBERT_{+Abs+Full} using the HuggingFace [139] library on a single V100 GPU (16 GB) for 3, and 5 days respectively. For hyperparameters, we used 512 for the maximum sequence length and 15% of the masked language model probability. In addition, we used Adam’s optimizer with 1e-3 learning weight and default betas. Moreover, we set batch size to 16, and we enabled the gradient accumulation with 4 steps for faster training. We made the pre-trained weights publicly available [140].

9.3 Fine-Tuning

With minimal architectural modification, BiodivBERT can be applied to various downstream text mining tasks. We fine-tuned it on both NER and RE using P100 (16 GB) by

Table 9.2: Pre-training Models setting corpora (from [135]).

Model	Corpora
BERT _{BASE} [74]	Wiki + Books
BioBERT _{v1.1} [8]	Wiki + Books + PubMed
BiodivBERT _{+Abs}	Wiki + Books + Abstracts
BiodivBERT _{+Abs+Full}	Wiki + Books + Abstracts + FullTxt

Table 9.3: Overview of the selected NER datasets (from [135]).

Dataset	Tags	#Docs	#Statements	#Annotations
QEMP [29]	4	50	2,226	5,154
COPIOUS [78]	5	668	26,277	26,007
Species800 [79]	1	800	14,756	5,330
Linnaeus [80]	1	100	34,310	3,884
BiodivNER [17]	6	150	2,398	9,982

Colab Pro⁹ using various state-of-the-art datasets. We made our pre-processed datasets publicly available [141].

9.3.1 Named Entity Recognition

Named Entity Recognition (NER) is the task of identifying the domain-specific proper nouns inside a given text. We leveraged the original BERT structure for NER such that it uses a single output layer based on the representations from its last layer to compute only token-level probabilities. We used entity-level precision, recall, and F1-score as the evaluation metrics of NER. We selected various state-of-the-art datasets to test the performance of BiodivBERT on NER, Table 9.3 summarizes them. BiodivNER [142] is constructed from biodiversity-specific metadata files and abstracts from PubMed and has five tags, e.g., organism and phenomena. We constructed this corpus and gave its details in Chapter 10. COPIOUS [78] is based on BHL documents and has six entity types, a.k.a. tags, including, e.g., Habitat and taxon names. QEMP [29] is created from biodiversity-related datasets metadata files and contains four tags, e.g., quality and material. Species800 [79] and Linnaeus [80] are designed for species names that are normalized to NCBI Taxonomy database¹⁰. We pre-processed all of them to follow the BIO¹¹ format for token classification.

9.3.2 Relation Extraction

Relation Extraction (RE) is the task of classifying relations among named entities in a corpus. We utilized the sentence classifier of the original version of BERT which uses a [CLS] token for the classification of relations. To the best of our knowledge, BiodivRE [142] is the only available RE corpus for the biodiversity domain, so we included it in our fine-tuning setting. We constructed this corpus as well, we give the details of it in Chapter 10. In addition, we included the BioRelEx [83], EU-ADR [81], and GAD [82] corpora from the biomedical domain. BiodivRE contains relations among entity types of BiodivNER like occur_in, and influence in a multi-class and a binary format. BioRelEx classifies the bindings between genes and diseases into three categories: exists (1), not exists (-1), and unsure (0). EU-ADR, and GAD include relations between gene and disease.

⁹https://colab.research.google.com/?utm_source=scs-index

¹⁰<https://www.ncbi.nlm.nih.gov/taxonomy>

¹¹https://natural-language-understanding.fandom.com/wiki/Named_entity_recognition#BIO

Table 9.4: Overview of the selected RE datasets (from [135]).

Dataset	#True Statements	#False Statements	Total
BioRelEx [83]	1,379	62	1,606
GAD [82]	25,209	22,761	53,300
EU-ADR [81]	2,358	837	3,550
BiodivRE [17]	1,369	2,631	4,000

Table 9.5: Fill-in mask task results by BERT-based models (from [135]).

Model	Rank	Result
BERT	1	Diversification and variation in brood pollination mutualisms.
	2	Diversification and variations in brood pollination mutualisms.
	3	Diversification and changes in brood pollination mutualisms.
BioBERT	1	Diversification and Image in brood pollination mutualisms.
	2	Diversification and dim in brood pollination mutualisms.
	3	Diversification and vasive in brood pollination mutualisms.
BiodivBERT	1	Diversification and change in brood pollination mutualisms.
	2	Diversification and diversity in brood pollination mutualisms.
	3	Diversification and evolution in brood pollination mutualisms.

In this chapter, we used the binary format that is provided by BiodivRE. For BioRelEx, we constructed a binary relation corpus by excluding the unsure relations. Moreover, and similar to BioBERT, we anonymized the target named entities in a sentence using their tags, e.g., @COMPLEXPROTEIN\$, and @GENE\$. For EU-ADR and GAD, we used the provided pre-processed version by BioBERT’s team since the original data are not available. Table 9.4 shows the selected RE datasets’ statistics.

9.4 Evaluation

To gain a first impression of the performance of our approach, we ran a mask-filling task on a typical biodiversity topic on BERT, BioBERT, and BiodivBERT using the following test case: ‘*Diversification and [MASK] in brood pollination mutualisms.*’. Table 9.5 shows that BiodivBERT has produced the most realistic results compared to the other two models. Such that, BiodivBERT generated both ‘diversity’ and ‘evolution’. Thus, it demonstrates the effectiveness of pre-training data.

Table 9.6 and Table 9.7 show the scores of fine-tuning BiodivBERT, BERT, and BioBERT models on two downstream tasks NER, and RE, respectively. In addition, we developed a single layer of the Bidirectional Long Short Term Memory (BiLSTM) with 10% dropout as a baseline approach. We micro-averaged the results per dataset to generate the scores of all systems. We fine-tuned these models on a single P100 GPU provided by Colab Pro. At first, we found that BioBERT_{v1.1} obtained higher scores than BERT_{BASE} on the downstream tasks, Second, BiodivBERT_{+Abs+Full} and BiodivBERT_{+Abs} gained the best results among all the others by achieving either first or second place on the given datasets. For NER, BiodivBERT_{+Abs+Full} exceeded BioBERT_{v1.1} for all datasets except QEMP, where BiodivBERT_{+Abs} outperformed BioBERT_{v1.1} with 1% F1 score. In addition, we noticed that all models gained higher scores in species-related datasets, e.g., LINNAEUS. A reason that could be behind these results is that these datasets are easier than those with fuzzy categories to identify. E.g., QEMP and BiodivNER have a class, ‘QUALITY’, that groups data measures that cover vast and various attributes of the biodiversity domain and would be harder to detect. For RE, we have mixed results; for ex-

Table 9.6: Fine-tuning scores on NER datasets. The highest score is marked in **bold** while the following score is marked underline. Evaluation Metrics (Met.) are Precision (P), Recall (R), and F1 score (F) (from [135]).

Dataset	Met.	BiLSTM	BERT _{BASE}	BioBERT _{v1.1}	BiodivBERT	
					+Abs	(+Abs+Full)
Spieces-800	P	0.49	0.80	0.87	<u>0.81</u>	0.79
	R	0.09	<u>0.81</u>	0.80	0.80	0.84
	F	0.16	<u>0.80</u>	<u>0.80</u>	0.81	0.81
LINNANUS	P	0.82	<u>0.93</u>	<u>0.93</u>	0.92	0.95
	R	0.22	<u>0.94</u>	<u>0.94</u>	0.90	0.95
	F	0.34	<u>0.94</u>	<u>0.94</u>	0.91	0.95
COPIOUS	P	0.77	<u>0.88</u>	<u>0.88</u>	<u>0.88</u>	0.89
	R	0.53	0.87	0.89	<u>0.88</u>	<u>0.88</u>
	F	0.63	0.88	0.88	0.88	0.88
QEMP	P	0.84	<u>0.90</u>	0.91	<u>0.90</u>	0.88
	R	0.53	0.73	<u>0.76</u>	0.78	0.72
	F	0.65	0.81	<u>0.83</u>	0.84	0.79
BiodivNER	P	0.66	<u>0.85</u>	0.86	0.86	<u>0.85</u>
	R	0.44	0.83	0.85	<u>0.86</u>	0.88
	F	0.53	0.84	<u>0.86</u>	<u>0.86</u>	0.87

Table 9.7: Fine-tuning scores RE datasets. The highest score is marked in **bold** while the following score is marked underline. Evaluation Metrics (Met.) are Precision (P), Recall (R), and F1 score (F) (from [135]).

Dataset	Met.	BiLSTM	BERT _{BASE}	BioBERT _{v1.1}	BiodivBERT	
					+Abs	(+Abs+Full)
BiodivRE	P	0.68	0.80	<u>0.79</u>	0.78	0.78
	R	0.68	0.81	0.81	<u>0.79</u>	0.77
	F	0.68	0.80	0.80	<u>0.79</u>	0.77
BioReLx	P	0.71	<u>0.83</u>	0.82	0.85	0.80
	R	0.78	0.89	0.70	<u>0.75</u>	0.74
	F	0.74	0.86	0.75	<u>0.79</u>	0.77
EU-ADR	P	0.71	<u>0.91</u>	0.56	0.92	0.92
	R	0.69	<u>0.62</u>	0.53	0.69	0.69
	F	0.60	<u>0.74</u>	0.54	0.79	0.79
GAD	P	0.66	0.77	0.81	0.77	<u>0.78</u>
	R	0.66	<u>0.77</u>	0.81	0.76	<u>0.77</u>
	F	0.66	0.77	0.81	0.77	<u>0.78</u>

ample, BiodivBERT_{+Abs+Full} outperforms BioBERT_{v1.1} with 2.5% F1-score for EU-ADR. However, BioBERT_{v1.1} overcomes BiodivBERT_{+Abs+Full} with 3% F1-score for BiodivRE. We plan to apply different fine-tuning settings on those datasets to enhance these scores.

To give a general overview of the performance of all models, we propose a simple arithmetic weighting score to demonstrate the effectiveness of each model. Equation 9.1 shows the task score where *task* is either NER or RE and is given by a weighted summation for the first three ranks. An overall score for a system is given by the summation of the tasks' weighted rank as shown in Equation 9.2. Figure 9.5 depicts that BiodivBERT_{+Abs+Full} is the best model in terms of the system score followed by BiodivBERT_{+Abs} with 17.17, and 16 system score, respectively.

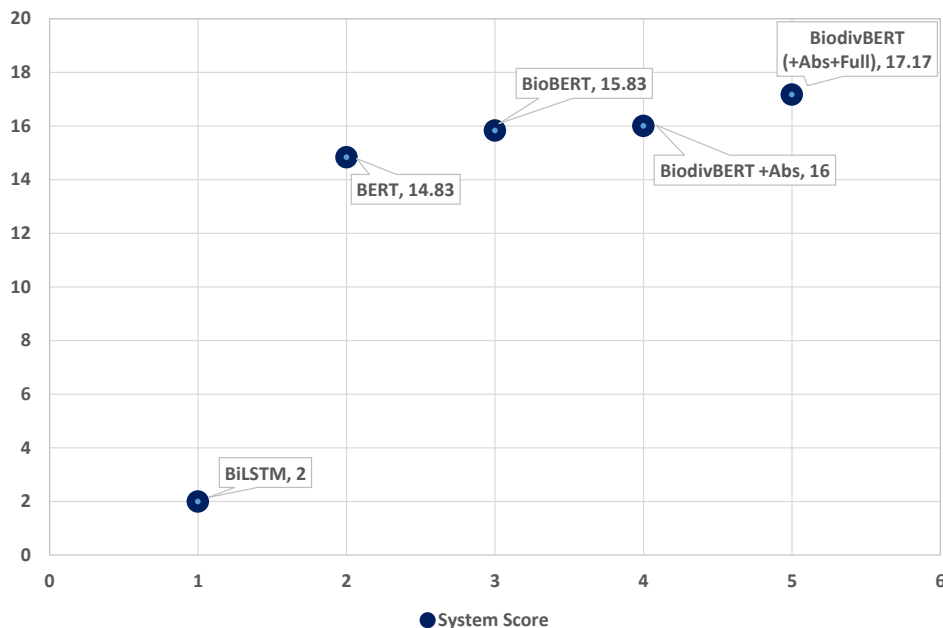


Figure 9.5: System scores of the selected models.

$$(9.1) \quad Score(task) = \#1_{rank}^{st}(task) + \frac{\#2_{rank}^{nd}(task)}{2} + \frac{\#3_{rank}^{rd}(task)}{3}$$

where $task \in \{NER, RE\}$

$$(9.2) \quad System_{Score} = Score(NER) + Score(RE)$$

Our constructed corpora BiodivNER and BiodivRE (see Chapter 10) are annotated using classes and relations from BiodivOnto we constructed in Chapter 8. This schema consists of six core concepts including, e.g., ‘Organism’, ‘Environment’, ‘Location’, and three core relations: *have*, *occure_in*, and *influence*. So far, the population of the ontology with instances is incomplete. We fine-tuned BiodivBERT on these two corpora. Thus, BiodivBERT could be used to auto-populate the ontology. For instance, ‘Seabass occurs in the western Atlantic Ocean’ would be classified as: [Seabass ORGANISM, occurs, Atlantic Ocean LOCATION] an . In a final step, the identified instances should be linked to existing knowledge graphs. e.g., ‘Seabass’ and ‘Atlantic Ocean’ would be mapped to <http://www.wikidata.org/entity/Q307102> and <https://www.wikidata.org/wiki/Q97>, respectively from Wikidata. This could be done, e.g. using our approach described in Chapter 6. With this, BiodivBERT has a potential to bring us closer to our ultimate goal, the creation of a comprehensive biodiversity knowledge graph out of textual data.

9.5 Summary

In this chapter, we introduced BiodivBERT as a pre-trained language model for the biodiversity domain. We pre-trained it using two domain-specific corpora that contain recent publications from both Springer and Elsevier publishers. We used ten keywords to crawl these data sources during [1990-2020]. We demonstrated the statistics of the constructed corpora. In addition, we fine-tuned BiodivBERT on two downstream tasks for text mining: Named Entity Recognition (NER) and Relation Extraction (RE). We included the closely

related state-of-the-art benchmarks for both tasks besides our developed biodiversity-specific datasets. We compared BiodivBERT to state-of-the-art approaches, including BERT, BioBERT, and a baseline approach to demonstrate its effectiveness. BiodivBERT outperforms the state-of-the-art approaches on task-specific datasets. Last but not least, we pointed out an application of BiodivBERT, it could auto-populate our constructed BiodivOnto ontology with instances. This means BiodivBERT represents the textual data interpreter to construct a knowledge graph from text. We made the outcome from this chapter publicly available under our GitHub repository¹². In addition, we released the collected DOIs for per-training corpora construction [136], the pre-trained weights [140], and the pre-processed datasets for fine-tuning [141] available at Zenodo.

¹²<https://github.com/fusion-jena/BiodivBERT>

Chapter 10

BiodivNERE Corpora

Natural Language Processing (NLP), with its sub-task Information Extraction, is a research field that uses structured data or scientific publications. The aim is to develop systems that automatically identify important terms and phrases in text. That supports scholars in getting a quick overview of unknown texts, e.g., in search or allows improved filtering. In Life Sciences, Information Extraction has a long history [143]. Driven by a series of workshops and shared tasks such as BioNLP¹, BioCreative², and BioASQ³ in the scope of CLEF⁴, multiple corpora and tools for various purposes were developed to extract main entities from text and relations among them automatically. However, determining what a relevant entity or relation in a document or data depends on the domain of focus. While scholars looking for biomedical data are mainly interested in data types such as diseases, biological processes and organisms [144], and related entities such as genes and proteins. In biodiversity research, other categories are of relevance, namely: organisms, environmental terms, geographic locations, measured data parameters, materials, biological, physical and chemical processes, and data types [28].

The increasing amount of scientific datasets in public data repositories calls for more intelligent systems that automatically analyze, process, integrate, connect or visualize data. An essential building block in the evolution of such computer-supported analysis tools is Information Extraction with its sub-tasks, Named Entity Recognition (NER) and Relation Extraction (RE). The former task aims to automatically identify important terms (entities) and groups of terms/expressions that fall in a certain category. The latter task extracts relationships that could occur among those entities (RE). However, the advancement of such tools is applicable if gold standards, manually labeled test corpora, are available. This supports the training of machines (for machine learning approaches) and allows an evaluation of the developed tools. For applied domains such as biodiversity research, gold standards are very rare.

In our scope, under the second research area **Textual data interpretation (TexI)**, we developed a textual data interpreter, BiodivBERT (see Chapter 9). We need to demonstrate the effectiveness of it using gold standard and domain-specific corpora. Since BiodivBERT is developed to construct a Knowledge Graph (KG) from text, the downstream tasks that achieve this goal are NER and RE. We present novel gold standards for biodiversity research in two downstream tasks (NER and RE). To hit our ultimate goal, KG construction from text, we align these corpora with the BiodivOnto that we discussed in Chapter 8. We provide a NER corpus based on scientific metadata files and abstracts with manual annotations of important terms from BiodivOnto such as species

¹<https://aclanthology.org/venues/bionlp/>

²<https://biocreative.bioinformatics.udel.edu/>

³<http://bioasq.org/>

⁴<http://clef2021.clef-initiative.eu/>

(ORGANISM), environmental terms (ENVIRONMENT), data parameters and measured variables (QUALITY), geographic locations (LOCATION), biological, chemical and physical processes (PHENOMENA) and materials (MATTER), e.g., chemical compounds. In addition, we provide an RE corpus based on a portion of the same data that consists of important binary and multi-class relations among entities such as *occur_in* (Organism, Environment), *influence* (Organism, Process), and *have/of* (Quality, Environment). We provide the results in formats that allow easy further processing for various NLP tasks based on machine learning and deep learning techniques.

In this chapter, we describe our used data, entity types and relations in Section 10.1. We explain our pipeline to create both corpora: BiodivNER, and BiodivRE in Section 10.2, and Section 10.3, respectively. We evaluate both corpora and explain their statistics in Section 10.4. We summarize and conclude this chapter in Section 10.5⁵.

10.1 Resources Reuse

To construct our corpora, we re-used the metadata and abstracts that we initially collected to develop BiodivOnto (see Chapter 8). Metadata files are gathered from two data sources with very different characteristics (BEFChina⁶ and data.world⁷). The Semedico search engine [126] retrieves relevant abstracts from PubMed⁸, a source with more than 32M abstracts. To ensure the relevance of the crawled data from Semedico, we have followed an iterative way of revision. We started with the initial keywords set that we used to crawl. Initially, these collected data were meant to extract biodiversity-related keywords. However, in this chapter, we use them for the purpose of developing NER and RE corpora.

Entity types and relations are settled in the last version of BiodivOnto. We use entity types, tags, or classes interchangeably in this chapter. We developed its final version after we integrated the outcome from the analysis of the Biodiversity Questions corpus. We summarize the classes we use to annotate the NER corpus as follows:

- **ORGANISM**: includes all individual life forms, e.g., mammal, insect, fungi, and bacteria.
- **PHENOMENA**: contains occurring natural, biological, physical, or chemical processes, including events, e.g., decomposition, colonization, and deforestation.
- **MATTER**: includes chemical and biological compounds and natural elements, e.g., carbon, H₂O, sediment, and sand.
- **ENVIRONMENT**: includes natural, or man-made environments ORGANISM live in, e.g., groundwater, garden, and aquarium.
- **QUALITY**: contains data parameters that are measured or observed, phenotypes and traits, e.g., volume, age, and structure.
- **LOCATION**: consists of geographic location (no coordinates), e.g., China, United States.

⁵This chapter is based on Abdelmageed et al [17]. Thanks to Leila Feddoul who participated in the manual annotation of both corpora (BiodivNER and BiodivRE). Thanks to Felicitas Löffler and Sheeba Samuel who participated in the manual annotation of BiodivNER. Thanks to Jitendra Gaikwad and Anahita Kazem, our biodiversity experts for validations and discussions.

⁶<https://data.botanik.uni-halle.de/bef-china/>

⁷<https://data.world/>

⁸<https://pubmed.ncbi.nlm.nih.gov/>

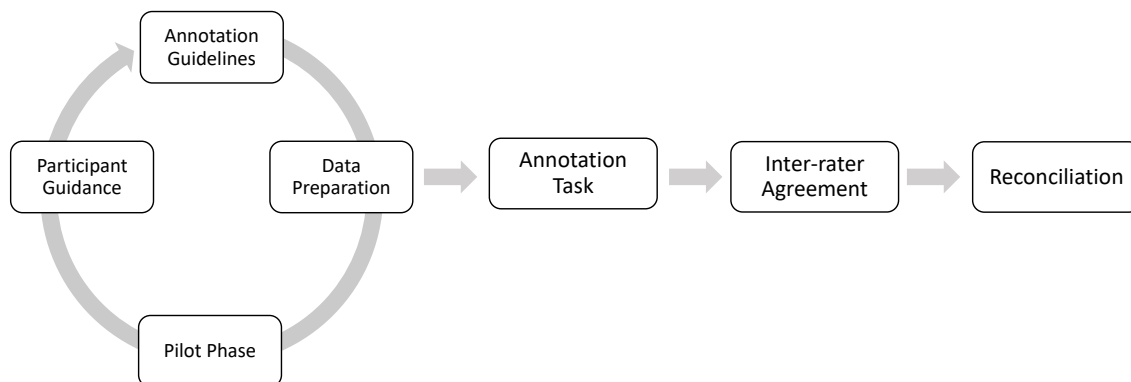


Figure 10.1: Our proposed NER corpus construction pipeline (from [17]).

We summarize the relations we use to annotate the RE corpus as follows:

- *occur_in* links both MATTER-ENVIRONMENT, ORGANISM-LOCATION, ORGANISM-ORGANISM, PHENOMENA-LOCATION, PHENOMENA-ENVIRONMENT, and ENVIRONMENT-LOCATION.
- *influence* relates ORGANISM-PHENOMENA, ORGANISM-MATTER, PHENOMENA-PHENOMENA, PHENOMENA-QUALITY, PHENOMENA-ENVIRONMENT, and QUALITY-QUALITY.
- *have*: appears between ORGANISM-ENVIRONMENT, ORGANISM-QUALITY, ENVIRONMENT-QUALITY, and MATTER-QUALITY.

10.2 BiodivNER Construction Pipeline

In this section, we explain the construction pipeline of the NER corpus as shown in Figure 10.1. Our process consists of seven steps. It starts with the annotation guidelines to describe what we annotate and is followed by the data preparation step in which the originally collected data is transformed into the required data format used for annotation. In the pilot phase, we carry out an initial annotation task to check whether we have to modify the annotation guidelines or whether we have to invest more time in the annotators' training. Afterwards, the actual annotation task takes place. The outcome is evaluated with the computation of the inter-rater agreement, precision, recall, and F1-score. Finally, we discuss the mismatches with biodiversity experts in the reconciliation phase.

10.2.1 Annotation Guidelines

We followed a modified version of our previous project guidelines to construct the QEMP corpus [29]. We set the current sentence as the only available context to annotate. We did not consider the entire document as in the gold standard construction process in NLP. Since the main purpose of this work is to develop a corpus for NER, we considered only noun entities and discarded adjective entities. In addition, we gave higher attention to the complex words that minimized the chance of having two valid annotations for one term. Thus, we followed the longest span annotation and avoided nested entities annotation. For example, 'benthic oxygen uptake rate' is annotated as [QUALITY] while we ignored

Sentence #	Word	Tag	Sentence #	Word	Tag
Sentence: 1	a	O	Sentence: 1	a	O
	silk	O		silk	O
	cover	O		cover	O
	prevents	O		prevents	O
	loss	O		loss	B-Phenomena
	of	O		of	O
	sand	O		sand	B-Matter
	and	O		and	O
	guarantees	O		guarantees	O
	free	O		free	O
	drainage	O		drainage	B-Phenomena
	of	O		of	O
	water	O		water	B-Matter

(a) initially prepared data

(b) while annotating data

Figure 10.2: NER annotation process (from [17]).

any simple word annotation inside this span. Conjunctions are handled as two separate entities. For example, ‘(phylogenetic diversity)[QUALITY] of (bacteria)[ORGANISM]’. We included more existing external resources than the ones used in QEMP to find proper annotations. For example, we considered the following ontologies that were used for constructing the original version of BiodivOnto: ECSO⁹ and ECOCORE¹⁰ for environmental-related terms, BCO¹¹ and CBO¹² for phenomena-related keywords. In addition, we utilize NCBITaxon¹³, FLOPO¹⁴ for species and phenotype annotation, respectively. Moreover, we used the SWEET¹⁵ ontology to capture any missing terms from the previous ontologies. Our last option to find annotations from existing sources is a reference to the ontological issues detected and summarized by [29]. Such kind of selected resources mixture facilitated the detection of a wide range of terms that vary in their granularity (too specific vs. too general terms).

10.2.2 Data Preparation

We parsed the original data collection into sentences. For each sentence, we tokenized it into a set of words using nltk¹⁶ library. Since our used annotation format is BIO-scheme¹⁷, where a word is annotated either with B-tag as a beginning of an entity or, I-tag as an inside of entity or, O as outside of the entity, each word is initialized with an O tag. Each sentence as a set of words with O tags is stored vertically in a CSV file, as shown in Figure 10.2a. Afterwards, we split the entire corpus into two halves to enable the double annotation process.

