Knowledge Extraction from Unstructured Data

von der Fakultät für Elektrotechnik und Informatik
der Gottfried Wilhelm Leibniz Universität Hannover
zur Erlangung des akademischen Grades

Doktor-Ingenieur
(abgekürzt Dr.-Ing.)
genehmigte Dissertation

von Herrn

Ahmad Sakor, M. Sc.
geboren am 09.04.1992
in Daraa, Syrien

2023

Referentin: Prof. Dr. Maria-Esther Vidal, Leibniz Universität Hannover
Korreferent: Prof. Dr. Ricardo Usbeck, University of Hamburg , Hamburg
Tag der Promotion: 05.05.2023

Abstract

Data availability is becoming more essential, considering the current growth of web-based data. The data available on the web are represented as unstructured, semi-structured, or structured data. In order to make the web-based data available for several Natural Language Processing or Data Mining tasks, the data needs to be presented as machine-readable data in a structured format. Thus, techniques for addressing the problem of capturing knowledge from unstructured data sources are needed. Knowledge extraction methods are used by the research communities to address this problem; methods that are able to capture knowledge in a natural language text and map the extracted knowledge to existing knowledge presented in knowledge graphs (KGs). These knowledge extraction methods include Named-entity recognition, Named-entity Disambiguation, Relation Recognition, and Relation Linking. This thesis addresses the problem of extracting knowledge over unstructured data and discovering patterns in the extracted knowledge. We devise a rule-based approach for entity and relation recognition and linking. The defined approach effectively maps entities and relations within a text to their resources in a target KG. Additionally, it overcomes the challenges of recognizing and linking entities and relations to a specific KG by employing devised catalogs of linguistic and domain-specific rules that state the criteria to recognize entities in a sentence of a particular language, and a deductive database that encodes knowledge in community-maintained KGs. Moreover, we define a Neuro-symbolic approach for the tasks of knowledge extraction in encyclopedic and domain-specific domains; it combines symbolic and sub-symbolic components to overcome the challenges of entity recognition and linking and the limitation of the availability of training data while maintaining the accuracy of recognizing and linking entities. Additionally, we present a context-aware framework for unveiling semantically related posts in a corpus; it is a knowledge-driven framework that retrieves associated posts effectively. We cast the problem of unveiling semantically related posts in a corpus into the Vertex Coloring Problem. We evaluate the performance of our techniques on several benchmarks related to various domains for knowledge extraction tasks. Furthermore, we apply these methods in real-world scenarios from national and international projects. The outcomes show that our techniques are able to effectively extract knowledge encoded in unstructured data and discover patterns over the extracted knowledge presented as machine-readable data. More importantly, the evaluation results provide evidence to the effectiveness of combining the reasoning capacity of the symbolic frameworks with the power of pattern recognition and classification of sub-symbolic models.

Keywords *Knowledge Extraction and Discovery, Semantic Web, Neuro-Symbolic*

Zusammenfassung

Die Datenverfügbarkeit wird in Anbetracht des derzeitigen Wachstums webbasierter Daten immer wichtiger. Die im Web verfügbaren Daten werden als unstrukturierte, semistrukturierte oder strukturierte Daten dargestellt. Um die webbasierten Daten für verschiedene Aufgaben der natürlichen Sprachverarbeitung oder des Data Mining verfügbar zu machen, müssen die Daten als maschinenlesbare Daten in einem strukturierten Format dargestellt werden. Es werden also Techniken benötigt, die das Problem der Wissenserfassung aus unstrukturierten Datenquellen lösen. In der Forschung werden Methoden zur Wissensextraktion eingesetzt, um dieses Problem zu lösen. Diese Methoden sind in der Lage, Wissen in einem natürlichsprachlichen Text zu erfassen und das extrahierte Wissen auf vorhandenes Wissen abzubilden, das in Wissensgraphen (KGs) dargestellt wird. Zu diesen Methoden der Wissensextraktion gehören die Erkennung von benannten Personen, die Disambiguierung von benannten Personen, die Erkennung von Beziehungen und die Verknüpfung von Beziehungen. Diese Dissertation befasst sich mit dem Problem der Wissensextraktion aus unstrukturierten Daten und der Entdeckung von Mustern in dem extrahierten Wissen. Wir entwickeln einen regelbasierten Ansatz für die Erkennung und Verknüpfung von Entitäten und Beziehungen. Der definierte Ansatz bildet Entitäten und Relationen in einem Text effektiv auf ihre Ressourcen in einer Ziel-KG ab. Darüber hinaus überwindet er die Herausforderungen bei der Erkennung und Verknüpfung von Entitäten und Relationen mit einem bestimmten KG, indem er ausgearbeitete Kataloge linguistischer und domänenspezifischer Regeln verwendet, die die Kriterien für die Erkennung von Entitäten in einem Satz einer bestimmten Sprache festlegen, sowie eine deduktive Datenbank, die das Wissen in von der Gemeinschaft verwalteten KGs kodiert. Darüber hinaus definieren wir einen neurosymbolischen Ansatz für die Aufgaben der Wissensextraktion in enzyklopädischen und domänenspezifischen Domänen. Er kombiniert symbolische und subsymbolische Komponenten, um die Herausforderungen der Erkennung und Verknüpfung von Entitäten und die Beschränkung der Verfügbarkeit von Trainingsdaten zu überwinden und gleichzeitig die Genauigkeit der Erkennung und Verknüpfung von Entitäten zu erhalten. Darüber hinaus stellen wir ein kontextbezogenes Framework zur Aufdeckung semantisch verwandter Beiträge in einem Korpus vor. Es handelt sich dabei um ein wissensbasiertes Framework, das assoziierte Beiträge effektiv auffindet. Wir haben das Problem der Aufdeckung semantisch verwandter Beiträge in einem Korpus in das Knotenfärbungsproblem übertragen. Wir bewerten die Leistung unserer Techniken anhand mehrerer Benchmarks, die sich auf verschiedene Domänen für Wissensextraktionsaufgaben beziehen. Außerdem wenden wir diese Methoden

in realen Szenarien aus nationalen und internationalen Projekten an. Die Ergebnisse zeigen, dass unsere Techniken in der Lage sind, in unstrukturierten Daten kodiertes Wissen effektiv zu extrahieren und Muster über das extrahierte Wissen zu entdecken, das als maschinenlesbare Daten präsentiert wird. Noch wichtiger ist, dass die Evaluierungsergebnisse einen Beweis für die Effektivität der Kombination der Schlussfolgerungskapazität der symbolischen Frameworks mit der Leistung der Mustererkennung und der Klassifizierung von subsymbolischen Modellen liefern.
Schlagwörter:*Internet der Dinge, Streamdaten, Semantic Web, RDF-Verdichtung, Semantische Anreicherung*

Acknowledgements

During this exciting Ph.D. journey, I met several people who inspired and motivated me all those years. First and foremost, I would like to thank Prof. Dr. Maria-Esther Vidal for her constant support and patience and for giving me the chance to pursue the degree under her supervision. Her support and advice in each step helped me to achieve this research work. Furthermore, I would like to thank the *Godfather* of my thesis Dr. Kuldeep Singh for his support and contributions to our research papers. My journey with Kuldeep started when he was the mentor of my master's thesis and still continues until today. Last but not least, a special thanks to Prof. Dr. Sören Auer for his support and the nice chats we always had and for encouraging me to submit my first research paper to one of the competitive conferences in NLP that refined the work presented in this thesis.

To my old friend Mohamad Yaser Jaradeh. We started off our study journey at the same time in Syria during our bachelor's studies, then moved to Germany to pursue our master's studies at the University of Bonn. Unsurprisingly, we both got the opportunity to start our Ph.D. journey at the Leibniz University of Hannover, and who knows, we may end up at the same place for our next career chapter.

Also, I'm grateful for knowing my colleague Sam. She is always a good listener to my stories (work-related or otherwise), and she is the one who brought me into jogging, which was really good after a long day of work.

I would also like to thank my colleagues Ariam, Emetis, Mayra, Kemele, Enrique, and Philipp, for whom I shared great moments throughout this journey. Not to mention all other SDM group members for their direct and indirect contributions to this adventure. Last but not least, a special thanks to Katja Bartel and Simone Matern for making our life easier and always supporting us with all the bureaucratics related to work.

My appreciation to my family, whom without their support, I would not have been able to achieve this milestone. My family that encourages me to travel to another country in order to pursue my Ph.D. To my mom Mrs. Souhad Almoustafa for her constant encouragement and pure love. To my dad Mr. Mohammad Fateh Sakor, for always being my idol without fail. To my brothers Rabee and Samer, for their continuous support and love, and to my best friend Mohammad Al Mahamed who has always been there for me and supported me during the hard times of this journey. Last but not least, many thanks to my friends who I met in Germany and helped me to start a new chapter in my life, Bilal, Tarek, Farhad, Linda, and many others. Thank you all again for your unconditional support.

I dedicate this thesis to my gurus: Prof. Dr. Maria-Esther Vidal, Prof. Dr. Sören Auer, and Dr. Kuldeep Singh.

Contents

1 Introduction **1**

1.1 Motivation 4

1.2 Existing Approaches 4

1.3 Problem Statement and Challenges 6

1.4 Research Questions 8

1.5 Thesis Overview 10

1.5.1 Contributions 10

1.5.2 List of Publications 15

1.6 Thesis Structure 18

1.7 Summary 19

2 Background **21**

2.1 The Semantic Web 21

2.1.1 Linked Data 22

2.1.2 Ontologies 23

2.1.3 Knowledge Bases and Knowledge Graphs 24

2.1.4 Resource Description Framework 25

2.1.5 OWL Web Ontology Language 28

2.1.6 SPARQL 28

2.2 Knowledge Representation 29

2.2.1 Horn Clauses 29

2.2.2 Deductive Database 30

2.3 Natural Language Processing 30

2.3.1 Information Extraction 31

2.3.2 Knowledge Extraction 31

2.4 AI Approaches for Knowledge Extraction 35

2.4.1 Rule-Based Systems 35

2.4.2 Machine Learning Approaches 36

2.4.3 Neuro-symbolic Systems 37

2.4.4 Heuristic Approaches 39

2.4.5 Greedy Algorithms 39

2.5 Knowledge Discovery . . . 39
2.6 Summary . . . 42

3 Related Work 45
3.1 Knowledge Recognition . . . 46
3.1.1 Named-Entity Recognition . . . 46
3.1.2 Relation Recognition . . . 47
3.2 Knowledge Linking . . . 48
3.2.1 Named-Entity Disambiguation . . . 48
3.2.2 Relation Linking . . . 51
3.2.3 Semantic Type Prediction . . . 52
3.3 Knowledge Discovery . . . 52
3.3.1 Recommending Related Posts . . . 52
3.3.2 Finding Sentence Similarity . . . 53
3.3.3 Finding Entity Relatedness . . . 53
3.3.4 Community Detection . . . 54
3.3.5 Graph Coloring . . . 55
3.4 Summary . . . 56

4 Symbolic Approach for Entity & Relation Recognition and Linking 57
4.1 Motivation . . . 61
4.2 Symbolic Approach for Knowledge Extraction . . . 63
4.2.1 DBpedia Use Case . . . 64
4.2.2 Wikidata Use Case . . . 67
4.2.3 Architecture . . . 67
4.2.4 Background Knowledge . . . 68
4.2.5 Catalog of Rules . . . 70
4.2.6 Recognition . . . 70
4.2.7 Linking . . . 70
4.3 Experimental Study - DBpedia Use case . . . 72
4.4 Experimental Study - Wikidata Use Case . . . 75
4.4.1 Experimental Results . . . 77
4.5 Impact . . . 81
4.6 Adoption and Reusability . . . 81
4.7 Maintenance and Sustainability . . . 82
4.8 Summary . . . 82

5 Neuro-symbolic approach for Named-Entity Recognition and Disambiguation 83
5.1 Motivating . . . 86

5.2 Problem Statement 86
5.3 Proposed Solution 87
5.3.1 Symbolic System 88
5.3.2 Candidates Generation 89
5.3.3 Sub-Symbolic System 90
5.3.4 Candidates Disambiguation Step 90
5.4 Experiment and Results 91
5.4.1 Datasets 91
5.4.2 Implementation 91
5.4.3 Baselines for Comparison 92
5.4.4 Main Results 93
5.5 Summary 96

6 Context-aware Framework for Unveiling Semantically related Posts in a Corpus 97
6.1 Motivating Example 99
6.2 Problem Statement and Solution 101
6.2.1 Preliminaries 101
6.2.2 The DSATUR Algorithm 103
6.2.3 Problem Statement 103
6.2.4 Proposed Solution 104
6.3 The PINYON Approach 109
6.3.1 Context-Aware Post Encoding 109
6.3.2 Context-Aware Post Decoding 111
6.3.3 Agnostic of the PINYON architecture 113
6.4 Experimental Study 114
6.4.1 Experimental Configuration 114
6.4.2 Experiment 1- Medical Domain 119
6.4.3 Experiment 2- Sport Domain 125
6.4.4 Discussion 127
6.5 Summary 128

7 Applications of Knowledge Extraction 129
7.1 Motivation 130
7.2 Projects 132
7.3 Use Cases 135
7.3.1 Knowledge Extraction from semi-structured data 135
7.3.2 Knowledge Extraction for Knowledge Graphs Creation Pipelines136
7.3.3 Knowledge Extraction for Entity Alignment 136
7.3.4 Knowledge Extraction for Capturing Contextual Knowledge 137

7.3.5 Knowledge Extraction for annotating Customized Semantic Data Models . . . 138
7.3.6 Knowledge Extraction and Discovery from Social Media Posts139
7.4 Evaluation . . . 140
7.4.1 Drug-Drug-Interactions Use Case . . . 140
7.4.2 Entity Alignment Use Case . . . 141
7.4.3 Annotating Customized Semantic Data Model Use Case . . . 143
7.5 Summary . . . 144

8 Conclusion and Future Directions 145
8.1 Revising the Research Questions . . . 146
8.2 Limitations . . . 148
8.3 Future Directions . . . 150
8.4 Closing Remarks . . . 150

Bibliography 151

A List of Publications 171
A.1 Peer-Reviewed International Journals . . . 171
A.2 Papers in Proceedings of Peer-Reviewed Conferences . . . 172
A.3 Under-Review . . . 173

B Appendix 175
B.1 Neuro-symbolic approach components . . . 175
B.1.1 Background Knowledge . . . 175
B.1.2 Catalog of linguistic rules . . . 175
B.1.3 Catalog of domain-specific rules . . . 177
B.1.4 Rule-based approach . . . 177
B.1.5 Wikidata dataset SPARQL query . . . 178
B.1.6 Implementation details . . . 179

C Appendix 181
C.1 Proofs and Propositions . . . 181
C.1.1 Lemma 4.1 . . . 181
C.1.2 DSATUR Propositions . . . 181

D Curriculum Vitae 183

List of Figures

1.1 Knowledge Extraction from Unstructured Data 3

2.1 **The Semantic Web Technology Stack**, this stack shows how RDF plays a key role as the base many building blocks of the technology [76]. . 22

2.2 The Linked Open Data Cloud as per November 2022 23

2.3 DBpedia Extraction Framework [77] 26

2.4 An RDF Graph for a Triple 26

2.5 An example of Named-Entity Recognition 32

2.6 An example of Relation Recognition [89] 33

2.7 An example of Linking Recognized Entities [90] 34

2.8 An example of Semantic Type Prediction [90] 35

2.9 An example of training a basic Perceptron Neural Network [97] 37

2.10 An example of a transformer model including an encoding and decoding phase [100] 38

2.11 Starting at A, a greedy algorithm will find the local maximum at "m", oblivious to the global maximum at "M". 40

2.12 Problem of reaching the largest sum 40

2.13 An example of a Bipartite Graph and the generated Complement Line Graph 42

3.1 Categories of the state-of-the-art Approaches 45

4.1 Performance of two Named-Entity Disambiguation (NED) and Relation Linking (RL) approaches on Specific Questions. TagMe and DBpedia Spotlight are the top-2 NED systems over the LC-QuAD QA dataset. However, considering short text questions, their behavior varies concerning question features, e.g., lowercase vs. uppercase, having implicit versus. explicit mappings, etc. Similar behavior has been observed for the top relation linking tools. 59

4.2 `Falcon 2.0` **API Web Interface**. 61

4.3 Overview of Falcon Approach. Falcon consists of two building blocks: 1) An extended knowledge graph which is built by merging information from various knowledge sources such as DBpedia, Wikidata, Oxford Dictionary, and WordNet. 2) Falcon architecture that has several modules focusing on surface form extraction and linking them to KG. 63

4.4 The `Falcon 2.0` **Architecture**. The boxes highlighted in Grey are reused from the DBpedia use case implementation. Grey boxes contain a linguistic pipeline for recognizing and linking entity and relation surface forms. The boxes in White are our addition to the Falcon pipeline to build a resource for the Wikidata entity and relation linking. The white boxes constitute what we refer to as BK specific to Wikidata. The text search engine contains the alignment of Wikidata entity/relation labels along with the entity and relation aliases. It is used for generating potential candidates for entity and relation linking. RDF triple store is a local copy of Wikidata triples containing all entities and predicates. 67

4.5 `Falcon 2.0` **Background Knowledge** is built by converting labels of entities and relations in Wikidata into pairs of alignments. It is a part of search engine (cf. Figure 4.4). 69

5.1 Exemplar sentence illustrating Biomedical Entity Linking challenges. The surface form "temperature" is used in two different contexts, and with the same label in UMLS, disambiguation will require additional domain-specific knowledge, such as entity descriptions. 85

5.2 The Noreen architecture with running example. The first component is the symbolic component. With the given sentence, Noreen first extracts the entities' surface form (in-case surface forms are present, first step can be skipped). The next step generates the candidates from the background knowledge (deductive database) from the surface forms. Domain-specific rules decide the rank of the candidates at this step. Now, a pre-trained sub-symbolic component (pre)trained using entity definitions component predicts the entity type. Using entity type, the final ranking of the candidate list generated by the symbolic component is decided for the correct output. 87

6.1 Motivating Example. A tweet announcing the promising results of combining Neoadjuvant chemotherapy and Nivolumab in patients with Non-small-cell lung cancer in stage IIIA. a) Related tweets resulting from searching using the Twitter API the criteria *Neoadjuvant chemotherapy and Nivolumab* and *Resectable non-small-cell lung cancer*. b) Results of semantically related tweets based on PINYON. Tweets retrieved from Twitter include at least one of the keywords in the input post. Tweets identified by PINYON are semantically related to the input post based on the shared context. 100

6.2 **Running Example**. Mapping the Problem of Context-Aware Post Recommendation into the Vertex Coloring Problem. Entities in a post are annotated with terms in different contexts (e.g., domain-specific or encyclopedic), relatedness between the terms in the same context are represented by edges of a Bipartite Graph *BP=(*$V_1 \cup V_2$*,E)*; a similarity measure quantifies relatedness of the terms in a context. (1) The complement of the line graph of BP is computed *Comp(LP(BP))=(F,K)*. (2) *Comp(LP(BP))=(F,K)* is colored with the minimal number of colors that maximize the value of *ColoredSimilarity(.)*, (3) Colored Graph in mapped into Communities. (4) Communities are used to create Context-Aware Communities of Posts. 104

6.3 Running Example. Entities in a post are annotated with concepts in various contexts (e.g., domain-specific or encyclopedic) derived from the knowledge graphs. 106

6.4 **The PINYON approach architecture**. The pipeline receives an input post $\mathcal{P}$, a dataset $\mathcal{R}$ of posts, and a set of contexts $\mathcal{C}$ (e.g., DBpedia, Wikidata, UMLS) and outputs related posts in the corresponding contexts *PostRelated(*$\mathcal{P}$,$\mathcal{P}_c(\mathcal{R})$*,c)*. During the encoding phase, the input post $\mathcal{P}$ and the Corpus $\mathcal{R}$ are annotated with terms from the provided contexts $\mathcal{C}$. The annotations are used to build the Bipartite Graph BP; then the Bipartite Graph is transformed to Complement Line Graph *Comp(LP(BP))=(F,K)*. The complement line graph is utilized to create context-aware communities during the decoding phase. The posts are grouped into the created communities, and community similarity is calculated to determine the related posts. 109

6.5 Comparison of Text- and Entity-based Embeddings of Biomedical Tweets. 119

6.6 Quality of the computed communities. Communities evaluated in terms of five metrics (higher values are better); Communities for the eight tweets in 6.4.2 are reported. *PINYON-CACD* exhibits the best performance. . . 119

6.7 Report of the Aggregated values for the community quality metrics. . . . 120

6.8 Context-Aware Related Post Decoding execution time for each tweet in Experiment 1 120

7.1 **Motivating Example**. Heterogeneous sources of knowledge. (a) Unstructured data sources, e.g., clinical notes, medical images, and clinical tests, encode invaluable knowledge about a patient medical condition. (b) Factors impact on the effectiveness of a treatment; they need to be identified to increase a patient survival time. (c) Various biomedical repositories maintain knowledge collected by the scientific community about facts that can contribute to the prescription of effective treatments. Data sources range from structured (e.g, COSMIC), to unstructured (e.g., PubMed); and short texts in structured data sources may encode also relevant knowledge (e.g., drug interactions). Heterogeneity problems across sources need to be solved for extracting the required knowledge [62]. 131

7.2 FALCON Recognizes Relevant Entities and Predicates. As a result, a Fine-Grained Representation of Drug-Drug Interactions is part of the Knowledge4COVID-19 KG [70]. 137

7.3 An exemplary RML Mapping Rule calling one FnO function named FalconDBpedia-Function. This function performs NER and NEL to link drug names to resources in DBpedia [268]. 138

7.4 A snippet of the data sources to be annotated by E-Falcon 139

7.5 A snippet of the data sources to be annotated by E-Falcon 140

7.6 A snippet of the data sources to be annotated by E-Falcon 144

List of Tables

4.1 Performance of the Falcon Framework compared to various entity linking tools. 73

4.2 Performance of the Falcon Framework compared to various Relation Linking tools. 74

4.3 Entity linking evaluation results on LC-QuAD 2.0 & SimpleQuestion datasets. Best values are in bold. 77

4.4 Entity linking evaluation results on the WEBQSP test dataset. Best values are in bold. 77

4.5 Entity Linking Performance of Falcon vs Falcon 2.0 on WEBQSP-WD. Best values are in bold. 79

4.6 Relation linking evaluation results on LC-QuAD 2.0 & SimpleQuestion datasets. 80

4.7 **Sample Questions from LC-QuAD 2.0 datset**. The table shows five sample questions and associated gold standard relations. These sentences do not include standard sentential relations in the English language. Considering Wikidata is largely authored by the crowd, the crowd often creates such uncommon relations. `Falcon 2.0` finds limitation in linking such relations, and most results are empty. 80

5.1 **Dataset Statistics**. |Mentions| is the number of mentions. |Entities| is the number of unique entities in the labeled partition, not the total Knowledge Base size. For EMS and Wikidata datasets, each surface form correspond to a unique entity in the corresponding knowledge base and it is only a test set. 92

5.2 Accuracy on the MedMention dataset (test) using two settings. Baseline values are from [151, 155]. First, in the absence of gold entity types and second (right side) with gold types. The seen subset of test data is when entities are seen during baseline training with corresponding sentences of the document. The unseen test set is the zero-shot setting. For *Noreen* both unseen and seen are the same behavior because we do not train our model with corresponding sentences of the entities. *Noreen* substantially outperforms baselines. 93

5.3 Generalization of Noreen. In this experiment, we follow an end-to-end setting to link a given keyword to a background knowledge base. 94
5.4 Ablation Study on MedMention Dataset. We clearly see that symbolic and sub-sybolic components complement each other's performance (accuracy). 94
5.5 End-to-end entity linking performance on MedMention dataset. 95
5.6 Recall values for Noreen for Entity Disambiguation with given entity mentions. 95

6.1 Summary of PINYON Notation 102
6.2 Overview of the tweets used for Experiment 1 118
6.3 **Size of the graphs *(*Comp(LP(BP)))** generated during the Context-Aware Post Encoding for each of the Tweets in Experiment 1 118
6.4 **PINYON Performance compared to the baselines for Experiment 1.** TweetKB COV19 is the input dataset. The Accuracy metric is reported. Best results are in bold. The columns in Orange are Tweets related to Lung Cancer. The columns in Yellow are tweets related to COVID-19. Three different solvers are utilized for the ablation study; METIS for graph partitioning and community detection, CACD and Welsh Powell for graph coloring. Three different BKs are considered for this experiment. 124
6.5 **PINYON Performance compared to the Baselines for Experiment 2**. FIFA dataset is the input dataset. The reported metrics are Precision, Recall, and F-Score. Best results are in bold. Three different solvers are utilized for the ablation study; METIS for graph partitioning and community detection, CACD and Welsh Powell for graph coloring. Two different BKs are considered. 125

7.1 Overview of extracted DDI patterns. Drugs mentions are in bold. Effect is in Italic. Impact is underlined 135
7.2 Number of entities, unique entities, and unique entities appearing two or more times obtained from Wikidata for each of the datasets used. 138
7.3 *Effectiveness.* The performance of *EABlock* is assessed in 15 datasets. The aligned entities are compared with the resources of the original labels. Misspelling errors are added with five types of transformations. Entity alignment is performed by the keyword-based functions over the keywords generated by the five transformations. The aligned entities are compared with the resources of the original labels. *EABlock* fails to recognize long labels generated from keywords with non-alphanumerical characters present in UMLS (e.g., n-((2r)-1-(((2r)-1-(((2r)-6-amino-1-(4-amino-4-carboxy-1-piperidinyl)-1-oxo-2-hexanyl)amino)-4-methyl-1-oxo-2-pentanyl)amino)-1-oxo-3-phenyl-2-propanyl)-d-phenylalaninamide). 142

Chapter 1

Introduction

A large amount of data is being produced online or offline daily. The produced data are represented as unstructured, semi-structured, or structured data. Unstructured data is data that does not have a pre-defined data model or is not organized in a pre-defined manner. Semi-structured data is a subtype of structured data that does not follow the tabular structure of data models found in relational databases or other types of data tables, but nevertheless contains tags to distinguish semantic components and impose hierarchies of records and fields within the data. Therefore, it can be called a self-describing structure. Structured data is data represented in a standardized format and has a well-defined structure that complies to a data model. Structured data is readable by both humans and computers. The current growth of data showed the need to have a machine-readable representation of the data where data must be structured or semi-structured data and represented in a format that can be processed by a computer. Most of the existing data on the web is unstructured data. Even some data sources are semi-structured (e.g., DrugBank[1]), they contain several textual descriptions that are not machine-readable. In order to fill the gap between unstructured and structured data, accurate techniques for knowledge extraction over unstructured and semi-structured data are required.

Knowledge extraction is the process of capturing knowledge from unstructured and semi-structured data sources. The captured knowledge must be presented in a machine-readable format in order to facilitate inferencing. While it follows a similar methodology to information extraction, where structured information is extracted from unstructured or semi-structured machine-readable data, the major difference is that the extraction results must go beyond the creation of structured information or the transformation into a relational schema. Either existing formal

[1] `https://drugbank.com`

information must be reused (e.g., in knowledge graphs [1] or knowledge bases), or a schema must be created based on the source data. A knowledge base (KB) is a mechanism that computers use to store complex structured and unstructured data. A knowledge base that integrates data using a graph-structured data model or topology is known as a knowledge graph. Knowledge graphs are commonly used to store interlinked descriptions of entities, events, or abstract concepts, as well as to encode the semantics behind the terminology that is utilized. Thus, performing knowledge extraction over unstructured data requires mapping the knowledge encoded in the unstructured data (natural language text) to an existing knowledge in a knowledge graph or knowledge base.

In real-world applications, extracting knowledge from a natural language text comprises several tasks. The tasks studied in this thesis are named-entity recognition (NER), named-entity disambiguation (NED) [2], relation recognition (RR), and relation linking (RL). Named-entity recognition [3] and relation recognition [4] extract surface forms (a.k.a mentions) from a natural language text. A surface form of an entity or a relation is the form of the entity or relation as it appears in the text. Surface forms can also be defined as the set of tokens that comprise an entity or a relation where each token represents a word in the natural language text. Named-entity disambiguation [2] is the task of assigning a unique identifier to recognized entities mentioned in a natural language text. It is a well-studied topic in scientific literature and finds applicability in specific domains such as question answering, social media, knowledge base construction, etc [5, 6]. Relation linking [7] maps the surface forms in a text representing relation to equivalent relations (predicates) of a knowledge graph or knowledge base. It has also attracted the interest of vast research communities since it complements NEDs approaches for the task of knowledge extraction [8, 9, 10, 11, 12]. Therefore, accurately recognizing entities and relation in a natural language text (unstructured data) is essential for knowledge extraction over unstructured data. Recognizing the wrong entity or relation, or linking to a resource that does not represent the recognized entities or relation will negatively impact the performance of knowledge extraction.

This thesis proposes techniques to extract knowledge from unstructured and semi-structured data and maps the extracted knowledge to existing knowledge in various sources (e.g., DBpedia [13], Wikidata [14], the Unified Medical Language System (UMLS) [15], and DrugBank). Furthermore, we propose new methods for entity & relation recognition, and entity & relation linking to different knowledge graphs. Figure 1.1 demonstrates the knowledge extraction methods defined by this thesis and the scientific contributions of each defined method. The defined methods comprise symbolic and neuro-symbolic approaches. The symbolic approaches resort to catalogs of linguistic and domain-specific rules and background knowledge that encodes the knowledge presented in community-maintained (e.g.,

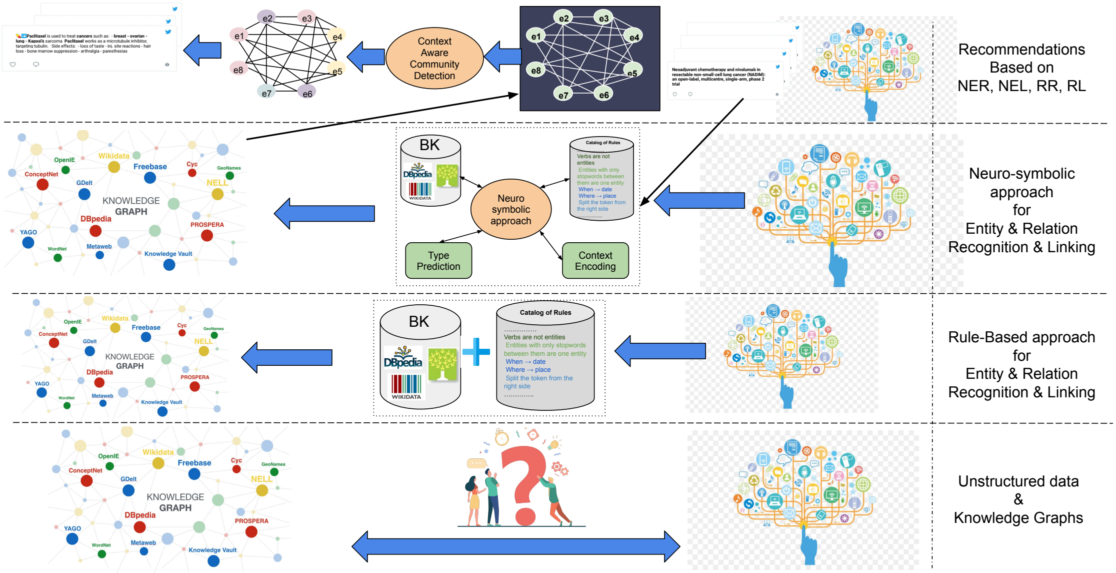

Paclitaxel is used to treat cancers such as: - breast - ovarian - lung - Kaposi's sarcoma Paclitaxel works as a microtubule inhibitor, targeting tubulin. Side effects: - loss of taste - inj. site reactions - hair loss - bone marrow suppression - arthralgia - paresthesias
e1
e2
e3
e4
e5
e6
e7
e8
Context Aware Community Detection
Neoadjuvant chemotherapy and nivolumab in resectable non-small-cell lung cancer (NADIM): an open-label, multicentre, single-arm, phase 2 trial
Recommendations Based on NER, NEL, RR, RL
OpenIE
Wikidata
Freebase
Cyc
GeoNames
ConceptNet
GDelt
NELL
KNOWLEDGE GRAPH
DBpedia
PROSPERA
YAGO
WordNet
Metaweb
Knowledge Vault
BK
WIKIDATA
Catalog of Rules
Verbs are not entities
Entities with only stopwords between them are one entity
When → date
Where → place
Split the token from the right side
Neuro symbolic approach
Type Prediction
Context Encoding
Neuro-symbolic approach for Entity & Relation Recognition & Linking
Rule-Based approach for Entity & Relation Recognition & Linking
Unstructured data & Knowledge Graphs

Figure 1.1: **Knowledge Extraction from Unstructured Data**. Mapping knowledge encoded in unstructured data to existing knowledge presented in a knowledge graph or a knowledge base is performed using symbolic and neuro-symbolic approaches for NER, NED, RE, and RL tasks. A background knowledge encodes knowledge from various sources is employed in symbolic and neuro-symbolic approaches. A recommendation system benefits from the defined approaches to map entities and relations in a corpus of posts to existing knowledge graphs for the task of unveiling semantically related posts in a corpus.

DBpedia, and Wikidata) and domain-specific (e.g., DrugBank, and UMLS) knowledge sources. The neuro-symbolic approach comprises symbolic and sub-symbolic components that empower the neuro-symbolic approach to take benefits of both symbolic and machine learning approaches. In addition, this thesis proposes techniques for unveiling semantically related posts in a corpus by resorting to context-aware background knowledge. We devise a knowledge-driven approach that exploits the semantics encoded in the context of the text describing a post and a context-aware background knowledge created by utilizing the defined methods for recognizing and linking entities and relations in a natural language text. Furthermore, we show how the defined methods can be used in real-world applications for extracting knowledge from unstructured and semi-structured data in different domains and what are the benefits of extracting such knowledge.

1.1 Motivation

Although a large volume of data is presented in structured formats (e.g., relational tables, Knowledge Graphs(KGs)/Bases(KBs)), a vast amount of data is still present in an unstructured format. Even some data sources are structured (e.g., DrugBank), they contain several textual descriptions that are not machine-readable. For example, the Drug-Drug-Interactions (DDIs) in DrugBank are represented as a textual description. Representing the DDIs as textual description limits the ability to infer knowledge on top of the DDIs (e.g., grouping drugs that share the same effect or impact). This highlights the need for knowledge extraction over unstructured data in order to represent the data in a structured format. The problem of bridging the gap between unstructured data and structured data has attracted the interest of a vast research community [16, 17, 18]. Representing the unstructured data in a structured or semi-structured format facilitates applying several other tasks like entity alignment, data integration, and related-posts recommendation. In order to apply knowledge extraction over unstructured data, several steps are required. In the first step, *recognition of entities and relations*, entities and relations in a natural language text are recognized. This allows the machine to understand large amounts of unstructured human language. The second step, *linking recognized entities and relations*, involves assigning a unique identifier to the entities and relations recognized in a natural language text. Typically, this unique identifier refers to a resource in a knowledge graph or knowledge base. Many resources in a single KG share the same label, generating ambiguity among resources to be linked to. The third step, *using extracted knowledge*, allows using the extracted knowledge in several tasks, such as unveiling semantically related posts in a corpus (Figure 1.1). All the mentioned steps consist of several challenges (e.g., ambiguity, lack of context, lack of available training data). Therefore, knowledge extraction demands accurate techniques for the mentioned steps.

1.2 Existing Approaches

Research communities have addressed the problem of knowledge extraction from unstructured data. They have proposed a variety of methods that fill the gap between structured and unstructured data [19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31]. Zaman et al. [27] summarizes existing state-of-the-art(SOTA) techniques in the domain of knowledge extraction to highlight their advantages and limitations. Most of the SOTA techniques suffer from the challenges explained in Section 1.3.

In the context of NER and NED, several surveys provide a detailed overview of the advancements of the techniques employed over community-maintained KGs [32, 33]. Various reading lists [34], online forums[2] and Github repositories[3] track the progress in this domain. Initially, web-based NED attempted to use Wikipedia [35] as the underlying knowledge source. The research field has evolved to the point where the SOTA approaches for NER and NED almost matches human-level performance [36]. With the advent of publicly available KGs such as DBpedia, Yago [37], and Freebase [38], the emphasis has shifted to developing tools on top of KGs. The developments in Deep Learning have introduced various models that perform both NER and NED as a single end-to-end step [39, 40]. Relation recognition and linking from unstructured data have been long-standing research field [41, 42, 43, 44, 45]. However, as an independent approach, linking a relation surface form in natural language text to its KG mention is a developing research problem. Several approaches were proposed by the research community [11, 12, 46, 47, 48]. The majority of the SOTA approaches relies on machine learning and deep learning approaches [49, 50]. Machine learning and deep learning approaches, while precise, require high-quality training annotations, which are not widely available for the tasks of NER, NED, RR, and RL [51, 52]. Their performance also drops for short text where not enough context is present. The defined methods of this thesis overcome the limitation of available training data by utilizing a symbolic approach that resorts to a catalog of linguistics and domain-specific rules, and a background knowledge that encodes the knowledge encoded in community-maintained knowledge sources (e.g., Wikidata).

Finding and recommending similar social media posts has attracted considerable attention in the scientific research community [53, 54, 55]. The work in [56] suggests a fuzzy inference system for categorizing tweets. Several other approaches, such as [54, 57, 58], rely on hashtag-based or user interaction history for tweet recommendations. These methods primarily focused on hashtag similarities. However, they also rely on detecting communities in social networks based on hashtags, followers, mentions, and topics. Besides social media, finding similar sentences in a document or a web article is a well-studied research domain. Kiros et al. [59] trains an encoder-decoder model to predict surrounding sentences of an encoded passage in a given document. Sentence-BERT [60] is a BERT [61] based model that generate semantically meaningful sentence embeddings that can be compared using cosine-similarity. Besides accurate the SOTA approaches, the majority require large amount of training data which is not always available in all the domains (e.g., biomedical domain) or emerging pandemics (e.g., COVID-

[2] `http://nlpprogress.com/english/entity_linking.html`

[3] `https://github.com/sebastianruder/NLP-progress/blob/master/english/entity_linking.md`

19). However, our defined framework is agnostic to training data for a specific domain and resorts to the backgound knowledge captured in publicly available KGs. Hence, it can easily adapt to varied domains without any training data.

1.3 Problem Statement and Challenges

For applying knowledge extraction over unstructured data, several tasks need to be applied. Moreover, surface forms of entities and relations in a natural language text need to be identified in order to be linked to a specific knowledge graph or knowledge base. This thesis aims at extracting knowledge from unstructured data and managing the extracted knowledge in a structured form. The following research problems are addressed; **1)** *recognizing entities and relations* in a natural language text; **2)** *linking entities and relations* to a knowledge graph or knowledge base; and **3)** *discovering patterns in the extracted knowledge*. To address the above mentioned problems, this thesis identifies several challenges in terms of knowledge extraction and management.

Challenge 1: Recognizing Entities and Relations in a Natural Language Text

Correctly identifying entities and relations in a natural language text involves several challenges. Complicated terms (medicine terms) are one of the challenges. For example, the term *chenodeoxycholic acid* represents an organic chemical in UMLS. The surface form of the term consists of two terms; *chenodeoxycholic* and *acid*. Thus, the challenge here is to recognize these two terms as one term *chenodeoxycholic acid* or two separated terms. Ambiguity and abbreviations are other challenges in identifying entities and relations in a language. For example, the surface form *vice president* can be recognized as an entity or a relation depending on the context of the sentence. Multiple words may be written in different ways. Abbreviations can be used to facilitate writing and comprehension. The same words can be written in both short and long forms. Words that will sometimes require some label for identification are another major challenge. Foreign terms that are not often used nowadays or rarely heard by many people are another major challenge in this area (e.g., words such as person names, locations). Thus, effective techniques for solving the described challenge are required.

Challenge 2: Linking Entities and Relations to a specific KG or KB

Correctly linking recognized entities and relations to a specific knowledge graph or knowledge base involves several challenges. Name variations is one of the challenges associated with linking. The same entity or relation may appear in different textual representations. These variations originate from acronyms (New York, NY), aliases (New York, Big Apple), or spelling variations and typographical errors (New yokr). Ambiguity without enough context is another major challenge for entity and relation linking. The same surface form of an entity or relation can often refer to many different entities or relations, depending on the context, as many entity names tend to be polysemous (i.e., have multiple meanings). For example, the word Apple could be referring to the company Apple Inc. or the fruit Apple. It is challenging to solve this ambiguity without enough context in the sentence. Another challenge for the task of linking is dark entities. Dark entities are entities that do not exist in any knowledge graph or knowledge base because they are very specific or are related to recent events that are not yet added to any knowledge graph. Thus, it is not possible to link such entities, which may lead to linking to the wrong entities. Therefore, accurate methods for solving the described challenge are demanding.

Challenge 3: Solving Heterogeneity across Unstructured Data

Integrating heterogeneous data sources involves several challenges. Data extraction is the first challenge. Data extraction is the initial step in the process of integrating. However, if data sources have diverse formats, structures, and types, the process might become complicated and time-consuming. In addition, once the data has been extracted, it must be transformed in order to be compatible with the target structure prior to integration. Data integrity is another major challenge in solving heterogeneity across unstructured data. In every data integration technique, data quality is a fundamental consideration. Inadequate data quality can be a compounding issue that impacts the entire integration process. Processing invalid or wrong data can result in inaccurate analytics, which, if integrated, can affect the performance of the systems that are going to use the data (e.g., entity linking approaches). Integrating heterogeneous unstructured data also entails a scalability challenge to solve heterogeneity across several data sources into a unified view [62], which might ultimately result in exponential data volume expansion. Thus, a robust integration technique that solves the mentioned challenges is required.

Challenge 4: Unveiling Related posts in a Corpus of Unstructured data

Several datasets and corpus exist on the web, and many of them contain patterns among the stored data. For example, social media networks consist of hundreds of millions of posts that share different patterns according to diverse perspectives. Typically, the data in such a corpus is presented as a textual description describing the post's content. Such a textual description is more human-readable other than machine-readable, making the task of unveiling related patterns over hundreds of millions of posts challenging. Moreover, the accuracy of discovering related patterns is also essential and affects the performance of the systems that utilize it (e.g., Tweets Recommender Systems). Another challenging aspect is the size of the studied corpus (hundreds of millions). Therefore, an efficient approach is required to ensure scalability, low execution time, and accuracy.

1.4 Research Questions

RQ1: What is the impact of the linguistics rules of a natural language on the tasks of knowledge extraction?

To answer this research question, we propose a catalog of linguistic rules. The rules are used in a symbolic approach for knowledge extraction over unstructured data, more specifically for the tasks of named-entity recognition and relation recognition. We evaluate the efficiency of using linguistics rules for the mentioned tasks in comparison to the state-of-the-art approaches. Furthermore, the effectiveness of using the defined linguistics rules is assessed by applying an ablation study for the defined symbolic approach. In the ablation study, we applied the same symbolic approach with and without the rules. The experimental results show that using the defined linguistic rules empowers the symbolic approach in recognizing surface forms (a.k.a mentions) of entities and relations in a natural language text effectively. Moreover, using the defined rules makes the symbolic approach independent of any further training data.

RQ2: How does the knowledge represented in community-maintained KGs and domain-specific KBs can be utilized for knowledge extraction tasks?

To answer this research question, we propose a deductive database [63] (background knowledge) built on top of community-maintained KGs and domain-specific KBs. We devise an alignment representation of the knowledge represented in the community-maintained KGs and domain-specific KBs. We exploit different properties of the resources in the used knowledge sources (e.g., labels or semantic types).

The deductive database consists of extensional and intensional databases. Furthermore, we evaluate the quality of the extracted knowledge by applying an ablation study where we use the different community-maintained KGs or domain-specific KBs individually or combined. The results show that the knowledge represented in community-maintained KGs and domain-specific KBs empowers knowledge extraction approaches by the knowledge encoded in these knowledge sources. Moreover, the results suggest that the choice of the underlying knowledge source depends on the domain of the studied unstructured data (e.g., the biomedical domain). The devised background knowledge is also utilized in a neuro-symbolic approach used for NER and NED. The sub-symbolic component of the neuro-symbolic approach resorts to the defined background knowledge to predict the semantic type of an entity in order to improve named-entity disambiguation.

RQ3: How contextual knowledge can be used to enhance knowledge extraction over unstructured data?

To answer this research question, we propose a knowledge-driven framework that captures the knowledge encoded in the context of a natural language text in order to discover semantically related posts in a corpus. The framework utilizes the defined approaches of this thesis; the symbolic and neuro-symbolic approaches that resort to the catalog of linguistic and domain-specific rules and the devised background knowledge. The defined framework annotates the corpus with entities and relations from existing knowledge graphs and knowledge bases (e.g., DBpedia, Wikidata, and UMLS). The annotations are used to build a bipartite graph in order to discover communities of related posts in the studied corpus. We evaluate the quality of the extracted knowledge using benchmarks from different domains. The results show that contextual knowledge can be used to enhance knowledge extraction over unstructured data. Moreover, the results highlight the importance of correctly recognizing and linking entities and relations in natural language text to improve knowledge extraction over unstructured data.

RQ4: How does the knowledge extracted from unstructured data can be used for real-world applications?

To answer this research question, we applied all the defined methods and techniques of this thesis in use-cases of real-world projects (e.g., iASiS[4], BigMedilytics[5],

[4] `https://project-iasis.eu/`

[5] `https://www.bigmedilytics.eu/`

P4-Lucat[6], CLARIFY[7], ImProVIT[8], CoyPu[9], PLATOON[10], Knowledge4Hubris[11], and K4COVID[12]). The methods and techniques are used to extract knowledge from unstructured data. For example, to link surface forms of terms to their corresponding resources in a KG/KB. The terms are related to the biomedical, energy, and non-specific (general) domains. We evaluate the quality of the extracted knowledge by asking experts to annotate the terms, then comparing it with our results. The results show the generality of the defined methods to the studied domain. Another application is to represent the textual description of Drug-Drug-Interactions in DrugBank in a structured form by identifying the drugs' mentions, effect, and impact in an interaction description, then linking them to their corresponding resources in DrugBank KB. The defined techniques are also used for data integration and entity alignment tasks [64]. Moreover, we utilize the defined framework for unveiling semantically related posts in a corpus to analyze tweets related to people who suffer from the hubris syndrome [65].

1.5 Thesis Overview

Intending to prepare the reader for the rest of the document, we present an overview of the main contributions of this thesis and references to the scientific publications covering this work. The thesis contributions are described below:

1.5.1 Contributions

- **Contribution 1: Catalogs of linguistic and domain-specific rules**. In this thesis, we devise catalogs of linguistic and domain-specific rules that are utilized by the symbolic and neuro-symbolic approaches for knowledge extraction over unstructured data. The linguistic rules state the criteria to recognize entities in a sentence of a particular language. The defined rules are designed to overcome the challenges of complicated terms, ambiguity, abbreviations, and foreign terms. As proof of concept, we have defined the linguistic rules for the English language. The domain-specific rules define

[6] https://p4-lucat.eu/

[7] https://www.clarify2020.eu/

[8] https://www.tib.eu/en/research-development/project-overview/project-summary/improvit

[9] https://coypu.org/

[10] https://platoon-project.eu/

[11] https://www.sgul.ac.uk/about/our-institutes/molecular-and-clinical-sciences/research-sections/neuroscience-research-section/knowledge-for-hubris

[12] https://github.com/SDM-TIB/Knowledge4COVID-19

what an entity is in a particular domain. Moreover, we resort to the properties of the resources in a KG to define entities. Thus, the defined rules are based on the assumption that entities have labels, definitions, and semantic types. Since these KGs can be community-maintained, the same resource may have several values of the same property (e.g., various labels or definitions). Additionally, a resource may have equivalent resources. To understand the impact of the defined rules, we conducted an ablation study where we excluded the domain-specific rules from the catalog of rules. In the ablation study, the performance of knowledge extraction drops when excluding the domain-specific rules. This observation highlights the impact of the defined rules for knowledge extraction tasks over unstructured data, answering the research question **RQ1**.