⁹<https://bioportal.bioontology.org/ontologies/ECSO>

¹⁰<https://bioportal.bioontology.org/ontologies/ECOCORE>

¹¹<https://bioportal.bioontology.org/ontologies/BCO>

¹²<https://bioportal.bioontology.org/ontologies/CBO>

¹³<https://bioportal.bioontology.org/ontologies/NCBITAXON>

¹⁴<https://bioportal.bioontology.org/ontologies/FLOPO>

¹⁵<https://bioportal.bioontology.org/ontologies/SWEET>

¹⁶<https://www.nltk.org/>

¹⁷https://natural-language-understanding.fandom.com/wiki/Named_entity_recognition#BIO

10.2.3 Trial and Pilot Phase

The author of this dissertation along with two PhD students and a Post Doc. were responsible for annotating the corpus. The four annotators received periodical guidance from two biodiversity experts. Initially, we established a trial or a pilot phase before the actual annotation process took place. The purpose of this phase is to ensure the training of the annotators (participant guidance) as well as, to revise the annotation guidelines. Around 2% (450 sentences) of the entire corpus is assigned to each annotator pair. Each annotator labeled a local copy of the pilot phase data in an Excel file. During this process, each annotator is asked to annotate a relevant term with one and only one tag from the provided tags. The results of this process are represented in Figure 10.2. After the end of the Pilot Phase, we held a ‘Share Thoughts’ meeting to discuss the outcome. At this stage, we realized that we need a modified version of the guidelines. For example, at the beginning, not all annotators followed the ‘longest span’ rule and annotated every single word separately. Thus, we have settled on the longest span sequence to avoid or minimize such inconsistencies. In addition, we have decided to add the SWEET ontology to include missing terms from the other used ontologies.

10.2.4 Annotation Process

After the pilot phase, we familiarized ourselves with the annotation process and the guidelines. Each half of the corpus was assigned to an annotator pair. We followed the same procedure as in the pilot phase. Each annotator from the annotators pair worked blindly on a local copy of the sheet. We refer to blindly as without access to the annotation of the other mate. This procedure ensures the higher quality of annotated data and allows the calculation of the inter-rater agreement. Each annotator was asked to complete the annotation of half of the corpus. This annotation process was time-consuming and lasted for several months. Annotating a term is considered to be done if the annotator found the target tag in the selected existing data sources. However, if the annotator was unsure about the correct annotation, the term with a suggested tag was kept in a separate sheet named ‘Open Issues’. We held various meetings with the biodiversity experts during this stage to solve the open issues. Since we had two annotator pairs, let’s say, team A and B for two different sheets, where each sheet represented half of the corpus, we were able to calculate the inter-rater agreement for each team. We used Kappa’s score for the agreement computation since it is one of the most common statistics to test inter-rater reliability [145]. The scores are 0.76 and 0.70 for teams A and B, respectively, with an average score of 0.73. In addition, we calculate both precision, recall, and F1-score for both teams, as shown in Figure 10.3, and Figure 10.4, respectively. Team A reached an average precision, recall, and F1-score of 0.73, 0.65, and 0.67 respectively. However, Team B gained average scores: 0.66, 0.74, and 0.67 for both precision, recall, and F1-score respectively.

10.2.5 Reconciliation

We have extracted the mismatches in a separate sheet per annotator pair. A sheet contained the actual sentence with each of the annotator’s answers. The task of each annotator pair was to reconcile their mismatches and to reach a final annotation that the two agreed on. We noticed that a significant cause for the mismatches was the rule of longest text span consideration in the annotation guidelines. For example, one annotator labeled the entire phrase ‘Secondary Metabolites’ as MATERIAL while the other tagged only ‘Metabolites’ as MATERIAL. Such cases were the easiest to solve. However, other cases, where an annotator pair could not agree on one correct annotation were discussed with the biodiversity experts. For example, ‘Soil lipid biomass’ seemed to be confusing as

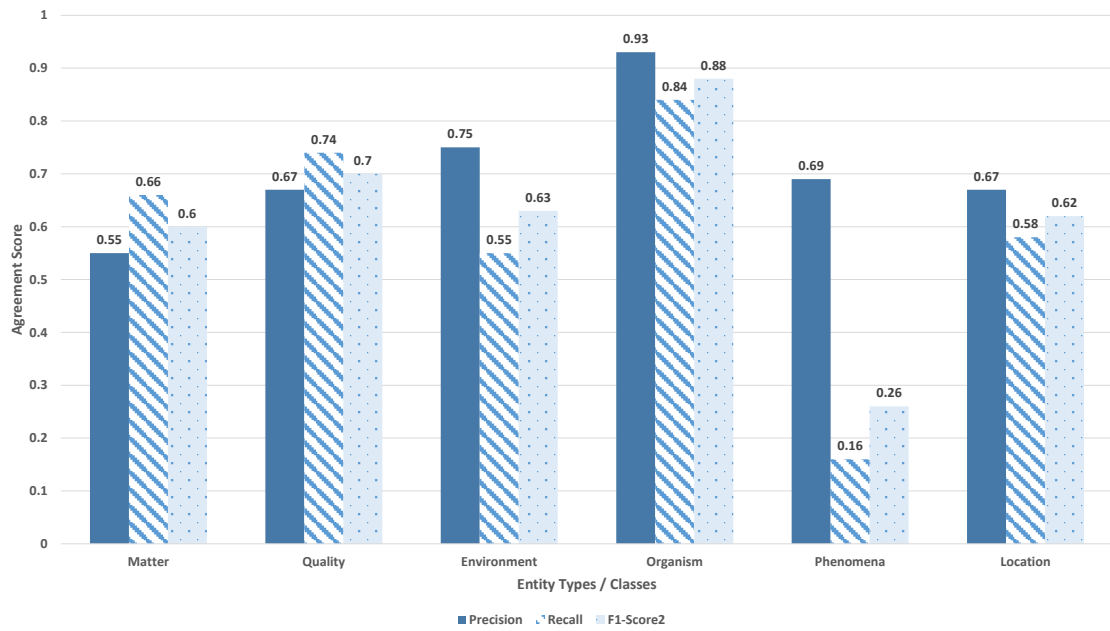


Figure 10.3: Team A: Agreement scores (from [17]).

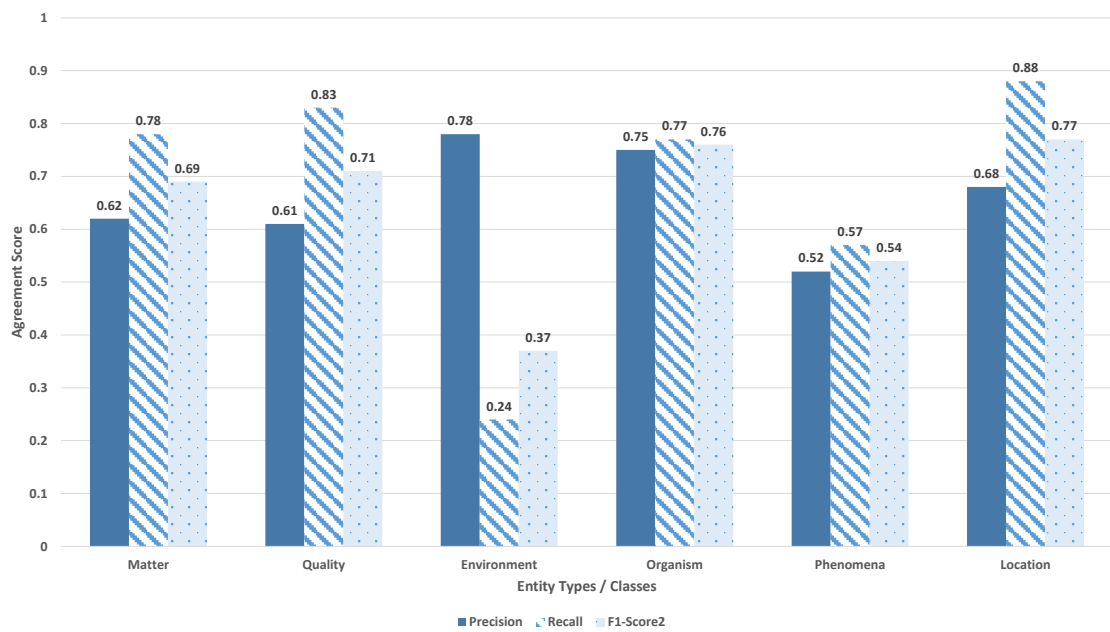


Figure 10.4: Team B: Agreement scores (from [17]).

density	of	soil	invertebrates	varies	in	response	to	earthworm	invasion
B-Quality	O	B-Organism	I-Organism	O	O	O	O	B-Phenomena	I-Phenomena

Figure 10.5: Creating sentence variations from a sentence containing more than two tags (from [17]).

it could be either classified as MATTER or QUALITY. In such a case, we followed the biodiversity expert’s opinion and settled on MATTER.

10.3 BiodivRE Construction Pipeline

In this section, we describe our pipeline of constructing the binary and multi-class RE corpus on top of the BiodivNER. Initially, we transformed the annotated data for NER to suit the RE annotations process. Then, we tried to sample a subset of sentences to obtain a reasonable size of the RE corpus to be annotated. For each sampling method, we detail its advantages and disadvantages. Afterwards, we explain the annotation process for the RE corpus.

10.3.1 Initial Construction

We considered the final NER corpus as an input for the RE corpus construction. We prepared the data in such a way to be more readable. Each sentence is represented by one row followed by its corresponding NER annotations in the following line. The NER corpus contains sentences with multiple tags. However, an RE corpus should be designed in a way that each sentence contains exactly two tags. We generated all possible combinations for sentences with more than two tags, including exactly two tags. Figure 10.5 illustrates an example where one sentence with three tags generates three sentences with two labels. This operation generated a large-scale corpus with more than 52K sentences. We expect a high rate of FALSE (no relation) statements in the generated corpus. However, our task aims at creating an RE corpus with a good balance between TRUE (existing relation) and FALSE sentences. To achieve this, we have to choose a suitable sampling strategy to achieve the best balance among the selected sentences. Therefore, we have explored two different sampling methods. We discuss them in the following sections.

10.3.2 Random Sampling

In the pilot phase of BiodivRE construction, we used a random sampling mechanism among the created corpus. We did not consider any selection criteria. We directly stacked the entire corpus in a list, shuffled it, and randomly picked ‘n’ sentences. We started annotating the resultant smaller corpus, and by doing so, we encountered two issues. At first, we found long sentences with too far tags which makes the existence of a relation between the two tags impossible. Second, some of the relation pairs in the ontology have not appeared in the corpus at all. There are two reasons for the second issue. Either such kinds of relations do not appear in the original corpus or they are missed by the sampler since it purely depends on the random selection. The conclusion from the pilot phase is the need of changing the sampling strategy.

10.3.3 Round Robin Sampling

We developed a balance-biased sampler using a round robin method to have more control over what to include in the final RE corpus. We grouped the sentences from the initial

construction by tag-pair, where a valid pair is the one appearing in the BiodivOnto, and the unsupported co-occurrences were grouped into a new category, ‘Other’. At this stage, we handled the relations bidirectionally between entities of interest to cover cases like ENVIRONMENT have QUALITY and QUALITY of ENVIRONMENT. Afterwards, we iterated over the groups, including the entire set of tag-pairs as well as the ‘Other’ group. We picked one sentence from each group until a threshold was reached. In our case, we selected 4000 sentences as a threshold. An additional criterion is that we limit the number of words between the two entities of interest to a certain value, e.g., 30 words. In this way, we solved the two problems that appeared using the random sampling method. At first, we guarantee that we cover all the relations of the BiodivOnto, if it exists in the text, in the final corpus. Second, we avoid cases with FALSE sentences due to too far entities since it is clear that no relation could exist between them.

10.3.4 Annotation Process

We directly referred to BiodivOnto and limited the accepted relations to those supported by the ontology. On the one hand, for each sentence, we checked whether there is a relation between its two named entities. On the other hand, whether this relation has a semantic correspondence in the BiodivOnto. For example, a verb relation ‘has an impact on’ is considered a synonym for the ontological relation ‘influence’. FALSE examples would be either the relation is not supported by the BiodivOnto or it has a different meaning than the ontological relation. For example, ‘Climate change (B-Phenomena I-Phenomena) impacts the carbon dioxide (B-Matter I-Matter)’ is a FALSE sentence since there is no ontological relation between PHENOMENA-MATTER. Such a sentence would appear since we also choose from the ‘Other’ group in the selected sampling method. Another FALSE example might occur between two entities with a relation in the BiodivOnto. ‘Trees (B-Organism) with extrafloral nectaries (B-Matter I-Matter)’ is a FALSE statement since the word with does not imply the relation influence between ORGANISM and MATTER.

Similar to our procedure to construct the NER corpus, we also applied a pilot phase for RE annotation. The author of this thesis and a PhD student annotated the same 50 sentences that were randomly picked. Afterwards, we calculated the inter-rater agreement (Kappa’s score), which resulted in 0.94. Due to this high score, we decided to split the corpus and individually continue the annotation.

During the real annotation phase, we encountered issues regarding the entity tags, especially for the longest span annotation. This rule does not seem to be correctly followed during the annotation of the NER corpus. For example, ‘earthworm invasion’ was annotated as [B-ORGANISM] [B-PHENOMENA], instead of [B-PHENOMENA] [I-PHENOMENA]. For those cases, we fixed them to follow the rule of the annotation declared originally in the NER guidelines. Figure 10.6 shows samples from an annotation sheet. The first column holds the actual relation label from BiodivOnto that will be used for the multi-class RE corpus. Then, it is followed by a binary relation tag (0- no relation, 1- existing relation). Yellow cells highlight the relation between the two entities of interest in the text. Red cell indicates that there is a relation based on the sentence but not supported by BiodivOnto. In this sentence, the verb ‘degrade’ has an ‘influence’ meaning implicitly. However, we expect to have a relationship that semantically means ‘have’; thus, the sentence is tagged with a 0. Other sentences like the last one indicate no relation at all.

10.4 Evaluation

In this section, we give an overview of our final NER and RE corpora. We illustrate the characteristics of each corpus, e.g., the class distribution in the NER corpus. In addition,

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	influence	1	...	soil	density	varies	in	response	to	earthworm	invasion	with
2	-	O	B-Quality	I-Quality	...	O	O	O	O	B-Phenomena	I-Phenomena	O	O	O
3	have	1	functional	richness	...	on	ecosystem	states	and	processes	and
4	-	B-Quality	I-Quality	O	O	O	B-Environment	O	O	O	O	O	O	O
5	occur_in	1	...	and	diversity	of	chemolithoautotrophic	bacteria	in	saline	barren	soils	.	.
6	-	O	O	O	O	O	B-Organism	I-Organism	O	B-Environment	I-Environment	I-Environment	O	O
7	have	1	ant	abundance	and	diversity	associated	with	natural	habitats	into	urban	habitats	.
8	-	B-Quality	I-Quality	O	O	O	O	O	B-Environment	I-Environment	O	O	O	O
9	0	for	understanding	how	microbial	communities	degrade	plant	biomass	in	natural	systems	.	.
10	-	O	O	O	B-Organism	I-Organism	O	B-Quality	I-Quality	O	O	O	O	O
11	0	density	of	soil	invertebrates	in	response	to	earthworm	invasion	is
12	-	B-Quality	O	O	O	O	O	O	O	B-Phenomena	I-Phenomena	O	O	O

Figure 10.6: A snippet of an RE sheet during annotation (from [17]).

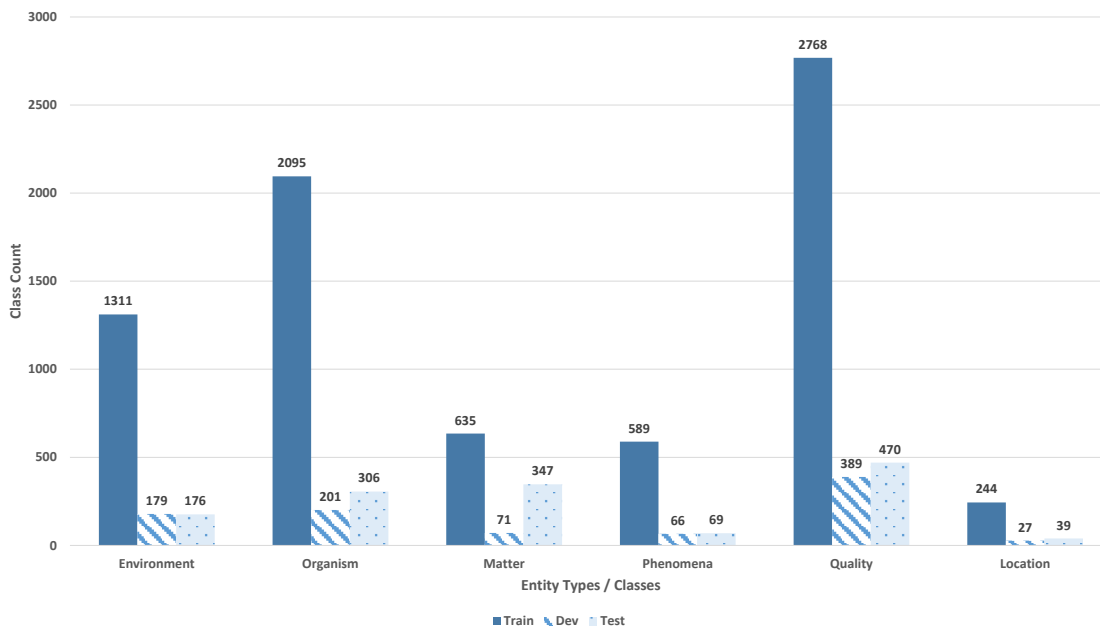


Figure 10.7: Category distribution of BiodivNER corpus (from [17]).

we compare them to existing state-of-the-art corpora.

10.4.1 BiodivNER Insights

The final version of the NER corpus consists of three folds: train, dev, and test because our corpus mainly addresses various tasks in NLP that could be solved based on machine-learning techniques. We followed the split of 80%, 10% and, 10% for the train, dev and, test sets, respectively. All files are given in a CSV format, each of which consists of three entries Sentence#, Word, and Tag, as shown in Figure 10.2.

Figure 10.7 provides an overview of the category distribution inside the BiodivNER corpus in the three data folds. QUALITY represents the most occurring mention in the corpus, followed by ORGANISM and ENVIRONMENT, respectively. However, LOCATION is the least frequent one. The overall distribution reflects a diverse corpus of the most important classes in the biodiversity domain.

Moreover, we compare our BiodivNER to the existing common corpora. Table 10.1 shows the comparison overview in terms of the data sources and document types. Table 10.2 demonstrate the number of annotated documents, number of statements, words, categories, and mentions. Mentions represent how many words are annotated. We also provide the number of unique mentions. COPIOUS corpus is the largest in terms of all aspects except the number of categories. However, BiodivNER covers the greatest number of categories. In addition, BiodivNER is the largest corpus that is based on metadata files of biodiversity datasets as a data source. COPIOUS has two categories closely related to

Table 10.1: State-of-the-art data sources comparison of NER corpora (from [17]).

Corpus	Data Source	Type
COPIOUS	BHL	Publications
QEMP	idiv, BEXIS, Pangeya, Dryad, BFCChina	Dataset Metadata
Species-800	PubMed	Abstracts
Linneaus	PubMed Central (PMC)	Publications
BiodivNER	iDiv, BExIS, Pangeya, Dryad, BEFChina, PubMed	Dataset Metadata, Abstracts

Table 10.2: State-of-the-art statistics comparison of NER corpora. Number of documents, statements, and categories are given by #Doc., #Stat. and, #Cate., respectively (from [17]).

Corpus	#Doc.	#Stat.	#Words (#Tokens)	#Cate	#Mentions (#Annotations)	#Unique Mentions
COPIOUS	668	26,277	502,507	5	26,007	6,753
QEMP	50	2,226	90,344	4	5,154	480
Species-800	800	14,756	381,259	1	5,330	1,441
Linneaus	100	34,310	828,278	1	3,884	324
BiodivNER	150	2,398	102,113	6	9,982	1,033

biodiversity (Habitat and Taxon) and two general Categories (Temporal Expression and Geographical Location). QEMP has four categories derived from the biodiversity domain (Environment, Material, Process, and Quality). As there are already a variety of corpora for species, we only concentrated on missing categories in QEMP. BiodivNER also covers such an essential category in addition to the same closely related classes as QEMP and a general domain LOCATION category.

10.4.2 BiodivRE Insights

Similar to BiodivNER, we created three folds in a CSV format for both binary and multi-class RE corpus. The files consist of two columns: (1) the relation either in a binary or label form, and (2) the sentence where the actual named entities are encoded with their tags. An example line in the file of binary relations: ‘1 Our study shows a significant decline of the @QUALITY\$ of @ENVIRONMENT\$’. However, it would be in the multi-relations files as: ‘have, Our study shows a significant decline of the @QUALITY\$ of @ENVIRONMENT\$’. This format will facilitate the training procedure for any machine learning technique. We followed the same split setting for 80%, 10%, 10% of the train, dev, and test sets respectively.

Figure 10.8 shows the category pairs distribution of the BiodivRE corpus. We calculated the frequencies in a bidirectional order. For example, ORG-ENV represents the total of such a pair and ENV-ORG as well. Since QUALITY is the most frequent class in the NER corpus, this is also reflected in the category pairs ORG-QUA and ENV-QUA. The self-relations that appear in ENV-ENV and PHE-PHE are the least frequent in our corpus. Other category pairs that the BiodivOnto support do not appear in the text used for creating the RE corpus. For example, ORG-ORG and ORG-LOC. The ‘Other’ group represents any co-occurrences that appear in the text and do not exist in the BiodivOnto. In addition, Figure 10.9, Figure 10.10 depict the binary and multi-class annotation distri-

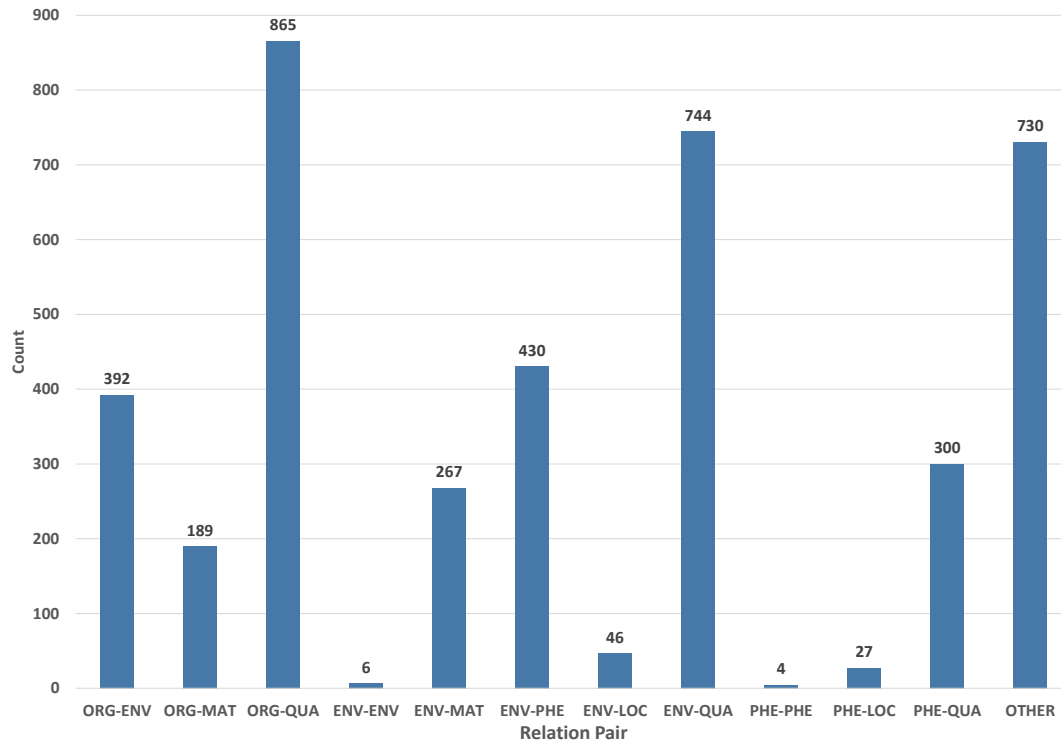


Figure 10.8: Category pairs distribution. For display purposes, category names are abbreviated to three letters (from [17]).

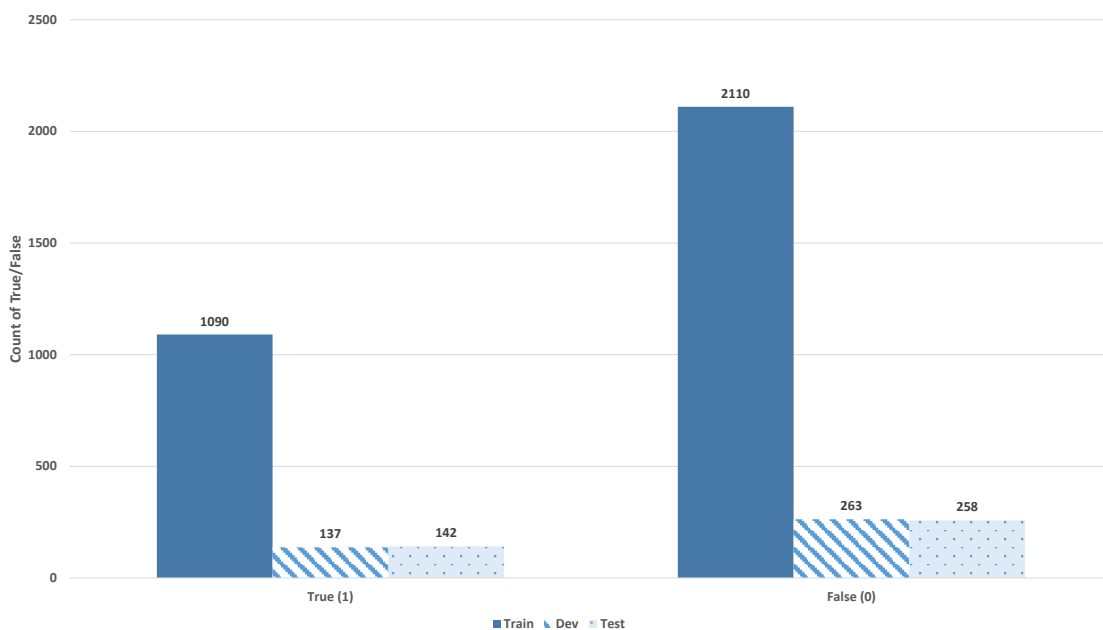


Figure 10.9: The binary distribution of the BiodivRE corpus (from [17]).

bution of the BiodivRE in the three folds of the benchmark. Such that *have* followed by *occur in* are the most common relations in the corpus.