- **Contribution 2: A Deductive Database (Background knowledge) that encodes knowledge in community-maintained KGs**. In this thesis, we propose a background knowledge that encodes knowledge in community-maintained knowledge graphs. Each resource (entity) in the background knowledge is described by its resourceID, labels, language, semantic type, *sameAs* link to other knowledge graphs, and a heuristic confidence score that counts the number of the shared labels of a resource among different sources or knowledge graphs to overcome the challenge of data heterogeneity. The *sameAs* link empowers the background knowledge by integrating knowledge from other knowledge graphs different from the target knowledge graph (e.g., more synonyms). The heuristic confidence score supports the neuro-symbolic approach in solving ambiguity by assigning a higher score to the resource that shares the same label among more providers. A deductive database defines the background knowledge. The background knowledge consists of extensional and intensional databases. The intensional database inductively defines new properties of the resources. It relies on the Leibniz Inference Rule [66] to align the properties of the equivalent resources (i.e., resources connected via *sameAs*). It is important to highlight that the defined technique of constructing the background knowledge is agnostic of the target knowledge graph and can be applied to any knowledge graph/base that have the required properties to construct the background knowledge. As a proof of concept, we implemented the defined background knowledge over DBpedia, Wikidata, and UMLS knowledge bases. The defined implementations of the background knowledge ensures scalability by the nature of the design of the deductive database and it only includes the necessary information extracted from the mentioned knowledge sources containing over 1,054,011,112 factual statements. Moreover, the defined implementations of the background knowledge allow linking entities and relations in a natural language text to three different knowledge bases (e.g., DBpedia, Wikidata, and UMLS) successfully, answering the research question **RQ2**.

- **Contribution 3: A Symbolic approach for entity and relation recognition**. In this thesis, we devise a symbolic approach for entity and relation recognition. The defined approach effectively identifies entities and relation in a natural language text. The defined approach relies on the defined catalog of linguistic rules to overcome the challenges of recognizing entities and relations in a natural language text. The defined approach performs joint entity and relation recognition by leveraging several fundamental principles of English morphology (e.g., compounding, headword identification). It uses the context of entities for finding relations and does not require training data. It is also important to highlight that the defined approach is agnostic of the studied natural language. For example, it can be utilized for the German language by replacing the English linguistic rules with the corresponding German linguistic rules. Our empirical study using several standard benchmarks and datasets shows that the defined approach outperforms state-of-the-art entity and relation recognition approaches for short text query inventories. These results allow us to answer **RQ1**.

- **Contribution 4: A symbolic approach for entity disambiguation and relation linking**. In this thesis, we devise a symbolic approach for entity disambiguation and relation linking. The defined approach effectively maps entities and relations within a text to their resources in a target knowledge graph or knowledge base. Additionally, it overcomes the challenges of linking entities and relations to a specific knowledge graph or knowledge base using a lightweight linguistic approach relying on the defined background knowledge. The defined approach performs joint entity and relation linking of a short text by utilizing an extended deductive database (background knowledge) created by integrating knowledge from various knowledge sources. It is also important to highlight that the defined approach is agnostic of the target knowledge graph or knowledge base. For example, the defined approach is implemented for different knowledge graphs (e.g., DBpededia, Wikidata) and knowledge bases (e.g., UMLS, DrugBank). The empirical evaluation using different standard benchmarks and datasets for the tasks of entity disambiguation and relation linking shows that the defined approach outperforms state-of-the-art entity and relation linking approaches for short text. These results allow us to answer the research questions **RQ1** and **RQ2**.

- **Contribution 5: A Neuro-symbolic approach for the tasks of NER and NED in encyclopedic and domain-specific domains**. This thesis proposes a neuro-symbolic approach for the tasks of NER and NED in encyclopedic and domain-specific domains; encyclopedic domain refers to a non-specific domain that contains large number of diverse topics, while domain-specific domain focuses on a specific topic (e.g., biomedical domain). The defined approach combines symbolic and sub-symbolic components to overcome the challenges of entity recognition and linking and the limitation of the availability of training data while maintaining the accuracy of recognizing and linking entities. The symbolic components consist of a symbolic entity extraction and linking approach backed by a catalog of linguistics and domain-specific rules, and the defined background knowledge of this thesis. The sub-symbolic component increases the accuracy of the symbolic components by employing the knowledge encoded in the definitions/descriptions of entities in a knowledge graph to overcome the challenges of name variations and ambiguity. Since the sub-symbolic component only depends on the description of entities, the defined neuro-symbolic approach does not need any further training data. We have empirically studied the performance of the defined approach on various benchmarks related to the encyclopedic and biomedical domains. The experiments show that the sub-symbolic component complements the performance of a symbolic system consisting of human-given rules. Moreover, the experiments show that our approach improves the state-of-the-art entity linking approaches' accuracy over different benchmarks. Furthermore, the defined approach works equally well over short text or keywords. The observed results allow us to answer the research questions **RQ1** and **RQ2**.

- **Contribution 6: Context-aware framework for unveiling semantically related posts in a corpus**. In this thesis, we propose a context-aware framework for unveiling semantically related posts in a corpus. The defined framework is a knowledge-driven framework that retrieves associated posts effectively. It implements a two-fold pipeline. First, it encodes, in a graph, a corpus of posts and an input post; posts are annotated with entities for existing KGs and connected based on the similarity of their entities to overcome the challenge of analyzing not machine-readable data. In the decoding phase, the encoded graph is used to discover communities of related posts. We cast the problem of unveiling semantically related posts in a corpus into the Vertex Coloring Problem, where communities of similar posts include the posts annotated with entities colored with the same colors. Applying this problem casting and taking advantage of the scalability of vertex coloring approaches allow the defined framework to overcome the challenge of scalability. Based on results reported in the graph theory [67], the defined framework implements the decoding phase guided by a heuristic-based method that determines relatedness among posts based on contextual knowledge and efficiently groups the most similar posts in the same communities. We empirically evaluated the defined framework over various datasets and compared it with state-of-the-art implementations of the decoding phase. The quality of the generated communities is also analyzed based on multiple metrics. The observed outcomes indicate that the defined framework accurately identifies semantically related posts in different contexts. Moreover, the reported results put in perspective the impact of known properties about the optimality of existing heuristics for vertex graph coloring and their implications on the defined framework scalability, answering the **RQ3**.

- **Contribution 7: Applications of the defined frameworks for knowledge extraction in real-world applications**. All the previous contributions of this thesis are used in use-cases of real-world projects. The defined frameworks are used for knowledge extraction over unstructured data. The symbolic approaches [68, 69] Falcon[13] and Falcon2[14] are being used by the community for the tasks of NER, NED, RR, and RL using their public APIs. Both APIs are being heavily used; 4,119,179 hits since April 2019 for Falcon and 5,664,204 hits since February 2020 for Falcon2. Both approaches were used in life science projects (e.g., iASiS, BigMedilytics, P4-Lucat, CLARIFY, ImProVIT, Knowledge4Hubris, and K4COVID [70]) to extract mentions of medical terms from a natural language text then link them to a domain-specific knowledge base like UMLS (e.g., drugs and comorbidities from doctors notes describing patients treatments). The symbolic approach and the background knowledge defined in this thesis are also implemented for a specific domain (energy domain)[15] where the goal is to identify entities in a natural language text and link them to a specific semantic data model described by experts in the EU project PLATOON [71, 72]. In the CoyPu project, the defined frameworks of this thesis are also used to extract knowledge from various datasets related to diverse domains. In the project Knowledge4Hubris, the defined framework for unveiling semantically related posts in a corpus is utilized to analyze tweets related to people who suffer from hubris disease. The experts in this project are interested in identifying people who suffer from hubris based on their way of writing posts and found it challenging to analyze hundreds of millions of tweets. The defined methods of this thesis are also used as components of knowledge graph creation and Semantic data integration pipelines [64, 73, 74]. All the mentioned applications of the defined contributions of this thesis allow us to answer the research question **RQ4**.

[13] `https://labs.tib.eu/falcon/`

[14] `https://labs.tib.eu/falcon/falcon2/`

[15] `https://labs.tib.eu/sdm/efalcon/`

1.5.2 List of Publications

This thesis is based on the following publications.

Peer-Reviewed International Journals

- José Alberto Benítez-Andrades, María Teresa García-Ordás, Mayra Russo, **Ahmad Sakor**, Luis Daniel Fernandes Rotger, Maria-Esther Vidal. *Empowering Machine Learning Models with Contextual Knowledge for Enhancing the Detection of Eating Disorders in Social Media Posts.* In: the Semantic Web journal (2022). This paper is a collaboration with researchers from the Universidad de León, Mayra Russo, a Ph.D. student at the Leibniz University of Hannover, and my supervisor, Prof. Dr. Maria-Esther Vidal. My contribution to this paper is using the techniques defined in this thesis to extract knowledge from social media posts related to the biomedical domain. ***Full research paper***

- Fotis Aisopos, Samaneh Jozashoori, Emetis Niazmand, Disha Purohit, Ariam Rivas, **Ahmad Sakor**, Enrique Iglesias, Dimitrios Vogiatzis, Ernestina Mena salvas, Alejandro Rodriguez Gonzalez, Guillermo Vigueras, Daniel Gomez-Bravo, Maria Torrente, Roberto Hernández López, Mariano Provencio Pulla, Athanasios Dalianis, Anna Triantafillou, Georgios Paliouras,Maria-Esther Vidal. *Knowledge Graphs for Enhancing Transparency in Health Data Ecosystems.* In: the Semantic Web journal (2022). This article is a collaboration with a group of Ph.D. students at the Leibniz University of Hannover, the partners of the EU project that I worked at, named BigMedylitics, and my supervisor, Prof. Dr. Maria-Esther Vidal. My contributions are using the defined techniques of this thesis to extract knowledge from unstructured data related to the topic Lung Cancer in order to create a KG, and designing and implementing knowledge graph exploration APIs over the created KG. ***Full research paper***

- **Ahmad Sakor**, Kuldeep Singh, Maria-Esther Vidal. *Resorting to Context-aware Background Knowledge for Unveiling Semantically Related Posts.* In: IEEE Access (2022). This article is a collaboration with Dr. Kuldeep Singh, a researcher at Zerotha research, and my supervisor, Prof. Dr. Maria-Esther Vidal. My contributions are the definition of the problem, defining the algorithm of the framework, defining the architecture of the framework, the implementation, and the evaluation of the defined framework. I also set up the testbeds and configurations for the evaluation study and performed them, describing the results and visualizing them. ***Full research paper***

- **Ahmad Sakor**, Samaneh Jozashoori, Emetis Niazmand, Ariam Rivas, Kosta ntinos Bougiatiotis, Fotis Aisopos, Enrique Iglesias, Philipp D Rohde, Trupti Padiya, Anastasia Krithara, Georgios Paliouras, Maria-Esther Vidal. *Knowledge4COVID-19: A semantic-based approach for constructing a COVID -19 related knowledge graph from various sources and analyzing treatments' toxicities.* In: Journal of Web Semantics (2022). This article is a collaboration with a group of Ph.D. students at the Leibniz University of Hannover, a group of researchers at Demokritos Institute of Greece, and my supervisor, Prof. Dr. Maria-Esther Vidal. My contributions are using the defined techniques of this thesis to extract knowledge from unstructured data related to the topic COVID-19 in order to create a KG, and designing and implementing knowledge graph exploration APIs over the created KG. ***Full research paper***

- Valentina Janev, Maria-Esther Vidal, Dea Pujić, Dušan Popadić, Enrique Iglesias, **Ahmad Sakor**, Andrej Čampa. *Responsible Knowledge Management in Energy Data Ecosystems.* In: Energies Journal (2022). This article is a collaboration with a group of researchers at University of Belgrade, a researcher at ComSensus company, Andrej Čampa, and my supervisor, Prof. Dr. Maria-Esther Vidal. My contribution is using the defined techniques of this thesis to extract knowledge from unstructured data related to the energy domain. ***Full research paper***

- Maria-Esther Vidal, Kemele M. Endris, Samaneh Jozashoori, **Ahmad Sakor**, Ariam Rivas. *Transforming Heterogeneous Data into Knowledge for Personalized Treatments—A Use Case.* In: Datenbank-Spektrum Journal (2019). This article is a collaboration with group of Ph.D. students at the Leibniz University of Hannover, and my supervisor, Prof. Dr. Maria-Esther Vidal. My contributions are using the defined techniques of this thesis to extract knowledge from unstructured data related to the biomedical domain, and implementing the knowledge graph creation pipeline with the corresponding exploration APIs. ***Full research paper***

Papers in Proceedings of Peer-Reviewed Conferences

- Samaneh Jozashoori, **Ahmad Sakor**, Enrique Iglesias, Maria-Esther Vidal. *EABlock: a declarative entity alignment block for knowledge graph creation pipelines.* In: Proceedings of the 37th ACM/SIGAPP Symposium On Applied Computing (2022). This paper is a joint work with Samaneh Jozashoori, a Ph.D. student at the Leibniz University of Hannover, and my supervisor, Prof. Dr. Maria-Esther Vidal. My contribution to this paper is using the defined techniques of this thesis to assist entity alignment for knowledge graph creation pipelines. ***Full research paper***

- Dušan Popadić, Enrique Iglesias, **Ahmad Sakor**, Valentina Janev, Maria-Esther Vidal. *Towards a Solution for an Energy Knowledge Graph.* In: ISIC 2022 International Semantic Intelligence Conference (2022). **Best paper award**. This paper is a collaboration with a group of researchers at the University of Belgrade, Enrique Iglesias, a research software developer, and my supervisor, Prof. Dr. Maria-Esther Vidal. In this paper, I contributed to the knowledge graph creation pipeline with the corresponding exploration APIs. ***Full research paper***

- Maria-Esther Vidal, Samaneh Jozashoori, **Ahmad Sakor**. *Semantic Data Integration Techniques for Transforming Big Biomedical Data into Actionable Knowledge.* 2019 IEEE 32nd International Symposium on Computer-Based Medical Systems (CBMS). This paper is a joint work with Samaneh Jozashoori, a Ph.D. student at the Leibniz University of Hannover, and my supervisor, Prof. Dr. Maria-Esther Vidal. My contributions are using the defined techniques of this thesis to extract knowledge from unstructured data related to the biomedical domain, and implementing the knowledge graph creation pipeline with the corresponding exploration APIs. ***Short research paper***

- **Ahmad Sakor**, Kuldeep Singh, Maria-Esther Vidal. *Falcon: An entity and relation linking framework over dbpedia.* In: ISWC2019, Demo track, CEUR Workshop Proceedings (2019). This paper is a joint work with Dr. Kuldeep Singh, a researcher at Zeroth research, and my supervisor, Prof. Dr. Maria-Esther Vidal. This paper is a collaboration with Kuldeep Singh, a researcher at Zerodha research, and my supervisor, Maria-Esther Vidal. My contributions are the designing and implementation of the demo, and evaluation of the defined demo. ***Demo paper***

- **Ahmad Sakor**, Kuldeep Singh, Anery Patel, Maria-Esther Vidal. *Falcon 2.0: An Entity and Relation Linking Tool over Wikidata.* In: CIKM '20: Proceedings of the 29th ACM International Conference on Information & Knowledge Management. This paper is a collaboration with Anery Patel, an intern student at the Leibniz University of Hannover, Dr. Kuldeep Singh, a researcher at Zeroth research, and my supervisor, Prof. Dr. Maria-Esther Vidal. My contributions are defining the architecture of the resource, implementation of the resource and the public API, and evaluation of the defined resource. I also set up the testbeds and configurations for the evaluation study and performed them. ***Resource paper***

- **Ahmad Sakor**, Isaiah Onando Mulang', Kuldeep Singh, Saeedeh Shekarpour, Maria-Esther Vidal, Jens Lehmann, Sören Auer. *Old is Gold: Linguistic Driven Approach for Entity and Relation Linking of Short Text.* In: NAACL 2019 Annual Conference of the North American Chapter of the Association for Computational Linguistics (2019). This paper is a collaboration with a group of researchers at the University of Bonn, Saeedeh Shekarpour, a researcher at the University of Dayton, Prof. Dr. Sören Auer, a professor at the Leibniz University of Hannover, and my supervisor, Prof. Dr. Maria-Esther Vidal. My contributions are the definition of the problem, defining the algorithm of the approach, defining the architecture of the approach, the implementation, and the evaluation of the defined framework. I also set up the testbeds and configurations for the evaluation study and performed them. ***Full research paper***

Under-Review

- **Ahmad Sakor**, Kuldeep Singh, Maria-Esther Vidal. *A Neuro-Symbolic Approach for Light-Weight Biomedical Entity Linking.* In: Anonymous submission. This paper is a joint work with Dr. Kuldeep Singh, a researcher at Zeroth research, and my supervisor, Prof. Dr. Maria-Esther Vidal. My contributions are the definition of the problem, providing the motivating example, defining the algorithm of the approach, defining the architecture of the approach, designing and formalizing the rules, implementation, and evaluation of the defined approach. I also set up the testbeds and configurations for the evaluation study and performed them, describing the results and visualizing them. ***Full research paper***

1.6 Thesis Structure

The rest of the thesis is structured as follows: Chapter 2 introduces the basic concepts in the fields of Knowledge Extraction, Natural Language Processing, Semantic Web, Knowledge Graphs, Community Detection algorithms, and Graph Coloring algorithms that are necessary to understand the work presented in the thesis. Chapter 3 discusses the state-of-the-art research work related to this thesis. The related approaches are categorized under three topics. First, we discuss state-of-the-art NER, NED, RR, and RL approaches. Then, we present the existing approaches for neuro-symbolic approaches designed for the tasks of knowledge extraction. Finally, we present the existing techniques for unveiling semantically related posts in a corpus. Furthermore, the approaches exploiting community detection and graph coloring algorithms as they are parts of the defined framework for unveiling semantically related posts in a corpus. Chapter 4 presents the defined symbolic approach for knowledge extraction over unstructured data. The defined approach's details and how it was implemented are described in this chapter. We present a detailed analysis of the defined techniques for knowledge extraction over the state-of-the-art benchmarks. The results show that our defined approach outperforms the state-of-the-art approaches without requiring training data. Furthermore, we discuss the success cases and the limitations of the defined approach. Chapter 5 presents the defined neuro-symbolic approach for efficient knowledge extraction on several domains, we describe the symbolic and sub-symbolic components of the defined approach in details. Moreover, the background knowledge, catalogs of linguistics, and domain-specific rules are described using safe horn clauses. The empirical evaluation using existing benchmarks shows that the defined approach is able to extract knowledge from unstructured data on different domains efficiently. Furthermore, we applied an ablation study to understand the behavior of each component of the neuro-symbolic approach, where we concluded that the sub-symbolic component complements the performance of a symbolic system consisting of human-given rule templates. In Chapter 6, we present a knowledge-driven framework that retrieves associated posts effectively; it implements a two-fold pipeline that consists of encoding and decoding phase. The experimental evaluation shows that the framework is able to accurately identify semantically related posts in different contexts (e.g., biomedical and sports domains). Chapter 7 describes the real-world applications of the defined contributions of this thesis. Moreover, in this chapter, we describe the characteristics of each use-case where our contributions are applied, highlighting the challenges of each use-case. Finally, Chapter 8 concludes the work presented in this thesis and discusses the limitations of the work. Moreover, it proposes some future work in related research areas.

1.7 Summary

Representing the unstructured data available on the web as structured data allows for the development of a wide variety of applications. Data can be searched, managed, and analyzed. For making data availability a reality, data produced by billions of devices or contributors need to be represented as structured data in order to allow various applications to query the data as they require. In real-world scenarios, connecting the knowledge encoded in unstructured data to existing knowledge in KGs/KBs is essential. Keeping the data as unstructured data limits the availability and connectivity to other existing data, preventing inferring knowledge on top of the unstructured data. Thus, an efficient knowledge extraction and management over unstructured data is required. Moreover, an efficient representation of unstructured data as machine-readable data is required in order to unveil related patterns in the data. These challenges, imposed by the increasing amount of generated data daily, need to be addressed in order to provide salable and efficient solutions for knowledge extraction over unstructured data. This thesis presents techniques for knowledge extraction over unstructured data based on symbolic, neuro-symbolic, and knowledge-driven approaches. The empirical evaluation results show that the defined techniques are able to effectively extract knowledge over unstructured data. Furthermore, the extracted knowledge is exploited to create a deductive database that enhance knowledge extraction approaches (the defined background knowledge). The experimental evaluation shows that the defined background knowledge improves the performance of NER, NER, RR, and RL tasks. In addition, we devise a knowledge-driven framework that retrieves semantically related posts in a corpus effectively. We empirically evaluate the results of the framework on general and domain-specific benchmarks. The experimental results suggest that the framework allows unveiling semantically related posts in corpus for recent topics where no enough training data is available (e.g., COVID-19). The findings of this thesis indicate that the defined techniques are able to improve knowledge extraction tasks over unstructured data by efficiently extracting knowledge on top of the data and linking it to existing knowledge graphs or knowledge bases.

Chapter 2

Background

In this chapter, we present the basic concepts and theoretical foundations for the research conducted in this thesis. In Section 2.1, the basic concepts for understanding the semantic web, its components, and applications are described. In Section 2.2, we explain the knowledge representation models used in this thesis and their related applications. This includes horn clauses and deductive databases that consists of extensional and intensional databases. In Section 2.3, we explore natural language processing and its relevant tasks. Furthermore, the tasks involved in knowledge extraction targeting a knowledge graph or knowledge base is exploited. Section 2.4 presents the AI approaches used for knowledge extraction. Moreover, we highlight the role of rule-based AI systems in this domain and compare it with machine learning approaches. In Section 2.5, we exploit the knowledge discovery task including community detection and vertex graph coloring algorithms employed for this task.

2.1 The Semantic Web

The Semantic web can be defined as the other version of the current web in which the data are structured to give a meaning for the content of the web [75]. The semantic web is standardized to help the exchange of data in a format set by the WorldWideWeb Consortium (W3C). Figure 2.1 illustrates the semantic web technology stack and the hierarchy used to present user interface and applications. The semantic web aim is to connect data from different sources which are not necessarily sharing data with other sources. The standardized format of the semantic web helps achieve integrating and combining data collected from various sources. The Semantic web also aims to give machines the ability of understanding the hyperlinked information. Hypertext Markup Language (HTML) is a use case of the semantic web – the language used to design web pages. HTML contains text

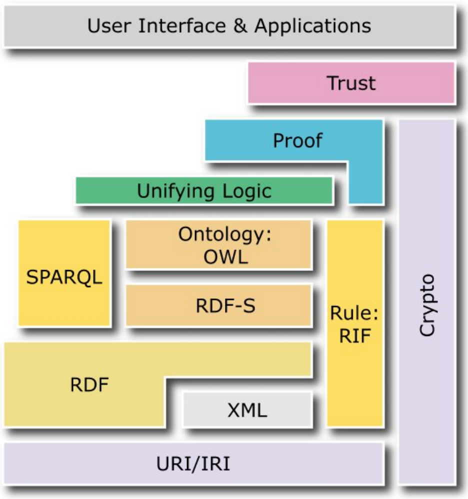


Figure 2.1: **The Semantic Web Technology Stack**, this stack shows how RDF plays a key role as the base many building blocks of the technology [76].

and multimedia elements in the form of tags which describe the layout and the content of the web page. The tags of the HTML language do not give any semantic description about the data. The semantic web provides semantic information for the HTML tags by using description languages like the Resource Description Framework (RDF) and the Web Ontology Language (OWL). The most common format used in semantic web is the Resource Description Format (RDF).

2.1.1 Linked Data

In order to publish structured data on the web, there are a set of guidelines[1] that must be followed:"

1. Use URIs as names for things.
2. Use HTTP URIs so that people can look up those names.
3. When someone looks up a URI, provide useful information, using the standards like RDF, RDFS, OWL ontologies, SPARQL.
4. Include links to other URI's so that a person or machine can discover more things."

[1] `https://www.w3.org/DesignIssues/LinkedData`

By following these guidelines, data from various sources are linked using typed relations just like the case of hyperlinks that link the HTML documents in the traditional web.

Linked data that has been released under an open license, which means it can be reused freely is called Linked Open Data (LOD). An example of the linked open dataset is DBpedia [77]. Figure 2.2[2] shows an example of all datasets that were published as linked data for the linked LOD community project; it includes more than 1,200 datasets in total.

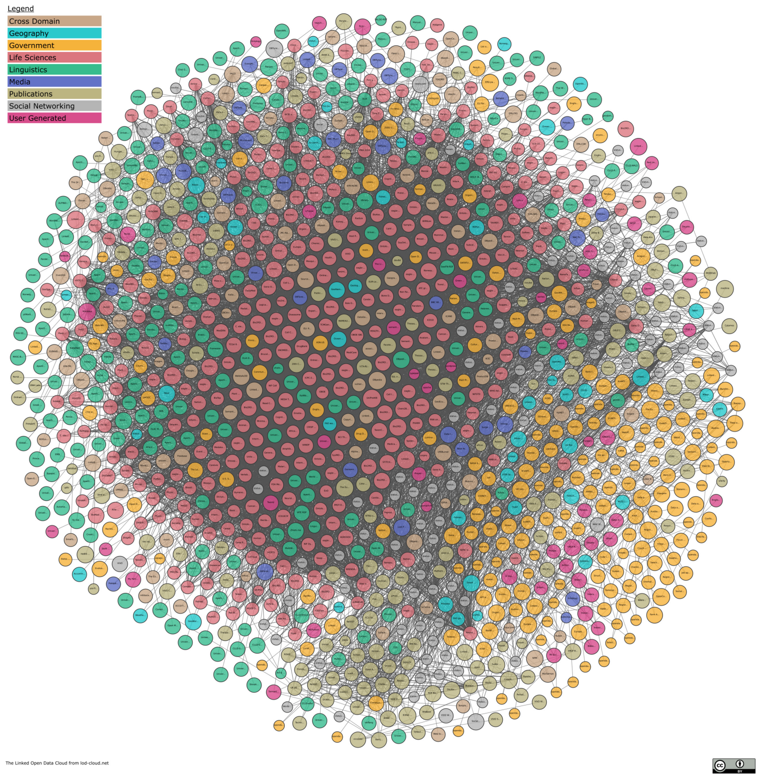


Figure 2.2: The Linked Open Data Cloud as per November 2022

2.1.2 Ontologies

An ontology is a formal representation of properties, categories, and relations between concepts. A concept may be an object, e.g., person, country, or city or it may be a relation like married, death place, etc. It depends on what the ontology is describing. The difference between Resource Description Framework (RDF) and ontology is that RDF describes relations between entities whereas ontology

[2] `http://lod-cloud.net/`

describes rules at the class level. For example, the phrase Men marry women is part of an ontology because it explains a general fact. On the other hand, Barack Obama marries Michelle Obama can be an RDF triple that is compatible with this ontology. A popular example of an ontology is the Basic Geonames[3] Vocabulary; it describes the format for storing records of places (e.g., London) as RDF data. Listing 2.1 is an example of a simple document stored according to the Geonames ontology.

Listing 2.1: Geonames Document Example

```
<gn:name>London</gn:name>
<gn:featureClass rdf:resource="http://www.geonames.org/ontology#P"/>
<gn:featureCode rdf:resource="http://www.geonames.org/ontology#P.PPLC"/>
<gn:countryCode>GB</gn:countryCode>
<gn:population>7556900</gn:population>
<wgs84_pos:lat>51.50853</wgs84_pos:lat>
<wgs84_pos:long>-0.12574</wgs84_pos:long>
```

2.1.3 Knowledge Bases and Knowledge Graphs

A knowledge base (KB) is a technology used by computer systems to store structured and unstructured information. A knowledge base may contain information about a specific domain or may contain data from all across the internet. A knowledge base that integrates data using a graph-structured data model or topology is known as a knowledge graph. Knowledge graphs are commonly used to store interlinked descriptions of entities, events, or abstract concepts, as well as to encode the semantics behind the terminology that is utilized. In 2012, Google used the term Knowledge Graph for the first time [78]. Paulheim [1] describes the features of knowledge graphs as follows:

A knowledge graph:

- mainly describes real world entities and their interrelations, organised in a structured graph;
- defines possible classes and relations of entities in a schema;
- allows for potentially interrelating arbitrary entities with each other;
- covers various topical domains.

Knowledge graphs have been created using a variety of computer science and mathematics principles. The Semantic web promoted the development of KGs

[3] http://www.geonames.org/ontology/

by providing a simple mechanism to organise data in graph-like structures using basic forms known as triples to express a single piece of factual information. Over a decade of research, the semantic web has made available several tools for building and processing information, hence facilitating research on the creation and usage of knowledge graphs. Similarly, a knowledge graph preserves the mathematical concepts underlying a graph and the operation it extends.

The most common KGs are DBpedia [77], YAGO [79] and Wikidata [14]. DBpedia is a knowledge graph of structured data extracted from Wikipedia. Figure 2.3 describes the extraction framework. The structured information contains images, geo-coordinates, external links, etc. OpenLink Virtuoso[4] is the host for RDF dataset of DBpedia. The data can be accessed either by SPARQL queries or HTTP requests. DBpedia contains 3 billion RDF triples which were extracted from 580 million pages from the English version of Wikipedia, and 2.46 billion RDF triples were extracted from other language versions[5]. Another Great Ontology (YAGO) is also a knowledge graph like DBpedia, but the data in YAGO are extracted from Wikipedia, Wordnet [80], and GeoNames [81]. YAGO has more than 10 million entities and more than 120 million facts about these entities[6]. By using the taxonomy of WordNet with Wikipedia categorization, YAGO contains more than 350,000 classes. YAGO is preferred to be used when it comes to classes and categories of entities, due to the massive number of classes. Wikidata [14] is a knowledge graph that include knowledge harvested from the web and curated by the crowd. Thus, Wikidata provides a rich context composed of encyclopedic and factual knowledge represented in the form of KG triples. Wikidata allows users to edit Wikidata pages directly, add newer entities, and define new relations between the objects. Wikidata is hugely popular as a crowdsourced collection of knowledge. Since its launch in 2012, over 1.7 billion edits have been made by the users across the world[7].

2.1.4 Resource Description Framework

Resource Description Framework (RDF) is a data model that represent triples as factual statements. RDF is used to describe resources like people, objects, concepts, etc., which helps to understand the semantics of the data on the web. The way of storing the data in RDF format came from the graph theory, which is a set of vertices (entities) connected by a set of edges (connections between entities).

RDF statement consists of three parts: the subject, the predicate, and the

[4] `https://virtuoso.openlinksw.com/`

[5] `https://wiki.dbpedia.org/about`

[6] `https://www.mpi-inf.mpg.de/departments/databases-and-information-systems/research/yago-naga/yago/`

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

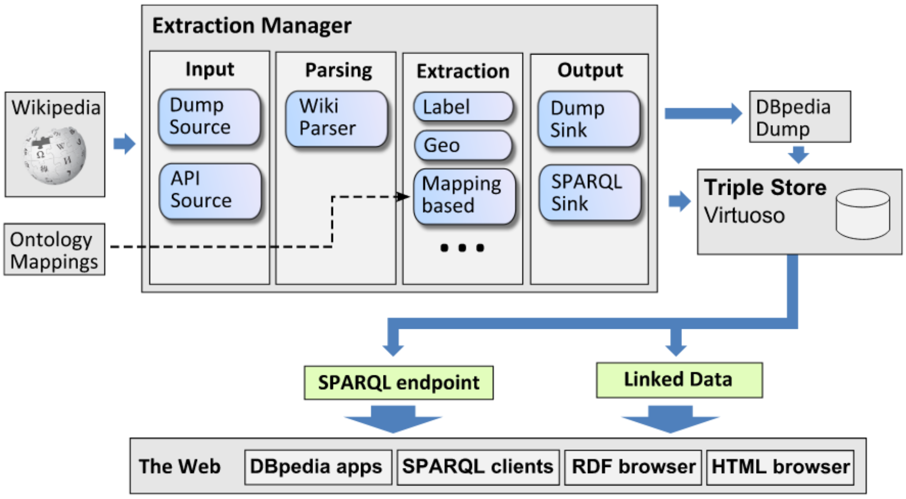


Figure 2.3: DBpedia Extraction Framework [77]

object. These three parts are called RDF triple. Figure 2.4 shows an RDF triple in the form of graph data.

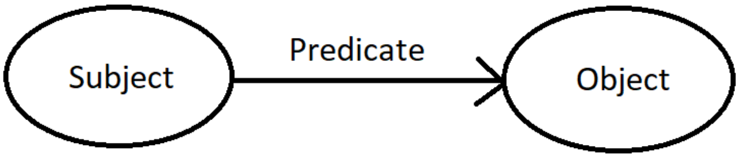


Figure 2.4: An RDF Graph for a Triple

The subject and object are the resources that are connected by the predicate. Predicates describe the relationship between two resources, e.g., deathPlace, spouse. Listing 2.2 shows an example of triples. From Listing 2.2, we can observe that a resource can be part of more than one triple.

Listing 2.2: Triples Example

```
<Barack Obama> <birthPlace> <Hawaii>
<Barak Obama> <vicePresident> <Joe Biden>
```

Every RDF resource is identified by an International Resource Identifier (IRI). For example, DBpedia uses IRI's of the form "http://dbpedia.org/resource/name" to indicate the thing described by the corresponding Wikipedia article.

Resource Description Framework Schema. The Resource Description Framework Schema (RDFS) is used to describe RDF vocabularies. RDFS contains core classes and core properties.

The Core classes of RDF Schema vocabulary contains the following resources:

1. **rdfs:Resource**: resources are all the things described by RDF expression, and they are considered as instances of the class rdfs:Resource.

2. **rdf:Property**: rdf:Property expresses the subset of RDF resources that are properties.

3. **rdfs:Class**: rdfs:Class describes a type or category. It is like the notion Class in object-oriented-programming languages. If we want to define a new class, the resource type representing that class must be rdfs:Class, which is achieved by assigning a property rdf:type that has the value rdfs:Class. RDF classes can be used to define almost anything. Such as people, places, and databases.

RDFS models include the following core properties:

1. **rdf:type**: when a resource has a property rdf:type, this means that the resource is a member of a class. The value of rdf:type may be a specific class, then the resource is called an instance of the specified class. Otherwise, if the value of rdf:type is another resource, then this other resource must be an instance of rdfs:Class. The resource rdfs:Class is itself a resource of rdf:type rdf:Class. A resource may be an instance of more than one class.

2. **rdfs:subClassOf**: rdfs:subClassOf is used to describe a subset or superset relation between classes. rdfs:subClassOf has the feature of transitivity. So if A is a subclass of B and B is a subclass of C, then A is a subclass of C. In this case resources that are instances of class A will also be instances of class C.

3. **rdfs:subPropertyOf**: the property rdfs:subPropertyOf is an instance of rdf:Property. A property may be a subproperty of zero, one, or more properties. If a property P2 is a rdfs:subPropertyOf another property P1, and if a resource A has a P2 property with a value of B, we can say that the resource A also has a P1 property with the value B.

4. **rdfs:seeAlso**: if we consider a resource as a subject resource in this property, then if there is another resource which provides additional information about the subject property, the relation between these two resources is the rdfs:seeAlso.

5. **rdfs:isDefinedBy**: rdfs:isDefinedBy is a subproperty of rdfs:seeAlso. The common usage of this property is to identify an RDF schema, by providing a name for one of the classes or properties defined by that schema.

2.1.5 OWL Web Ontology Language

OWL Web Ontology Language is a semantic web language designed to expresses ontologies. OWL is designed by the World Wide Web Consortium. Similarly to RDFS, the OWL language has classes and class extensions. OWL has many concepts, in the following, we list the common concepts of OWL [82]:

- **Class**: group of objects that share the same characteristics
- **Class extension**: all instances of a class
- **Class axioms**: like rdfs:subClassOf, owl:equivalentClass, and owl:disjointWith.
- **Property**: like RDFS, properties describe the class
- **Property extension**: all pairs of subjects and objects that can be connected with the property
- **Property axioms**: like rdfs:subPropertyOf.
- **Instance axioms**: like owl:sameAs.

2.1.6 SPARQL

SPARQL is an RDF query language designed to query and retrieve RDF data [83]; it can be used to find relations between resources, or which resources are related to a relation or another resource. SPARQL query may contain conjunctions, disjunctions, and optional patterns; its query structure is similar to RDF triples. However, it contains variables, which are substituted during the process of getting the result. SPARQL consists of four different query variations for different goals:

- **SELECT** query: raw values returned from a SPARQL endpoint, the format of the result is a table format.
- **CONSTRUCT** query: used to get information from the SPARQL endpoint, the format of the result is valid RDF.
- **ASK** query: Boolean results after hitting the SPARQL endpoint.
- **DESCRIBE** query: used to get RDF graph from the SPARQL endpoint.

Here is a **SELECT** SPARQL query that retrieves information about a person using the foaf ontology.

Listing 2.3 SPARQL query that fetches a person and their title and last name using the friend of a friend vocabulary.

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>.
PREFIX foaf: <http://xmlns.com/foaf/0.1/>.
SELECT ?title ?family_name
WHERE {
    ?person rdf:type  foaf:Person;
            foaf:title ?title ;
            foaf:foaf:family_name ?family_name.

}
```

2.2 Knowledge Representation

Knowledge representation is the subject of representing information about the world in a format that a computer system can utilize to perform complex tasks such as diagnosing a medical condition or having a dialog in a natural language. Knowledge representation includes psychological studies about how humans solve problems and represent knowledge in order to establish formalisms that facilitate the design and construction of complex systems. Horn clauses, deductive databases, extensional databases, and intensional databases are examples of knowledge representation formalisms.

2.2.1 Horn Clauses

A Horn clause is a logical formula of a specific rule-like form in mathematical logic and logic programming that has useful properties for use in logic programming, formal specification, and model theory [63]. Horn clauses are named after logician Alfred Horn, who identified their significance for the first time in 1951. A Horn clause is a clause with at most one positive, (i.e. literal). Horn clauses consist of three main types; definite clause, fact, and goal clause. A Horn clause containing exactly one positive literal is a definite clause or a strict Horn clause; a definite clause including no negative literals is a unit clause, and a unit clause containing no variables is a fact. A Horn clause without a positive literal is a goal clause [63]. In the following is an example of Horn Clauses used in the deductive database devised in this thesis.

```
label(resourceID,label,language,provider,confidenceScore).
definition(resourceID,label,language,provider).
type(resourceID,type).
sameAs(resourceID1,resourceID2).
```

2.2.2 Deductive Database

A deductive database [84] is a database system that may make deductions (i.e., deduce extra facts) based on rules and facts recorded in the database (deductive database).Datalog is the language generally used in deductive databases to define facts, rules, and queries. Deductive databases emerged from the need to merge logic programming with relational databases to create systems that support a sophisticated formalism and are nevertheless capable of handling very big datasets quickly and efficiently. Deductive databases are more expressive than relational databases but not as expressive as logic programming systems. In recent years, Deductive databases, such as Datalog, have seen increased use in emerging fields like data integration, information extraction, networking, program analysis, and cloud computing [85]. A deductive database contains extensional and intensional databases. An extensional database represents the facts stored in a deductive database. While an intensional database, consists of the rules in a deductive database from which new facts may be inferred [86]. In the following is an example of horn clauses used in the intensional database created in this thesis.

```
sameAs(resourceID1,resourceID2),
label(resourceID1,label,language,provider,confidenceScore)=>
label(resourceID2,label,language,provider,confidenceScore)
```

2.3 Natural Language Processing

Natural Language Processing (NLP) is part of Artificial Intelligence (AI). NLP main tasks are making computers understand, process, and manipulate human language. One example of NLP applications is speech recognition. NLP uses different techniques to perform its tasks, some of these techniques are statistical, machine learning, and rules-based algorithms. Standard NLP tasks consist of tokenization, parsing, lemmatization/stemming, language detection, identification of semantic relationships, and part-of-speech tagging [87]. NLP tasks are also used to assist information and knowledge extraction in various domains.

2.3.1 Information Extraction

Information extraction is the task of extracting data from unstructured data sources in order to identify entities, classify them, and store them in a data model. A primary objective of information extraction is to enable computation on unstructured material. A more specific objective is to enable logical reasoning to generate inferences from the logical content of the input data. Structured data is semantically well-defined data from a specified domain, comprehended in terms of category and context. Information extraction is partially connected to natural language processing as it seeks to discover concealed information or offer structure to text written in natural language. Information extraction is a component of a larger puzzle. Typically, it operates in the background of larger fields such as "Information Retrieval" and "Natural Language Processing." [88].

2.3.2 Knowledge Extraction

Knowledge extraction is the technique of extracting information from unstructured and semi-structured data sources. To allow inference, the created knowledge must be presented in a machine-readable format. While it follows a similar methodology to information extraction, in which structured information is extracted from unstructured or semi-structured machine-readable data, the primary difference is that the extraction result must go beyond the creation of structured information or the transformation into a relational schema. Either existing formal information must be reused (e.g., in knowledge graphs or knowledge bases) or a schema must be derived from the source data. In real-world applications, extracting knowledge from a natural language text comprises several tasks; Named-entity recognition (NER), Named-entity Disambiguation (NED), Relation Recognition (RR), and Relation Linking (RL).

Named-entity Recognition. Named-entity recognition (NER) [3], also known as entity identification or entity extraction, is a knowledge extraction technique that recognizes named entities in a natural language text. Entities include things like names of people, organizations, places, time and dates, quantities, monetary values, percentages, etc. Figure 2.5[8] shows an example of named-entity recognition over a natural language text. Using named-entity recognition, the essential knowledge encoded in a natural language text can be extracted in order to deter-

[8] https://www.analyticsvidhya.com/blog/2021/11/a-beginners-introduction-to-ner-named-entity-recognition/

mine what a natural language text is describing. Formally, a given text is a set of tokens $\mathcal{T} = \{t_1, ..., t_n\}$. A text consists of a set of entities E. $\mathcal{P}(\mathcal{T})$ and $\mathcal{P}(\mathcal{E})$ are, respectively, the power set of tokens and the power set of entities. Mapping the set of tokens in a text to the power set of entities $\mathcal{P}(\mathcal{E})$ is defined by the partial function $(p : \mathcal{P}(\mathcal{T}) \rightarrow \mathcal{P}(\mathcal{E}))$. Let ϕ be a function that represents an oracle that correctly associates a set of tokens with entities; $\phi : \mathcal{P}(\mathcal{T}) \rightarrow \mathcal{P}(\mathcal{E})$. Let $acc(.,.)$ be a utility function that assigns a value of accuracy to the linking assigned by $p(.)$ in comparison to the one produced by the oracle $\phi(.)$. The function $p(.)$ should solve the following optimization conditions.

Accuracy of the mappings is maximized

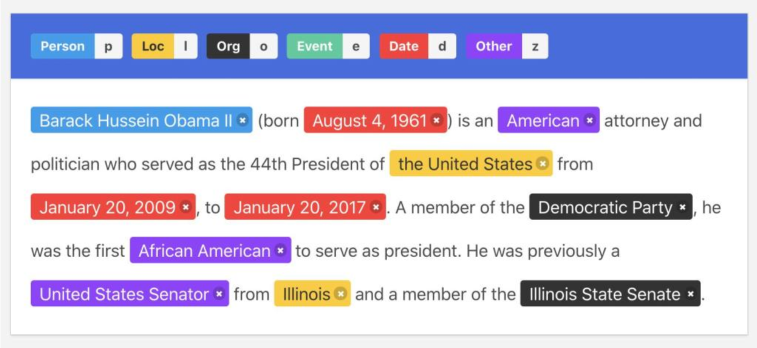
$$\arg \max_{token \in \mathcal{P}(\mathcal{T})} acc(\phi(token), p(token))$$

Maximizing a mapping's number of tokens

$$\arg \max_{\substack{token \in \mathcal{P}(\mathcal{T}) \\ entities \in \mathcal{P}(\mathcal{E}) \\ p(token) = entities}} |token|$$

Minimizing a mapping's number of entities

$$\arg \min_{\substack{token \in \mathcal{P}(\mathcal{T}) \\ entities \in \mathcal{P}(\mathcal{E}) \\ p(token) = entities}} |entities|$$



Barack Hussein Obama II (born August 4, 1961) is an American attorney and politician who served as the 44th President of the United States from January 20, 2009, to January 20, 2017. A member of the Democratic Party, he was the first African American to serve as president. He was previously a United States Senator from Illinois and a member of the Illinois State Senate.

Figure 2.5: An example of Named-Entity Recognition

Relation Recognition. Relation recognition [4], also known as relation extractions is a knowledge extraction method that identify relations in a natural language text. Relations connects existing entities in a natural language text and describe their relationship. Figure 2.6 shows an example of recognizing relations in a natural language text; the relation *grew up* in the sentence *Bill Gates was born on Oct, 28 in 1955 and grew up in Seattle.*, describes the relationship between the entity *Bill Gates* and the entity *Seattle* in the given sentence. Relation recognition allows knowledge extraction techniques to connect the recognized entities and play a major role in question answering systems. Formally, the problem of relation recognition tries to solve the same optimization conditions described for named-entity recognition considering that E is a set of predicates in the text.

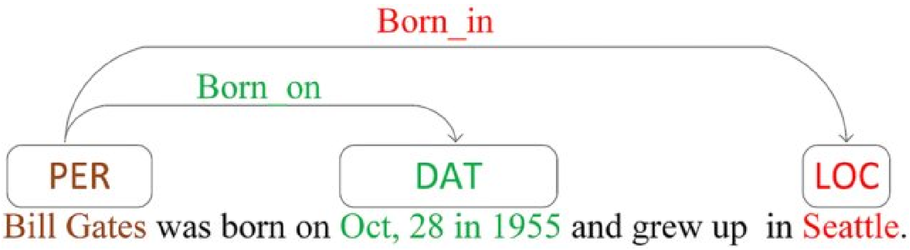

Born_in
Born_on
PER
DAT
LOC
Bill Gates was born on Oct, 28 in 1955 and grew up in Seattle.

Figure 2.6: An example of Relation Recognition [89]

Named-Entity Disambiguation. Named-Entity Disambiguation [2] maps the surface forms (a.k.a mentions) within a text into the textual representation of entities in a knowledge graph. Figure 2.7 shows an example of linking recognized entities in a sentence to Wikipedia; it also highlights the challenge of ambiguity among possible candidates that have similar labels. This mapping follows a particular strategy which is formalized in the following. Formally, a given text is a set of tokens $\mathcal{T} = \{t_1, ..., t_n\}$. A knowledge graph consists of a set of resources U. $\mathcal{P}(\mathcal{T})$ and $\mathcal{P}(\mathcal{U})$ are, respectively, the power set of tokens and the power set of resources. Mapping the set of tokens in a text to the power set of resources $\mathcal{P}(\mathcal{U})$ is defined by the partial function $(p : \mathcal{P}(\mathcal{T}) \rightarrow \mathcal{P}(\mathcal{U}))$. Let ϕ be a function that represents an oracle that correctly associates a set of tokens with resources/entities; $\phi : \mathcal{P}(\mathcal{T}) \rightarrow \mathcal{P}(\mathcal{U})$. Let $acc(.,.)$ be a utility function that assigns a value of accuracy to the linking assigned by $p(.)$ in comparison to the one produced by the oracle $\phi(.)$. The function $p(.)$ should solve the following optimization conditions.

Accuracy of the mappings is maximized

$$\arg \max_{token \in \mathcal{P}(\mathcal{T})} acc(\phi(token), p(token))$$

Maximizing a mapping's number of tokens

$$\underset{\substack{token \in \mathcal{P}(\mathcal{T}) \\ entities \in \mathcal{P}(\mathcal{U}) \\ p(token) = entities}}{\arg\max} |token|$$

Minimizing a mapping's number of entities

$$\underset{\substack{token \in \mathcal{P}(\mathcal{T}) \\ entities \in \mathcal{P}(\mathcal{U}) \\ p(token) = entities}}{\arg\min} |entities|$$

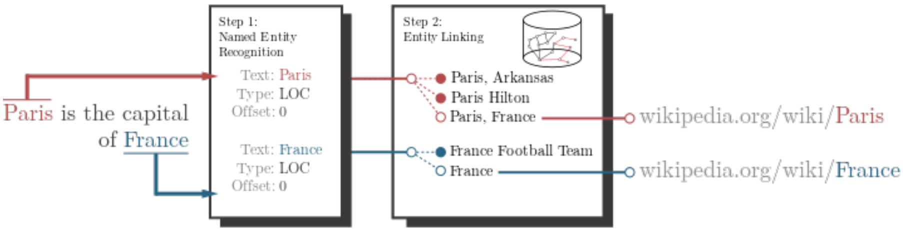
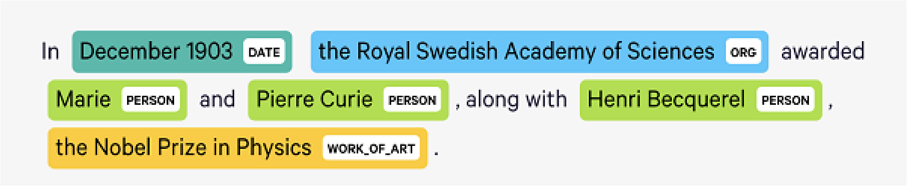


Figure 2.7: An example of Linking Recognized Entities [90]

Relation Linking. Relation linking is about linking surface forms in a natural language text representing a relation to equivalent relations (predicates) of a knowledge graph. Relation linking task complements the task of named-entity disambiguation for knowledge extraction over unstructured data; it allows knowledge extraction methods to connect the linked entities from a natural language text using an existing relation (predicate) in the same knowledge graph used for linking the entities. This allows to map the knowledge encoded in the natural language text to an existing knowledge in a knowledge graph or knowledge base. For example, linking the relation *vice president* in the question *who is the vice president of USA?* to its resource in a knowledge graph (e.g., DBpedia), allows questions answering systems to find an answer to the mentioned question using the predicate *VicePresident*. Formally, the problem of relation linking solves the same optimization conditions described for named-entity disambiguation considering that E is the set of relation in a knowledge graph.

Semantic Type Prediction. Semantic type prediction is the task of predicting the type of an entity with respect to the context of the text where the entity is mentioned. Predicting the correct semantic type of an entity allows to solve the challenge of ambiguity among the candidates of the named-entity disambiguation problem. For example, in the sentence *What is the annual income of Apple ?*, correctly predicting the semantic type (company) of the entity *Apple* assist named-entity disambiguation approaches in excluding the candidate Apple (the fruit) from the list of candidates and keeping the correct candidate Apple Inc (the company). Figure 2.8[9] illustrates how a semantic type prediction method assign a type (class) for each recognized entity.



In December 1903 DATE the Royal Swedish Academy of Sciences ORG awarded Marie PERSON and Pierre Curie PERSON , along with Henri Becquerel PERSON , the Nobel Prize in Physics WORK_OF_ART .

Figure 2.8: An example of Semantic Type Prediction [90]

2.4 AI Approaches for Knowledge Extraction

2.4.1 Rule-Based Systems

A rule-based system (symbolic system) is a system that stores, sorts, and manipulates data using rules created by humans; it simulates human intelligence in this way. A rule-based system is used to store and manage knowledge in order to effectively analyze input. It is frequently utilized in applications and research involving artificial intelligence. Typically, the term rule-based approach refers to approaches that feature human-crafted or curated rule sets. Rule-based systems built employing automatic rule inference, such as neuro-symbolic, are typically omitted from this system category. Rule-based approaches can also be called rule-based AI. A rule-based AI system is a system designed to prompt artificial intelligence through a model that is exclusively based on predefined rules. Such a rule-based AI system consists of a catalog of human-coded rules that produce predetermined outcome. If-then coding statements describe these AI system models. A catalog of rules and a catalog of facts are two essential components of rule-based AI models, and by utilizing them, developers can design a basic artificial intelligence model. In the following is an example of linguistic rules used in this thesis:

[9] https://odsc.medium.com/building-named-entity-recognition-and-relationship-extraction-components-with-huggingface-77d233e27e65