Table 10.3 compares our RE corpus and the biomedical corpora GAD, EU-ADR, and BioRelEx. We selected these corpora for comparison since the data is publicly available and the scope of the annotation is limited to only one sentence, as was the case of our BiodivRE corpus. For example, the COPIOUS corpus discusses the RE part, but the data are unavailable. In addition, BioCreative V [146] uses the entire abstract as a context of

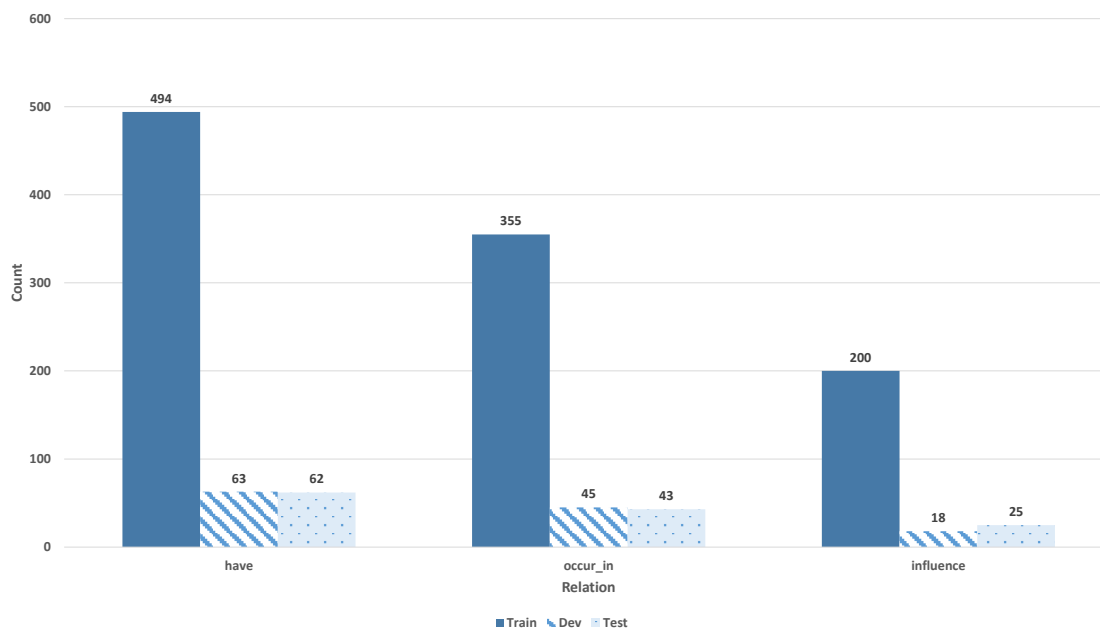


Figure 10.10: Multi-class relations distribution of BiodivRE corpus (from [17]).

Table 10.3: RE corpora comparison (from [17]).

Corpus	Relations	#TRUE Statements	#FALSE Statements	Total
GAD	Binary	25,209	22,761	53,300
EU-ADR	Binary	2,358	837	3,550
BioRelEx	Multi-class	1,379	62	1,606
BiodivRE	Binary, Multi-class	1,369	2,631	4,000

annotation, and thus, we skip it here. For BioRelEx, in the original dataset paper, they have -1, 1, and 0 classes. We use them here as the former two classes map to TRUE while the latter maps to FALSE classes. BiodivRE has a second-place among the existing corpora concerning the number of sentences (4K) with a higher rate of FALSE sentences. There are two reasons behind this high number of FALSE statements. On the one hand, we found that most metadata sentences have a listing format of entities, and we could not guess the relation among them (the most frequent sentences). On the other hand, BiodivOnto is still incomplete; some relations are missing from it. For example, ‘Trees (B-ORGANISM) with extrafloral nectaries (B-MATTER, I-MATTER)’ holds a meaning of *contains*, but we look for *influence*.

10.4.3 Availability and Licensing

Resources should be easily accessible to allow replication and reuse. We follow the FAIR (Findable, Accessible, Interoperable, and Reusable) guidelines to publish our contributions [2]. We release our dataset [142] in such a way that researchers in the community can benefit from it. Our benchmarks are released under the Creative Commons Attribution 4.0 International (CC BY 4.0) License.

10.5 Summary

We introduced BiodivNERE as a package for two corpora for Named Entity Recognition (NER) and Relation Extraction (RE) tasks. Both are based on abstracts and

metadata from the biodiversity domain. We manually annotated and revised them with the aid of biodiversity experts. BiodivNER, the NER corpus, consists of six important classes in the biodiversity domain. Such that these entity types include ORGANISM, ENVIRONMENT, QUALITY, LOCATION, PHENOMENA, MATTER. BiodivRE, the RE corpus, is a binary and multi-class classification benchmark. It contains three relations from the domain including *occur_in*, *influence*, and *have/of*. Both classes and relations are represented in the final version of the BiodivOnto schema. We use both corpora to fine-tune and evaluate our developed textual data interpreter BiodivBERT in Chapter 9. Given our developed framework BiodivBERT and our constructed corpora in this chapter, we see the potential use of BiodivBERT to auto-populate the BiodivOnto ontology and create a knowledge graph from textual data. We released our code publicly available under our GitHub repository¹⁸. In addition, we made both corpora available at Zenodo [142].

¹⁸<https://github.com/fusion-jena/BiodivNERE>

Part IV

Metadata Interpretation

Chapter 11

Meta2KG Framework

Scientific data generated in biodiversity research are very heterogenous and can occur in multiple formats. This is an obstacle for machine processing, which needs additional information for data integration, data search, or data visualization. Therefore, primary research data are described by metadata, and descriptive information about W-questions (what, who, when, where and why). Such metadata are mostly provided in structured formats such as JSON or XML. A metadata file contains essential information for various applications, like dataset search [28]. One way to exploit this untapped wealth is by transforming this raw metadata into Knowledge Graphs (KGs). With this, we can increase the FAIRness [2] of the data by enhancing its re-usability. Ideally, it should be in a machine-understandable format like RDF. This enables data queries using structured query languages like SPARQL and empowers further data usage.

Embeddings are a well-established technique that captures the semantics of a given word or sentence. Previous works have shown their significant impact on many Natural Language Processing (NLP) applications like Word2Vec [128], Glove [129], and fast-Text [95]. Thus, embeddings are our base method for ontology matching. In this chapter, under the third research area of this thesis, **Metadata interpretation (MI)**, we investigate metadata as a source for generating KGs. We introduce a fully-automated approach that transforms raw metadata files into a KG using an embedding-based matching technique. We develop the Biodiversity Metadata Ontology (BMO) as an underlying schema for our technique. We demonstrate the effectiveness of this matching method and discuss common challenges in the automatic transformation process. We tested our technique on a biodiversity use case; however, we expect our method to be domain-independent since we do not rely on any domain-specific mapping rules. We populate the resultant KG with instances from several metadata files as a unified KG. Our results show that metadata files are a promising source for KG construction.

In this chapter, we outline the entire approach for transforming metadata into KGs including the details of our selected data repositories, pre-processing steps, the underlying schema development, embeddings-based ontology matching techniques, and artifacts release in Section 11.1.

We demonstrate the effectiveness and matching results of our developed unsupervised techniques in Section 11.2. We conclude and summarize this chapter in Section 11.3¹.

¹We published a summarized and detailed versions of the methodology explained in this chapter at Abdelmageed and König-Ries, Ontology Matching Workshop, ISWC, 2022 [18], and Knowledge Graph Construction Workshop, ESWC 2023 [19].

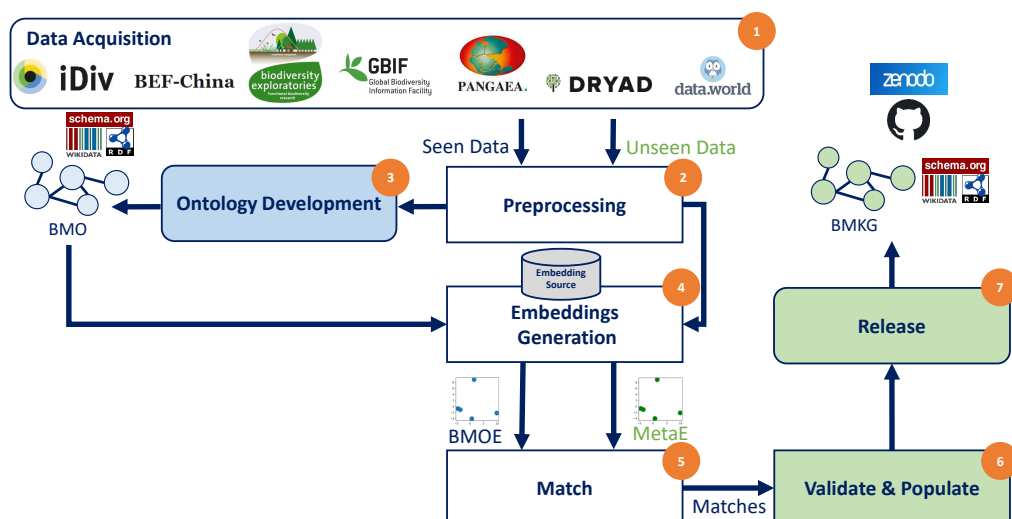


Figure 11.1: Overview of our raw metadata to KG transformation workflow.

11.1 Approach

Figure 11.1 shows the seven phases of our pipeline that we detail in the following sections. It consists of 1) A description of the data sources we used to develop the data model and evaluate our matching technique (Data Acquisition). 2) The preprocessing that we applied on the collected metadata files to facilitate its interpretation (Preprocessing). 3) The process of our data modeling (Ontology Development). 4) The embedding sources we used to generate the word embeddings in addition to vectors construction methods (Embeddings Generation). 5) Our similarity measurement and ontology matching techniques (Match). 6) Our auto-population technique with supported datatype validations (Validate & Populate), and finally, 7) how we published and indexed our contributions, including the Biodiversity Metadata Knowledge Graph (BMKG) (Release). We used the first fold of the collected metadata, ‘Seen Data’ to develop the underlying ontology Biodiversity Metadata Ontology (BMO) and to generate the ontological embeddings BMOE. We used the second fold of the collected metadata, ‘Unseen Data’ for ontology matching and auto-population. Both ‘Data Acquisition’, ‘Ontology Development’, and ‘Release’ stages involve manual labor. The rest of the modules are fully-automated.

11.1.1 Data Acquisition

The first step in this work is to decide the sources of metadata files. We decided to collect them from seven biodiversity data portals that have various characteristics. These portals are German Centre for Integrative Biodiversity Research (iDiv)², BEF-China³, Biodiversity Exploratories (BExIS)⁴, Global Biodiversity Information Facility (GBIF)⁵ and data.world⁶. In addition, we included biodiversity-related metadata from PANGAEA⁷ and, Dryad⁸, both are well established data publishers for ecological data. We queried these portals using keywords identified as typical for the biodiversity domain, the same

²<https://data.botanik.uni-halle.de/bef-china/>

³<https://bef-china.com/>

⁴<https://www.biodiversity-exploratories.de/en/>

⁵<https://www.gbif.org/>

⁶<https://data.world/>

⁷<https://www.pangaea.de/>

⁸<https://datadryad.org/stash>

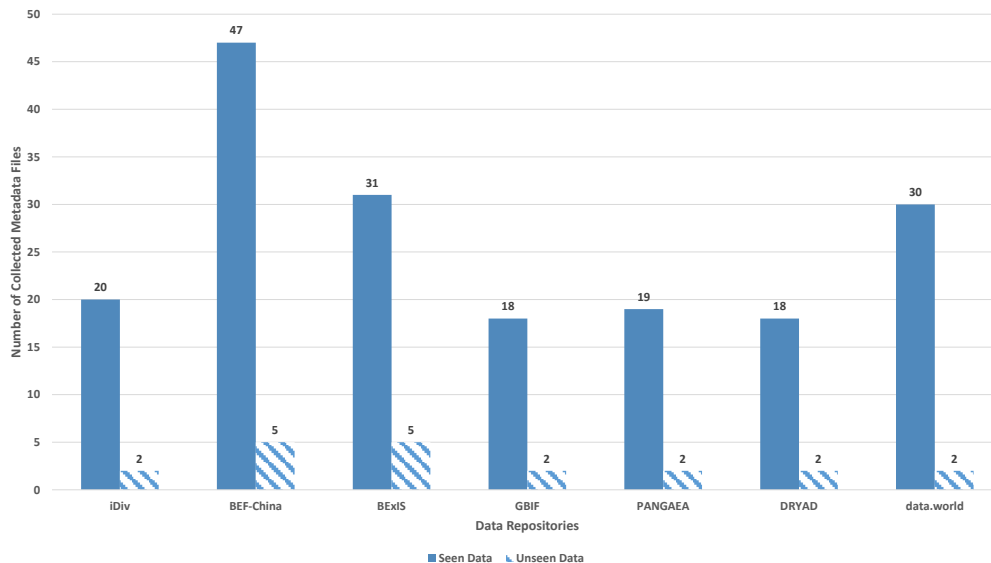


Figure 11.2: Seen & Unseen metadata distribution.

keywords to develop BiodivOnto (see Chapter 8). Such keywords include, e.g., ‘abundance’, ‘benthic’, ‘biomass’, ‘carbon’, ‘climate change’, ‘decomposition’, ‘earthworms’, ‘ecosystem’. We picked the first 50 datasets from each repository from the search results and selected the most complete ones. Those that have mostly completed their metadata values are defined as complete. Figure 11.2 shows the overall distribution of the selected metadata files over the repositories. We divided the collected data into **Seen** and **Unseen** data. For the Unseen data, we picked five files from each repository with the most samples: BEF-China and BExIS, and we selected two files from each of the rest. We considered the remaining metadata samples as Seen data. We use Seen data for modeling the underlying ontology and creating its embeddings. However, the Unseen data is used to create the ground truth, validation and the final KG population.

11.1.2 Preprocessing

We applied a preprocessing step to the ‘Seen data’. It included the conversion of the XML files into a key-value data structure. That way, a key encodes the entire hierarchy of a metadata field. For example, the key `dataset.temporalCoverage...calendarDate` corresponds to the XML in Listing 11.1. Moreover, we cleaned the keys from generic words, e.g., `dataset`, `calendarDate`, `id`, `#text`. We decided on these generic terms by manual analysis of the entire repositories. So, a clean key for this example is `temporalCoverage.beginDate`. We keep the key-value structure, ‘flat dictionary’, in a separate file, and we use it to pre-train word embeddings later in this work.

Listing 11.1: Metadata field XML snippet

```
<dataset id="171">
  <temporalCoverage>
    <rangeOfDates>
      <beginDate>
        <calendarDate>
          2009/07/31
        </calendarDate>
      </beginDate>
    </rangeOfDates>
  </temporalCoverage>
</dataset>
```

Table 11.1: Auto-generated keys, frequencies, and the selected key name.

Auto-generated	Frequency	Selected
versionID	19	version
version	49	version
Short_Abstract	2	abstract
Abstract.Abstract	18	abstract
abstract	362	abstract
DOI	15	doi
doi	2	doi
contact.phone	8	contactPerson_phone
contacts.contactPerson.phone	12	contactPerson_phone
coverage		geographicCoverage
.geographicCoverage		_boundingCoordinates
.boundingCoordinates	57	_eastBoundingCoordinate
.eastBoundingCoordinate		
SpaceBoundingBoxes		geographicCoverage
.BoundingBox	250	_boundingCoordinates
.eastBoundingCoordinate		_eastBoundingCoordinate

11.1.3 Ontology Development

The target of this phase is to find a common vocabulary for the seven data repositories we decided to work with. After the preprocessing step, we calculated the frequency of each key in the Seen data to analyze the used keys for each data repository and get insights on the most common keys in the biodiversity metadata in general. Table 11.1 shows a sample of the auto-generated keys after we apply the cleaning steps and their frequencies. The last column depicts our chosen key that would appear in the ontology. The selected format would be the shared vocab among all data portals. The selected repositories use various syntactic representations for the same semantic meaning. For example, ‘abstract’ conveys the information from both fields: ‘Short_Abstract’, and ‘Abstract.Abstract’. We manually analyzed the resultant cleaned and grouped keys to develop a shared schema that aligns our data repositories. We kept the ‘Selected’ key with all its synonyms. Such selected keys represent our schema. We use its synonyms to generate the embedding of the key later in this work. We held several meetings with a biodiversity expert to validate and review such schema. During those meetings, we integrated the biodiversity expert’s opinion, e.g., we included other vocabularies for one data repository, i.e., BExIS. Thus, this phase, ontology development is an iterative process where we integrated the feedback from the domain expert.

We used the python module, `rdflib`⁹ to create the RDF file for the schema, the Biodiversity Metadata Ontology (BMO). We reused existing vocabulary from `schema.org`. In addition, we defined a new concept under BMO namespace if it did not exist. For example, we reused ‘Organization’, ‘Person’, and ‘Address’ from `schema.org`. However, we defined both ‘Taxonomic Coverage’, and ‘Geographic Coverage’ using BMO namespace. In addition, we used datatype properties from Wikidata and Dublin Core¹⁰.

Figure 11.3 depicts the concepts and relations of the Biodiversity Metadata Ontology (BMO). The dashed lines represent the `subClassOf` relation where the dashed node

⁹<https://rdflib.readthedocs.io/en/stable/>

¹⁰<https://www.dublincore.org/specifications/dublin-core/dcmi-terms/>

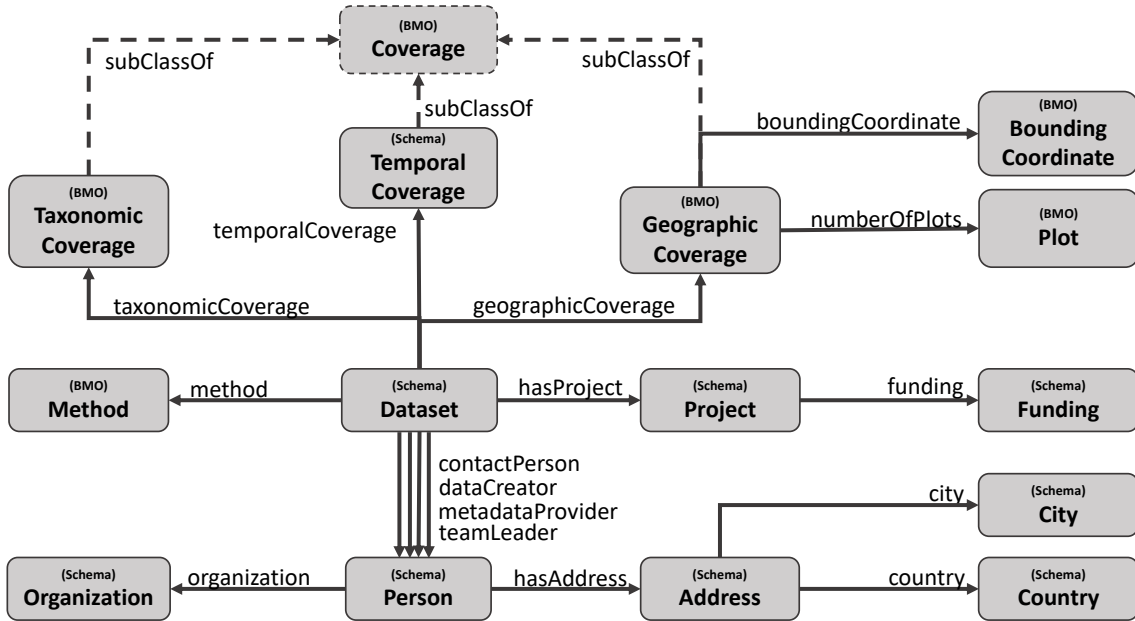


Figure 11.3: Biodiversity Metadata Ontology (BMO) concepts and relations.

notes the parent class. Other nodes and lines represent concrete classes and relations, respectively. We demonstrate the properties of our main concept `Dataset` in Table 11.2. The ‘Match’ column denotes the `skos:exactMatch` from the corresponding source except for both `license` and `accessRights`, they represent `skos:closeMatch` due to a range mismatch between our properties and those defined in Dublin Core.

11.1.4 Embeddings Generation

In this section, we explain the embeddings sources and the methods we developed to transform the keywords into embedding space.

Embedding Sources We supported two variations of embeddings. On the one hand, for domain-specific embeddings, we trained a fasttext [95] model on the Seen data by converting the key-value pairs, ‘flat dictionary’ (see Subsection 11.1.2), into synthetic sentences. Iteratively, we used both the key and its value in such a dictionary to create the corresponding sentence. On the other hand, for the pre-trained embeddings¹¹, we used the publicly available Wikipedia-based embeddings. We used these resources to generate both ontological embeddings (*BMOE*) and metadata embeddings (*MetaE*) for the unseen data. We compare both embedding sources during our experiments.

Generation Method Since our selected repositories use different keywords representing precisely the same thing. For example, BEF-China, GBIF, and BExIS use `geographicCoverage`, and DRYAD uses only `Coverage` to describe the geographical specs of a study. The same applies for `taxonomicCoverage` that BEF-China, GBIF use, and BExIS, whereas `Taxonomic_Scope` and `TaxonCoverage` are used by iDiv and PANGAEA, respectively. Thus, we used the list of synonyms for each selected key that was created during the BMO development. We aim to obtain embedding vectors of BMO relations that encode information from synonyms. For example, a vector for ‘version’, represents the version of the dataset, would be a function of all its synonyms: ‘verion’, ‘versionID’.

¹¹<https://fasttext.cc/docs/en/english-vectors.html>

Table 11.2: Properties of our main concept: **Dataset**. Short forms; SCH, DCT, and WD map to schema.org, Dublin Core Terms, and Wikidata respectively.

Type	Property	Match	Source	Meaning
datatype	title	name	SCH	The title of the dataset
	abstract	abstract	SCH	The short text that summarizes a dataset
	description	description	SCH	The summary of the dataset
	language	inLanguage	SCH	The language of the data provided
	intellectualRights	<i>accessRights</i>	DCT	Specify if the data is public or private
	license	<i>license</i>	DCT	The license of the dataset
	citation	citation	SCH	How to cite the dataset
	dataFormat		BMO	The dataset format, e.g., delimiter
	version	version	SCH	The version of the dataset
	keywordsSet	keywords	SCH	Dataset tags
	doi	P356	WD	Dataset DOI
	alternateIdentifier	identifier	SCH	Dataset download or home URL
	publicationDate	datePublished	SCH	When the dataset is published
	numberOfRecords	P4876	WD	How many records in the dataset
object	contactPerson		BMO	The contact person of the dataset
	metadataProvider		BMO	Who provided the metadata
	dataCreator		BMO	Who created the data
	project		BMO	The associated project of the dataset
	geographicCoverage		BMO	The geo specs of the study
	temporalCoverage		BMO	The time duration of the study
	taxonomicCoverage		BMO	The included taxons of the study

We developed two methods for approaching such an idea: 1) *Mean*: An embedding vector of a given key e_{key} is determined as the mean vector of all its synonyms set as defined as SE in Equation 11.1. 2) *Weighted Mean*: Similar to the Mean method and inspired from TF-IDF¹², we gave higher weight to the more specific words that form an entire key. For example, `temporalCoverage.startDate`, `startDate` would have double the weight of `temporalCoverage`. `temporalCoverage` is a less discriminative word since it would appear with another term like `endDate`. This method is described in Equation 11.2 where es_{ij} is the individual word vector of a given key of synonyms set SE , and we use the word position j as the weight. We use the embeddings generation methods to transform BMO ontology and the Unseen data keys into the embeddings space. Both would be

¹²<https://en.wikipedia.org/wiki/Tf-idf>

mapped into ‘BMOE’, and ‘MetaE’, respectively. We use both embeddings to perform the matching operation in the following section.

$$(11.1) \quad e_{key} = \frac{\sum_i^{|SE|} \sum_j^{|Words|} es_{ij}}{|SE|}$$

$$(11.2) \quad e_{key} = \frac{\sum_i^{|SE|} \sum_j^{|Words|} es_{ij} \times j}{|SE|}$$

11.1.5 Match

In this phase, we worked with the Unseen data. We performed the same procedure of pre-processing to obtain clean keys. One significant difference between this step and BMO embeddings is that this step has no synonyms; however, the mean-based operations are only done on the key level. For matching, we used cosine similarity in the embedding space between the ontological embeddings, *BMOE*, and metadata embeddings, *MetaE*. For each *MetaE*, we retrieve the closest BMO vector that has $\geq 70\%$ similarity. We avoid the closest assignment for better recall. We chose such a threshold to balance the precision and recall. We tried higher thresholds; however, it misses a lot of true matches. This makes sense since the target ontological embeddings are created using a mean or weighted mean operation; thus, a 100% similarity will never be achieved. This step matches the unseen data to the ontology concepts and properties; however, it lacks the instances.

11.1.6 Validate & Populate

To populate the BMO with instances, we rely on the ‘flat dictionary’. In that sense, the key has mapped to, e.g., ontology property, and its value represents the instance we add to the ontology. Auto-populating such ontology given only matches from the step above is not accurate for two reasons: 1) invalid entries in the metadata fields, and 2) misclassification that yields datatype violations. We limit the population of a triple if and only if its value has the expected datatype. For example, we populate `dataCreator_Phone` if the corresponding value is a phone. We cover basic datatype validations using regular expressions for the following datatypes: *Phone*, *Email*, *Coordinate*, *URL*, *Decimal*, and *Date*. In addition, we validate the resultant KG using the W3C RDF Validation Service¹³.

11.1.7 Release

Resources should be easily accessible to allow replication and reuse. We follow the FAIR (Findable, Accessible, Interoperable, and Reusable) guidelines [2] to release our contributions. We release our ground truth [147], ontological embeddings [148], BMO [149], and BMKG [150] in Turtle, N-triples, and RDF-XML format in Zenodo, so researchers in the community can benefit from them. We published our resources and code under the Creative Commons Attribution 4.0 International (CC BY 4.0) and Apache License 2.0, respectively.