```
Rule 1:  Stopwords are not entities or relations
Rule 2:  Verbs are not entities
Rule 3:  A single compound word comprises words without stopwords
Rule 4:  Entities with only stopwords between them are one entity
```

2.4.2 Machine Learning Approaches

Machine Learning (ML) [91] is a core component of Artificial Intelligence, with numerous applications utilizing ML algorithms to identify patterns in data. The recent expansion in hardware and computational capacity of machines, together with the growing availability of web-based data, have facilitated a substantial increase in the research and applications of ML techniques. Deep Learning [92] is a subfield of ML that has recently attracted considerable interest due to its capacity to extract latent semantic features from data. This behavior permits the learning of representations that can be reused for several knowledge extraction tasks. This section provides an overview of the machine learning techniques relevant to this thesis. Specifically, models of Neural Networks and Transformers.

Neural Networks

Artificial Neural Networks (ANN) [93], also known as Neural Networks (NN), are statistical learning algorithms that discover linear or nonlinear functions from provided data. The Perceptron is the simplest neural network; it is a combining function that receives several real-valued inputs and produces a single output after putting the combined value via an activation function. Given a set of values $(y_1; y_2; y_3)$, for instance, each of these values is multiplied by a learnable weight. Thus, the weight set is provided by: $(w_1; w_2; w_3)$. These weights are the real-valued parameters that characterize the underlying mathematical function that transfers the provided inputs to the desired output value. Typically, a function receives its output via a weighted sum of its inputs. A transfer or activation function is applied to the generated value. The threshold activation (also known as the step function) is utilized by the basic Perceptron, where the neuron's output is adjusted to 1.0 if the sum $(\sum w_i y_i)$ is larger than a real number threshold value, and zero otherwise. The activation function for a perceptron used as a supervised binary classifier is important for guaranteeing that the output is mapped between the required values (0,1) or (-1,1) (Figure 2.9). Moreover, the input's learned weight indicates its strength or contribution to the output's overall value. Likewise, the bias value of an input provides the capacity to adjust the activation function curve up or down. Thus, neural networks can be applied for several knowledge extractions tasks due to its learning technique [94, 95, 96].

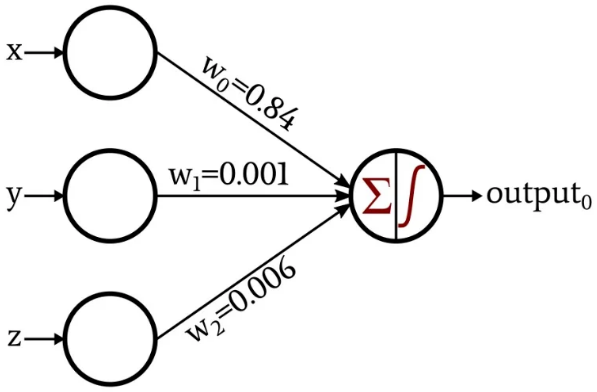


Figure 2.9: An example of training a basic Perceptron Neural Network [97]

Transformers Models

A transformer model [98] is a deep learning model that employs the mechanism of self-attention by differently ranking the importance of each portion of the input data. It is largely utilized in the natural language processing and computer vision research areas. Transformers, similar to recurrent neural networks (RNNs), are meant to interpret sequential input data, such as natural language, and have applications in translation and text summarization. Unlike RNNs, transformers process the full input at once. The attention mechanism gives context for each input sequence location. If the incoming data is a natural language sentence, for instance, the transformer does not need to process each word individually. This permits greater parallelization than RNNs and hence reduces training durations. The architecture of a transformer consists of an encoder and a decoder (Figure 2.10). The encoder is responsible for mapping an input sequence to a sequence of continuous representations, that is later passed to a decoder. The decoder combines the output of the encoder and the output of the decoder at the prior step (iteration) in order to produce an output sequence. The design of transformers models makes it fit for knowledge extraction tasks such as semantic type prediction [99].

2.4.3 Neuro-symbolic Systems

Neuro-symbolic systems, also know as Neuro-symbolic artificial intelligence is an area of research that integrates in its architecture machine learning (sub-symbolic)

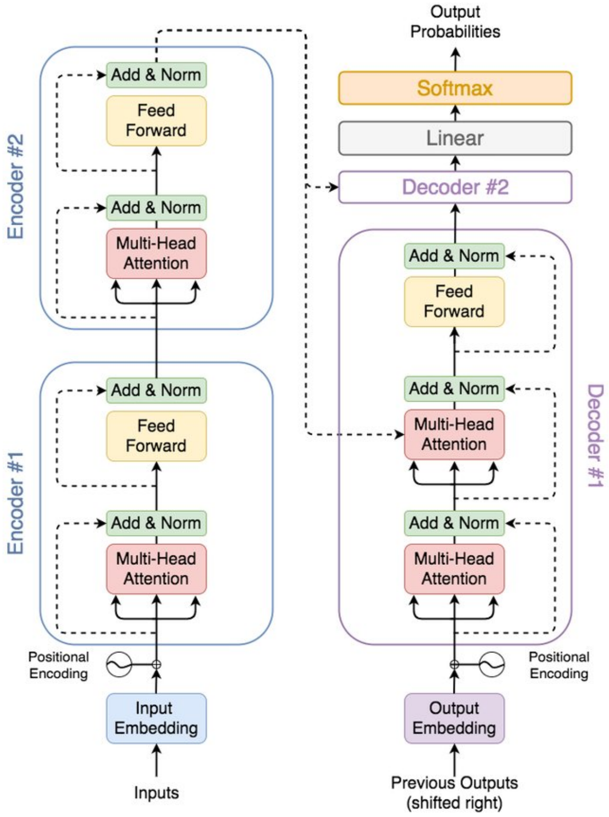


Figure 2.10: An example of a transformer model including an encoding and decoding phase [100]

techniques based on neural networks (e.g., deep learning) with symbolic systems (e.g., rule-based). For example, the field of knowledge extraction and reasoning. Neuro-symbolic AI is not a new research area; however, it remained a specialist field [101] of research until recent breakthroughs in machine learning and deep learning techniques triggered a significant increase in interest from research communities in integrating both symbolic and sub-symbolic techniques [102]. Neuro-symbolic artificial intelligence is an area of AI that integrates sub-symbolic and symbolic systems. Sub-symbolic systems are based on neural networks (e.g., deep learning); it has reported noticeable breakthrough results in the recent decade, and is leading the current general interest in AI [103]. Symbolic systems rely on the explicit representation of knowledge using formal statements (e.g., linguistic rules) to produce its outcomes. Thus, neuro-symbolic systems can leverage the advantages of symbolic and sub-symbolic systems to perform knowledge extraction.

2.4.4 Heuristic Approaches

A heuristic approach [104] is an approach that follows practical methods based on previous experience. However, a heuristic approach does not guarantee to give the optimal solution. A heuristic approach is used to solve problems faster than classic approaches that are known to be slow. Also, it is used to find approximate solutions when the classic methods slip to find a correct solution. Some applications of heuristic approach are travelling salesman problem and virus scanning. A heuristic approach finds solutions faster than classic methods because it depends on the practical experience which leads to correct answers other than the classic slow approaches which also lead to correct results but after a long execution time.

2.4.5 Greedy Algorithms

A greedy algorithm [105] solves the problem by considering the locally optimal choice at each step with the purpose of finding a global optimum. In other words, a greedy algorithm always considers the choice that seems to be the best at that moment. Examples of greedy algorithms are Huffman encoding which its usage is data compression, or Dijkstra's algorithm, which is used to discover the shortest path through a graph. Greedy algorithms do not guarantee the optimal solution because as mentioned before, once it finds a solution it considers the solution as the best solution at that moment. Figure 2.11[10] and Figure 2.12[11] illustrate how greedy algorithms may fail to get the optimal solution.

2.5 Knowledge Discovery

Knowledge discovery [106] is the process of exploring vast amounts of data for patterns that might describe the knowledge encoded in the data; it is commonly defined as deriving knowledge from input data. Knowledge discovery emerged from the field of data mining and is strongly tied to its approach and terminology. One popular application of knowledge discovery is discovering knowledge encoded in graphs [107]. In this thesis, we apply knowledge discovery over different types of graphs (e.g., Line Graphs, Bipartite Graphs, and Complement Line Graphs) by utilizing community detection and vertex graph coloring techniques over the mentioned graphs.

Community Detection in Graphs. Detecting communities [108] within a graph is one of the most major challenge in graph analysis. For instance, there could be millions of nodes and edges in a huge graph reflecting an online social

[10] https://en.wikipedia.org/wiki/File:Greedy_Glouton.svg

[11] https://en.wikipedia.org/wiki/File:Greedy-search-path-example.gif

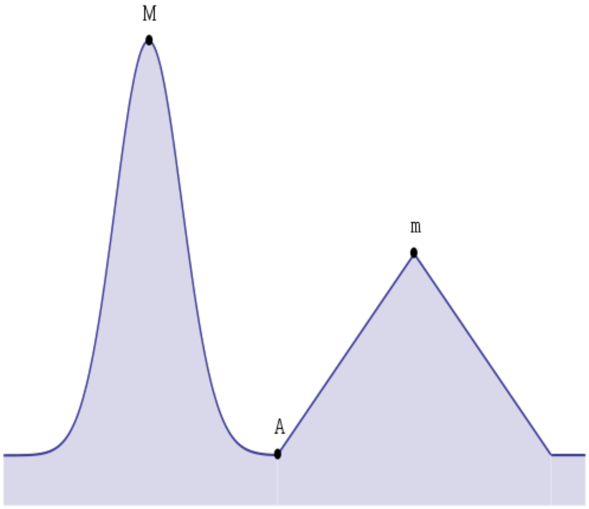


Figure 2.11: Starting at A, a greedy algorithm will find the local maximum at "m", oblivious to the global maximum at "M".

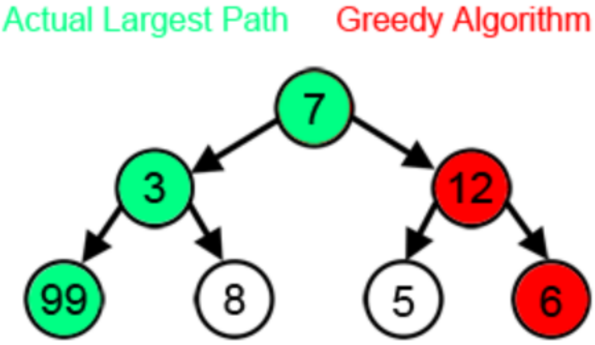


Figure 2.12: Problem of reaching the largest sum

network. Therefore, we require community detection algorithms capable of segmenting the graph into multiple communities. There are basically two types of community detection techniques for graphs; Agglomerative techniques and Divisive techniques. Agglomerative techniques begin with a graph consisting of the original graph's nodes but no edges. Next, edges are added one by one to the graph, beginning with "stronger" edges and progressing to "weaker" edges. This edge's strength or weight can be determined in various ways. Contrariwise, divisive techniques are employed in reverse. They begin with the whole graph and iteratively remove the edges. First, the edge with the highest weight is eliminated. The edge-weight computation is recalculated at each step since the weight of the remaining edges varies when one edge is removed. After a given number of iterations, densely connected node are clustered together. Community detection methods have been used widely for detecting communities in graphs representing social media networks [109, 110, 111].

Line Graphs The line graph [112] of an undirected graph G in graph theory is another graph L(G) that depicts the adjacencies between edges of G. Formally, Given a graph *G=(V,J)* such that $J \subseteq V \times V$, the *line graph LP(G)=(F,T)* of G comprises a) a vertex f_{e_q} in F per each edge e_q in J, and b) an edge (f_{e_q},f_{e_k}) in T if e_q and $e_k \in J$ and share a vertex in common, i.e., the following edges belong to J: e_q=(v_i, v_z) and e_k=(v_j, v_z), or e_q=(v_z, v_i) and e_k=(v_z, v_j).

Bipartite Graphs A bipartite graph [113] is a graph whose vertices are partitioned into two distinct and independent sets U and U‘, such that every edge connects a vertex in U to one in U‘(Figure 2.13). Formally, A *bipartite graph BP=(*$V_1 \cup V_2$*,E)* comprises vertices in $\mathrm{V}_1 \cup \mathrm{V}_2$ and edges are in $E \subseteq \mathrm{V}_1 \times \mathrm{V}_2$; the intersection of V_1 and V_2 is empty, i.e., $\mathrm{V}_1 \cap \mathrm{V}_2 = \emptyset$.

Complement Line Graphs The complement line graph [114] of a graph G is another graph G' that shares the same set of vertices in G but an edge exist between two vertices (v, u) in G', if and only if there is no edge in between (v, u) in G (Figure 2.13). Formally, given a graph *G=(V,J)*, the *complement graph* of *G* is a graph *Comp(G)=(V,K)*, where vertices of *G* and *Comp(G)* are the same, and K is the complement of J, i.e., *K=((V× V)-J)*.

Vertex Graph Coloring The vertex coloring problem [115] corresponds to the coloring of the vertices in a graph *G=(V,J)* with the minimal number of colors such that adjacent vertices are colored with distinct colors. Formally, let $\mathcal{SC}$ be a set of colors and $\mu(.)$ is a mapping from V to $\mathcal{SC}$. The function $\mu(.)$ is a solution to the vertex coloring problem for G if $\mu(.)$ is defined as follows:

- Adjacent vertices are in distinct colors, i.e., if v_i and $v_j \in V$ are adjacent then, $\mu(v_i) \neq \mu(v_j)$.

- Number of colors in $\mu(.)$ is minimized, i.e., the optimization objective is

formally defined as follows
$\arg\min_{\mathcal{USC} \subseteq \mathcal{SC}} \mid \{\mu(n_i)/n_i \in V \land \mu(n_i) \in \mathcal{USC}\} \mid$

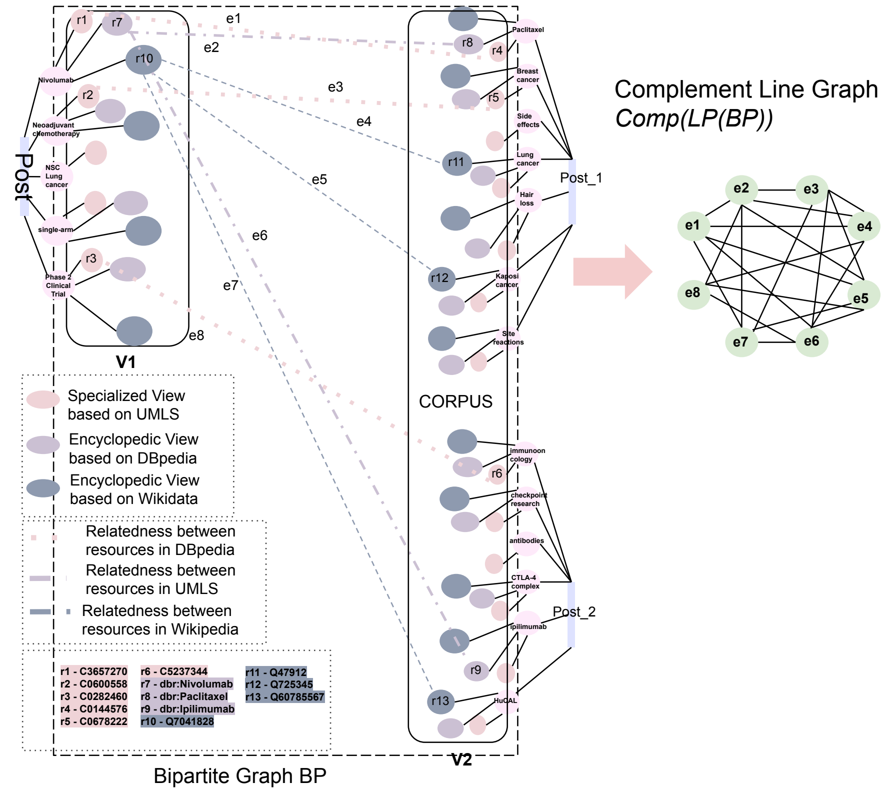


Figure 2.13: An example of a Bipartite Graph and the generated Complement Line Graph

2.6 Summary

Extracting knowledge over unstructured data in various domains, and the particular research challenges described in Chapter 1 demand comprehensive solutions from different angles. The concepts and current methods outlined in this chapter provide a solid basis for addressing the described challenges. The semantic web concept described in Section 2.1 provide formalism for representing the data on the web. Knowledge representation concepts exploited in Section 2.2 describe the current techniques for representing knowledge in a machine-readable format in order to be used by knowledge extraction approaches to address the **RQ2**. Knowledge extraction methods and the related concepts presented in Section 2.3.2 define the foundations for extracting knowledge from unstructured data and exploit the related tasks in this direction. The systems and methods presented in Section 2.4 provide an overview of the utilized techniques for knowledge extraction exploring symbolic, sub-symbolic, and neuro-symbolic methods, for answering the research questions **RQ1**. Section 2.5 presents the concepts and methods used for knowledge discovery in this thesis, to answer the **RQ3**.

Chapter 3

Related Work

In this chapter, we present a detailed analysis of the state-of-the-art approaches related to the main research problems and questions defined in Chapter 1. We initially describe the topics, shown in Figure 3.1, identified for the review of existing approaches. These topics include state-of-the-art approaches proposed by research communities to solve the issues related to Knowledge Extraction from unstructured data. Section 3.1 exploits state-of-the-art approaches for Knowledge Recognition. Furthermore, Named-Entity Recognition, and Relation Recognition approaches are reviewed. Section 3.2 describes the existing approaches for Named-Entity Disambiguation, Relation Linking, and Semantic Type Prediction. In section 3.3,

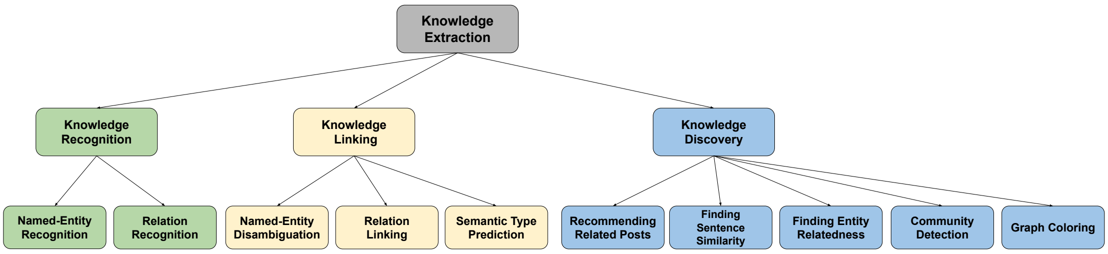


Figure 3.1: **Categories of the state-of-the-art Approaches**. The related work presented in this thesis is categorized under three topics; Knowledge Recognition, Knowledge Linking, and Knowledge Discovery. The Knowledge Recognition topic, encompasses existing techniques related to Named-Entity Recognition and Relation Recognition. The Knowledge Linking topic describes existing approaches related to Named-Entity Disambiguation, Relation Linking, and Semantic Type Prediction. The Knowledge Discovery topic covers the existing approaches related to Recommending Related Posts, Finding Sentence Similarity, Finding Entity Relatedness, Community Detection, and Graph Coloring.

the state-of-the-art methods for Knowledge Discovery that are related to the techniques used in this thesis are elaborated. Furthermore, the approaches related to Recommending Related Posts, Finding Sentence Similarity, Finding Entity Relatedness, Community Detection, and Graph Coloring are discussed.

3.1 Knowledge Recognition

3.1.1 Named-Entity Recognition

The problem of named-entity recognition involves extracting surface forms (a.k.a mentions) that are related to entities from a natural language text. Stanford NER [116] is a named-entity recognition tool developed using JAVA; it is also called CRFClassifier. Stanford NER approach is based on training models with labeled data; it uses simulated annealing in place of Viterbi decoding in sequence models such as HMMs, CMMs, and CRFs to incorporate non-local structure while preserving tractable inference. Thus, Stanford NER augments an existing CRF-based information extraction system with long-distance dependency models, enforcing label consistency and extraction template consistency constraints. TextRazor [117] is another named-entity recognition approach; it is based on a massive knowledge base of entity details extracted from different sources (e.g., Wikipedia and Wikidata). TextRazor uses their matching engine to recognize entities by looking to a dictionary of millions of different possible entities. For identifying entities that have never been mentioned before, TextRazor uses a rule-based tagger. TextRazor applies many steps of disambiguation after identifying the entity to get the correct entity finally. Entity and Relation Linking (EARL) [11] is a joint entity and relation recognition and linking framework; it uses a graph connection based solution. EARL resorts to a shallow parsing technique to extract all the keyword phrases in a natural language text; it uses SENNA [118] as the keyword extractor approach in order to link the extracted entities later. TagMe[1] is an approach that can identify meaningful short-phrases in natural language text and link them to a relevant Wikipedia page. TagMe supports English, German, and Italian languages for named-entity recognition. TagMe reports very good performance especially when annotating short texts. DBpedia Spotlight [119] is one of the collective named-entity recognition and linking approaches; it uses tokenization technique in order to identifies entities during its spotting stage. DBpedia Spotlight is able to recognize entities in a natural language text for 16 different languages and it has an online API for public use[2]. Spacy [120] is an open-source software library

[1] https://tagme.d4science.org/tagme/

[2] https://demo.dbpedia-spotlight.org/

for advanced natural language processing; it features convolutional neural network models for performing the task of named-entity recognition and it is widely used by the research community. SciSpacy [121] based on Spacy library for biomedical text processing; it utilizes fast and robust models for biomedical natural language processing. It is an end-to-end named-entity recognition and linking approach. Notwithstanding the mentioned named-entity recognition approaches are able to identify entities in a natural language text, the majority of them have limitations with the challenges mentioned in Section 1.3. In this thesis, we present a rule-based approach for the task of named-entity recognition that utilize catalogs of linguistic rules designed to overcome the mentioned challenges by leveraging several fundamental principles of English morphology.

3.1.2 Relation Recognition

The problem of relation recognition includes extracting surface forms that represent relations in a natural language text. Relation recognition have been longstanding research field [41, 42, 43, 44, 45]. Entity and Relation Linking (EARL) [11] is a framework that performs entity and relation recognition and linking as a joint task; it follows the same described strategy for named-entity recognition for relation recognition. RelationMatcher [12] identifies relations in a natural language text. RelationMatcher uses an index-based approach for extracting the relations. RelationMatcher approach is based on two-step phases. The first phase builds a semantically indexed bi-partite knowledge graph (SIBKB). The second phase is utilizing SIBKB in a pipeline for relation linking. RelationMatcher knowledge graph is based on PATTY [122], which is a background knowledge base containing semantically-typed relations. ReMatch [46] is a relation linking tool that extracts and links the relations in the question to the DBpedia knowledge graph. ReMatch's approach models KG relations with their related part of speech, then improves the model with the help of Wordnet and dependency parsing. Starting from this model, ReMatch uses similarity measurements to get the most similar property from the model to the one in the question. Probabilistic models have also been implemented for relation recognition. Initially, researchers such as TEXT RUNNER [123] employ a probabilistic model in the form of a Naive Bayes model for analyzing the textual characteristics of language. While Markov Logic Networks [44] and Conditional Random Fields [43] have been proposed as potential methods for enhancing the accuracy of data extraction. RelEx [45] utilizes dependency parse trees and a few simple heuristics to recognize relations from a natural language text. The current progress in relation recognition is attributable to the availability of enormous training data curated via distant supervision [124]. Despite the fact that the mentioned relation recognition approaches are able to recognize relations in a natural language text, the majority of them are limited by the challenges described in Section 1.3. In this thesis, we propose a rule-based approach for the task of relation recognition that employs linguistic rule catalogs tailored to address the aforementioned challenges by .

3.2 Knowledge Linking

3.2.1 Named-Entity Disambiguation

The named-entity disambiguation (also known as Entity Linking) task is a widely studied topic in the scientific literature ranging from graph traversal [125] to Neural Network-based approaches [5, 17, 126]. De Cao et al. [127] propose a generative model that removes hard negative sampling during training. Many of these approaches improve NED performance by incorporating extra knowledge such as entity definitions, entity types, and KG triples to assist the training process [128, 129]. A majority of these models rely on a fixed candidate set, treat the problem as a classification task, and require large training data. To mitigate dependency on labeled corpus, approaches such as [130, 131] use unlabeled corpora. Recently, researchers attempted zero-shot models to generalize NED to unseen entities [132, 133]. Entity and Relation Linking (EARL) [11] is a named-entity and relation disambiguation approach that uses a graph connection based solution. EARL uses Generalised Travelling Salesman Problem (GTSP) and GTSP approximate algorithm solutions for solving the disambiguation task. In addition, EARL determines the best semantic connection between the entities and the relations in a natural language text by exploiting the *connection density* between entity candidates and relation candidates. DBpedia Spotlight [119] is one of the collective named-entity disambiguation approaches; it identifies and links entities to DBpedia KG by applying a predifned pipeline. Ambiverse [134] is also a named-entity disambiguation tool that recognizes and links entities to YAGO knowledge graph. Ambiverse has an API for developers or anyone who is interested in using their tool. Ambiverse also supports using personal knowledge graphs, like a company-specific knowledge. Babelfy [135] is another tool for named-entity recognition; it merges the two domains of Word Sense Disambiguation (WSD) and named-entity disambiguation. Babelfy approach is based on a loose identification of candidate meanings combined with a densest sub-graph heuristic that chooses high-coherence semantic statements. NCEL [49] learns both local and global features from Wikipedia articles, hyperlinks, and entity links to derive joint embeddings of words and entities. These embeddings are used to train a deep Graph Convolutional Network (GCN) that integrates all the features through a Multi-layer Perceptron. The output is passed through a Sub-Graph Convolution Network, which finally resorts to a fully connected decoder. The decoder maps the output states to linked entities. The BI-LSTM+CRF model [50] formulates named-entity disambiguation as a sequence learning task in which the entity mentions are a sequence whose length equals the series of the output entities. The majority of the described named-entity disambiguation approaches have limitation disambiguating entities in short text where no enough context is presented.

NED for short text. There is concrete evidence in the literature that the machine learning-based models trained over generic datasets such as WikiDisamb30 [136], and CoNLL (YAGO) [137] do not perform well when applied to short texts. Singh et al. [138] evaluated more than 20 named-entity disambiguation tools over DBpedia for short text (e.g., questions) and concluded that issues like capitalization of surface forms, implicit entities, and multi-word entities affect the performance of NED tools in a short input text. Cetoli et al. [51] propose a neural network-based approach for linking entities to Wikidata KG. The authors also align an existing Wikipedia corpus-based dataset to Wikidata. However, this work only targets entity disambiguation and assumes that the entities are already recognized in the sentences. Arjun [52] is an approach for entity linking over Wikidata; it uses an attention-based neural network for linking Wikidata entity labels. OpenTapioca [139] is another attempt that performs end-to-end entity linking over Wikidata. OpenTapioca is also available as an API. S-Mart [140] is a tree-based structured learning framework based on multiple additive regression trees for linking entities in a tweet. The model was later adapted for linking entities in the questions. TagMe [136] is one of the popular works in this area, and uses a dictionary of entity surface forms extracted from Wikipedia to detect entity mentions in the parsed input text. These mentions passed through a voting scheme that computes the score for each mention-entity pair as the sum of votes given by candidate entities of all other mentions in the text [136], finally a pruning step filters out less relevant annotations. However, TagMe considers sentence length 30 for referring it as short text. VCG [141] is another attempt which is a unifying network that models contexts of variable granularity to extract features for an end to end entity linking. Albeit precise, deep learning approaches demand *high-quality* training annotations, which are not extensively available for Wikidata entity linking [51, 52]; it is important to note that most of these approaches use state-of-the-art sub-symbolic methods and require a large amount of training data. However, when these tools are applied to short text in a new domain such as question answering (QA) or keyword based search, the performance is limited [138, 142]. However, our defined rule-based approach does not require any training data and the predefined linguistic rules are able to overcome the challenge of having not enough context.

Neuro-symbolic approaches for NED. The neuro-symbolic approaches in the AI domain became a relatively hot topic of research [102]. In terms of symbolic approaches, [139] proposed a lightweight entity linking over Wikidata, which also relies on heuristic rules. Jiang et al. [143] uses first-order logic rules to boost the performance of a neural component for entity linking. However, the approach only focuses on short text (questions) as rules are designed for questions and find limitations to be generalized to a specific domain or a full sentence. Nevertheless, our defined neuro-symbolic approach is implemented for a specific domain use case and the empirical evaluation shows the effectiveness of the defined approach. Neuro-symbolic approaches also find their effectiveness in other domains such as vision QA, KG reasoning, and semantic embeddings [144, 145, 146].

Domain-specific NED approaches. Domain-specific biomedical NED is an active research area. Limsopatham and Collier [147] adds a softmax layer for classification. Other works encode surface forms and entity candidates into a common space and disambiguated candidates by nearest neighbors [148, 149]. Yuan, Yuan, and Yu [150] proposed a generative model by injecting synonyms and definition knowledge into the model by KG-guided pre-training. Angell et al. [151] and Varma et al. [152] focused on retrieve-and re-rank approach. Varma et al. [152] proposed a cross-domain data integration method that transfers structural knowledge from a general text KB to the medical domain. KrissBERT [153] introduced a simple yet effective method for NED based on distant supervision (self-supervision), which can work in a zero-shot regime without prior knowledge about instances in the test set. Sung et al. [154] improve biomedical NED by learning representations of entities solely based on the entity synonyms. Agarwal et al. [155] proposed a novel training mechanism using mention and entity representations that is based on building directed spanning trees over mentions and entities across documents; it allows learning mention coreference relationship. Overall, the literature is rich with several biomedical NED approaches. In contrast with existing approaches and as a proof of concept, our proposed neuro-symbolic approach relies on the idea of the availability of a closed-form KB (e.g., UMLS), to encode all the entity knowledge (e.g., entity definitions) in a sub-symbolic component (e.g., BERT [61]). Then, once a model is pre-trained similar to [156], use this knowledge to boost the performance of a symbolic component inspired by human-given rule templates [69, 143]. By this process, our proposed neuro-symbolic approach does not require any labeled training data at inference time (sentence, entity mentions, and corresponding links to a KB), considering the symbolic component only relies on linguistic rules for generating candidate entities during inference time. This is the fundamental difference with existing biomedical NED approaches.

3.2.2 Relation Linking

As mentioned before, relation extraction have been long-standing research field. However, linking relation label to its KG resources as independent approach is a relatively new field of research. Mulang', Singh, and Orlandi [46] had the first attempt in this direction and developed Rematch. ReMatch characterizes both the properties in a KG and the relations in a question as comparable triples, then leverages both synonyms and semantic similarity measures based on graph distances from the lexical knowledge base - Wordnet [80]. SIBKB [12] approach for relation linking uses PATTY to derive word embeddings for a bipartite semantically indexed knowledge base which assist in RL, likewise also in full QA systems such as AskNow [47] where PATTY is deployed as an underlying source of relation patterns. RelationMatcher [12] identifies and links relation in a text to the proper KG. RelationMatcher uses an index-based approach for extracting properties from a knowledge base. RelationMatcher approach is based on two-step phases. The first one builds a semantically indexed bi-partite knowledge graph (SIBKB). The second phase is utilizing SIBKB in a pipeline for relation linking. RelationMatcher knowledge graph is based on PATTY [122], which is a background knowledge base containing semantically-typed relations. ReMatch [46] is a relation linking tool that extracts and links the relations in the question to the DBpedia knowledge graph. The ReMatch approach models KG relations with their related part of speech, then improves the model with the help of Wordnet and dependency parsing. Starting from this model, ReMatch uses similarity measurements to get the most similar property from the model to the one in the question. RelMatch [157] is the disambiguation module (DM) of OKBQA framework. This module is the based on Agnostic Named Entity Disambiguation (AGDISTIS) [158] and disambiguation module of AutoSPARQL project [159]; it is integrated in Frankenstein framework [157] using Qanary [160, 161, 162]. Since NER/D and RE/L are parallel tasks and the occurrence of a named entity is often accompanied by relations, recent research has attempted to perform NED and RL as a joined process. EARL [11] is a tool for joined NED and RL that relies on Generalized Travelling Salesman Problem to find the right path between entities in the question. Several techniques exist in the literature for the collective entity and relation extraction in a text [8, 9, 10]. Despite effective, the described related approach for relation linking have limitations related to the challenges described in Section 1.3. In this thesis, we devise a rule-based approach for the task of relation linking that employs the proposed background knowledge to overcome the challenges of relation linking in a natural language text.

3.2.3 Semantic Type Prediction

Semantic type prediction has been used by many researcher to improve knowledge extraction tasks. For example, it can be used to improve named-entity disambiguation in various domains [163, 164, 165]. Chen et al. [164] proposes to inject latent entity type information into the entity embeddings based on pre-trained BERT. In addition, they integrate a BERT-based entity similarity score into the local context model of a state-of-the-art model to better capture latent entity type information. Vashishth et al. [163] presents a semantic type prediction module for biomedical NLP pipelines and two automatically-constructed, large-scale datasets with broad coverage of semantic types to improve medical entity linking. Biswas et al. [166] proposes an approach for entity typing leveraging different graph walk strategies in RDF2vec [167] together with textual entity descriptions. RDF2vec first generates graph walks and then uses a language model to obtain embeddings for each node in the graph. Moon, Jones, and Samatova [168] uses semantic type prediction for knowledge graph completion task. They propose an approach to address the entity type prediction problem. All the mentioned approaches for semantic type prediction suffer from the limitation of available training data especially in specific domains (e.g., biomedical domain). In this thesis, we devise a semantic type prediction approach that relies on the definition/description of entities for training. Thus, our proposed approach does not require any further training data.

3.3 Knowledge Discovery

3.3.1 Recommending Related Posts

Finding and recommending similar social media posts has attracted a considerable attention in the scientific research community [53, 54]. Work in [56] proposes a fuzzy inference system that learns the interests of the source and target users to categorize tweets. Tweet-Recommender system [55] provides tweets related to the news using topic similarities and language modeling. Several other approaches, such as [54, 57, 58], focus on hashtag-based or user interaction history for tweet recommendations. These methods primarily focused on hashtag similarities. However, they also rely on detecting communities in social network based on followers, mention, hashtag, and topic.

3.3.2 Finding Sentence Similarity

Besides social media, finding similar sentences in a document or a Web article is a well-studied research domain. Kiros et al. [59] train an encoder-decoder model to predict surrounding sentences of an encoded passage in a given document. Cer et al. [169] introduce a transformer-based model for encoding sentences into embedding vectors for calculating semantic similarity between two sentences. Other approaches such as in [170] propose sentential embedding-based techniques for computing similarity between two sentences. Sentence-BERT [60] is a BERT [61] based model that uses siamese and triplet network structures to derive semantically meaningful sentence embeddings that can be compared using cosine-similarity. We position our proposed approach for unveiling semantically related posts in a corpus with Sentence-BERT and its similar extensions in biomedical domain: BERTweet [171], COVID-Twitter-BERT [172]. However, there are two fundamental differences. First, our proposed approach aims to discover post relatedness based on the context and meaning of the entities in a post. Second, our proposed approach is agnostic to the domain-specific training data and resorts to the proposed background knowledge captured in publicly available KGs. Hence, our proposed approach can adapt to varied domains without any training data.

3.3.3 Finding Entity Relatedness

As our approach for unveiling semantically related posts in a corpus relies on entity relatedness, we describe the research work aiming to predict if two entities are similar. The problem of finding related entities and patterns from an unstructured text has not been investigated widely. Some approaches learn patterns in dependency representations of sentences to find similarities between the entities and the predicates mention in different sentences [173, 174]. Other approaches compare entities based on the semantic meaning [175], or by extracting linguistic patterns [176]. Stevenson et al. Stevenson and Greenwood [175] propose an approach that automatically learns patterns using WordNet [80] to find the similarity between entities and the predicates. For instance, in a given document, this approach aims at labeling the patterns "president resign" and "executive leave job" semantically similar. The approach starts with a small set of sample extraction patterns; it uses a similarity metric based on a version of the vector space model augmented with information from WordNet to learn similar patterns; however, it does not aim to label entities based on their similarity. Sematch [177] is a framework for the development, evaluation, and application of semantic similarity for KGs. Sematch is used to compute semantic similarity scores of the concepts, words, and entities in a

KG. Sematch focuses on specific knowledge-based semantic similarity metrics; they rely on both structural knowledge in a taxonomy (e.g., depth, path length, least common subsumer) and statistical information contents (corpus-IC and graph-IC). Sematch only calculates the similarity between two entities at a time. Some other approaches such as [55, 178, 179] aim to find relevant tweets in domains/topics such as London Riots and news. However, due to unavailability of public code, these approaches have been omitted from our experiments. Researchers from Facebook [180] released graph embeddings trained on the Wikidata knowledge graph. These embeddings can be utilized to calculate similarities between entities. However, contextual knowledge is not considered during the computation of embeddings. Recently, researchers have employed neural networks and deep-learning to capture similar patterns in unstructured text. EquatorNLP [173] is an approach that combines deep natural language processing and advanced machine learning for the task of extracting facts related to disaster response. Another deep-learning approach [174] recognizes mentions of Adverse Drug Reactions (ADR) in social media using knowledge-infused recurrent models. This approach solves the challenge of extracting ADR entities with characteristics including long surface forms, varied, and unconventional descriptions, as compared to more formal medical symptom terminology. Since graph embeddings report promising results for entity relatedness [181], we use RDF2Vec [167] in our framework for unveiling semantically related posts for the task of entity relatedness.

3.3.4 Community Detection

Existing community detection approaches have focused on the fundamental problem of grouping nodes in a network in the way that very densely connected nodes are placed in one community. In contrast, nodes in different communities are sparsely connected. As a result, detected communities provide the basis for uncovering connections that would be detected by simply traversing a network. Community detection relies on an objective function that captures the intuition that a community is a set of nodes with better internal connectivity than external connectivity. The exact optimization of this objective function is typically NP-hard. Heuristic-based (e.g., METIS [182] and semEP [183]) and approximation algorithms (e.g., [184, 185, 186]) aim at identifying sets of nodes with good values of the objective function and that can be understood good communities [187]. METIS [182] generates a graph partition guided by a heuristic that aims at creating a coarse graph whose size is within a small factor of the size of the final partition obtained after multilevel refinement. This process is conducted in three stages: coarsening, partition of the coarsest graph, and refinement, implemented following multilevel and multi-constraint partitioning schemes to scale up to large graphs;

this makes METIS a suitable method for detecting communities in large graphs. SemEP [183] is a graph partitioning method that identifies a minimal partition of a weighted bipartite graph; it is guided for optimizing an objective function that combines the values of similarity among the nodes in a community with the density of the connections. The problem of graph partitioning implemented by SemEP is matched to the Vertex Coloring Problem. The experimental studies in Chapter 6 show that this encoding enables the detection of high-quality communities in real-world graphs of various topologies [188, 189, 190]. Built upon these results, our proposed approach formalizes the problem of finding related posts as a problem of community detection. Our proposed approach resembles SemEP and resorts to well-known heuristic-based algorithms –e.g., DSATUR[191] and Welsh Powell [192]– for the Vertex Coloring Problem and semantic similarity measures to identify context-aware communities. However, since our proposed approach is agnostic of community detection techniques, METIS is also evaluated as a potential implementation. The results of our experimental studies indicate the empirical advantage of our approach against METIS and Welsh Powell.

3.3.5 Graph Coloring

The field of graph coloring is relatively well studied, spanning over the last few decades [192]; it is one of the most studied NP-HARD problem in computer science [193]. Initial approaches for graph coloring such as Welsh Powell [192] introduced vertex coloring algorithm. The vertex coloring algorithm finds application in various computer science sub-fields such as scheduling [194], frequency assignment [195] and communication network [196]. Vertex coloring problem has many applications in information extraction and link discovery tasks [183, 197, 198, 199]. In this thesis we map the studied problem of recommending semantically related posts to a graph coloring problem, which has not been done in the literature thus far. Hence, to do so, we revert too few of the fundamental approaches in vertex coloring problem such as DSATUR [191] and extend them to our use-case. We are not using the latest vertex coloring algorithms because: 1) DSATUR is one of the most fundamental algorithms in vertex coloring. Proving its extensibility to our studied problem solves purpose of application of graph coloring algorithm for unveiling semantically related posts. 2) In recent works of vertex coloring algorithms, it is a usual practice to compare proposed algorithm to DSATUR to show empirical effectiveness [200, 201]. 3) Our focus in this thesis is not to empirically compare hundreds of already proposed vertex coloring algorithms, instead to show if vertex coloring can be a solution for the studied problem in this thesis. Initial approaches for graph coloring such as Welsh Powell [192] introduced vertex coloring algorithm. Other graph partitioning approaches such as METIS can only partition undirected graphs, whereas our proposed approach does not have such dependencies. Empirical studies clearly illustrate the empirical advantage of our approach against METIS and Welsh Powell.

3.4 Summary

Subject to the analysis mentioned above of the existing approaches, this thesis focuses on methods for knowledge extraction over unstructured data. These methods should be able to extract knowledge over unstructured data efficiently. Furthermore, we exploit the knowledge recognition and linking techniques for entities and relations used for knowledge extraction. The knowledge recognition and linking techniques should improve knowledge extraction and discovery tasks. Finally, for a better understanding of knowledge discovery related work, we exploit the related work in the topics of recommending related posts, finding sentence similarity, finding entity relatedness, community detection, and graph coloring.

Chapter 4

Symbolic Approach for Entity & Relation Recognition and Linking

Short texts challenge NLP tasks such as named entity recognition, disambiguation, linking and relation inference because they do not provide sufficient context or are partially malformed (e.g., w.r.t. capitalization, long tail entities, implicit relations). Entity Linking (EL) task annotates surface forms in the text with the corresponding reference mentions in knowledge bases such as Wikipedia. It involves the two sub-tasks, i.e. Named Entity Recognition and Disambiguation (NER and NED) tasks. The state of the art contains considerable research body for EL from text to its Wikipedia mention [136, 137, 202, 203, 204, 205]. With the emergence of Knowledge Graphs (KGs) which represent data in a higher structured and semantic format such as DBpedia [77], Freebase [38] and Wikidata [14] that utilize Wikipedia as familiar knowledge source, retrieval-based applications such as question answering (QA) systems or keyword-based semantic search systems are empowered to provide more cognitive capabilities. Entity linking is a crucial component for a variety of applications built on knowledge graphs. For instance, an ideal NED tool on DBpedia recognizes the entities embedded in the question 'Who wrote the book The Pillars of The Earth?' and links them to the corresponding DBpedia entity (e.g. 'Pillars of The Earth' to `dbr:The_Pillars_of_the_Earth`)[1]. Another important NLP task is relation linking; it is about linking surface forms in text representing a relation to equivalent relations (predicates) of a KG. In our example question, an ideal relation linking (RL) tool links '`wrote`' to `dbo:author`[2]. There are existing approaches which address EL and RL tasks either jointly or independently [8, 9, 10, 11, 12]. However, they mostly fail in case of short text (e.g.

[1] dbr is the prefix for `http://dbpedia.org/resource/`

[2] dbo is the prefix for `http://dbpedia.org/ontology/`

question or key words based query) because the short text does not provide sufficient context which is essential for the disambiguation process. More importantly, a short text is often malformed meaning the text is incomplete, inexpressive, or implicit which is the case, particularly for relations in short sentences. Figure 4.1 shows the challenge tackled in this chapter with respect to the existing approaches. We have reported the content of this chapter in these publications [68, 69, 206]. The results of this chapter provide an answer to the following research question:

RQ1: What is the impact of the linguistics rules of a natural language on the tasks of knowledge extraction?

To answer this research question, we contribute to proposing a novel approach for jointly linking entities and relations within a short text into the entities and relations of a KG. The proposed approach is implemented for DBpedia (Falcon) and Wikidata (Falcon2.0) KGs. The proposed approach effectively maps entities and relations within a short text to its mentions of a background knowledge graph; it overcomes the challenges of short text using a light-weight linguistic approach relying on a background knowledge graph. Moreover, the proposed approach performs joint entity and relation linking of a short text by leveraging several fundamental principles of English morphology (e.g. compounding, headword identification) and utilizes an extended knowledge graph created by merging entities and relations from various knowledge sources. It uses the context of entities for finding relations and does not require training data. Our empirical study using several standard benchmarks and datasets show that Falcon and Falcon2.0 significantly outperform state-of-the-art entity and relation linking for short text query inventories. In essence, this chapter makes the following contributions to the problem of recognizing and linking entities and relations:

- A catalog of linguistic rules utilized by the proposed rule-based approach for knowledge extraction tasks (Contribution 1 in Chapter 1).

- A background knowledge that encodes knowledge extracted from various knowledge sources (Contribution 2 in Chapter 1).

- The Falcon approach that recognize entities and relations in a natural language text and link them to the DBpedia KG (Contributions 3 and 4in Chapter 1).

- The Falcon2.0 approach that performs the tasks of NER, NED, RR, and RL over a natural language text and use Wikidata as the target KG (Contributions 3 and 4 in Chapter 1).

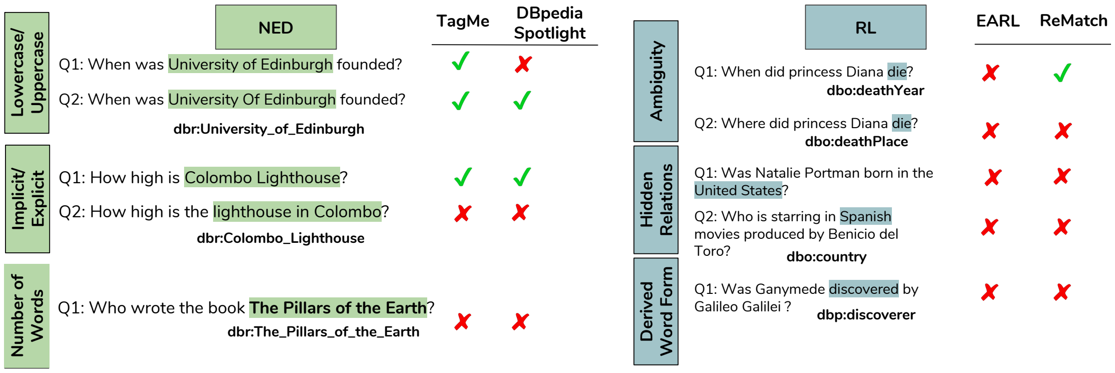

NED
TagMe
DBpedia Spotlight
Lowercase/ Uppercase
Q1: When was University of Edinburgh founded?
✓ ✗
Q2: When was University Of Edinburgh founded?
✓ ✓
dbr:University_of_Edinburgh
Implicit/ Explicit
Q1: How high is Colombo Lighthouse?
✓ ✓
Q2: How high is the lighthouse in Colombo?
✗ ✗
dbr:Colombo_Lighthouse
Number of Words
Q1: Who wrote the book The Pillars of the Earth?
dbr:The_Pillars_of_the_Earth
✗ ✗
RL
EARL
ReMatch
Ambiguity
Q1: When did princess Diana die?
dbo:deathYear
✗ ✓
Q2: Where did princess Diana die?
dbo:deathPlace
✗ ✗
Hidden Relations
Q1: Was Natalie Portman born in the United States?
✗ ✗
Q2: Who is starring in Spanish movies produced by Benicio del Toro?
dbo:country
✗ ✗
Derived Word Form
Q1: Was Ganymede discovered by Galileo Galilei ?
dbp:discoverer
✗ ✗

Figure 4.1: Performance of two Named-Entity Disambiguation (NED) and Relation Linking (RL) approaches on Specific Questions. TagMe and DBpedia Spotlight are the top-2 NED systems over the LC-QuAD QA dataset. However, considering short text questions, their behavior varies concerning question features, e.g., lowercase vs. uppercase, having implicit versus. explicit mappings, etc. Similar behavior has been observed for the top relation linking tools.

- An empirical evaluation of the proposed approaches among several benchmarks and datasets demonstrating the effectiveness of the proposed approaches over the state-of-the-art approaches.

Research Objectives. Existing approaches and systems for NER, NED, EL, and RL resort to machine learning and deep learning approaches that require a large training data [207, 208]. These approaches achieve high performance on data similar to seen data. For instance, Singh et al. [138] evaluated 20 NED tools for question answering over the DBpedia KG including TagMe [209], DBpedia Spotlight [119], Babelfy [135], and several APIs released by industry including Ambiverse [134], TextRazor [117], and Dandelion [210]. Among all, TagMe reports the highest F-score (0.67) over the complex question answering dataset LC-QuAD (TagMe is one of the top performing tools with an F-score of 0.91 on the generic WikiDisamb30 dataset [209]). Please be noted that TagMe was explicitly released for short text. However, when the input text is from a domain different from the training domain, its performance significantly falls down. Regarding the performance of various RL approaches such as ReMatch [46], SIBKB [12] is still low concerning accuracy and run-time even if they are purposefully developed for a particular domain or task. This deficiency is due to disregarding the context of the entities [138, 211]. Therefore, when aiming for annotating entities and relations of short text, it is important to develop an approach which a) is agnostic of the requirement of large training data and b) jointly links entities and relations to its KG equivalence.

Approach. We target the problem of joint entity and relation linking within short text using DBpedia and Wikidata KGs as background knowledge. We propose a novel approach that resorts to several fundamental principles of English morphology such as compounding [212], right-hand rule for headword identification [213] and utilizes an extended knowledge graph created by merging entities and relations from various knowledge sources. The approach focuses on capturing semantics underlying the input text by using the context of entities for finding relations and does not require any training data. Albeit simple, to the best of our knowledge, the combination of strategies and optimization of our approach is unique. Our evaluations show that it leads to substantial gains in recall, precision, and F-score on various benchmarks and domains.

Resources. Falcon is available as an open Web API[3], and its source code is released to ensure reproducibility. Another open source contribution is an extended knowledge graph which we built by merging information from several sources, e.g. DBpedia, Wikidata, Oxford dictionary, and Wordnet. These contributions are in our public Github[4]. The Falcon API already has over four million hits since April 2019.

`Falcon 2.0`: The first resource for joint entity and relation linking over Wikidata. `Falcon 2.0` relies on fundamental principles of English morphology (tokenization and compounding) and links entity and relation surface forms in a short sentence to its Wikidata mentions. `Falcon 2.0` is available as an online API[5] (Figure 4.2). Falcon 2.0 is also able to recognize entities in keywords such as Barack Obama, where there is no relation. We empirically evaluate `Falcon 2.0` on three datasets tailored for Wikidata. According to the observed results, `Falcon 2.0` significantly outperforms all the existing baselines. For the ease of use, we integrate the Falcon API[6] into Falcon 2.0. This option is available in case Wikipedia contains an equivalence entity (Wikidata is a superset of DBpedia). The `Falcon 2.0` API already has over five million hits since February 2020, which shows its gaining usability (excluding self-access of the API while performing the evaluation).

This chapter is structured as follows: the next section motivates our work by illustrating several limitations of state of the art over short text and highlighting the challenges related to the problem studied. Section 4.2 details the Falcon approach and explains the implementation details related to the components of the Falcon architecture for two use cases. In Section 4.3, we conduct the evaluation study for the DBpedia use of the proposed approach(Falcon). Section 4.4 presents the experimental study for the Wikidata use case (Falcon2.0). Section 4.3 and Section 4.4 allow us to answer the research question **RQ1**. Section 4.5 presents the

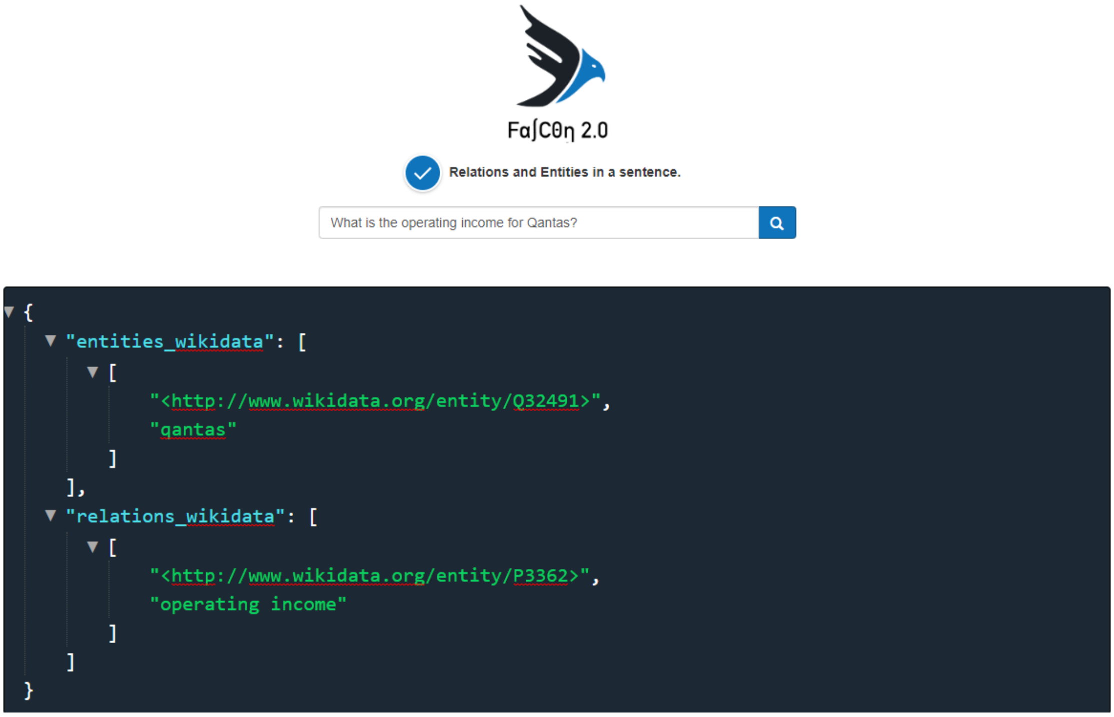
[3] `https://labs.tib.eu/falcon/`

[4] `https://github.com/AhmadSakor/falcon`

[5] `https://labs.tib.eu/falcon/falcon2/`

[6] `https://labs.tib.eu/falcon/`

importance and impact of this work for the research community. The availability and sustainability of resources is explained in Section 4.6, and its maintenance related discussion is presented in Section 4.7. Finally, we present the closing remarks of this chapter in Section 4.8.


FalCon 2.0
☑ Relations and Entities in a sentence.
What is the operating income for Qantas?