11.2 Evaluation

We conducted several experiments to demonstrate the effectiveness of the generated embeddings. Besides the two mean-based methods (mean and weighted mean) for embeddings

¹³<https://www.w3.org/RDF/Validator/>

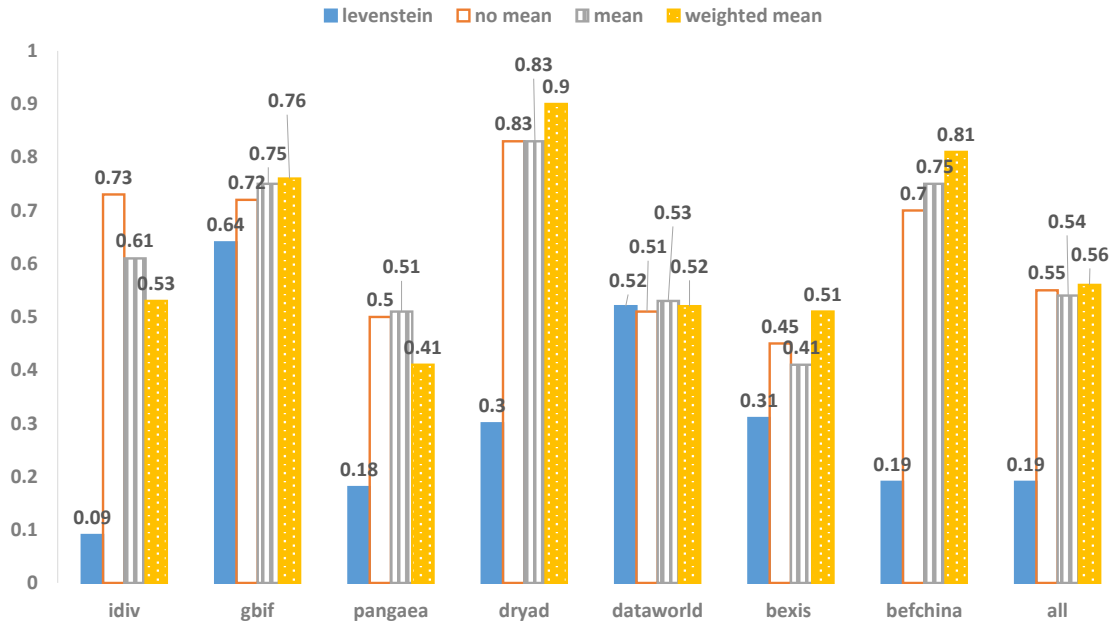


Figure 11.4: Matching F1-score on Unseen data using our embeddings.

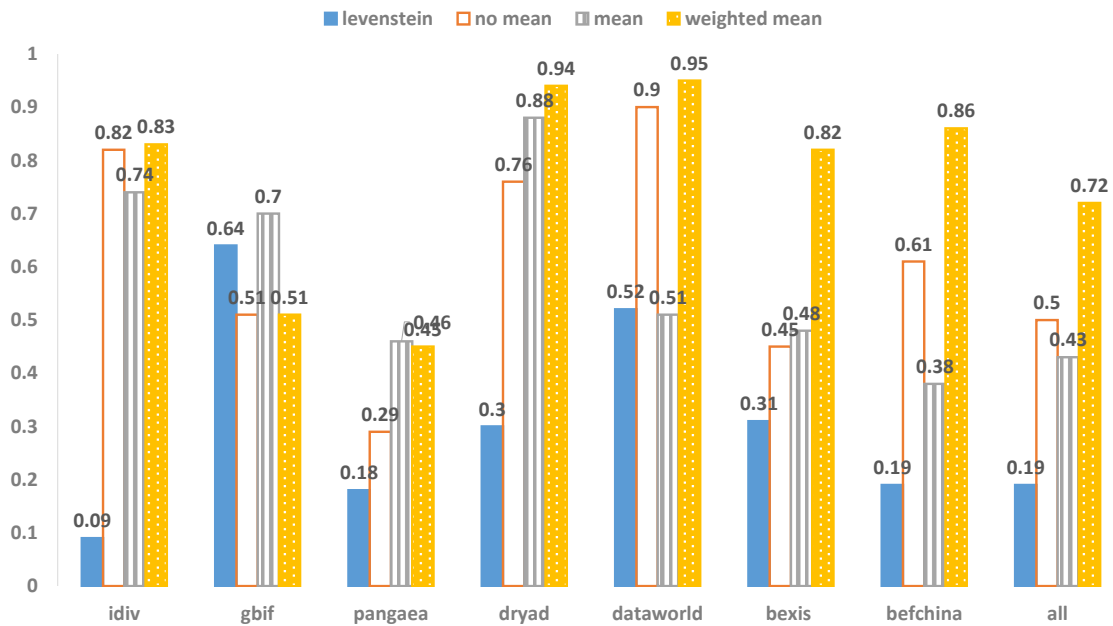


Figure 11.5: Matching F1-score on Unseen data using Wiki-based embeddings.

generation, we developed a two baseline approaches. On one hand, to test the effectiveness of embeddings we developed a base line approach based on string similarity using Levenshtein distance (levenstein). On other hand, to test the effectiveness of mean operations, we handled each key in the Unseen data as a single word without any kind of splitting, then, transform that word to a vector (key to vec). For evaluation, we manually annotated the cleaned Unseen Data with the correct match from the ontology. We use such ground truth to evaluate our matching technique. We considered the value of the auto-generated key to classify it. In the following, we show our matching results and give insights about the resultant auto-generated KG.

11.2.1 Matching Results

Since we have two embedding sources (our custom embeddings and pre-trained Wikipedia-based embeddings) and three techniques (baseline (no mean), mean, and weighted mean) to obtain an embedding vector with an additional lexical baseline (levenstein), we conducted seven experiments to cover all combinations. Figure 11.4 and Figure 11.5 show the F1-score for all experimental settings using our own and wikipedia-based embeddings. We calculated the scores per data repository and the accumulated them as well (all). We found that the Weighted Mean approach combined with the pre-trained Wikipedia-based embeddings yielded the best scores. This proves that our developed mean-based method successfully captured a wide range of syntactic representations from metadata keywords. Since we used synthetic sentences that are derived from a combination of metadata key and value, ‘flat dictionary’ items combined, we lacked proper natural text during the training. Thus, it justifies the lower scores with our custom embeddings. From the repositories perspective, our approach gained the lowest scores on PANGAEA due to the lack of proper metadata fields, thus, confusing our matching procedure. However, our method reaches, at some times, 100% precision on Dryad due to its relatively more straightforward fields to match, e.g., ‘title’.

11.2.2 Resultant Knowledge Graph

Our resultant BMKG represents BMO with instances. It contains those instances from the Unseen data. Figure 11.6 represents the frequency of triples in the BMKG. Darker colors depict higher triple frequency. `Dataset` datatype properties, e.g., `keywords`, `citation`, and `title`, are the most occurred triples in the graph from the unseen metadata files. They are auto-populated correctly with valid instances. The data properties are followed by the `DataCreator` and `ContactPerson`. The `NumberOfPlots` seems to be more frequently used than the `BoundingCoordinates` under the `GeographicCoverage`. The `MetadataProvider` is frequently incomplete compared to both `DataCreator` and `ContactPerson` since it is usually described by `givenName` and `phone` only.

We gave a closer look to the BMKG where we manually investigated the populated `Dataset` instances using Protégé¹⁴. Our regular expressions-based validations for datatype managed to filter invalid triples. For example, all `phone`, `email`, `startDate`, and `endDate` triples are correct. However, we still need a more sophisticated method to validate more triples. For example, we found a `citation` triple that contains only the year of publication. In addition, since we rely on a fully-automated procedure for matching, it would yield false statements. For example, a `license` is classified as `intellectualRights` or vice versa.

11.3 Summary

We investigated the construction of a Knowledge Graph (KG) using metadata as the only source of data. Our pipeline is tested on, but not limited to, a biodiversity domain use case. We demonstrated our used data repositories: seven biodiversity data portals. We manually collected the metadata files from them. We divide them into Seen and Unseen data. We used the Seen data to construct the underlying data model that aligns the selected data portals. In addition, we used them to transform the constructed ontology into the embedding space. We used the Unseen data to evaluate our unsupervised matching techniques and auto-populate the BMO with instances. Such embeddings-based techniques are based on the mean operation where the similarity measure is the cosine similarity. We demonstrated the effectiveness of the developed matching and population

¹⁴<https://protege.stanford.edu/>

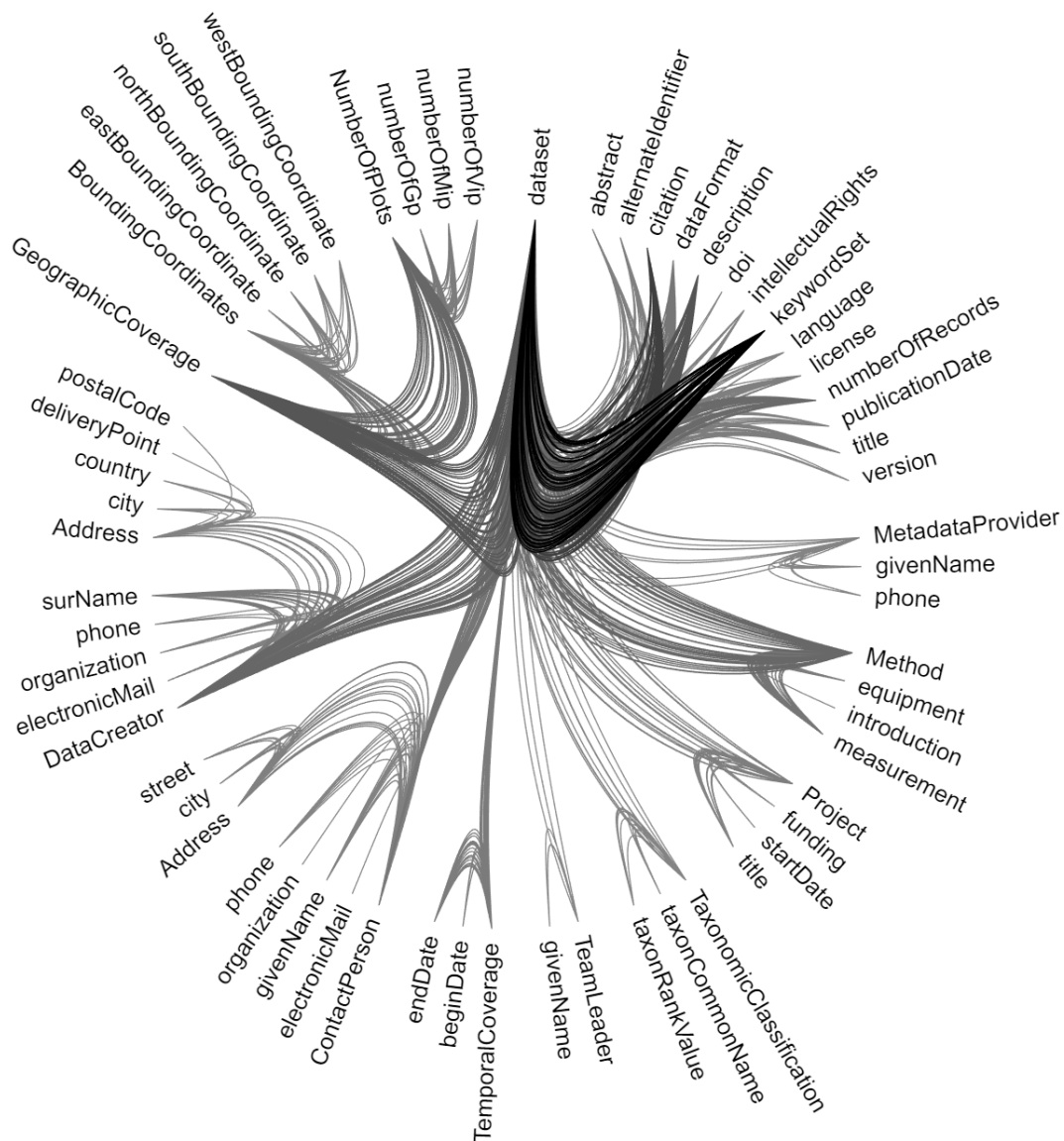


Figure 11.6: Biodiversity Metadata Knowledge Graph (BMKG) triples frequencies. capital and lower nodes represent objects, and literals respectively.

techniques. In addition, we showed the current limitations of the methodology, and we pointed out possible solutions for them. Besides the transformation pipeline, we presented the Biodiversity Metadata Ontology (BMO) and Biodiversity Metadata Knowledge Graph (BMKG) as byproducts of this work. We release our resources and code publicly under our GitHub repository¹⁵. In addition, we release our ground truth [147], ontological embeddings [148], BMO [149], and BMKG [150] in Turtle, n-triples, and RDF-XML format in Zenodo.

¹⁵<https://github.com/fusion-jena/Meta2KG>

Part V
The Final

Chapter 12

Evaluation

We defined three research areas covered by this dissertation in Chapter 2. We repeat them here to have a clear picture of what we evaluate. In the field of tabular data interpretation (TabI), we match the individual table components (cell, column, column-pair) to their counterparts from existing KGs. Regarding textual data interpretation (TexI), we extract both entities and relations of interest from unstructured text. Concerning metadata understanding (MI), we also grasp the entities and relations from semi-structured data (metadata).

In the last three parts, we demonstrated our key contributions to the three research areas behind this work. Part II contains two contributions: i) **JenTab** is a framework that matches tabular data to KGs. It solves three Semantic Table Interpretation (STI) tasks to achieve its ultimate goal. ii) **BiodivTab** is the biodiversity-specific benchmark for STI tasks. Part III shows our core contributions to interpreting textual data. We extract structured information from unstructured text. Thus, we have i) **BiodivOnto**, which represents concepts and their relations of interest. We demonstrated our data-driven approach to reach those concepts and relations with the continuous enhancements we did using other data sources. ii) **BiodivBERT** is a BERT-based model that we pre-trained on domain-specific corpora. We fine-tuned it in two downstream tasks, Named Entity Recognition (NER) and Relation Extraction (RE). Both tasks are essential to extract structured information from the text (entities/classes and properties). iii) **BiodivNERE** is a package that contains biodiversity-specific NER and RE corpora. We explained our methodology to construct both of them. Part IV includes our contribution to transform the semi-structured metadata files into a KG. Such framework is **Meta2KG** that introduces matching and auto-population embedding-based techniques for an ontology given a semi-structured XML or JSON file.

In Parts II to IV, we already evaluated the individual contributions. In addition, this chapter takes a more holistic view and discusses in how far the overall objectives and requirements of the thesis identified in Chapter 2 could be met. So, we evaluate our contributions concerning the pre-defined objectives (Objective1-Objective8) and requirements. We have ten requirements for the first two research areas TabI and TexI, (R1.1-R1.10), and (R2.1-R2.10), respectively. We defined five requirements (R3.1-R3.5) for the last research area TabI. Table 12.1 summarizes these requirements.

In this chapter, we evaluate our contributions concerning the three research areas in Section 12.1. We discuss the shortcomings and limitations of the current solutions in Section 12.2 We summarize this chapter in Section 12.3.

Table 12.1: The summary of this thesis requirements.

Requirements	Objective	Summary
Tabular data interpretation (TabI)		
R1.1	Objective 1	Tabular data to KG matching framework.
R1.2	Objective 1	The framework should support easy extensions and configurable components.
R1.3	Objective 1	The framework should be scalable.
R1.4	Objective 1	The framework should have a reasonable amount of dependencies and easy to configure locally.
R1.5	Objective 1	The framework should provide an analysis of the current processing input.
R1.6	Objective 1	The framework should store the results and able to extract them in a specific format.
R1.7	Objective 1	The framework should have trusted accuracy scores and effective in the biodiversity domain.
R1.8	Objective 2	The domain-specific benchmark should reflect real-world challenges.
R1.9	Objective 2	The benchmark should have annotations that convey human-level knowledge.
R1.10	Objective 2	The benchmark should be diverse.
Textual data interpretation (TexI)		
R2.1	Objective 3	The underlying schema of the technique should reflect the important concepts and relations of the domain.
R2.2	Objective 4	The model should be trained on domain-specific textual data.
R2.3	Objective 4	The model should detect the most important entities and relations in the domain.
R2.4	Objective 4	The model should be easy to use.
R2.5	Objective 4	The model should have trusted scores.
R2.6	Objective 5	The evaluation corpora should be diverse.
R2.7	Objective 5	The evaluation corpora should be gold-standard
R2.8	Objective 5	The evaluation corpora should be aligned with the schema in R2.1.
R2.9	Objective 5	The evaluation corpora should support machine learning format.
R2.10	Objective 5	The evaluation corpora should be demonstrate a good balance for classes and relations.
Metadata interpretation (MI)		
R3.1	Objective 6	The underlying schema of the technique that should reflect the most common vocabulary in the domain.
R3.2	Objective 7	The technique should be able to capture the most important entities and relations in the domain.
R3.3	Objective 7	The technique should effectively auto-populate the underlying ontology.
R3.4	Objective 7	The technique should produce a machine-readable format output.
R3.5	Objective 8	The technique should show the effectiveness of the extracted information.

12.1 Achievements

In this section, we evaluate our two key contributions for the three areas of research. First, we evaluate the two contributions regarding the Tabular data interpretation (TabI). We explain how they meet the predefined requirements under this research area. On the one hand, JenTab represents our framework that matches tabular data to KGs. On the other hand, BiodivTab demonstrates our manually annotated biodiversity-specific benchmark for Semantic Table Interpretation (STI) tasks.

JenTab We developed a modular approach called JenTab to solve STI tasks. This framework aims to match the individual components of tabular data to a KG (R1.1). JenTab is easily extensible and its components are highly configurable. For example, it provides an easy way to change the target KG and enable various settings for solving the STI tasks. Currently, it contains six pipelines that we developed based on various dataset characteristics. In addition, it supports annotations from both Wikidata and DBpedia. By this means, JenTab fulfills R1.2. JenTab adopts a distributed approach that leverages the capabilities of the individual client machines. This allows for a scalable framework that can process large-scale tabular data benchmarks. We tested JenTab on more than 156,000 tables in the scope of the SemTab challenge. This demonstrates the requirement fulfillment of R1.3. JenTab fulfills Requirement R1.4 by having a reasonable amount of open-source dependencies and is well-documented. In addition, it supports an easy-to-use execution via Docker containers, as seen in Listing 12.1.

Listing 12.1: JenTab docker commands

```
1. docker-compose -f docker-compose.manager.yml up
2. docker-compose -f docker-compose.yml up
3. docker run --network="host" runner
```

JenTab won the second prize in the ‘Usability track’ by IBM research in 2021. In addition, JenTab is the only system participant that obtained the ‘Artifact Availability Badge’ by SemTab 2022. This badge is awarded for systems that are open-source data and code, well-documented, and have open-source and reasonable dependencies. The open-source data includes the pre-computed lookup and the system solutions. We release both of them to be publicly available to enable further analysis by future work. For example, Avgardro et al. [151] used the output of our system in a deeper quality check framework. JenTab demonstrated effectiveness in this work compared to other state of the art. In addition, Chaves-Fraga and Dimou [152] used our artifacts to compare the fully automatic systems versus the declarative mapping rules.

JenTab provides an analysis of the current processing input tables. Figure 12.2 depicts a screenshot of the centralized node of JenTab. It shows how many tables are in progress, successfully completed, failed to complete, or the errors that were returned. In addition, it stores the results and has the feature to extract them in the required format. We demonstrate the output format in Figure 12.1 where Wikidata is the target KG. First, CEA results, it includes the file/table name without extension, row id, column id, the combination of them yield into a specific cell and the mapped entity from KG. Second, CTA results, it consists of file/table name, target column id, and the mapped semantic type or class from the KG. Finally, CPA output, it shows the file/table name, subject and object columns id, and their semantic property or relation from the KG. The Manager enables a debug feature by retrieving tables with no or incomplete annotations to help us investigate these hard cases closely. Such features helped us identify each dataset’s characteristics and facilitate error analysis. Moreover, the Manager demonstrates audit statistics, i.e., the loads on individual system components (e.g., CEA Creation Module). Thus, it makes the system more transparent and achieves the requirement R1.6.

filename	row_id	col_id	entity	filename	col_id	type	filename	subj_id	obj_id	prop
A	B	C	D	A	B	C	A	B	C	D
1	X3QM4E4N	1	0 Q7886316	1	X3QM4E4I	0 Q9035798	1	X3QM4E4N	0	1 P1082
2	X3QM4E4N	2	0 Q2221050	2	D57LSWY2	0 Q17201685	2	D57LSWY2	0	1 P2046
3	X3QM4E4N	3	0 Q2221433	3	9XIWHWO	0 Q17201685	3	9XIWHWO4	0	1 P2046
4	X3QM4E4N	4	0 Q2221050	4	KGKU5BIK	0 Q17201685	4	KGKU5BIK	0	1 P2046
5	X3QM4E4N	5	0 Q7886269	5	T20QAA67	0 Q17201685	5	T20QAA67	0	1 P2046
6	X3QM4E4N	6	0 Q7886269	6	PLELE23Q	0 Q17201685	6	PLELE23Q	0	1 P2046
7	D57LSWY2	1	0 Q968756	7	PE41RDSF	0 Q17201685	7	PE41RDSF	0	1 P2046
8	D57LSWY2	2	0 Q7886173	8	EP61J3CK	0 Q17201685	8	EP61J3CK	0	1 P2046
9	D57LSWY2	3	0 Q5638890	9	3ZXBWWC	0 Q17201685	9	3ZXBWWOU	0	1 P2044
10	D57LSWY2	4	0 Q5638890	10	R6FOATDC	0 Q17201685	10	R6FOATDO	0	1 P2044

Figure 12.1: JenTab: Output snippet. (a) CEA, (b) CTA, and (c) CPA.

Currently Processing

- Tables (total): 4649
- Results returned: 4025
- Errors returned: 98
- Unfinished: 624
- Finished: No

Average time until ...

- Results: 47 seconds
- Errors: 11 minutes 13 seconds
- Finished: 8 hours 15 minutes 10 seconds

Clients connect within ...

- Last minute: 0
- Last 15 minutes: 0
- Last hour: 0
- Last day: 0

Mappings

- CEA: 19179 / 20411 (93.96%)
- CTA: 3861 / 4278 (90.25%)
- CPA: 3256 / 3919 (83.08%)

Submission Data

This does not check whether data for all tables has been received. It is only a snapshot of the current state of submitted solutions.

- CEA: 1 solved, 1 missing mappings
- CPA: 1 solved, 1 missing mappings
- CTA: 1 solved, 1 missing mappings

Missing Mappings

This is not concerned whether the obtained mappings are correct, but only lists the tables where mappings are missing. Lists at most 10 tables per category. Mind the dependencies: if fewer CEA mappings are found, also the CTA and CPA mappings will deteriorate.

CEA

No Mappings	Missing Some Mappings		
Name	Name	Missing	Total
KTG02AJT	SSHFDU1X	8	12
MT9YBQC1	WSD27W2Y	5	12
O37CZFX8	MUSOT714	5	6
HIUTFRBX	BMUH600P	5	7
UC00CVH3	SMHJNQ3	5	6
8WQAZFK1	EJ40TU7J	5	10
NFT3OLYI	CY56B1WH	4	5
OY8XY765	YVTS9L5B	4	5
BDIJMTQD	FPV9NTYO	4	7
3UPWOYHZ	J7W9A0NE	4	6

CPA

No Mappings	Missing Some Mappings		
Name	Name	Missing	Total
S332WBIC	BKRL2AKL	2	3
A3YQ16EC	5QNC4HK0	2	3
FUL936TS	9396Z88H	2	3
WJRYA69B	N224517P	2	3
TVXFU6CJ	LG9C41BR	2	3
GXOMA66V	9ZYUUTWM	2	3
TXC52I8Q	5P47EXDI	2	3
41TGT669	F2G9RECM	2	3
SYD5RIJ	K33SSOS4	2	3
5HCEGPV0	9LKATUTG	2	3

CTA

No Mappings	Missing Some Mappings		
Name	Name	Missing	Total
F2BKSSWX	60QZBSXD	2	3
S332WBIC	NL104UTT	2	3
8CUEHHYY	USSROQ4T	1	3
M21Z4QUF	ISYRVQ1E	1	2
HRJ155MZ	B606N5T7	1	2
41TGT669	YUG0GFQD	1	2
SYD5RIJ	YDIQAKBS	1	2
AAW06NXC	7VLJ4ANI	1	2
33GARHA5	QZZ69YKK	1	2
KTG02AJT	OISVRHPV	1	4

Audit Analysis

The data here shows the effectiveness for each method of pipeline in each step.

	CEA	CTA	CPA
Creation	genericLookup, 1325446 tokens, 62074 fullCell, 6084	default, 1110	default, 921
Selection	stringSimilarity, 1371987 colSimilarity, 24314 missingCeaByCta, 17682	LCS, 662 directParents, 0 popularity, 0	majorityVote, 547

Figure 12.2: JenTab: Manager screenshot.

JenTab is a top performer system that solves the STI tasks. During its years of development, it has high accuracy scores in the general domain benchmarks. On average, and given all the Automatically Generated (AG) datasets, it achieved 0.91, 0.93, and 0.97 F1-scores for CEA, CTA, and CPA tasks, respectively. In the biodiversity domain, JenTab was found to be less effective than in the general domain. It achieves 60% and 41% for CEA and CTA tasks for the BiodivTab dataset. Such scores are quite low compared to those gained for the AG datasets due to the unique characteristics a biodiversity dataset could have. We discuss such differences in the following section. The obtained scores on both domains achieve the last requirement, R1.7.

BiodivTab We constructed a biodiversity-specific benchmark for STI-tasks called BiodivTab. It is manually annotated tabular with Wikidata and DBpedia KGs. BiodivTab

is based on real biodiversity research tables and data augmentation. However, the latter is applied with real-world challenges that we obtained during the analysis phase of data collection. So, we argue that no artificial challenges were added to the benchmark, and thus, it reflects the common real-world challenges in the domain. So, BiodivTab fulfills the requirement R1.8. Since the benchmark is manually annotated and a domain expert partially verified the annotations, we argue it is a gold-standard level benchmark and reflects the expert level of annotations; thus, it achieves requirement R1.9. In addition, we collected various biodiversity data tables from three data portals with multitudinous characteristics. Thus, BiodivTab represents a diverse collection of biodiversity tables and completes R1.10.

BiodivTab brought real-world challenges to the STI community over the last two years. It won the first place prize from IBM research during the ISWC 2021. The average scores achieved on BiodivTab by existing methods, including JenTab, are much lower than the synthetic datasets. Its Wikidata version that we published in 2021 has, on average, 0.40 and 0.28 F1-scores for CEA and CTA, respectively. For its DBpedia version, which we released in 2022, it has, on average, 0.73 and 0.65 F1-score for CEA and CTA tasks, respectively. The participating systems solving the DBpedia version are KGCODE-Tab and JenTab only.

We identify the characteristics of synthetic dataset and BiodivTab as an example domain-specific benchmark. On the one hand, we summarize the challenges we encountered in the AG datasets in Figure 12.3 as follows:

- (a) missing or not descriptive table metadata, like column headers.
- (b) spelling mistakes.
- (c) ambiguity in cell values. For example, *UK* has (*Ukrainian (Q8798)*, *United Kingdom (Q145)*, *University of Kentucky (Q1360303)* and more) as corresponding entities in Wikidata.
- (d) missing spaces, causing tokenizers to perform poorly.
- (e) inconsistent format of date and time values.
- (f) nested pieces of information in *Quantity* fields, interfere in the corresponding CPA tasks.
- (g) redundant columns.
- (h) encoding issues.
- (i) seemingly random noise in the data. *Berlin* would be expected in the context of the given an example.

Subject Column	← Object Columns / Properties →			
Country	Inception (LITERAL)	Area (LITERAL)	Label (LITERAL)	Capital (IRI)
Country	Col1			
Egypt	1922February, 28	1,010,407.87 km2 (... ft2)	Egypt	Cairo
Germa?ny	3 October 1990 (03.10.1990)	357,400 km2 (... ft2)	Germany	TÃ¼bingen
UK	??	NA	United Kingdom	London
...

Figure 12.3: AG challenges.

Sample_Collector	Species	hucname	Exp_Plot	Exp_Plot_Position	N	K
David Eichenberg (University of Halle-Wittenberg)	C. eyrei	5040004100	B34	212	1.243	undetrmind
Wenzel Kroeber (University of Hamburg)	Ch. axillaris	Cumberland	B34	505	1.367	undetrmind
David Eichenberg (University of Halle-Wittenberg)	Acer davidii	Kentucky	B34	704	1.62	undetrmind
David Eichenberg (University of Halle-Wittenberg)	Li. formosana	Kentucky	B34	1009	1.456	31.826

Figure 12.4: BiodivTab challenges.

- (j) missing values including nulls, empty strings or special characters like (? , - , -) to the same effect.
- (k) tables of excessive length.

On the other hand, we give an overview of the encountered challenges in the biodiversity data tables in Figure 12.4 as follows:

1. *Nested Entities*: more than one proper entity in a single cell, e.g., a chemical compound is combined with a unit of measurement.
2. *Typos*: Data is predominantly collected manually by humans, so misspellings will occur, e.g., ‘Dead Leav’ is used for ‘Dead Leaves’.
3. *Acronyms*: Abbreviations of different sorts are common, e.g., ‘Canna glauca’, a particular kind of flower, is often referred to as ‘C.glauca’ or ‘Ca.glauce’.
4. *Synecdoche*: Scientists may use a general entity as a short form to a more particular one, e.g., ‘Kentucky’ is used instead of ‘Kentucky River’.
5. *Lack of Context*: The collected data may barely provide any informative context for semantic annotations. e.g., a column with a missing or severely misspelled header.
6. *Specimen Data*: The collected datasets contain observations of particular specimens or groups but do not pertain to the species as a whole.
7. *Numerical Data*: Most of the collected datasets describe the specimen by various measurements in numerical form.
8. *Missing Values*: Data collected can be sparse and may include gaps, e.g., a column ‘super kingdoms’ may consist of ‘unknown’ values for the most part.

Second, we evaluate our three key contributions regarding the Textual data interpretation (TextI). We explain how they meet the predefined requirements under this research area. First, BiodivOnto is our conceptual data model determining the most common terminology in the biodiversity domain. Second, BiodivBERT is the framework that extracts both concepts and relations based on the BiodivOnto from unstructured text. Finally, BiodivNERE (NER and RE) are the evaluation corpora we constructed and used to evaluate BiodivBERT.

BiodivOnto The conceptual model captures the most common and high-level concepts and relations. We constructed this schema using a data-driven approach with the aid of biodiversity experts. We decided to collect the relevant data from various biodiversity data sources that are well-established for ecological data. These data sources are Semedico to crawl abstracts and BEF-China, and data.world to collect tabular and metadata. We used a collection of typical biodiversity keywords to crawl these data. We manually extracted

the relevant terms of the biodiversity from the collected data. After applying cleaning and filtration steps, we developed a clustering-based technique to assign these keywords into groups that share the same semantic meaning. Thus, we converted the resultant keywords into embeddings-space. We applied further steps to construct the hierarchy from the seeds that resulted from the clustering technique. We relied on WordNet to identify the unique seeds which, in return, represent our core concepts. We involved biodiversity experts in revising the derived classes and asked them to label the relations among them manually. Thus, BiodivOnto reflects the commonly used vocabulary in the biodiversity domain. This contribution achieves the requirement R2.1.

BiodivBERT We developed a BERT-based model to extract entities and their relations from unstructured data. BiodivBERT is a biodiversity-specific language model that could be easily adapted for various applications. We pre-trained it on domain-specific data. That outcome fulfills the requirement R2.2. We fine-tuned BiodivBERT on two downstream tasks for NER and RE using gold-standard corpora that are annotated with classes and relations from the BiodivOnto. Since such an ontology captures the essential entities and relations of the biodiversity domain, the fine-tuned model on these corpora can detect those entities with their relations. Thus, BiodivBERT can be used to auto-populate the ontology. This contribution fulfills requirement R2.3. We released BiodivBERT in a way that is compatible with the HuggingFace library. This library is the most common one for dealing with transformers-based models. We show the usage of BiodivBERT in three different applications. First, ‘Masked language model’ as shown in Listing 12.2. It could be used for a fill-in mask task or domain-specific word embedding extraction. Second, ‘Token classification’ as shown in Listing 12.3. This variant loads BiodivBERT for word tagging or Named Entity Recognition (NER) task. Finally, ‘Sequence classification’ as shown in Listing 12.4. This way is meant for relation detection or RE as a binary classification task. In addition, we released the Jupyter Notebooks that we used for fine-tuning under our GitHub repository. The way we released the model is easy to use and achieves requirement R2.4. BiodivBERT achieved better results than the general domain models (baselines and BERT-based ones). BiodivBERT (+Abs+Full), which we pre-trained using both abstracts and full publications, achieved an average 0.86 F1-score on the Named Entity Recognition (NER) task. The same model gained an average 0.77 F1-score on the Relation Extraction (RE) task. Our model achieved first or second place compared to the state-of-the-art models on both tasks. Given the overall evaluation using the simple arithmetic weighting score, BiodivBERT is the best model for the two tasks. It outperforms `BERT_base_cased`, and `BioBERT` with 2.34%, and 1.34% F1-score, respectively. Such scores demonstrate the effectiveness of BiodivBERT in the biodiversity domain and complete requirement R2.5.

BiodivNERE We constructed two evaluation corpora for Named Entity Recognition (NER) and Relation Extraction (RE) to assess the effectiveness of BiodivBERT. We collected data from several abstracts and biodiversity data portals that have various characteristics. Thus, they are diverse and reflect various aspects of the biodiversity research field. This property completes requirement R2.6. We manually annotated both corpora and verified our annotations using various metrics. Thus, they are gold-standard benchmarks, and we ensure the high quality of the annotations. By these means, both fulfill requirement R2.7. We manually annotated both corpora using concepts and relations of the BiodivOnto schema model. This ensures the alignment with the developed model and the ontology and fulfills requirement R2.8. We released both corpora as a package. On the one hand, the BiodivNER corpus adopts the format BIO-tag that annotates each token in a given sentence. On the other hand, the BiodivRE corpus lists the sentence with a boolean value that determines whether there is a relation between the two entities

of that sentence. In addition, we split both corpora into train/dev/test splits that facilitate the training and testing of machine learning models. Thus, both corpora complete the requirement R2.9. We investigated the class/relation balance of the three data folds; train/dev/test. The evaluation corpora demonstrated a good balance for classes and the relation between data folds. For example, all classes that are used for training appear in other data folds, dev, and test. This achieved the last requirement in this area, R2.10.

Listing 12.2: BiodivBERT Masked Language Model (MLM)

```
> from transformers import AutoTokenizer,
    AutoModelForMaskedLM

> tokenizer = AutoTokenizer
    .from_pretrained("NoYo25/BiodivBERT")

> model = AutoModelForMaskedLM.
    from_pretrained("NoYo25/BiodivBERT")
```

Listing 12.3: BiodivBERT token classification

```
> from transformers import AutoTokenizer,
    AutoModelForTokenClassification

> tokenizer = AutoTokenizer.
    from_pretrained("NoYo25/BiodivBERT")

> model = AutoModelForTokenClassification.
    from_pretrained("NoYo25/BiodivBERT")
```

Listing 12.4: BiodivBERT relation extraction

```
> from transformers import AutoTokenizer,
    AutoModelForSequenceClassification

> tokenizer = AutoTokenizer.
    from_pretrained("NoYo25/BiodivBERT")

> model = AutoModelForSequenceClassification.
    from_pretrained("NoYo25/BiodivBERT")
```

Finally, we evaluate our key contribution regarding the Metadata interpretation (MI). We explain how it meets the predefined requirements under this research area.

Meta2KG In the first step of this work, we collected data from seven biodiversity data portals, e.g., iDiv, BEF-China, and BExIS. These heterogeneous data sources include a wide range of biodiversity metadata vocabulary. We manually aligned these diverse terminologies under a shared schema; Biodiversity Metadata Ontology (BMO) ontology. This underlying model reflects the most common vocabulary used in biodiversity repositories metadata. The BMO ontology completes the requirement R3.1. We developed an unsupervised learning method that transforms semi-structured data, e.g., XML or JSON, into KG. It is an embedding-based approach that relies on mean operations. This technique aims to capture the identified entities and relations of the BMO from the metadata files regardless of the original data source of this file. This feature fulfills the requirement R3.2. We provided a validation layer that ensures the correctness of matched data triple. For example, we validated both Phone, Email, Coordinate, URL, Decimal, and Date using regular expressions. If the validation has passed successfully, we auto-populate the triple. This step will avoid the population of empty triples or wrongly filled ones from the ori-

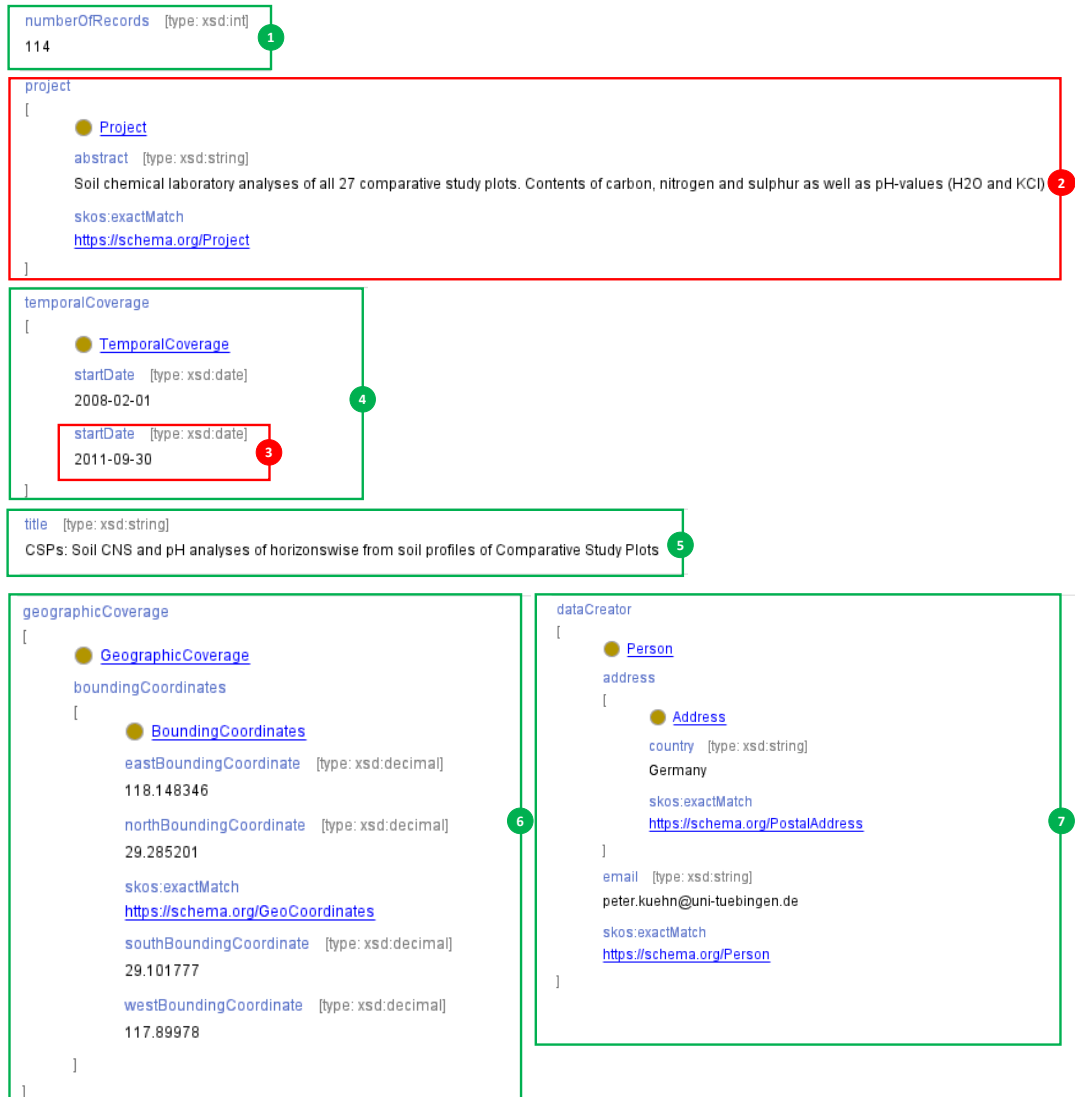


Figure 12.5: BMKG snippet analysis.

ginal metadata. Thus, it demonstrates a minimum quality level of the auto-population results and partially fulfills the requirement (R3.3). The output of the developed technique (match + populate) is a unified RDF file. This output represents Biodiversity Metadata Knowledge Graph (BMKG); i.e., the BMO with the auto-generated instances from metadata files. We released the BMKG in three formats: turtle (.ttl), n-triples, and RDF; thus, machine-readable format. That way, the output fulfills requirement R3.4.

The average F1-score that our embedding-based technique achieved given unseen ground truth data (metadata files) is 73%. Figure 12.5 shows a snippet of the automatically generated KG. We picked a random instance of the core concept ‘Dataset’ and investigated the auto-populated triples manually using Protégé with its original corresponding metadata file. In the figure, ‘numberOfRecords’, ‘temporalCoverage’, ‘title’, ‘geographicCoverage’, ‘dataCreator’ are correctly matched and populated (green rectangles 1, 4, 5, 6, 7). However, our technique mismatches the ‘Project’ and the ‘startDate’ under the ‘TemporalCoverage’ triple (red rectangles 2 and 3). From the original file, the former is just a ‘description’, and the latter should be ‘endDate’. In addition, we identified missing triples under the ‘dataCreator’, e.g., phone value. This means that our validation layer failed to validate a phone value. The previously mentioned score and the manual analysis demonstrate a promising technique to match and auto-populate an ontology using a

completely unsupervised technique which fulfills the requirement R3.5.

12.2 Retrospective; Limitations of the Solutions

In this section, we discuss the current limitations and shortcomings of each contribution.

JenTab We developed a modular approach to tackle the Semantic Table Interpretation (STI) tasks. It matches the individual components of tabular data into KGs. JenTab relies on the live lookup API and SPARQL query endpoint of the target KG, e.g., Wikidata. Thus, the performance of JenTab is bounded by these external dependencies, i.e., the maximum requests per minute. The provided datasets with their ground truth are created ahead at some point in time. However, JenTab always looks for annotations from the current and live version of the target KG. This could cause a loss of accuracy scores. Using the live APIs would be recommended in case of the deployment and live usage of JenTab, but for targeted datasets, an offline dump might be better.

CTA solutions are crucial. We investigated the effect of CTA selection on the scores of the other tasks. We developed three strategies that all rely on the direct types and their parents. Our findings showed that these direct parents, e.g., ‘P31’, would suit the datasets that are easier to solve and auto-generated. However, we found that integrating one more level to these parents ‘2Hops’ would suit these more challenging datasets. We do not recommend using higher levels of the parents like the case of ‘MultiHops’ since it produces very general inaccurate solutions. An additional limitation is that we assume that the semantic types are always derived via the direct parents or their higher level. However, in real-world scenarios, as shown in the BiodivTab, the most fine-grained type is given by another property, e.g., `wdt:P105` in the case of taxon-related columns. JenTab detected these columns as ‘taxon’, e.g., ‘`wd:Q16521`’, which is also a correct class but not the most fine-grained type.

BiodivTab We constructed a tabular data annotation benchmark based on real biodiversity research and data augmentation. We collected various biodiversity data tables from three data portals with multitudinous characteristics. Thus, BiodivTab represents a diverse collection of biodiversity tables. We manually annotated the collected data from Wikidata concerning two STI tasks; CEA and CTA. BiodivTab is relatively small compared to the existing state-of-the-art benchmarks. It consists of only 50 tables. The small size might be a limitation for BiodivTab to be used with large machine-learning models, but it has no issue with feature-based methods. In that sense, we need to increase its size by including more tables.

BiodivTab lacks the annotations for CPA task due to its ‘Specimen Data’ characteristic. Given the selected data portals, we could not provide CPA annotations. The reason is that most of the collected tables represent local data that belong to biodiversity experiments or laboratories. However, it might be possible to include more tables from diverse data sources that focus on ‘Species Data’ instead.

BiodivTab is annotated by a single annotator (the author of the thesis) and involves a biodiversity expert to review these annotations. The expert managed to check a round of 1/3 of the mappings. We used this revision round to estimate the error rate, which is 1%. However, this does not guarantee that BiodivTab is an error-free dataset. We believe that neither the existing benchmarks nor BiodivTab is free from errors. There is an upcoming trend to investigate the quality of the ground truth of STI benchmarks [100, 35]. Nevertheless, so far, BiodivTab did not receive any issues regarding its ground truth data during its participation in the SemTab challenge.

BiodivOnto We constructed a conceptual model for the biodiversity domain using a data-driven approach with the aid of biodiversity experts. We decided to collect the relevant data from various biodiversity data sources that are well-established for ecological data. We derived the concepts of BiodivOnto using a clustering-based approach followed by several manual steps. We evaluated the final outcome with biodiversity experts, which verified the concepts. However, we should support qualitative evaluation of the clustering contents and learned embeddings to develop a generic methodology. We did a rough investigation on the created clusters using visualizations and manual inspection; however, we needed to support quantitative results.

BiodivOnto is in its conceptual model view. It lacks the actual ontology file, e.g., the OWL file. To complete this work and to deliver a final and reusable core ontology, we should merge the resultant ontologies' modules into a single ontology. This step requires a manual check of the quality of the resultant ontology.

BiodivBERT We developed a BERT-based model that extracts entities and their relations from unstructured data. We pre-trained BiodivBERT on domain-specific data, a biodiversity-specific language model that could be easily adapted for various applications. We fine-tuned BiodivBERT on two downstream tasks for Named Entity Recognition (NER) and Relation Extraction (RE) using gold-standard corpora, BiodivNERE, as one package. Both are annotated with classes and relations from the BiodivOnto. Given the overall evaluation using the simple arithmetic weighting score, BiodivBERT is the best model for the two tasks. It outperforms `BERT_base_cased`, and `BioBERT` with 2.34%, and 1.34% F1-score, respectively. Such scores demonstrate the effectiveness of BiodivBERT in the biodiversity domain. BiodivBERT has mixed scores on the RE task. It did not achieve the best performance on all datasets regarding RE. Also, its average score in this task is lower than that on NER (0.77 vs. 0.86). The reason could be one or more of the following: 1) the selected benchmarks have low-quality annotations. 2) the selected benchmarks are fuzzy and way harder to solve. 3) fine-tuning settings need to be adapted. Thus, we need to profoundly investigate the reason behind the low scores in this task.

BiodivBERT is purely a language model that could be adapted to any application. We developed it to extract entities and relations from unstructured text. The ultimate goal of developing BiodivBERT is to create a KG from textual data. We fine-tuned BiodivBERT on BiodivNERE (NER and RE corpora), but the output is still not in triples format, e.g., RDF. BiodivBERT can be used for token and sequence classification; however, this output lacks semantics. For example, the detected entity should be mapped to one of the existing KGs, e.g., Wikidata. This step needs entity disambiguation methods. Those could be inspired by our developed techniques in JenTab, e.g., CEA generation module.

BiodivNERE We constructed two gold-standard evaluation corpora to assess the effectiveness of BiodivBERT. We collected data from various abstracts and metadata from diverse biodiversity data portals with various characteristics. Both corpora reflect different aspects of the biodiversity research field. We manually annotated both corpora using concepts and relations of the BiodivOnto schema model. That way both benchmarks are aligned with this ontology. From a sample inspection, we found that metadata could contain too long sentences with unuseful information like field names comma-separated. For example, a sentence could be as follows: 'dataset_id, tree_id, lab_id, pH, NO3'. These cases work for NER task; however, they are impossible to hold a meaning for the RE. Two possibilities to solve this issue, especially for RE are on the one hand, we manually investigate the constructed corpus for relations, BiodivRE, then remove these cases. On the other hand, in the Data Collection phase, we select long text fields from metadata only, e.g., 'description', 'abstract', and 'title'. The latter is a solution that could be done on

the level of the methodology itself and would require a new manual annotation. We conclude from this issue that abstracts or natural texts are more trusted sources for relations annotations.

Meta2KG We developed an unsupervised learning method that transforms semi-structured data, e.g., XML or JSON, into KG. We collected data from seven biodiversity data portals, e.g., iDiv, BEF-China, and BExIS. These heterogeneous data sources include a wide range of biodiversity metadata vocabulary. We manually aligned these diverse terminologies under a shared schema, Biodiversity Metadata Ontology (BMO) ontology, representing the domain’s most common vocabulary. We developed an ontology matching technique that is an embedding-based approach and relies on mean operations. The average F1-score that our embedding-based technique achieved given unseen ground truth data (metadata files) is 73%. The results of our experiments and the resultant BMKG analysis showed that metadata is a promising data source to create a KG in using a fully automatic approach. However, some repositories provide weak and incomplete metadata fields like PANGAEA. Such repositories introduce noise that we omit as much as possible to generate a clean KG. We report the obstacles we faced during the automatic ontology matching and population. On the one hand, we give an overview of the common obstacles with our provided solutions as follows: 1) A resultant triple might violate datatype constraints due to a mismatch by our approach or originally filed with a wrong datatype. We proposed validations that are based on regular expressions for several datatypes. E.g., a triple like (*dataCreator*, *phone*, *X*) is considered valid if and only if *X* is a valid *phone* value. 2) Inconsistent value format of metadata attributes. *Keywords* are used either in a word-by-word form or a list separated by a delimiter like commas and semicolons. We set the granularity to a word level for consistency; thus, we split any given list by its delimiter. Finally, 3) Embeddings failed to differentiate between values like *surName* and *givenName* since both are names. Thus, we consider the actual string value to obtain the correct match for such cases. We believe that a hybrid approach that uses both embeddings and string similarity would yield better results in general and not only limited to names.

On the other hand, we list the following limitations that are not solved yet. In our future work, we will consider the sketched solutions: 1) We discovered more inconsistencies regarding some metadata fields. Currently, *license* and *intellectualRights* properties accept a literal as a range. However, Dublin Core defines both of them where the expected range is an actual ‘license’, and ‘right statement’ objects, respectively. We plan to change that to follow the Dublin Core definitions where we support entity linking. 2) Currently, *citation* is a data property accepts a string as a range. We chose based on the options commonly used in the selected repositories. However, a typical citation contains more fine-grained data like authors, volume, and issue, which PANGAEA partially adopts. Thus, we consider a further analysis of the *citation* field by recognizing its individual parts. By this means, it would yield more fine-grained KG and better description. 3) Metadata fields might contain several (semi)redundant information across various fields, e.g., BEF-China might have these duplicates under *description*, *abstract*, *introduction*, *measurement*. A semi-automatic approach could overcome this issue. Finally, 4) We found complex fields that have multiple semantic concepts. For example, the *description* that is used in ‘data.world’ often contain information about *citation* or *license*. So, detecting those nested entities would yield more self-encapsulated information.

12.3 Summary

In this chapter, we evaluated our contributions. On the one hand, we discussed the degree to which we fulfilled the requirements and predefined objectives. We introduced both of

them in Chapter 2. We gave an overview of the overall performance, e.g., the accuracy of the developed frameworks and tools under each research area. We showed sample outputs for each of them. We explained the required execution steps for the developed frameworks. In addition, we discussed our crafted domain-specific benchmarks for each research area. We gave an overview of their characteristics. For example, we listed the unique challenges in BiodivTab compared to the commonly used general domain and Automatically Generated (AG) datasets. On the other hand, we discussed the shortcomings and demonstrated a retrospective of our contributions. Table 12.2 demonstrates the summary of our key contributions under each research area with their corresponding objectives and requirements.

Table 12.2: The summary of the evaluation of our research contributions.