```
▼ {
  ▼ "entities_wikidata": [
    ▼ [
        "<http://www.wikidata.org/entity/Q32491>",
        "qantas"
      ]
    ],
  ▼ "relations_wikidata": [
    ▼ [
        "<http://www.wikidata.org/entity/P3362>",
        "operating income"
      ]
    ]
}
```

Figure 4.2: `Falcon 2.0` **API Web Interface**.

4.1 Motivation

We motivate our work by analyzing the performance of state-of-the-art EL and RL tools regarding query inventories on the DBpedia KG. In the following, we categorize the observed limitations.

Effect of Capitalization on EL tools TagMe and DBpedia Spotlight are the best two performing EL systems for question answering over DBpedia [138]. Considering the question ‘When was University of Edinburgh founded’, where the entity `University of Edinburgh` has one word (i.e. ‘of’) starting with lowercase letters. TagMe can identify this entity and link to its corresponding DBpedia entity `dbr:University_of_Edinburgh` but DBpedia Spotlight fails. However, when all words in the entity label are in uppercase, both tools recognize and link entities correctly (cf. Figure 4.1).

Effect of Implicit/Explicit Entities on EL tools The vocabulary mismatch problem [214] is common for text paraphrasing and significantly affects the performance of EL approaches. In Figure 4.1, both EL tools can correctly link the entity in the question 'How high is Colombo Lighthouse?' but fail when the question is rephrased to 'How high is the lighthouse in Colombo?' due to the vocabulary mismatch problem. In the first representation of the question, the entity label `Colombo Lighthouse` exactly matches to the DBpedia entity `dbr:Colombo_Lighthouse` which is not the case in the rephrased question (`dbr:Colombo_Lighthouse` is expected entity for `lighthouse in Colombo`).

Effect of the Number of Words in an Entity Label on EL tools Long tail entities were studied as a separate phenomenon such as in news [215]. For question answering, an increasing number of words jeopardizes entity linking performance. In our motivating example, both EL tools can not link the entity present from the question 'Who wrote the book The Pillars of the Earth?' where the entity label ('The Pillars of the Earth') has five words (a question from LC-QuAD dataset [216]).

Effect of Ambiguity of Question on RL tools EARL [11] and Rematch [46] are the two top performing relation linking tools for question answering over two different datasets QALD-5 [217] and LC-QuAD respectively. In Figure 4.1, for the question 'When did princess Diana die', Rematch correctly recognizes the relation `die` and links it to `dbo:deathYear`. However, when the question slightly changed to "Where did princess Diana die?" in which the expected relation is `dbo:deathPlace`, both tools fail to understand the ambiguity of the question intent and cannot provide the correct DBpedia IRIs.

Effect of Hidden Relation in a Question on RL tools Questions are typically relatively short and sometimes there is no natural language label for the relation. For example, to correctly answer the LC-QuAD question 'Was Natalie Portman born in the United States?' contains two relations: 1) the relational label `born` needs to be linked to `dbo:birthPlace` and 2) `dbo:country` is the hidden relation for which no relation surface form is present. A similar case can be observed in another question from the same dataset 'Who is starring in Spanish movies produced by Benicio del Toro?' where one of the expected relations is `dbo:country` for which no relation label is present. For both questions, EARL and ReMatch cannot identify hidden relations.

Effect of Derived Word Form of Relation Label on RL tools Consider the question 'Was Ganymede discovered by Galileo Galilei?' in which the relation label `discovered` is expected to link to the DBpedia ontology `dbo:discoverer`. The word `discoverer` is the derived word form of relation label `discovered`, and due to this, both tools fail to provide correct relation linking.

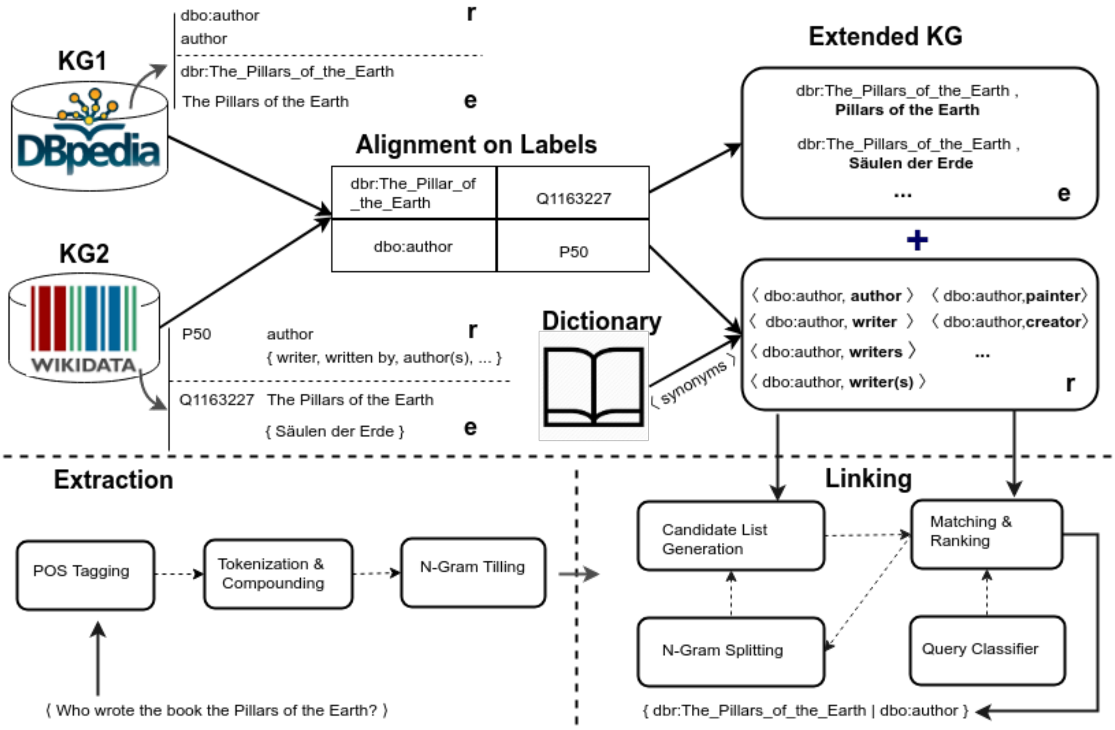


Figure 4.3: Overview of Falcon Approach. Falcon consists of two building blocks: 1) An extended knowledge graph which is built by merging information from various knowledge sources such as DBpedia, Wikidata, Oxford Dictionary, and WordNet. 2) Falcon architecture that has several modules focusing on surface form extraction and linking them to KG.

4.2 Symbolic Approach for Knowledge Extraction

The Falcon approach maps the surface forms within the short text into the textual representation of entities in KG. This mapping follows a particular strategy which is formalized in the following. Formally, a given short text is a set of tokens $\mathcal{T} = \{t_1, ..., t_n\}$. The set of entities in KG is the union of all KG resources $\mathcal{E} = C \cup P \cup I$ (where C, P, I are respectively a set of classes, properties, and instances), and L is the set of literals associated with entities. The task of entity linking is about mapping a subset of the input tokens denoted by $\mathcal{S} \in \mathcal{P}(\mathcal{T})$ (where $\mathcal{P}(\mathcal{T})$ is the power set of $\mathcal{T}$) to a set of entities denoted by $\mathcal{S}' \in \mathcal{P}(\mathcal{E})$ (where $\mathcal{P}(\mathcal{E})$ is the power set of $\mathcal{E}$), this mapping formally is represented as $\rho : \mathcal{S} \rightarrow \mathcal{S}'$. The Falcon approach deals with two optimization tasks as while it tries to maximize the number of tokens included in the set $\mathcal{S}$ (equation 1), it reduces the number of mapped entities in the set $\mathcal{S}'$ (eq. 2).

$$\gamma = \arg \max_{t_i \in \mathcal{A} | \mathcal{S} \in \mathcal{S}} \{\#t_i\} \tag{1}$$

$$\omega = \arg \min_{e_i \in \mathcal{A} | \mathcal{A} \in \mathcal{S}'} \{\#e_i\} \tag{2}$$

As a proof of concept, the proposed approach is implemented for two use cases. The first use case implements the Falcon approach targeting the DBpedia KG resources for linking. In the second use case, the Falcon approach links recognized entities and relations to the Wikidata KG (*Falcon 2.0*). The following sections explain the implementation details related to the components of the proposed approach.

4.2.1 DBpedia Use Case

Extended Knowledge Graph The DBpedia KG contains over 5.6 million entities and 111 million facts (consisting of subject-predicate-object triples) which require overall 14.2GB storage [77]. A major portion of this large information is not useful for EL/RL. Therefore we sliced DBpedia and extracted all the entity and relation labels to create a local KG. For example, the entity `Barack Obama`[7] in DBpedia has the natural language label 'Barack Obama' but DBpedia does not contain another representation of this label. However, the Wikidata KG is much richer and contains several aliases (or `known_as` labels) of Barack Obama such as Hussein Obama II, Barack Obama II , Obama, Barak Obama, President Obama, BHO and others[8]. We extended our local KG with this information from Wikidata. Similarly, for relation labels, the local KG is enriched with traditional linguistic resources such as Oxford dictionary [218], and semantic dictionaries like WordNet [80] to provide synonyms, derived word forms, etc. Use of background knowledge is common in question answering over DBpedia such as AskNow [47] uses Wordnet to support relation linking. However, we also propose extending entity labels using Wikidata which is not yet used in literature. These two separate extended KGs with a total size of 1.4GB are used as an underlying source of knowledge and act as the core of our approach (cf. Figure 4.3).

POS Tagging In the first module illustrated in Figure 4.3, short input text annotated with POS tag information using spaCy [120]. This step is used primarily to identify verb and noun phrases in the sentence.

[7] `http://dbpedia.org/page/Barack_Obama`

[8] `https://www.wikidata.org/wiki/Q76`

Tokenization and Compounding The next module creates tokens from the input sentence removing the stop words. In the first step, we break the sentence into potential tokens by removing all the stop words, and we use the stopword list provided by Fox [219]. For creating tokens, we also reuse basic compounding principle of English morphology. Compound words are lexeme that contains two or more stems [212]. The words which do not have any stop words between them considered as one compound word during token formation. For example, in question "Who is the wife of Barack Obama?", Barack Obama is noun phrases which do not have any stop word between, they considered as a single compound word. Compounding allows us to reduce the total number of tokens.

N-gram Tiling Typically, approaches described in [214, 220] dealing with short text start with the shortest token (or N-gram) to search associated candidates in the knowledge graph. This approach is not effective when an entity has many words in its label as it creates several additional tokens. For example in question "Who wrote the book The Pillars of the Earth?", it may generate several little tokens such as book, Pillars, Earth and it will result in several potential candidates in KG. In contrast, Bill et al. [221] applied an N-gram tiling algorithm in a question answering system to find the long answer in case of overlapping small answers. For example, answers "PQR" and "QRS" merged into single long answer "PQRS." This algorithm proceeds greedily until high scoring longest tilled N-gram found. We applied a similar approach to find the longest possible token for extracting the potential entity label. In the exemplary question " Who wrote the book The Pillars of the Earth?", The previous module generates tokens "wrote, book, Pillars, Earth." In N-gram tiling algorithm, we do not consider identified verbs of the sentence because in most cases a verb cannot be an entity label. Hence three tokens "book, Pillars, Earth" are merged as a single token. Also, verb token acts as a division point of the sentence in case of two entities, and we do not merge tokens from either side of the verb. In this process, the N-gram tiling algorithm starts with the first token from either side of the verb (which is a case of two entities in a sentence) and ends at the last non-stop word. The tiling algorithm also considers the stop words and provide the longest tilled N-gram. After N-gram tiling, we have two tokens: "wrote" and "book The Pillars of the Earth."

Candidate List Generation From the tokens, we create two list 1) potential relation candidates which contain verbs ("wrote") 2) potential entity candidates ("book The Pillars of the Earth"). We first search tokens of potential relation candidates in an extended KG of relations and get all the possible `DBpedia relation candidates`. Similar process has been repeated separately for `potential entity candidates` and all the `DBpedia entity candidates` are generated. For search, we use elastic search [222] over indexed extended KG. The reason behind the use of elastic search is its effectiveness over indexed KGs as reported by Dubey et al. [11]. In few cases, it is also possible that there is no verb in a sentence (e.g. Who is the prime minister of USA?). Then, we keep the list `potential relation candidates` empty, and search all the tokens of `potential entity candidates` into extended KG of DBpedia relations because number of relations in DBpedia are very less and when tokens in `potential entity candidates` find any match, they are pushed to `potential relation candidates`.

Candidate Ranking To rank best DBpedia candidates, we utilize the fundamental principle of knowledge graph creation. In any knowledge graph, a sentence is represented as triple with <subject, predicate, object>. Therefore, we rank the candidates by creating a triple consisting of the relation and entity candidates from `DBpedia entity candidates` and `DBpedia relation candidates`, then check if these triples exist in the DBpedia KG. We do it by passing the triple to DBpedia SPARQL endpoint. This can be done by executing a simple Ask query against a KG endpoint which would return a boolean value indicative of the existence of triple or otherwise of this triple. For each existing triple, we increase the weight of the entities and relations involved in the triple.

While ranking, we also consider question headwords (who, what, when, etc.) for question classification [223]. Each relation in DBpedia has its domain and range associated with an entity such as person, place, date, etc. The headwords are used to determine the correct range and domain of the DBpedia relation. For example in the question "Who is starring in Spanish movies produced by Benicio del Toro?" there is a hidden relation `dbo:country` for which no surface form is present. While checking the domain of each token in relation and entity candidate lists, we can extract that word "Spanish" has the domain country; therefore, it is also an expected relation.

N-Gram Splitting In the previous module, if we do not get any triple in DBpedia for candidates present in `potential entity candidates` and `potential relation candidates`, we split the tokens (N-grams). To split the tokens, we again use the fundamentals of English morphology. The compound words in English have their headword always towards right side [213]. Therefore, we start splitting tokens from "N-Gram tiling" module from the right side and pass these tokens to candidate generation module. This greedy algorithm stops when it finds triple(s) of DBpedia candidate list.

4.2.2 Wikidata Use Case

In this section, we describe Falcon 2.0 in detail. First the architecture of Falcon 2.0 is depicted. Next, we discuss the BK used to match the surface forms in the text to the resource in a specific KG. In the chapter's scope, we define "short text" as grammatically correct questions (up to 15 words).

4.2.3 Architecture

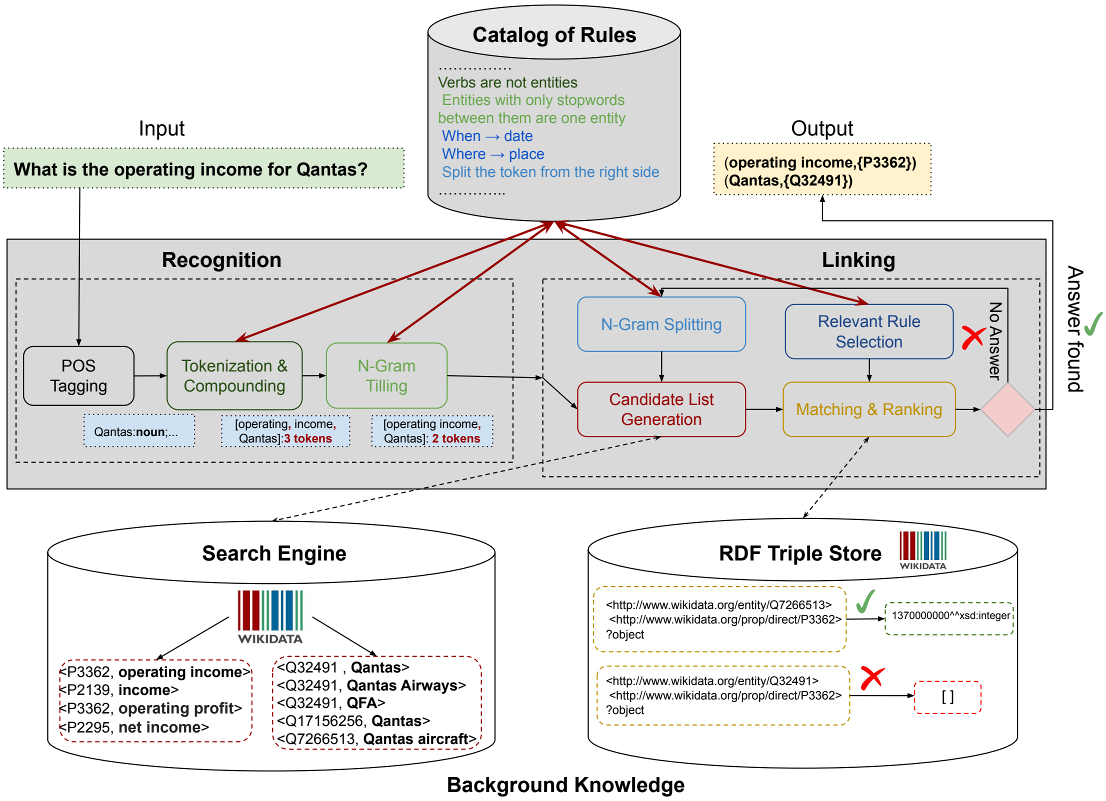

Catalog of Rules
....
Verbs are not entities
Entities with only stopwords between them are one entity
When → date
Where → place
Split the token from the right side
....
Input
What is the operating income for Qantas?
Output
(operating income,{P3362})
(Qantas,{Q32491})
Recognition
Linking
POS Tagging
Tokenization & Compounding
N-Gram Tilling
Qantas:noun;...
[operating, income, Qantas]:3 tokens
[operating income, Qantas]: 2 tokens
N-Gram Splitting
Relevant Rule Selection
Candidate List Generation
Matching & Ranking
No Answer
Answer found
Search Engine
WIKIDATA
<P3362, operating income>
<P2139, income>
<P3362, operating profit>
<P2295, net income>
<Q32491 , Qantas>
<Q32491, Qantas Airways>
<Q32491, QFA>
<Q17156256, Qantas>
<Q7266513, Qantas aircraft>
RDF Triple Store
WIKIDATA
<http://www.wikidata.org/entity/Q7266513> <http://www.wikidata.org/prop/direct/P3362> ?object
1370000000^^xsd:integer
<http://www.wikidata.org/entity/Q32491> <http://www.wikidata.org/prop/direct/P3362> ?object
[ ]
Background Knowledge

Figure 4.4: The Falcon 2.0 **Architecture**. The boxes highlighted in Grey are reused from the DBpedia use case implementation. Grey boxes contain a linguistic pipeline for recognizing and linking entity and relation surface forms. The boxes in White are our addition to the Falcon pipeline to build a resource for the Wikidata entity and relation linking. The white boxes constitute what we refer to as BK specific to Wikidata. The text search engine contains the alignment of Wikidata entity/relation labels along with the entity and relation aliases. It is used for generating potential candidates for entity and relation linking. RDF triple store is a local copy of Wikidata triples containing all entities and predicates.