Contribution	Objective	Requirements	Summary
Tabular data interpretation (TabI)			
JenTab	Objective 1	R1.1-1.7	The framework demonstrated effectiveness on general and biodiversity domain.
BiodivTab	Objective 2	R1.8-10	The first real-world benchmark for STI from the biodiversity domain.
Textual data interpretation (TexI)			
BiodivOnto	Objective 3	R2.1	A conceptual data model that capture the dominant concepts and relations in the biodiversity domain.
BiodivBERT	Objective 4	R2.2-R2.5	A BERT-based model that extract entities and relation from text given BiodivOnto as a data model.
BiodivNERE	Objective 5	R2.6-R2.10	Two corpora for NER and RE tasks based on biodiversity related abstracts and metadata.
Metadata interpretation (MI)			
BMO	Objective 6	R3.1	Ontology that aligns seven biodiversity portals metadata.
Meta2KG	Objective 7-8	R3.2,R3.3,R3.5	Unsupervised technique that transforms metadata file in a KG.
BMKG	Objective 7	R3.4	Unified biodiversity KG; RDF automatically generated from metadata.

Chapter 13

Conclusions and Future Work

The research problem we addressed in this dissertation is how we enhance the data re-usability as one of the FAIR principles. The motivation behind our work comes from the biodiversity experts and scientists who want to search for the data. Typically, they search for all datasets relevant to their research questions. This happens via searches in data repositories, literature, and personal connections. The publications found are then read to find essential references. Metadata about datasets are extracted from them. All this information is extracted manually, and this process can take several months. We divided this problem into three research questions: 1) How can we use tabular datasets for KG construction? (RQ1). 2) How can we benefit from the information in the associated publications to enrich the constructed KG? (RQ2). 3) How can we leverage the existing metadata to enrich the constructed KG? (RQ3). Each of these research questions is mapped to a research area.

We introduced seven key contributions to answer these questions. At first, in the scope of ‘Tabular data interpretation (TabI)’, we developed a framework, JenTab, that tackles the Semantic Table Interpretation (STI) tasks. In addition, we constructed a biodiversity-specific benchmark, BiodivTab, for evaluating Semantic Table Interpretation (STI) systems. We evaluated JenTab using BiodivTab and various existing benchmarks for general domain. Both JenTab and BiodivTab are developed and evaluated in the scope of ‘Semantic Web Challenge on Tabular Data to Knowledge Graph Matching (SemTab)’ challenge during 2020-2022. BiodivTab and JenTab won the first and second prizes, respectively, by IBM research during ISWC 2021. In addition, JenTab won the Artifacts Availability Badge, by SemTab community in ISWC 2022. JenTab is a top performer system that solves the STI tasks.

Second, in the sense of ‘Textual data interpretation (TexI)’, we constructed a conceptual model, BiodivOnto, representing the most important categories in the biodiversity domain. We developed a data-driven approach to reach these core concepts and integrated the biodiversity experts’ opinions to develop their relations. BiodivOnto consists of six classes and three relations. In addition, we developed a BERT-based model, BiodivBERT, that detects the named entities and their relations from unstructured text. We pre-trained it using domain-specific corpora from two well-known publishers using modern research data. We fine-tuned BiodivBERT on various state-of-the-art datasets for two downstream tasks Named Entity Recognition (NER) and Relation Extraction (RE). Moreover, we constructed two evaluation corpora for NER and RE. We released them under one package, BiodivNERE. We manually labeled the collected biodiversity-specific metadata and abstracts with classes and relations of the BiodivOnto.

Third, in the context of ‘Metadata interpretation (MI)’, we developed an unsupervised learning and embedding-based technique, Meta2KG, that transforms semi-structured data into a Knowledge Graph (KG). We manually crafted the Biodiversity Metadata Onto-

logy (BMO) ontology that aligns seven biodiversity data repositories using a shared vocabulary. In addition, we developed mean-based methods for BMO matching. Moreover, we investigated the auto-population of this ontology with triple validation techniques.

Table 13.1: The summary of the online materials of this dissertation.

Repository	Artifact	Link	Description
JenTab [9, 10, 11, 12]			
GitHub	Code	https://github.com/fusion-jena/JenTab	JenTab's code base (recent).
Zenodo	Precomputed Lookup	https://doi.org/10.5281/zenodo.7229246	The pre-computed lookup for handling the misspellings
GitHub	Precomputed Lookup	https://github.com/fusion-jena/JenTab_precomputed_lookup	Previous editions
Zenodo	JenTab Solutions	https://doi.org/10.5281/zenodo.7229243	System solutions for CEA, CTA, and CPA.
GitHub	JenTab Solutions	https://github.com/fusion-jena/JenTab_solution_files	Previous editions.
YouTube	Talk	https://youtu.be/sbBpFQoFqg	JenTab - SemTab @ ISWC 2020.
YouTube	Talk	https://youtu.be/btO9wYzWtqo	JenTab - KGC @ ESWC 2021.
YouTube	Talk	https://youtu.be/sbCicYsHuKM	JenTab - SemTab @ ISWC 2021.
BiodivTab [13, 14]			
GitHub	Code	https://github.com/fusion-jena/BiodivTab	Code base to construct the benchmark.
Zenodo	Wikidata Version	https://doi.org/10.5281/zenodo.6461556	Wikidata Version of BiodivTab (tables, targets, and ground truth).
Zenodo	DBpedia Version	https://doi.org/10.5281/zenodo.7319654	DBpedia Version of BiodivTab (tables, targets, and ground truth). (train/test).
YouTube	Talk	https://youtu.be/JY1kZai2F04	BiodivTab in OM @ ISWC 2022.
BiodivOnto [15, 16]			
GitHub	Code	https://github.com/fusion-jena/BiodivOnto	Code base to construct the BiodivOnto.
GitHub	Poster	https://fusion-jena.github.io/BiodivOnto/	BiodivOnto @ESWC 2021.
BiodivBERT [135]			
GitHub	Code	https://github.com/fusion-jena/BiodivBERT	Pre-training scripts and fine-tuning notebooks.
Zenodo	Pre-training DOIs	https://doi.org/10.5281/zenodo.6555690	Pre-training DOIs to construct +Abs and +Abs+Full corpora.
Zenodo	Pre-processed Datasets	https://doi.org/10.5281/zenodo.6554208	Datasets for fine-tuning (NER and RE).
Zenodo	Pre-trained Model	https://doi.org/10.5281/zenodo.6554141	Pretrained weights and model configuration.
BiodivNERE [17]			
GitHub	Code	https://github.com/fusion-jena/BiodivNERE	Code utils to construct NER and RE corpora.
Zenodo	Data	https://doi.org/10.5281/zenodo.6575865	Benchmarks files for train/dev/test
Meta2KG [18, 19]			
GitHub	Code	https://github.com/fusion-jena/Meta2KG	Code base to transform metadata into KGs.
Zenodo	BMO	https://doi.org/10.5281/zenodo.6948519	Biodiversity Metadata Ontology (BMO) files.
Zenodo	BMOE	https://doi.org/10.5281/zenodo.6951658	BMO embeddings JSON files.
Zenodo	BMKG	https://doi.org/10.5281/zenodo.6948573	Biodiversity Metadata Knowledge Graph (BMKG), 18 dataset instances.
Zenodo	Ground Truth	https://doi.org/10.5281/zenodo.6951623	Manually labeled metadata files.

13.1 Available Materials

Table 13.1 lists the publicly available artifacts, datasets, code, and pre-recorded talks for each contribution. We used GitHub to manage the code, whereas Zenodo hosted data-related contributions like datasets, embeddings, and ontologies. In addition, we added links for the recorded talks.

13.2 Future Directions

In the following, we list possible future directions for this thesis.

JenTab Extensions

- **Pre-computed lookup as a service** The pre-computed lookup is our primary source to tackle the misspellings. Due to its high resource requirements, we build this lookup before the actual start of JenTab’s pipeline. It would be essential to convert this step into a live service that we integrate within the pipeline execution. This would facilitate deploying and running JenTab faster and decreases its dependencies.
- **JenTab as a service** Currently, we support two executions mode to run JenTab, either by local setup or via docker containers. However, deploying JenTab as a public service would increase its audience and users. It would have a broader scope of community to benefit from it.
- **RDF as an output** JenTab supports the required format for each STI by the SemTab challenge. For example, for CEA task, JenTab produces a CSV file that contains: ‘filename’, ‘row_id’, ‘col_id’, and the annotation itself (see Chapter 12 for all tasks’ output). However, for generating a KG over tabular data, RDF would be a better format to use than the current CSV file.
- **JenTab interactive UI** Currently, we support the visibility of the current processing dataset in the ‘Manager’ or the central node of JenTab. For example, we display the successfully annotated tables, statistics of the results, and so on. However, we pre-configured JenTab on the input tables and targets. An interactive UI should support, e.g., uploading a table and its target or a batch of them. In addition, it should give the option to select the target KG to annotate a table. Since we provide multiple pipelines to solve the STI tasks, the UI should provide a list of all supported pipelines to enable users to select one of them.
- **Annotate as much as possible** JenTab relies on the given targets to annotate the table counterparts from KGs. It would make sense to guess the targets in a real-life scenario rather than ask the user for them.
- **Scoring system** The most powerful pipeline we developed in ‘JenTab’ is the `pipeline_full` that implements all modules from the Create, Filter and Select (CFS) pattern. However, the filter functions might be too harsh on the candidates. I.e., drops correct mappings due to high support threshold. Switching to a scoring system that preserves all candidates and keeps them until a final selection would even enhance the scores of JenTab.

BiodivTab Extensions

- **large-scale benchmark** BiodivTab contains 50 tables. It is a relatively small benchmark compared to the existing datasets. We should include more biodiversity tables from other projects to cover a broader domain spectrum and increase the benchmark size.
- **CEA task** Due to the specimen data issue we encountered during the ‘Data Collection’ phase, we were not able to provide annotations for entities, i.e., CEA annotations for table cells. We should include more tables from other data sources to check if this issue still exists.
- **Domain-specific ground truth** Currently, we support Wikidata and DBpedia as target KGs. To diversify the target and be closer to the domain, we should provide ground truth data from other KGs, particularly domain-specific ones, e.g., biodiversity-specific ontologies.
- **Multi-rater agreement** Currently, we depend on the partial revision of the provided annotations by a domain expert. We should enable a double annotation process and calculate the inter-rater agreements for additional quality insurance.

BiodivOnto Extensions

- **More domain experts** During the development of the initial version of BiodivOnto, we relied on one domain expert. However, in the later stage of development and after the involvement of two more experts, they had different influences on the developed model. Thus, for more reliable model development, we should include more biodiversity experts and follow the majority vote of them. It requires a setup of fixed survey questions to enable such voting mechanism. Our Questionnaire in BiodivTab (see Chapter 7) inspires this step.
- **Quantitative evaluation** To determine the core concepts of the BiodivOnto we relied on an unsupervised clustering-based technique in the embedding space. We verified the final outcome by domain experts, ensuring the method’s correctness. However, to convert the mechanism into a fully automated technique we should support a qualitative evaluation of the constructed embeddings. Standard techniques that are developed to ensure the quality of the embeddings are in [128, 129].

BiodivBERT Extensions

- **Lightweight model** Currently, we rely on the BERT_base_cased. We plan to pre-train and fine-tune on a lightweight model, e.g., distilBERT [153].
- **Robust variant** Similar to the above point, we consider pre-training and fine-tuning a more robust model e.g., RoBERTa [154].
- **Enhance RE low scores** We also plan to investigate the reasons behind the low scores on the RE task, especially with the BiodivRE and BioRelEx datasets. For example, we could try different settings for fine-tuning.
- **Biodiversity-specific RE datasets** Due to the lack of the available biodiversity-specific datasets for the RE task, we used benchmarks from the biomedical domain. For specificity, we plan to fine-tune BiodivBERT on more biodiversity-specific datasets whenever they are available.

- **RDF as an output** We currently support the native output by any transformers-based model for the two downstream tasks (NER and RE). To enable KG creation from unstructured text, we should transform the triple components: named entities and their relations we detect from text into an RDF.
- **BiodivBERT as a service** Currently, we wrap the usage of BiodivBERT through the HuggingFace library. Ideally, it costs three lines of code to load the model in the local space. For more expansive usage of BiodivBERT by non-tech users, e.g., biodiversity experts, we plan to deploy BiodivBERT for actual token and sequence prediction for the biodiversity literature.
- **BiodivOnto Auto-population** Since BiodivBERT is fine-tuned on corpora that contain annotations from BiodivOnto. We plan to deploy the model to construct the KG on top of it. In this scope, we need to develop an entity disambiguation mechanism to fill the ontology with instances. We can rely on the CEA disambiguation process we developed in JenTab to achieve this task.

BiodivNERE Extensions

- **Diverse classes and relations** Currently, we support six classes and three relations from the current version of the BiodivOnto. To cover a broader range of the domain, we should include annotations by more classes and relations. For example, restore the dropped relations from BiodivOnto, e.g., ‘part_of’ and ‘is_a’.
- **Fine-grained annotations** We facilitated the annotation process by including only the top-level classes in the BiodivOnto. For example, we merged both ‘Ecosystem’ and ‘Landscape’ and used their parents ‘Environment’ for annotating. To enable fine-grained classification models, we should support fine-grained annotations in correspondence. These fine-grained annotations will support fine-grained KG as a result.
- **Large-scale benchmarks** Additionally, we should include more data sources to cover a broader range of the domain.

Meta2KG Extensions

- **Enhance the scores** We plan to enhance our matching technique by using an ensemble-based method that relies on both embeddings and string similarity.
- **Fine-grained representation** We parse complex fields into more fine-grained pieces for better representation. For example, the ‘Citation’ field contains multiple entities, e.g., author, institution, volume, and so on. Thus, detecting those entities would yield a more-fine grained KG and enable more sophisticated applications like Question Answering (QA).
- **Sophisticated triple verification** Currently, we support a validation step based on the associated value of the given class. We support primitive data validation that validates phone, email, coordinates, and URL using regular expressions. However, we found miss-classified triples and false negative examples in the resultant KG. We need to investigate more sophisticated techniques for validation purposes.

The Integrated Tool

- **Unified framework** To enable KG construction on top of the three heterogeneous data sources (tabular, textual data, and metadata). We should integrate all the developed techniques (JenTab, BiodivBERT, and Meta2KG) into a framework, such that it receives one or more inputs from the supported sources and outputs a unified KG.
- **Triplestore** We should also consider the persistent storage and the exposure of the resultant KG via the SPARQL endpoint. A strong candidate to consider is Blazegraph¹, which handles large-scale KGs efficiently, e.g., Wikidata Query service. By this means, we enhance the data re-usability.

¹<https://blazegraph.com/>

List of Figures

1.1	Entities of interest per data source.	4
2.1	Abstract view of our research methodology	17
3.1	A triple representation.	20
3.2	Inner-relationship examples.	20
3.3	A summary of Semantic Table Interpretation (STI) tasks.	22
3.4	SemTab tasks summary.	23
3.5	A NER example.	23
3.6	An RE example.	24
4.1	Related work of the three subareas overview.	38
5.1	Schema of contribution pillar.	39
5.2	Contributions' map.	41
6.1	Illustration of Semantic Table Interpretation (STI) tasks.	46
6.2	JenTab: Current system architecture	47
6.3	Disambiguation contexts overview	49
6.4	CFS Pattern	51
6.5	Example for CTA selection by LCS	54
6.6	pipeline_full as an arrangement of CFS building blocks.	55
6.7	Confusion matrix for type prediction	57
6.8	Audit statistics for CEA creation	59
6.9	Audit statistics for CEA creation (BiodivTab)	59
6.10	Audit statistics for CEA selection.	60
6.11	Audit statistics for CEA selection (BiodivTab).	60
6.12	Audit statistics for CTA selection.	61
6.13	Audit statistics for CTA selection (BiodivTab).	61
6.14	JenTab F1-scores Automatically Generated (AG) datasets [2020-2022].	65
6.15	JenTab F1-scores Tough Tables (2T) datasets [2020-2022].	65
6.16	Runners and runtime of JenTab [2020-2022].	66
7.1	Steps of BiodivTab construction	70
7.2	Domain distribution in BiodivTab benchmark	75
8.1	Proposed four-phase pipeline.	83
8.2	Crawling phase.	84
8.3	Extracted keywords vs. existing projects	85
8.4	Filtration effect on the selected data sources	85
8.5	Simple vs. compound keywords	86
8.6	A sample of seeds WordNet similarity	86
8.7	'Quality' cluster in the final iteration	87

8.8	Core concepts and their relations	88
8.9	Occurrence frequency of relations in questions related to biodiversity research	90
8.10	Final version of BiodivOnto	91
9.1	BiodivBERT pre-training and fine-tuning Overview	94
9.2	Keywords results statistics (+Abs Corpus)	96
9.3	Open access keywords results statistics (+Abs+Full Corpus)	97
9.4	Data loss during full Text corpus construction	97
9.5	System scores of the selected models.	101
10.1	Our proposed NER corpus construction pipeline.	105
10.2	NER annotation process	106
10.3	Team A: Agreement scores	108
10.4	Team B: Agreement scores	108
10.5	RE sentence variations creation.	109
10.6	A snippet of an RE sheet during annotation	111
10.7	Category distribution of BiodivNER corpus	111
10.8	Category pairs distribution.	113
10.9	The binary distribution of the BiodivRE corpus	113
10.10	Multi-class relations distribution of BiodivRE corpus	114
11.1	Meta2KG workflow.	120
11.2	Seen & Unseen metadata distribution.	121
11.3	Biodiversity Metadata Ontology (BMO) concepts and relations.	123
11.4	Matching F1-score on Unseen data using our embeddings.	126
11.5	Matching F1-score on Unseen data using Wiki-based embeddings.	126
11.6	Biodiversity Metadata Knowledge Graph (BMKG) triples frequencies.	128
12.1	JenTab: Output snippet.	134
12.2	JenTab: Manager screenshot.	134
12.3	AG challenges.	135
12.4	BiodivTab challenges.	136
12.5	BMKG snippet analysis.	139
A.1	JenTab Usability Track Certificate ISWC 2021.	179
A.2	BiodivTab Applications Track Certificate ISWC 2021.	179

List of Tables

4.1	Summary of the current state-of-the-art STI approaches.	26
4.2	Existing STI benchmarks: Method, source, and target KG.	31
4.3	Summary of existing STI benchmarks.	32
6.1	Generic Strategy: Unique labels and ratio of resolved labels per round.	58
6.2	SemTab 2020 top participants' scores (AG-datasets)	62
6.3	JenTab & SOTA scores (HardTables 2021-2022)	63
6.4	JenTab and existing systems scores (2T dataset)	63
6.5	JenTab & SOTA scores (BioTables 2021)	64
6.6	JenTab & SOTA scores (BiodivTab 2021-2022)	64
6.7	Execution time for different setups (SemTab2020 benchmark).	66
6.8	Execution time for different setups (SemTab 2020-2022 benchmarks).	66
7.1	Prevalence of challenges among the selected datasets	71
7.2	Questionnaire: Which type would be correct for the given taxons?	72
7.3	Data augmentation technique per dataset	73
7.4	Original and selected tables sizes, and entity and type mappings	75
7.5	Most and least frequent semantic types in BiodivTab	75
7.6	Data sources for existing benchmarks and their corresponding targets	76
7.7	Comparison with existing benchmarks.	77
8.1	Core concepts in existing ontologies	88
8.2	Summary of the categories used in NER annotation.	89
9.1	Final Pre-training Corpora Statistics	96
9.2	Pre-training Models setting corpora	98
9.3	Overview of the selected NER datasets	98
9.4	Overview of the selected RE datasets	99
9.5	Fill-in mask task results by BERT-based models	99
9.6	Fine-tuning scores on NER datasets	100
9.7	Fine-tuning scores on RE datasets	100
10.1	NER benchmarks data sources comparison.	112
10.2	NER benchmarks statistics comparison.	112
10.3	RE corpora comparison	114
11.1	Auto-generated keys, frequencies, and the selected key name.	122
11.2	Properties of the <code>Dataset</code> main concept	124
12.1	The summary of this thesis requirements.	132
12.2	The summary of the evaluation of our research contributions.	143
13.1	The summary of the online materials of this dissertation.	146

Listings

11.1 Metadata field XML snippet	121
12.1 JenTab docker commands	133
12.2 BiodivBERT Masked Language Model (MLM)	138
12.3 BiodivBERT token classification	138
12.4 BiodivBERT relation extraction	138

Bibliography

- [1] Sören Auer, Viktor Kovtun, Manuel Prinz, Anna Kasprzik, Markus Stocker, and Maria-Esther Vidal.
Towards a Knowledge Graph for Science.
In *Proceedings of the 8th International Conference on Web Intelligence, Mining and Semantics, WIMS 2018, Novi Sad, Serbia, June 25-27, 2018*, pages 1:1–1:6. ACM, 2018.
doi:10.1145/3227609.3227689.
- [2] Mark D Wilkinson, Michel Dumontier, IJsbrand Jan Aalbersberg, Gabrielle Appleton, Myles Axton, Arie Baak, Niklas Blomberg, Jan-Willem Boiten, Luiz Bonino da Silva Santos, Philip E Bourne, et al.
The fair guiding principles for scientific data management and stewardship.
Scientific data, 3, 2016.
doi:10.1038/sdata.2016.18.
- [3] Luiz M. R. Gadelha et al.
A survey of biodiversity informatics: Concepts, practices, and challenges.
Wiley Interdiscip. Rev. Data Min. Knowl. Discov., 11(1), 2021.
doi:10.1002/widm.1394.
- [4] Brend Wanders.
Repurposing and probabilistic integration of data.
SIKS dissertation series, Universiteit Twente, Jun 2016.
isbn:978-90-365-4110-7, number:2016-24.
URL: <https://doi.org/10.3990/1.9789036541107>.
- [5] Roderic Page.
Towards a biodiversity knowledge graph.
Research Ideas and Outcomes, 2:e8767, 2016.
URL: <https://doi.org/10.3897/rio.2.e8767>.
- [6] Riza Batista-Navarro, Chrysoula Zerva, and Sophia Ananiadou.
Construction of a Biodiversity Knowledge Repository using a Text Mining-based Framework.
In *Proceedings of the 3rd Annual International Symposium on Information Management and Big Data - SIMBig 2016, Cusco, Peru, September 1-3, 2016*, volume 1743 of *CEUR Workshop Proceedings*, pages 22–25. CEUR-WS.org, 2016.
URL: <http://ceur-ws.org/Vol-1743/paper1.pdf>.
- [7] Helen RP Phillips, Elizabeth M Bach, Marie LC Bartz, Joanne M Bennett, Rémy Beugnon, Maria JI Briones, George G Brown, Olga Ferlian, Konstantin B Gongalsky, Carlos A Guerra, et al.
Global data on earthworm abundance, biomass, diversity and corresponding environmental properties.
Scientific Data, 8(1):1–12, 2021.
Nature Publishing Group.

- URL: <https://www.nature.com/articles/s41597-021-00912-z>.
- [8] Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang.
BioBERT: a pre-trained biomedical language representation model for biomedical text mining.
Bioinform., 36(4):1234–1240, 2020.
doi:10.1093/bioinformatics/btz682.
- [9] Nora Abdelmageed and Sirko Schindler.
Jentab: Matching tabular data to knowledge graphs.
In *Proceedings of the Semantic Web Challenge on Tabular Data to Knowledge Graph Matching (SemTab 2020) co-located with the 19th International Semantic Web Conference (ISWC 2020), Virtual conference (originally planned to be in Athens, Greece), November 5, 2020*, volume 2775 of *CEUR Workshop Proceedings*, pages 40–49. CEUR-WS.org, 2020.
URL: <http://ceur-ws.org/Vol-2775/paper4.pdf>.
- [10] Nora Abdelmageed and Sirko Schindler.
JenTab Meets SemTab 2021’s New Challenges.
In *Proceedings of the Semantic Web Challenge on Tabular Data to Knowledge Graph Matching co-located with the 20th International Semantic Web Conference (ISWC 2021), Virtual conference, October 27, 2021*, volume 3103 of *CEUR Workshop Proceedings*, pages 42–53. CEUR-WS.org, 2021.
URL: <http://ceur-ws.org/Vol-3103/paper4.pdf>.
- [11] Nora Abdelmageed and Sirko Schindler.
Jentab: A toolkit for semantic table annotations.
In *Proceedings of the 2nd International Workshop on Knowledge Graph Construction co-located with 18th Extended Semantic Web Conference (ESWC 2021), Online, June 6, 2021*, volume 2873 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2021.
URL: <http://ceur-ws.org/Vol-2873/paper5.pdf>.
- [12] Nora Abdelmageed and Sirko Schindler.
JenTab: Do CTA solutions affect the entire scores? .
In *Proceedings of the Semantic Web Challenge on Tabular Data to Knowledge Graph Matching co-located with the 21th International Semantic Web Conference (ISWC 2022), Virtual conference, October 24, 2022*, volume 3320 of *CEUR Workshop Proceedings*, pages 72–79. CEUR-WS.org, 2022.
URL: <https://ceur-ws.org/Vol-3320/paper8.pdf>.
- [13] Nora Abdelmageed, Sirko Schindler, and Birgitta König-Ries.
Biodivtab: A table annotation benchmark based on biodiversity research data.
In *Proceedings of the Semantic Web Challenge on Tabular Data to Knowledge Graph Matching co-located with the 20th International Semantic Web Conference (ISWC 2021), Virtual conference, October 27, 2021*, volume 3103 of *CEUR Workshop Proceedings*, pages 13–18. CEUR-WS.org, 2021.
URL: <http://ceur-ws.org/Vol-3103/paper1.pdf>.
- [14] Nora Abdelmageed, Sirko Schindler, and Birgitta König-Ries.
BiodivTab: Semantic table annotation benchmark construction, analysis, and new additions.
In *Proceedings of the 17th International Workshop on Ontology Matching co-located with the 21st International Semantic Web Conference (ISWC 2022)*, CEUR Workshop Proceedings. CEUR-WS.org, 2022.

- [15] Nora Abdelmageed, Alsayed Algergawy, Sheeba Samuel, and Birgitta König-Ries. BiodivOnto: Towards a Core Ontology for Biodiversity. In *The Semantic Web: ESWC 2021 Satellite Events*, volume 12739, pages 3–8. Springer, 2021.
doi:10.1007/978-3-030-80418-3_1.
- [16] Nora Abdelmageed, Alsayed Algergawy, Sheeba Samuel, and Birgitta König-Ries. A data-driven approach for core biodiversity ontology development. In *Proceedings of the Joint Ontology Workshops 2021 Episode VII: The Bolzano Summer of Knowledge co-located with the 12th International Conference on Formal Ontology in Information Systems (FOIS 2021), and the 12th International Conference on Biomedical Ontologies (ICBO 2021), Bolzano, Italy, September 11-18, 2021*, volume 2969 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2021.
URL: <http://ceur-ws.org/Vol-2969/paper5-s4biodiv.pdf>.
- [17] Nora Abdelmageed, Felicitas Löffler, Leila Feddoul, Alsayed Algergawy, Sheeba Samuel, Jitendra Gaikwad, Anahita Kazem, and Birgitta König-Ries. Biodivnere: Gold standard corpora for named entity recognition and relation extraction in the biodiversity domain. *Biodiversity Data Journal*, 10:e89481, 2022.
doi:10.3897/BDJ.10.e89481.
- [18] Nora Abdelmageed and Birgitta König-Ries. Meta2KG: Transforming metadata to knowledge graphs. In *Proceedings of the 17th International Workshop on Ontology Matching co-located with the 21st International Semantic Web Conference (ISWC 2022)*., volume 3324 of *CEUR Workshop Proceedings*, pages 226–228. CEUR-WS.org, 2022.
URL: https://ceur-ws.org/Vol-3324/om2022_poster3.pdf.
- [19] Nora Abdelmageed and Birgitta König-Ries. Meta2KG: An embeddings-based approach for transforming metadata to knowledge graphs. In *Proceedings of the Fourth International Workshop on Knowledge Graph Construction co-located with the 20th Extended Semantic Web Conference (ESWC 2023)*, CEUR Workshop Proceedings. CEUR-WS.org, 2023.
- [20] Aidan Hogan, Eva Blomqvist, Michael Cochez, Claudia d’Amato, Gerard de Melo, Claudio Gutierrez, 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. *ACM Comput. Surv.*, 54(4):71:1–71:37, 2022.
<https://doi.org/10.1145/3447772>.
- [21] 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*. Number 22 in Synthesis Lectures on Data, Semantics, and Knowledge. Morgan & Claypool, 2021.
URL: <https://kgbook.org/>, doi:10.2200/S01125ED1V01Y202109DSK022.
- [22] Ernesto Jiménez-Ruiz, Oktie Hassanzadeh, Vasilis Efthymiou, Jiaoyan Chen, and Kavitha Srinivas.

- SemTab 2019: Resources to benchmark tabular data to knowledge graph matching systems.
In *The Semantic Web - 17th International Conference, ESWC 2020, Heraklion, Crete, Greece, May 31-June 4, 2020, Proceedings*, pages 514–530. Springer International Publishing, 2020.
URL: https://doi.org/10.1007/978-3-030-49461-2_30.
- [23] Ernesto Jiménez-Ruiz, Otkie Hassanzadeh, Vasilis Efthymiou, Jiaoyan Chen, Kavitha Srinivas, and Vincenzo Cutrona.
Results of semtab 2020.
In *Proceedings of the Semantic Web Challenge on Tabular Data to Knowledge Graph Matching (SemTab 2020) co-located with the 19th International Semantic Web Conference (ISWC 2020), Virtual conference (originally planned to be in Athens, Greece), November 5, 2020*, volume 2775 of *CEUR Workshop Proceedings*, pages 1–8. CEUR-WS.org, 2020.
URL: <http://ceur-ws.org/Vol-2775/paper0.pdf>.
- [24] Vincenzo Cutrona, Jiaoyan Chen, Vasilis Efthymiou, Otkie Hassanzadeh, Ernesto Jiménez-Ruiz, Juan Sequeda, Kavitha Srinivas, Nora Abdelmageed, Madelon Hulsebos, Daniela Oliveira, and Catia Pesquita.
Results of SemTab 2021.
In *Proceedings of the Semantic Web Challenge on Tabular Data to Knowledge Graph Matching co-located with the 20th International Semantic Web Conference (ISWC 2021), Virtual conference, October 27, 2021*, volume 3103 of *CEUR Workshop Proceedings*, pages 1–12. CEUR-WS.org, 2021.
URL: <http://ceur-ws.org/Vol-3103/paper0.pdf>.
- [25] Yalin Wang and Jianying Hu.
Detecting tables in HTML documents.
In *Document Analysis Systems V, 5th International Workshop, DAS 2002, Princeton, NJ, USA, August 19-21, 2002, Proceedings*, volume 2423 of *Lecture Notes in Computer Science*, pages 249–260. Springer, 2002.
URL: https://doi.org/10.1007/3-540-45869-7_29.
- [26] Gerald Penn, Jianying Hu, Hengbin Luo, and Ryan T. McDonald.
Flexible Web Document Analysis for Delivery to Narrow-Bandwidth Devices.
In *6th International Conference on Document Analysis and Recognition (ICDAR 2001), 10-13 September 2001, Seattle, WA, USA*, pages 1074–1078. IEEE Computer Society, 2001.
URL: <https://doi.org/10.1109/ICDAR.2001.953951>.
- [27] Jixiong Liu, Yoan Chabot, Raphaël Troncy, Viet-Phi Huynh, Thomas Labbé, and Pierre Monnin.
From tabular data to knowledge graphs: A survey of semantic table interpretation tasks and methods.
Journal of Web Semantics, page 100761, 2022.
URL: <https://doi.org/10.1016/j.websem.2022.100761>.
- [28] Felicitas Löffler, Valentin Wesp, Birgitta König-Ries, and Friederike Klan.
Dataset search in biodiversity research: Do metadata in data repositories reflect scholarly information needs?
PloS one, 16(3):e0246099, 2021.
doi:10.1371/journal.pone.0246099.
- [29] Felicitas Löffler, Nora Abdelmageed, Samira Babalou, Pawandeep Kaur, and Birgitta König-Ries.

- Tag Me If You Can! Semantic Annotation of Biodiversity Metadata with the QEMP Corpus and the BiodivTagger.
In *Proceedings of The 12th Language Resources and Evaluation Conference, LREC 2020, Marseille, France, May 11-16, 2020*, pages 4557–4564. European Language Resources Association, 2020.
URL: <https://aclanthology.org/2020.lrec-1.560/>.
- [30] Jing Li, Aixin Sun, Jianglei Han, and Chenliang Li.
A Survey on Deep Learning for Named Entity Recognition.
IEEE Trans. Knowl. Data Eng., 34(1):50–70, 2022.
doi:10.1109/TKDE.2020.2981314.
- [31] David Nadeau and Satoshi Sekine.
A survey of named entity recognition and classification.
Linguisticae Investigationes, 30(1):3–26, 2007.
- [32] Xavier Schmitt, Sylvain Kubler, Jérémy Robert, Mike Papadakis, and Yves Le Traon.
A replicable comparison study of NER software: Stanfordnlp, nltk, opennlp, spacy, gate.
In *Sixth International Conference on Social Networks Analysis, Management and Security, SNAMS 2019, Granada, Spain, October 22-25, 2019*, pages 338–343. IEEE, 2019.
doi:10.1109/SNAMS.2019.8931850.
- [33] Alisa Smirnova and Philippe Cudré-Mauroux.
Relation extraction using distant supervision: A survey.
ACM Comput. Surv., 51(5):106:1–106:35, 2019.
doi:10.1145/3241741.
- [34] Yong Shi, Yang Xiao, and Lingfeng Niu.
A brief survey of relation extraction based on distant supervision.
In *Computational Science - ICCS 2019 - 19th International Conference, Faro, Portugal, June 12-14, 2019, Proceedings, Part III*, volume 11538 of *Lecture Notes in Computer Science*, pages 293–303. Springer, 2019.
doi:10.1007/978-3-030-22744-9_23.
- [35] Nora Abdelmageed, Jiaoyan Chen, Vincenzo Cutrona, Vasilis Eftthymiou, Oktie Hassanzadeh, Madelon Hulsebos, Ernesto Jiménez-Ruiz, Juan Sequeda, and Kavitha Srinivas.
Results of SemTab 2022.
In *Proceedings of the Semantic Web Challenge on Tabular Data to Knowledge Graph Matching co-located with the 21st International Semantic Web Conference (ISWC 2022)*., volume 3320 of *CEUR Workshop Proceedings*, pages 1–13. CEUR-WS.org, 2022.
URL: <http://ceur-ws.org/Vol-3320/paper0.pdf>.
- [36] Phuc Nguyen, Natthawut Kertkeidkachorn, Ryutaro Ichise, and Hideaki Takeda.
Mtab: Matching tabular data to knowledge graph using probability models.
In *Proceedings of the Semantic Web Challenge on Tabular Data to Knowledge Graph Matching co-located with the 18th International Semantic Web Conference, SemTab@ISWC 2019, Auckland, New Zealand, October 30, 2019*, volume 2553 of *CEUR Workshop Proceedings*, pages 7–14. CEUR-WS.org, 2019.
URL: <http://ceur-ws.org/Vol-2553/paper2.pdf>.
- [37] Phuc Nguyen, Ikuya Yamada, Natthawut Kertkeidkachorn, Ryutaro Ichise, and Hideaki Takeda.
MTab4Wikidata at SemTab 2020: Tabular Data Annotation with Wikidata.

- In *Proceedings of the Semantic Web Challenge on Tabular Data to Knowledge Graph Matching (SemTab 2020) co-located with the 19th International Semantic Web Conference (ISWC 2020)*, Virtual conference (originally planned to be in Athens, Greece), November 5, 2020, volume 2775 of *CEUR Workshop Proceedings*, pages 86–95. CEUR-WS.org, 2020.
URL: <http://ceur-ws.org/Vol-2775/paper9.pdf>.
- [38] Phuc Nguyen, Ikuya Yamada, Natthawut Kertkeidkachorn, Ryutaro Ichise, and Hideaki Takeda.
SemTab 2021: Tabular Data Annotation with MTab Tool.
In Ernesto Jiménez-Ruiz, Vasilis Efthymiou, Jiaoyan Chen, Vincenzo Cutrona, Oktie Hassanzadeh, Juan Sequeda, Kavitha Srinivas, Nora Abdelmageed, Madelon Hulsebos, Daniela Oliveira, and Catia Pesquita, editors, *Proceedings of the Semantic Web Challenge on Tabular Data to Knowledge Graph Matching co-located with the 20th International Semantic Web Conference (ISWC 2021)*, Virtual conference, October 27, 2021, volume 3103 of *CEUR Workshop Proceedings*, pages 92–101. CEUR-WS.org, 2021.
URL: <http://ceur-ws.org/Vol-3103/paper8.pdf>.
- [39] Viet-Phi Huynh, Jixiong Liu, Yoan Chabot, Thomas Labbé, Pierre Monnin, and Raphaël Troncy.
DAGOBAN: Enhanced Scoring Algorithms for Scalable Annotations of Tabular Data.
In Ernesto Jiménez-Ruiz, Oktie Hassanzadeh, Vasilis Efthymiou, Jiaoyan Chen, Kavitha Srinivas, and Vincenzo Cutrona, editors, *Proceedings of the Semantic Web Challenge on Tabular Data to Knowledge Graph Matching (SemTab 2020) co-located with the 19th International Semantic Web Conference (ISWC 2020)*, Virtual conference (originally planned to be in Athens, Greece), November 5, 2020, volume 2775 of *CEUR Workshop Proceedings*, pages 27–39. CEUR-WS.org, 2020.
URL: <http://ceur-ws.org/Vol-2775/paper3.pdf>.
- [40] Viet-Phi Huynh, Jixiong Liu, Yoan Chabot, Frédéric Deuzé, Thomas Labbé, Pierre Monnin, and Raphaël Troncy.
DAGOBAN: table and graph contexts for efficient semantic annotation of tabular data.
In *Proceedings of the Semantic Web Challenge on Tabular Data to Knowledge Graph Matching co-located with the 20th International Semantic Web Conference (ISWC 2021)*, Virtual conference, October 27, 2021, volume 3103 of *CEUR Workshop Proceedings*, pages 19–31. CEUR-WS.org, 2021.
URL: <http://ceur-ws.org/Vol-3103/paper2.pdf>.
- [41] Renat Shigapov, Philipp Zumstein, Jan Kamlah, Lars Oberländer, Jörg Mechnich, and Irene Schumm.
bbw: Matching CSV to Wikidata via Meta-lookup.
In *Proceedings of the Semantic Web Challenge on Tabular Data to Knowledge Graph Matching (SemTab 2020) co-located with the 19th International Semantic Web Conference (ISWC 2020)*, Virtual conference (originally planned to be in Athens, Greece), November 5, 2020, volume 2775 of *CEUR Workshop Proceedings*, pages 17–26. CEUR-WS.org, 2020.
URL: <http://ceur-ws.org/Vol-2775/paper2.pdf>.
- [42] Gilles Vandewiele, Bram Steenwinkel, Filip De Turck, and Femke Ongenaë.
CVS2KG: transforming tabular data into semantic knowledge.

- In *Proceedings of the Semantic Web Challenge on Tabular Data to Knowledge Graph Matching co-located with the 18th International Semantic Web Conference, SemTab@ISWC 2019, Auckland, New Zealand, October 30, 2019*, volume 2553 of *CEUR Workshop Proceedings*, pages 33–40. CEUR-WS.org, 2019.
URL: <http://ceur-ws.org/Vol-2553/paper5.pdf>.
- [43] Avijit Thawani, Minda Hu, Erdong Hu, Husain Zafar, Naren Teja Divvala, Aman-deep Singh, Ehsan Qasemi, Pedro A. Szekely, and Jay Pujara.
Entity linking to knowledge graphs to infer column types and properties.
In *Proceedings of the Semantic Web Challenge on Tabular Data to Knowledge Graph Matching co-located with the 18th International Semantic Web Conference, SemTab@ISWC 2019, Auckland, New Zealand, October 30, 2019*, volume 2553 of *CEUR Workshop Proceedings*, pages 25–32. CEUR-WS.org, 2019.
URL: <http://ceur-ws.org/Vol-2553/paper4.pdf>.
- [44] Marco Cremaschi, Roberto Avogadro, and David Chiericato.
MantisTable: an Automatic Approach for the Semantic Table Interpretation.
In *Proceedings of the Semantic Web Challenge on Tabular Data to Knowledge Graph Matching co-located with the 18th International Semantic Web Conference, SemTab@ISWC 2019, Auckland, New Zealand, October 30, 2019*, volume 2553 of *CEUR Workshop Proceedings*, pages 15–24. CEUR-WS.org, 2019.
URL: <http://ceur-ws.org/Vol-2553/paper3.pdf>.
- [45] Marco Cremaschi, Roberto Avogadro, Andrea Barazzetti, and David Chiericato.
MantisTable SE: an Efficient Approach for the Semantic Table Interpretation.
In *Proceedings of the Semantic Web Challenge on Tabular Data to Knowledge Graph Matching (SemTab 2020) co-located with the 19th International Semantic Web Conference (ISWC 2020), Virtual conference (originally planned to be in Athens, Greece), November 5, 2020*, volume 2775 of *CEUR Workshop Proceedings*, pages 75–85. CEUR-WS.org, 2020.
URL: <http://ceur-ws.org/Vol-2775/paper8.pdf>.
- [46] Marco Cremaschi, Flavio De Paoli, Anisa Rula, and Blerina Spahiu.
A fully automated approach to a complete semantic table interpretation.
Future Gener. Comput. Syst., 112:478–500, 2020.
URL: <https://doi.org/10.1016/j.future.2020.05.019>.
- [47] Roberto Avogadro and Marco Cremaschi.
Mantistable V: A novel and efficient approach to semantic table interpretation.
In *Proceedings of the Semantic Web Challenge on Tabular Data to Knowledge Graph Matching co-located with the 20th International Semantic Web Conference (ISWC 2021), Virtual conference, October 27, 2021*, volume 3103 of *CEUR Workshop Proceedings*, pages 79–91. CEUR-WS.org, 2021.
URL: <http://ceur-ws.org/Vol-3103/paper7.pdf>.
- [48] Daniela Oliveira and Mathieu d’Aquin.
ADOG - Annotating Data with Ontologies and Graphs.
In *Proceedings of the Semantic Web Challenge on Tabular Data to Knowledge Graph Matching co-located with the 18th International Semantic Web Conference, SemTab@ISWC 2019, Auckland, New Zealand, October 30, 2019*, volume 2553 of *CEUR Workshop Proceedings*, pages 1–6. CEUR-WS.org, 2019.
URL: <http://ceur-ws.org/Vol-2553/paper1.pdf>.
- [49] Shuang Chen, Alperen Karaoglu, Carina Negreanu, Tingting Ma, Jin-Ge Yao, Jack Williams, Andy Gordon, and Chin-Yew Lin.
LinkingPark: An Integrated Approach for Semantic Table Interpretation.

- In *Proceedings of the Semantic Web Challenge on Tabular Data to Knowledge Graph Matching (SemTab 2020) co-located with the 19th International Semantic Web Conference (ISWC 2020), Virtual conference (originally planned to be in Athens, Greece), November 5, 2020*, volume 2775 of *CEUR Workshop Proceedings*, pages 65–74. CEUR-WS.org, 2020.
URL: <http://ceur-ws.org/Vol-2775/paper7.pdf>.
- [50] Donguk Kim, Heesung Park, Jae Kyu Lee, and Wooju Kim.
Generating Conceptual Subgraph from Tabular Data for Knowledge Graph Matching.
In *Proceedings of the Semantic Web Challenge on Tabular Data to Knowledge Graph Matching (SemTab 2020) co-located with the 19th International Semantic Web Conference (ISWC 2020), Virtual conference (originally planned to be in Athens, Greece), November 5, 2020*, volume 2775 of *CEUR Workshop Proceedings*, pages 96–103. CEUR-WS.org, 2020.
URL: <http://ceur-ws.org/Vol-2775/paper10.pdf>.
- [51] Bram Steenwinckel, Filip De Turck, and Femke Ongenaes.
MAGIC: mining an augmented graph using ink, starting from a CSV.
In *Proceedings of the Semantic Web Challenge on Tabular Data to Knowledge Graph Matching co-located with the 20th International Semantic Web Conference (ISWC 2021), Virtual conference, October 27, 2021*, volume 3103 of *CEUR Workshop Proceedings*, pages 68–78. CEUR-WS.org, 2021.
URL: <http://ceur-ws.org/Vol-3103/paper6.pdf>.
- [52] Rabia Azzi and Gayo Diallo.
AMALGAM: A matching approach to fairfy tabular data with knowledge graph model.
In Álvaro Rocha, Hojjat Adeli, Gintautas Dzemyda, Fernando Moreira, and Ana Maria Ramalho Correia, editors, *Trends and Applications in Information Systems and Technologies - Volume 2, WorldCIST 2021, Terceira Island, Azores, Portugal, 30 March - 2 April, 2021*, volume 1366 of *Advances in Intelligent Systems and Computing*, pages 76–86. Springer, 2021.
URL: https://doi.org/10.1007/978-3-030-72651-5_8.
- [53] Girija Limaye, Sunita Sarawagi, and Soumen Chakrabarti.
Annotating and Searching Web Tables Using Entities, Types and Relationships. *Proc. VLDB Endow.*, 3(1):1338–1347, 2010.
URL: http://www.vldb.org/pvldb/vldb2010/pvldb_vol3/R118.pdf.
- [54] Sebastian Neumaier, Jürgen Umbrich, Josiane Xavier Parreira, and Axel Polleres.
Multi-level semantic labelling of numerical values.
In *The Semantic Web - ISWC 2016 - 15th International Semantic Web Conference, Kobe, Japan, October 17-21, 2016, Proceedings, Part I*, volume 9981 of *Lecture Notes in Computer Science*, pages 428–445, 2016.
URL: https://doi.org/10.1007/978-3-319-46523-4_26.
- [55] Vasilis Efthymiou, Oktie Hassanzadeh, Mariano Rodriguez-Muro, and Vassilis Christophides.
Matching Web Tables with Knowledge Base Entities: From Entity Lookups to Entity Embeddings.
In *The Semantic Web - ISWC 2017 - 16th International Semantic Web Conference, Vienna, Austria, October 21-25, 2017, Proceedings, Part I*, volume 10587 of *Lecture Notes in Computer Science*, pages 260–277. Springer, 2017.
URL: https://doi.org/10.1007/978-3-319-68288-4_16.

- [56] Yoan Chabot, Thomas Labbé, Jixiong Liu, and Raphaël Troncy.
DAGOBAH: an end-to-end context-free tabular data semantic annotation system.
In *Proceedings of the Semantic Web Challenge on Tabular Data to Knowledge Graph Matching co-located with the 18th International Semantic Web Conference, SemTab@ISWC 2019, Auckland, New Zealand, October 30, 2019*, volume 2553 of *CEUR Workshop Proceedings*, pages 41–48. CEUR-WS.org, 2019.
URL: <http://ceur-ws.org/Vol-2553/paper6.pdf>.
- [57] Jiaoyan Chen, Ernesto Jiménez-Ruiz, Ian Horrocks, and Charles Sutton.
ColNet: Embedding the Semantics of Web Tables for Column Type Prediction.
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*, pages 29–36. AAAI Press, 2019.
URL: <https://doi.org/10.1609/aaai.v33i01.330129>.
- [58] Jiaoyan Chen, Ernesto Jiménez-Ruiz, Ian Horrocks, and Charles Sutton.
Learning semantic annotations for tabular data.
In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019*, pages 2088–2094. ijcai.org, 2019.
URL: <https://doi.org/10.24963/ijcai.2019/289>.
- [59] Madelon Hulsebos, Kevin Zeng Hu, Michiel A. Bakker, Emanuel Zraggen, Arvind Satyanarayan, Tim Kraska, Çagatay Demiralp, and César A. Hidalgo.
Sherlock: A Deep Learning Approach to Semantic Data Type Detection.
In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2019, Anchorage, AK, USA, August 4-8, 2019*, pages 1500–1508. ACM, 2019.
URL: <https://doi.org/10.1145/3292500.3330993>.
- [60] Oliver Lehmberg, Dominique Ritze, Robert Meusel, and Christian Bizer.
A Large Public Corpus of Web Tables containing Time and Context Metadata.
In *Proceedings of the 25th International Conference on World Wide Web, WWW 2016, Montreal, Canada, April 11-15, 2016, Companion Volume*, pages 75–76. ACM, 2016.
doi:10.1145/2872518.2889386.
- [61] Xiang Deng, Huan Sun, Alyssa Lees, You Wu, and Cong Yu.
TURL: table understanding through representation learning.
SIGMOD Rec., 51(1):33–40, 2022.
URL: <https://doi.org/10.1145/3542700.3542709>.
- [62] 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*, pages 5998–6008, 2017.
URL: <https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html>.
- [63] Varish Mulwad, Tim Finin, and Anupam Joshi.
Semantic Message Passing for Generating Linked Data from Tables.

- In *The Semantic Web - ISWC 2013 - 12th International Semantic Web Conference, Sydney, NSW, Australia, October 21-25, 2013, Proceedings, Part I*, volume 8218 of *Lecture Notes in Computer Science*, pages 363–378. Springer, 2013.
URL: https://doi.org/10.1007/978-3-642-41335-3_23.
- [64] Chandra Sekhar Bhagavatula, Thanapon Noraset, and Doug Downey.
TabEL: Entity Linking in Web Tables.
In *The Semantic Web - ISWC 2015 - 14th International Semantic Web Conference, Bethlehem, PA, USA, October 11-15, 2015, Proceedings, Part I*, volume 9366 of *Lecture Notes in Computer Science*, pages 425–441. Springer, 2015.
URL: https://doi.org/10.1007/978-3-319-25007-6_25.
- [65] Dominique Ritze, Oliver Lehmberg, and Christian Bizer.
Matching HTML Tables to DBpedia.
In *Proceedings of the 5th International Conference on Web Intelligence, Mining and Semantics, WIMS 2015, Larnaca, Cyprus, July 13-15, 2015*, pages 10:1–10:6. ACM, 2015.
doi:10.1145/2797115.2797118.
- [66] Michael J. Cafarella, Alon Y. Halevy, Daisy Zhe Wang, Eugene Wu, and Yang Zhang.
WebTables: exploring the power of tables on the web.
Proc. VLDB Endow., 1(1):538–549, 2008.
URL: <http://www.vldb.org/pvldb/vol1/1453916.pdf>, doi:10.14778/1453856.1453916.
- [67] Oktie Hassanzadeh, Vasilis Efthymiou, Jiaoyan Chen, Ernesto Jiménez-Ruiz, and Kavitha Srinivas.
SemTab2019: Semantic Web Challenge on Tabular Data to Knowledge Graph Matching - 2019 Data Sets, October 2019.
doi:10.5281/zenodo.3518539.
- [68] Oktie Hassanzadeh, Vasilis Efthymiou, Jiaoyan Chen, Ernesto Jiménez-Ruiz, and Kavitha Srinivas.
SemTab 2020: Semantic Web Challenge on Tabular Data to Knowledge Graph Matching Data Sets, November 2020.
doi:10.5281/zenodo.4282879.
- [69] Vincenzo Cutrona, Federico Bianchi, Ernesto Jiménez-Ruiz, and Matteo Palmonari.
Tough Tables: Carefully Evaluating Entity Linking for Tabular Data, November 2020.
doi:10.5281/zenodo.4246370.
- [70] Oktie Hassanzadeh, Vasilis Efthymiou, Jiaoyan Chen, Ernesto Jiménez-Ruiz, and Kavitha Srinivas.
SemTab 2021: Semantic Web Challenge on Tabular Data to Knowledge Graph Matching Data Sets, November 2021.
- [71] Madelon Hulsebos, Çağatay Demiralp, and Paul Demiralp.
GitTables for SemTab 2021 - CTA task, November 2021.
doi:10.5281/zenodo.5706316.
- [72] Madelon Hulsebos, Çağatay Demiralp, and Paul Groth.
GitTables: A Large-Scale Corpus of Relational Tables.
CoRR, abs/2106.07258, 2021.
URL: <https://arxiv.org/abs/2106.07258>.
- [73] Lin Yao, Hong Liu, Yi Liu, Xinxin Li, and Muhammad Waqas Anwar.
Biomedical named entity recognition based on deep neural network.

- Int. J. Hybrid Inf. Technol*, 8(8):279–288, 2015.
URL: <http://dx.doi.org/10.14257/ijhit.2015.8.8.29>.
- [74] 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)*, pages 4171–4186. Association for Computational Linguistics, 2019.
doi:10.18653/v1/n19-1423.
- [75] Erik F. Tjong Kim Sang and Fien De Meulder.
Introduction to the conll-2003 shared task: Language-independent named entity recognition.
In *Proceedings of the Seventh Conference on Natural Language Learning, CoNLL 2003, Held in cooperation with HLT-NAACL 2003, Edmonton, Canada, May 31 - June 1, 2003*, pages 142–147. ACL, 2003.
URL: <https://aclanthology.org/W03-0419/>.
- [76] Xavier Carreras and Lluís Màrquez.
Introduction to the CoNLL-2005 Shared Task: Semantic Role Labeling.
In *Proceedings of the Ninth Conference on Computational Natural Language Learning, CoNLL 2005, Ann Arbor, Michigan, USA, June 29-30, 2005*, pages 152–164. ACL, 2005.
URL: <https://aclanthology.org/W05-0620/>.
- [77] Ralph Weischedel, Martha Palmer, Mitchell Marcus, Eduard Hovy, Sameer Pradhan, Lance Ramshaw, Nianwen Xue, Ann Taylor, Jeff Kaufman, Michelle Franchini, Mohammed El-Bachouti, Robert Belvin, and Ann Houston.
Ontonotes release 5.0 ldc2013t19 web download.
Philadelphia: Linguistic Data Consortium, 2013.
doi:10.35111/xmhb-2b84.
- [78] Nhung T.H. Nguyen Roselyn S. Gabud and Sophia Ananiadou.
COPIOUS: A gold standard corpus of named entities towards extracting species occurrence from biodiversity literature.
Biodiversity data journal, e29626(7), 2019.
doi:10.3897/BDJ.7.e29626.
- [79] Evangelos Pafilis, Sune P Frankild, Lucia Fanini, Sarah Faulwetter, Christina Pavloudi, Aikaterini Vasileiadou, Christos Arvanitidis, and Lars Juhl Jensen.
The SPECIES and ORGANISMS resources for fast and accurate identification of taxonomic names in text.
PloS one, 8(6):e65390, 2013.
URL: <https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0065390>.
- [80] Martin Gerner, Goran Nenadic, and Casey M. Bergman.
LINNAEUS: A species name identification system for biomedical literature.
BMC Bioinform., 11:85, 2010.
doi:10.1186/1471-2105-11-85.
- [81] Erik M. van Mulligen, Annie Fourrier-Réglat, David Gurwitz, Mariam Molokhia, Ainhoa Nieto, Gianluca Trifirò, Jan A. Kors, and Laura Inés Furlong.
The EU-ADR corpus: Annotated drugs, diseases, targets, and their relationships.
J. Biomed. Informatics, 45(5):879–884, 2012.
doi:10.1016/j.jbi.2012.04.004.

- [82] Àlex Bravo, Janet Piñero González, Núria Queralt-Rosinach, Michael Rautschka, and Laura Inés Furlong.
Extraction of relations between genes and diseases from text and large-scale data analysis: implications for translational research.
BMC Bioinform., 16:55:1–55:17, 2015.
doi:10.1186/s12859-015-0472-9.
- [83] Hrant Khachatrian, Lilit Nersisyan, Karen Hambarzumyan, Tigran Galstyan, Anna Hakobyan, Arsen Arakelyan, Andrey Rzhetsky, and Aram Galstyan.
BioRelEx 1.0: Biological Relation Extraction Benchmark.
In Dina Demner-Fushman, Kevin Bretonnel Cohen, Sophia Ananiadou, and Junichi Tsujii, editors, *Proceedings of the 18th BioNLP Workshop and Shared Task, BioNLP@ACL 2019, Florence, Italy, August 1, 2019*, pages 176–190. Association for Computational Linguistics, 2019.
doi:10.18653/v1/w19-5019.
- [84] Emily Alsentzer, John R Murphy, Willie Boag, Wei-Hung Weng, Di Jin, Tristan Naumann, and Matthew McDermott.
Publicly available clinical bert embeddings.
In *Proceedings of the 2nd Clinical Natural Language Processing Workshop*, pages 72–78, Minneapolis, Minnesota, USA, 2019. Association for Computational Linguistics.
URL: <https://www.aclweb.org/anthology/W19-1909>, doi:10.18653/v1/W19-1909.
- [85] Afshin Sadeghi, Christoph Lange, Maria-Esther Vidal, and Sören Auer.
Integration of scholarly communication metadata using knowledge graphs.
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*, volume 10450 of *Lecture Notes in Computer Science*, pages 328–341. Springer, 2017.
doi:10.1007/978-3-319-67008-9_26.
- [86] Said Fathalla and Christoph Lange.
EVENTS: A dataset on the history of top-prestigious events in five computer science communities.
In Alejandra N. González-Beltrán, Francesco Osborne, Silvio Peroni, and Sahar Vahdati, editors, *Semantics, Analytics, Visualization - 3rd International Workshop, SAVE-SD 2017, Perth, Australia, April 3, 2017, and 4th International Workshop, SAVE-SD 2018, Lyon, France, April 24, 2018, Revised Selected Papers*, volume 10959 of *Lecture Notes in Computer Science*, pages 110–120. Springer, 2018.
doi:10.1007/978-3-030-01379-0_8.
- [87] Said Fathalla and Christoph Lange.
EVENTSKG: A knowledge graph representation for top-prestigious computer science events metadata.
In Ngoc Thanh Nguyen, Elias Pimenidis, Zaheer Khan, and Bogdan Trawinski, editors, *Computational Collective Intelligence - 10th International Conference, ICCCI 2018, Bristol, UK, September 5-7, 2018, Proceedings, Part I*, volume 11055 of *Lecture Notes in Computer Science*, pages 53–63. Springer, 2018.
doi:10.1007/978-3-319-98443-8_6.
- [88] Said Fathalla, Christoph Lange, and Sören Auer.
EVENTSKG: A 5-star dataset of top-ranked events in eight computer science communities.

- In Pascal Hitzler, Miriam Fernández, Krzysztof Janowicz, Amrapali Zaveri, Alasdair J. G. Gray, Vanessa López, Armin Haller, and Karl Hammar, editors, *The Semantic Web - 16th International Conference, ESWC 2019, Portorož, Slovenia, June 2-6, 2019, Proceedings*, volume 11503 of *Lecture Notes in Computer Science*, pages 427–442. Springer, 2019.
doi:10.1007/978-3-030-21348-0_28.
- [89] Said Fathalla, Sahar Vahdati, Sören Auer, and Christoph Lange.
The scientific events ontology of the openresearch.org curation platform.
In Chih-Cheng Hung and George A. Papadopoulos, editors, *Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing, SAC 2019, Limassol, Cyprus, April 8-12, 2019*, pages 2311–2313. ACM, 2019.
doi:10.1145/3297280.3297631.
- [90] Markus Schröder, Christian Jilek, and Andreas Dengel.
A human-in-the-loop approach for personal knowledge graph construction from file names.
In *Proceedings of the 3rd International Workshop on Knowledge Graph Construction (KGCW 2022) co-located with 19th Extended Semantic Web Conference (ESWC 2022), Heronissos, Greece, May 30, 2022*, volume 3141 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2022.
URL: <http://ceur-ws.org/Vol-3141/paper2.pdf>.
- [91] Oktie Hassanzadeh, Vasilis Efthymiou, and Jiaoyan Chen.
SemTab 2022: Semantic Web Challenge on Tabular Data to Knowledge Graph Matching Data Sets - "Hard Tables" R1 and R2, December 2022.
doi:10.5281/zenodo.7416036.
- [92] Denny Vrandečić and Markus Krötzsch.
Wikidata: a free collaborative knowledgebase.
Commun. ACM, 57(10):78–85, 2014.
URL: <https://doi.org/10.1145/2629489>.
- [93] Sören Auer, Christian Bizer, Georgi Kobilarov, Jens Lehmann, Richard Cyganiak, and Zachary G. Ives.
DBpedia: A Nucleus for a Web of Open Data.
In *The Semantic Web, 6th International Semantic Web Conference, 2nd Asian Semantic Web Conference, ISWC 2007 + ASWC 2007, Busan, Korea, November 11-15, 2007*, volume 4825 of *Lecture Notes in Computer Science*, pages 722–735. Springer, 2007.
URL: https://doi.org/10.1007/978-3-540-76298-0_52.
- [94] Jan Martin Keil.
Efficient bounded jaro-winkler similarity based search.
In Torsten Grust, Felix Naumann, Alexander Böhm, Wolfgang Lehner, Theo Härder, Erhard Rahm, Andreas Heuer, Meike Klettke, and Holger Meyer, editors, *Datenbanksysteme für Business, Technologie und Web (BTW 2019), 18. Fachtagung des GI-Fachbereichs „Datenbanken und Informationssysteme“ (DBIS), 4.-8. März 2019, Rostock, Germany, Proceedings*, volume P-289 of *LNI*, pages 205–214. Gesellschaft für Informatik, Bonn, 2019.
doi:10.18420/btw2019-13.
- [95] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomáš Mikolov.
Enriching word vectors with subword information.
Trans. Assoc. Comput. Linguistics, 5:135–146, 2017.
doi:10.1162/tac1_a_00051.

- [96] W. E. Winkler.
String Comparator Metrics and Enhanced Decision Rules in the Fellegi-Sunter Model of Record Linkage., 1990.
ERIC.
- [97] Vladimir I Levenshtein.
Binary codes capable of correcting deletions, insertions and reversals.
Doklady. Akademii Nauk SSSR, 163(4):845–848, 1965.
- [98] Wiem Baazouzi, Marouen Kachroudi, and Sami Faiz.
Kepler-aSI at SemTab 2021.
In *Proceedings of the Semantic Web Challenge on Tabular Data to Knowledge Graph Matching co-located with the 20th International Semantic Web Conference (ISWC 2021), Virtual conference, October 27, 2021*, volume 3103 of *CEUR Workshop Proceedings*, pages 54–67. CEUR-WS.org, 2021.
URL: <http://ceur-ws.org/Vol-3103/paper5.pdf>.
- [99] Xinhe Li, Shuxin Wang, Wei Zhou, Gongrui Zhang, Chenghuan Jiang, Tianyu Hong, and Peng Wang.
KGCODE-Tab Results for SemTab 2022.
In *Proceedings of the Semantic Web Challenge on Tabular Data to Knowledge Graph Matching co-located with the 21st International Semantic Web Conference (ISWC 2022), Virtual conference, October 24, 2022*, volume 3320 of *CEUR Workshop Proceedings*, pages 37–44. CEUR-WS.org, 2022.
URL: <http://ceur-ws.org/Vol-3320/paper5.pdf>.
- [100] Viet-Phi Huynh, Yoan Chabot, Thomas Labbé, Jixiong Liu, and Raphaël Troncy.
From Heuristics to Language Models: A Journey Through the Universe of Semantic Table Interpretation with DAGOBAN.
In *Proceedings of the Semantic Web Challenge on Tabular Data to Knowledge Graph Matching co-located with the 21st International Semantic Web Conference (ISWC 2022), Virtual conference, October 24, 2022*, volume 3320 of *CEUR Workshop Proceedings*, pages 45–58. CEUR-WS.org, 2022.
URL: <http://ceur-ws.org/Vol-3320/paper6.pdf>.
- [101] Marco Cremaschi, Roberto Avogadro, and David Chiericato.
s-elBat: a Semantic Interpretation Approach for Messy taBle-s.
In *Proceedings of the Semantic Web Challenge on Tabular Data to Knowledge Graph Matching co-located with the 21st International Semantic Web Conference (ISWC 2022), Virtual conference, October 24, 2022*, volume 3320 of *CEUR Workshop Proceedings*, pages 59–71. CEUR-WS.org, 2022.
URL: <http://ceur-ws.org/Vol-3320/paper7.pdf>.
- [102] Nora Abdelmageed and Sirko Schindler.
fusion-jena/JenTab, October 2021.
doi:10.5281/zenodo.5584721.
- [103] Nora Abdelmageed and Sirko Schindler.
fusion-jena/JenTab: KG CW 2021, April 2021.
doi:10.5281/zenodo.4730314.
- [104] Nora Abdelmageed.
fusion-jena/JenTab: JenTab code for SemTab 2022, October 2022.
doi:10.5281/zenodo.7229238.
- [105] Nora Abdelmageed and Sirko Schindler.
JenTab_precomputed_lookup_2021, October 2021.
doi:10.5281/zenodo.5584447.

- [106] Nora Abdelmageed.
fusion-jena/JenTab_precomputed_lookup: SemTab2022, October 2022.
doi:10.5281/zenodo.7229246.
- [107] Nora Abdelmageed and Sirko Schindler.
JenTab Solution Files for SemTab 2021, October 2021.
doi:10.5281/zenodo.5584538.
- [108] Nora Abdelmageed and Sirko Schindler.
fusion-jena/JenTab_solution_files: SemTab2022, October 2022.
doi:10.5281/zenodo.7229243.
- [109] Ulrich Küster and Birgitta König-Ries.
Towards standard test collections for the empirical evaluation of semantic web service approaches.
Int. J. Semantic Comput., 2(3):381–402, 2008.
doi:10.1142/S1793351X0800052X.
- [110] Gordon Müller-Seitz and Guido Reger.
'Wikipedia, the Free Encyclopedia' as a role model? Lessons for open innovation from an exploratory examination of the supposedly democratic-anarchic nature of Wikipedia.
Int. J. Technol. Manag., 52(3/4):457–476, 2010.
URL: <https://doi.org/10.1504/IJTM.2010.035985>.
- [111] Nora Abdelmageed, Sirko Schindler, and Birgitta König-Ries.
fusion-jena/biodivtab, November 2022.
doi:10.5281/zenodo.7319654.
- [112] Nora Abdelmageed, Sirko Schindler, and Birgitta König-Ries.
fusion-jena/biodivtab: Benchmark data and code, December 2021.
doi:10.5281/zenodo.5749340.
- [113] Rudi Studer, V. Richard Benjamins, and Dieter Fensel.
Knowledge engineering: Principles and methods.
Data & Knowledge Engineering, 25(1):161 – 197, 1998.
URL: <http://www.sciencedirect.com/science/article/pii/S0169023X97000566>, doi:
[https://doi.org/10.1016/S0169-023X\(97\)00056-6](https://doi.org/10.1016/S0169-023X(97)00056-6).
- [114] Viktor Senderov, Kiril Simov, Nico M. Franz, Pavel Stoev, Terry Catapano, Donat Agosti, Guido Sautter, Robert A. Morris, and Lyubomir Penev.
OpenBiodiv-O: ontology of the OpenBiodiv knowledge management system.
J. Biomed. Semant., 9(1):5:1–5:15, 2018.
doi:10.1186/s13326-017-0174-5.
- [115] Nicola Guarino.
Formal ontology in information systems: Proceedings of the first international conference (FOIS'98), June 6-8, Trento, Italy, volume 46.
IOS press, 1998.
- [116] Ansgar Scherp, Carsten Saathoff, Thomas Franz, and Steffen Staab.
Designing core ontologies.
Appl. Ontology, 6(3):177–221, 2011.
doi:10.3233/A0-2011-0096.
- [117] Helena Sofia Andrade N. P. Pinto and João Pavão Martins.
Ontologies: How can they be built?
Knowl. Inf. Syst., 6(4):441–464, 2004.
doi:10.1007/s10115-003-0138-1.

- [118] Said Fathalla, Sahar Vahdati, Sören Auer, and Christoph Lange.
Semsur: A core ontology for the semantic representation of research findings.
In Anna Fensel, Victor de Boer, Tassilo Pellegrini, Elmar Kiesling, Bernhard Haslhofer, Laura Hollink, and Alexander Schindler, editors, *Proceedings of the 14th International Conference on Semantic Systems, SEMANTiCS 2018, Vienna, Austria, September 10-13, 2018*, volume 137 of *Procedia Computer Science*, pages 151–162. Elsevier, 2018.
doi:10.1016/j.procs.2018.09.015.
- [119] Luan Fonseca Garcia, Mara Abel, Michel Perrin, and Renata dos Santos Alvarenga.
The GeoCore ontology: A core ontology for general use in Geology.
Comput. Geosci., 135:104387, 2020.
doi:10.1016/j.cageo.2019.104387.
- [120] Cássia Trojahn, Renata Vieira, Daniela Schmidt, Adam Pease, and Giancarlo Guizzardi.
Foundational ontologies meet ontology matching: A survey.
Semantic Web, 13(4):685–704, 2022.
doi:10.3233/SW-210447.
- [121] Robert Arp, Barry Smith, and Andrew D Spear.
Building ontologies with basic formal ontology.
Mit Press, 2015.
- [122] Heinrich Herre.
General Formal Ontology (GFO): A foundational ontology for conceptual modelling.
In *Theory and applications of ontology: computer applications*, pages 297–345, 2010.
- [123] Ian Niles and Adam Pease.
Towards a standard upper ontology.
In *2nd International Conference on Formal Ontology in Information Systems, FOIS 2001, Ogunquit, Maine, USA, October 17-19, 2001, Proceedings*, pages 2–9. ACM, 2001.
doi:10.1145/505168.505170.
- [124] Ivan Terziev, Atanas Kiryakov, Dimitar Manov, et al.
Base upper-level ontology (bulo) guidance.
SEKT deliverable, 1(1), 2005.
- [125] Patrícia M. C. Campos, Cássio Chaves Reginato, and João Paulo A. Almeida.
Towards a Core Ontology for Scientific Research Activities.
In *Advances in Conceptual Modeling - ER 2019 Workshops FAIR, MREBA, EmpER, MoBiD, OntoCom, and ER Doctoral Symposium Papers, Salvador, Brazil, November 4-7, 2019, Proceedings*, volume 11787 of *Lecture Notes in Computer Science*, pages 3–12. Springer, 2019.
doi:10.1007/978-3-030-34146-6_1.
- [126] Erik Faessler and Udo Hahn.
Semedico: A comprehensive semantic search engine for the life sciences.
In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, System Demonstrations*, pages 91–96. Association for Computational Linguistics, 2017.
doi:10.18653/v1/P17-4016.
- [127] Vladimir Udovenko and Alsayed Algergawy.
Entity extraction in the ecological domain—a practical guide.
BTW 2019–Workshopband, 2019.

- [128] Yoav Goldberg and Omer Levy.
Word2Vec Explained: deriving Mikolov et al.'s negative-sampling word-embedding method.
CoRR, abs/1402.3722, 2014.
URL: <http://arxiv.org/abs/1402.3722>, arXiv:1402.3722.
- [129] 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*, pages 1532–1543. ACL, 2014.
doi:10.3115/v1/d14-1162.
- [130] 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)*, pages 2227–2237. Association for Computational Linguistics, 2018.
doi:10.18653/v1/n18-1202.
- [131] Ted Pedersen, Siddharth Patwardhan, and Jason Michelizzi.
WordNet: : Similarity - Measuring the Relatedness of Concepts.
In *Proceedings of the Nineteenth National Conference on Artificial Intelligence, Sixteenth Conference on Innovative Applications of Artificial Intelligence, July 25-29, 2004, San Jose, California, USA*, pages 1024–1025. AAAI Press / The MIT Press, 2004.
URL: <http://www.aaai.org/Library/AAAI/2004/aaai04-160.php>.
- [132] Erik Faessler, Friederike Klan, Alsayed Algergawy, Birgitta König-Ries, and Udo Hahn.
Selecting and tailoring ontologies with JOYCE.
In *Knowledge Engineering and Knowledge Management - EKAW 2016 Satellite Events, EKM and Drift-an-LOD, Bologna, Italy, November 19-23, 2016, Revised Selected Papers*, volume 10180 of *Lecture Notes in Computer Science*, pages 114–118. Springer, 2016.
doi:10.1007/978-3-319-58694-6_12.
- [133] J Richard Landis and Gary G Koch.
The measurement of observer agreement for categorical data.
biometrics, pages 159–174, 1977.
doi:10.2307/2529310.
- [134] Eduardo S Brondizio, Josef Settele, Sandra Díaz, and Hien T Ngo.
Global assessment report on biodiversity and ecosystem services of the Intergovernmental Science-Policy Platform on Biodiversity and Ecosystem Services.
Intergovernmental Science-Policy Platform on Biodiversity and Ecosystem Services, 2019.
URL: <https://ipbes.net/global-assessment>.
- [135] Nora Abdelmageed, Felicitas Löffler, and Birgitta König-ries.
BiodivBERT: a Pre-Trained Language Model for the Biodiversity Domain.
In *14th International Semantic Web Applications and Tools for Health Care and Life Sciences Conference SWAT4HCLS Basel, Switherland, February 2023*.
Accepted.

- [136] Nora Abdelmageed, Felicitas Löffler, and Birgitta König-Ries. BiodivBERT: Pre-training Corpora DOIs, May 2022. doi:10.5281/zenodo.6555690.
- [137] GROBID. <https://github.com/kermitt2/grobid>, 2008–2021.
- [138] Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Lukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. Google’s neural machine translation system: Bridging the gap between human and machine translation. *CoRR*, abs/1609.08144, 2016. URL: <http://arxiv.org/abs/1609.08144>.
- [139] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online, oct 2020. Association for Computational Linguistics. URL: <https://www.aclweb.org/anthology/2020.emnlp-demos.6>.
- [140] Nora Abdelmageed, Felicitas Löffler, and Birgitta König-Ries. BiodivBERT: Pre-trained weights, configuration, and training arguments, May 2022. doi:10.5281/zenodo.6554141.
- [141] Nora Abdelmageed, Felicitas Löffler, and Birgitta König-Ries. BiodivBERT: Pre-processed Datasets for NER and RE Downstream Tasks, May 2022. doi:10.5281/zenodo.6554208.
- [142] Nora Abdelmageed, Felicitas Löffler, Leila Feddoul, Alsayed Algergawy, Sheeba Samuel, Jitendra Gaikwad, Anahita Kazem, and Birgitta König-Ries. BiodivNERE: Gold Standard Corpora for Named Entity Recognition and Relation Extraction in Biodiversity Domain, April 2022. doi:10.5281/zenodo.6458503.
- [143] Anne E Thessen, Hong Cui, and Dmitry Mozzherin. Applications of natural language processing in biodiversity science. *Advances in bioinformatics*, 2012, 2012.
- [144] Kirk Roberts, Anupama E Gururaj, Xiaoling Chen, Saeid Pournejati, William R Hersh, Dina Demner-Fushman, Lucila Ohno-Machado, Trevor Cohen, and Hua Xu. Information retrieval for biomedical datasets: the 2016 biocaddie dataset retrieval challenge. *Database*, 2017, 2017.
- [145] Kenneth J Berry and Paul W Mielke Jr. A generalization of Cohen’s kappa agreement measure to interval measurement and multiple raters.

- Educational and Psychological Measurement*, 48(4):921–933, 1988.
doi:10.1177/0013164488484007.
- [146] Fabio Rinaldi, Tilia Renate Ellendorff, Sumit Madan, Simon Clematide, Adrian Van der Lek, Heinz-Theodor Mevissen, and Juliane Fluck.
BioCreative V track 4: a shared task for the extraction of causal network information using the Biological Expression Language.
Database J. Biol. Databases Curation, 2016, 2016.
doi:10.1093/database/baw067.
- [147] Nora Abdelmageed and Birgitta König-Ries.
Biodiversity Metadata Ground Truth, August 2022.
version: v1.0.0, Zenodo.
doi:10.5281/zenodo.6951623.
- [148] Nora Abdelmageed and Birgitta König-Ries.
Biodiversity Metadata Ontology Embeddings (BMOE), August 2022.
version: v1.0.0, Zenodo.
doi:10.5281/zenodo.6951658.
- [149] Nora Abdelmageed and Birgitta König-Ries.
Biodiversity Metadata Ontology (BMO), August 2022.
version: v1.0.0, Zenodo.
doi:10.5281/zenodo.6948519.
- [150] Nora Abdelmageed and Birgitta König-Ries.
Biodiversity Metadata Knowledge Graph (BMKG), August 2022.
doi:10.5281/zenodo.6948573.
- [151] Roberto Avogadro, Marco Cremaschi, Ernesto Jiménez-Ruiz, and Anisa Rula.
A Framework for Quality Assessment of Semantic Annotations of Tabular Data.
In *The Semantic Web - ISWC 2021 - 20th International Semantic Web Conference, ISWC 2021, Virtual Event, October 24-28, 2021, Proceedings*, volume 12922 of *Lecture Notes in Computer Science*, pages 528–545. Springer, 2021.
doi:10.1007/978-3-030-88361-4_31.
- [152] David Chaves-Fraga and Anastasia Dimou.
Declarative Description of Knowledge Graphs Construction Automation: Status & Challenges.
In *Proceedings of the 3rd International Workshop on Knowledge Graph Construction (KGCW 2022) co-located with 19th Extended Semantic Web Conference (ESWC 2022), Hersonissos, Greece, May 30, 2022*, volume 3141 of *CEUR Workshop Proceedings*, 2022.
URL: <http://ceur-ws.org/Vol-3141/paper5.pdf>.
- [153] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf.
DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter.
CoRR, abs/1910.01108, 2019.
URL: <http://arxiv.org/abs/1910.01108>.
- [154] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov.
RoBERTa: A Robustly Optimized BERT Pretraining Approach.
CoRR, abs/1907.11692, 2019.
URL: <http://arxiv.org/abs/1907.11692>.

Appendices

Appendix A

Certificates and Awards

Figure A.1 shows the awarded certificate for JenTab as a second place winner of the Usability Track. Figure A.2 depicts the awarded certificate for BiodivTab as a first place winner of the Applications Track. Both are awarded by IBM Research at SemTab, ISWC 2021.



Figure A.1: JenTab Usability Track Certificate ISWC 2021.



Figure A.2: BiodivTab Applications Track Certificate ISWC 2021.