The `Falcon 2.0` architecture is depicted in Figure 4.4. `Falcon 2.0` receives short input texts and outputs a set of entities and relations extracted from the text; each entity and relation in the output is associated with a unique Internationalized Resource Identifier (IRI) in Wikidata. `Falcon 2.0` resorts to BK and a catalog of rules for performing entity and relation linking. The BK combines Wikidata labels and their corresponding aliases. Additionally, it comprises alignments between nouns and entities in Wikidata. Alignments are stored in a text search engine, while the knowledge source is maintained in an RDF triple store accessible via a SPARQL endpoint. The rules that represent the English morphology are in a catalog; a forward chaining inference process is performed on top of the catalog during the extraction and linking tasks. `Falcon 2.0` also comprises several modules that identify and link entities and relations to the Wikidata. These modules implement POS Tagging, Tokenization & Compounding, N-Gram Tiling, Candidate List Generation, Matching & Ranking, Query Classifier, and N-Gram Splitting and are reused from the implementation of Falcon.

4.2.4 Background Knowledge

Wikidata contains over 52 million entities and 3.9 billion facts (in the form of subject-predicate-object triples). Since Falcon 2.0 background knowledge only depends on labels, a significant portion of this extensive information is not useful for our approach. Hence, we only extract all the entity and relation labels to create a local background KG, A.K.A "alias background knowledge base.". For example, the entity `United States of America`[9] in Wikidata has the natural language label 'United States of America' and several other aliases (or `known_as` labels) of `United States of America` such as "the United States of America, America, U.S.A., the U.S., United States, etc.". We extended our background KG with this information from Wikidata. Similarly, for relation's labels, the background KG is enriched with `known_as` labels to provide synonyms and derived word forms. For example, the relation spouse [10] in Wikidata has the label `spouse` and the other known as labels are husband, wife, married to, wedded to, partner, etc. This variety of synonyms for each relation empowers `Falcon 2.0` to match the surface form in the text to a relation in Wikidata. Figure 4.5 illustrates the process of building background knowledge.

[9] https://www.wikidata.org/wiki/Q30

[10] https://www.wikidata.org/wiki/Property:P26

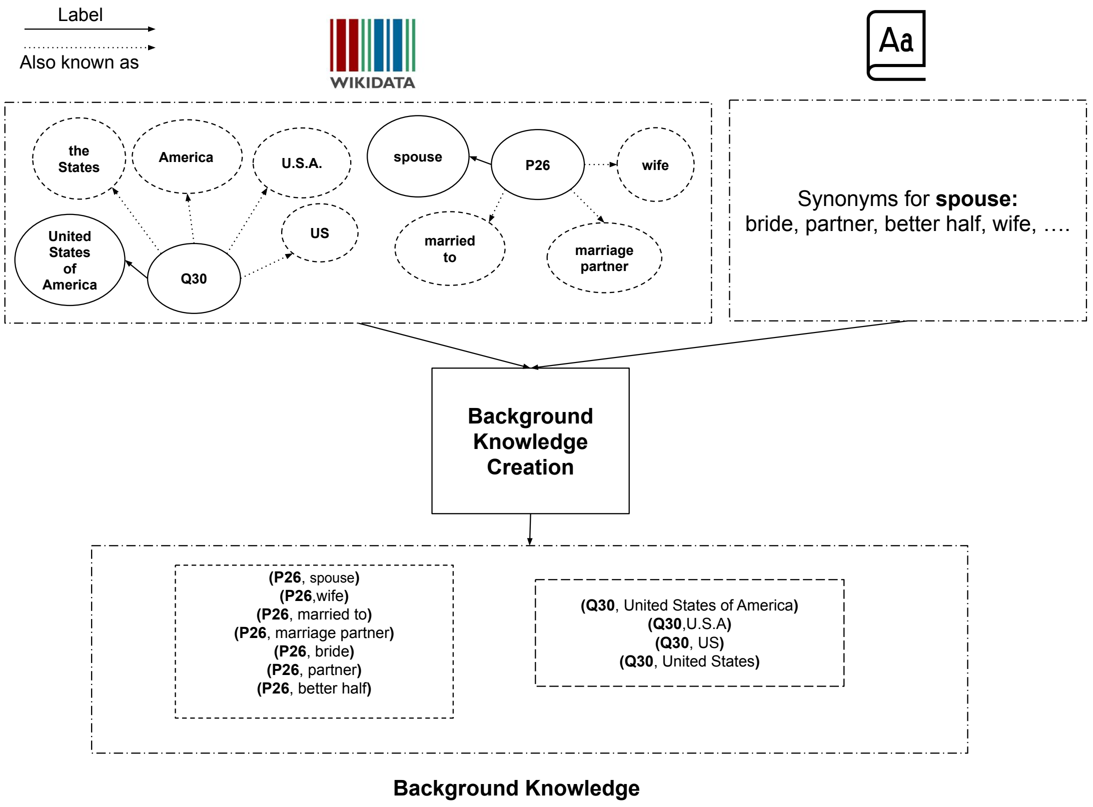


Figure 4.5: `Falcon 2.0` **Background Knowledge** is built by converting labels of entities and relations in Wikidata into pairs of alignments. It is a part of search engine (cf. Figure 4.4).

4.2.5 Catalog of Rules

`Falcon 2.0` is a rule-based approach. A catalog of rules is predefined to extract entities and relations from the text. The rules are based on the English morphological principles. For example, `Falcon 2.0` excludes all verbs from the entities candidates list based on the rule `verbs are not entities`. For example, the N-Gram tiling module in the `Falcon 2.0` architecture resorts to the rule: `entities with only stopwords between them are one entity`. Another example of such rule `When -> date, Where -> place` solves the ambiguity of matching the correct relation in case the short text is a question by looking at the questions headword. For example, give the two questions `When did Princess Diana die?` and `Where did Princess Diana die?`, the relation died can be the death place or the death year. The question headword (When/Where) is the only insight to solve the ambiguity here. When the question word is `where`, `Falcon 2.0` matches only relations that have a place as a range of the relation.

4.2.6 Recognition

Extraction phase in `Falcon 2.0` consists of three modules. POS tagging, tokenization & compounding, and N-Gram tiling. The input of this phase is a natural language text. The output of the phase is the list of surface forms related to entities or relations.

Part-of-speech (POS) Tagging receives a natural language text as an input. It tags each word in the text with its related tag, e.g., noun, verb, and adverb. This module differentiates between nouns and verbs to enable the application of the morphological rules from the catalog. The output of the module is a list of the pairs of (word, tag).

Tokenization & Compounding builds the tokens list by removing the stopwords from the input and splitting verbs from nouns. For example, if the input is `What is the operating income for Qantas`, the output of this module is a list of three tokens [operating, income, Qantas].

N-Gram Tilling module combines tokens with only stopwords between them relying on one of the rules from a catalog of rules. For example, if we consider the previous module's output as an input for the n-gram tilling module, `operating` and `income` tokens will be combined in one token. The output of the module is a list of two tokens [operating income, Qantas].

4.2.7 Linking

This phase consists of four modules: candidate list generation, matching & ranking, relevant rule selection, and n-gram splitting.

Candidate List Generation receives the output of the recognition phase. The module queries the text search engine for each token. Then, tokens will have an associated candidate list of resources. For example, the retrieved candidate list of the token `operating income` is [(P3362, operating income), (P2139, income), (P3362, operating profit)]; where the first element is the Wikidata predicate identifier and the second is the list of labels associated with the predicates which match the query "operating income."

Matching & Ranking ranks the candidate list received from the candidate list generation module and matches candidates' entities and relations. Since, in any KG, the facts are represented as triples, the matching and ranking module creates triples consisting of the entities and relationships from the candidates' list. Then, for each pair of entity and relation, the module checks if the triple exists in the RDF triple store (Wikidata). The checking is done by executing a simple ASK query over the RDF triple store. For each triple, the module increases the rank of the involved relations and entities. The output of the module is the ranked list of the candidates.

Relevant Rule Selection interacts with the matching & ranking module by suggesting increasing the ranks of some candidates relying on the catalog of rules. One of the suggestions is considering the question headword to clear the ambiguity between two relations based on the range of relationships in the KG.

N-Gram Splitting module is used if none of the triples tested in the matching & ranking modules exists in the triple store, i.e., the compounding the approach did in the tokenization & compounding module led to combining two separated entities. The module splits the tokens from the right side and passes the tokens again to the candidate list generation module. Splitting the tokens from the right side resorts to one of the fundamentals of the English morphology; the compound words in English have their headword always towards the right side [213].

Text Search Engine stores all the alignments of the labels. A simple querying technique [222] is used as the text search engine over background knowledge. It receives a token as an input and then returns all the related resources with labels similar to the received token.

RDF Triple store is a local copy of the Wikidata endpoint. It consists of all the RDF triples of Wikidata labeled with the English language. An RDF triple store is used to check the existence of the triples passed from the **Matching & Ranking** module. The RDF triple store keeps around 3.9 billion triples.

4.3 Experimental Study - DBpedia Use case

Experiment Setup. We used a local laptop machine, with eight cores and 16GB RAM running Ubuntu 18.04 for implementation. Falcon is deployed as public API on a server with 723GB RAM, 96 cores (Intel(R) Xeon(R) Platinum 8160 CPU with 2.10GHz) running Ubuntu 18.04. This API is used for calculating all the results. The EL systems have been evaluated on different settings in literature, therefore to provide a fair evaluation we utilize Gerbil [224], which is a benchmarking framework for EL systems and integrated Falcon API into the Gerbil architecture. We report macro precision (P), macro recall (R), and macro F-score[11] in the tables. Falcon average run time is 1.9 seconds per question. Gerbil does not benchmark RL systems; therefore, RL systems are benchmarked using Frankenstein platform [157]. Our code, Extended KG, and data is in Github.[12]

Datasets. We employ two distinct datasets: 1) the LC-QuAD [216] dataset comprises 5,000 complex questions for DBpedia (80 percent questions are with more than one entity and relation) where average question length is 12.29 words. 2) QALD-7 [225] is the most popular benchmarking dataset for QA over DBpedia comprising 215 questions. In QALD, the average question length is 7.41 words and over 50% of the questions include a single entity and relation. For our linguistic based approach, we randomly selected 100 questions each from SimpleQuestions dataset [226] and complex questions[13] for the formation of rules.

Baselines. The state-of-the-art outperforming tools are TagMe and DBpedia Spotlight reported in [138]. These two systems in addition to the systems already integrated in Gerbil i.e., KEA [227], FOX [228], Babelfy [135], AIDA [137] are included in our benchmark. We also report the performance of EARL [11] for entity linking as it jointly performs EL and RL. For relation linking, the EARL system is our baseline. We evaluate NED and RL systems on the LC-QuAD3253 subset of the LC-QuAD dataset (containing 3,253 LC-QuAD questions) to compare the performance with the 20 NED and five RL systems evaluated by Singh et al. [138]. Many of these 20 tools are APIs from industry (Ambiverse [134], TextRazor [117], and Dandelion [210]) which use state of the art machine learning approaches.

Performance Evaluation Table 4.1 summarizes Falcon's performance compared to state-of-the-art systems integrated in Gerbil. For the QALD and LC-QuAD datasets, Falcon significantly outperforms the baseline. Similar observa-

[11] https://github.com/dice-group/gerbil/wiki/Precision,-Recall-and-F1-measure

[12] https://github.com/AhmadSakor/falcon

[13] http://qa.mpi-inf.mpg.de/comqa/

tions are made for relation linking, where the performance of Falcon is approximately twice as high as the next best competitor on all datasets (cf. Table 4.2).
Success cases of Falcon: Falcon overcomes several major issues of short text such as capitalization of surface forms, derived word forms of relation labels and successfully handles long tail entities. For entity linking, we achieve slightly better performance on LC-QuAD than QALD. This is due to the fact that LC-QuAD questions mostly contain more than one entity and relation and thus provide more context to understand the short text. Also, major failure cases of state-of-the-art EL systems over these datasets are due to the short length and limitation to exploit the context. For example the question 'Give me the count of all people who ascended a peak in California.' (`dbr:California` is correct entity), TagMe provides two entities: `dbr:California` (for surface form California) and `dbr:Give_In_to_Me` (for "Give me"). Fundamental principles such as compounding and N-gram tiling have positive impact on the Falcon performance and we can correctly annotate several long tail entities and entities containing compound words. For example, Falcon cor-

Table 4.1: Performance of the Falcon Framework compared to various entity linking tools.

System	**Dataset**	**P**	**R**	**F**
KEA [227]	QALD-7	0.06	0.06	0.06
EARL [11]	QALD-7	0.58	0.60	0.58
FOX [228]	QALD-7	0.59	0.57	0.57
Babelfy [135]	QALD-7	0.40	0.55	0.44
AIDA [137]	QALD-7	0.61	0.58	0.59
DBpedia Spotlight [119]	QALD-7	0.68	0.72	0.69
TagMe [209]	QALD-7	0.64	0.76	0.67
Falcon	QALD-7	**0.78**	**0.79**	**0.78**
KEA [227]	LC-QuAD	0.001	0.001	0.001
EARL [11]	LC-QuAD	0.53	0.55	0.53
FOX [228]	LC-QuAD	0.53	0.51	0.51
Babelfy [135]	LC-QuAD	0.43	0.50	0.44
AIDA [137]	LC-QuAD	0.50	0.45	0.47
DBpedia Spotlight [119]	LC-QuAD	0.60	0.65	0.61
TagMe [209]	LC-QuAD	0.65	0.77	0.68
Falcon	LC-QuAD	**0.81**	**0.86**	**0.83**
[138]	LC-QuAD3253	0.69	0.66	0.67
Falcon	LC-QuAD3253	**0.73**	**0.74**	**0.73**

rectly annotates question from LC-QuAD: 'Name the military unit whose garrison is Arlington County, Virginia and command structure is United States Department of Defense' where expected entities are `dbr:Arlington_County,_Virginia` and `dbr:United_States_Department_of_Defense`. Also, extended local KG has provided several interpretation of entities and their derived forms. The extended KG act as source of background knowledge during the linking process and provide extra information about entities. Generally, other entity linking tools directly map surface forms to the underlying KG using several novel techniques. However, this concept of enriching a local extended KG is not exploited in the literature and it has positively impacted the performance of the Falcon.

For relation linking, taking the context of the entities into account improved the overall performance of the Falcon. In our example question 'Who wrote the book The Pillars of the Earth?', EARL, SIBKB and Rematch aim for directly mapping `wrote` to DBpedia which results in several wrong relations such as `dbo:writer`, `dbo:creator` but when Falcon considers entity references of the question to verify which triples exist with the given entity `dbr:The_Pillars_of_the_Earth`, Falcon determines the correct relation `dbo:author`. It is important to note that existing relation linking tools completely ignore the context of the entities. Secondly, Falcon uses a fundamental principle of creating an RDF knowledge graph. While ranking the candidates in the Candidate List Ranking step, Falcon verifies the presence of the correct triple containing entity and associated relation in the KG. It has been done by cross-checking all the combinations of `potential entity candidates` and `potential relation candidates` as triple using an ASK query. Three con-

Table 4.2: Performance of the Falcon Framework compared to various Relation Linking tools.

QA Component	**Dataset**	**P**	**R**	**F**
SIBKB [12]	QALD-7	0.29	0.31	0.30
ReMatch [46]	QALD-7	0.31	0.34	0.33
EARL [11]	QALD-7	0.27	0.28	0.27
Falcon	QALD-7	**0.58**	**0.61**	**0.59**
SIBKB [12]	LC-QuAD	0.13	0.15	0.14
ReMatch [46]	LC-QuAD	0.15	0.17	0.16
EARL [11]	LC-QuAD	0.17	0.21	0.18
Falcon	LC-QuAD	**0.42**	**0.44**	**0.43**
[138]	LC-QuAD3253	0.25	0.22	0.23
Falcon	LC-QuAD3253	**0.56**	**0.57**	**0.56**

cepts (utilization of entity context, ranking the candidates based on the presence of triple in the KG, and use of extended KG) have collectively resulted into a significant jump over other relation linking tools as observed in the Table 4.2.
Failure cases of Falcon: There are few EL cases where Falcon fails. For example, in question 'How many writers worked on the album Main Course?', the expected entity is `dbr:Main_Course`. However, Falcon returns `dbr:Critters_2:_The_Main_Course`. This is caused by compounding and the resulting token for this question was 'album Main Course'. For the same question Falcon correctly links the relations. We further analyzed failure cases of Falcon for RL. We found that more than half of the questions which were unanswered have implicit relations. For example, for the question 'In what city is the Heineken brewery?' with the two relations `dbo:locationCity` and `dbo:manufacturer`, Falcon returns `dbo:city` as relation. There are few types of questions ('Count all the scientologists.') for which Falcon fails both for EL and RL tasks. This question is relatively short and requires reasoning to provide correct entities and relations (`dbr:Scientology` and `dbo:religion`).

4.4 Experimental Study - Wikidata Use Case

We study three research questions: RQ1) What is the performance of `Falcon 2.0` for entity linking over Wikidata? RQ2) What is the impact of Wikidata's specific background knowledge on the performance of a linguistic approach? RQ3) What is the performance of `Falcon 2.0` for relation linking over Wikidata?

Metrics We report the performance using the standard metrics of Precision, Recall, and F-measure. Precision is the fraction of relevant resources among the retrieved resources.

$$precision = \frac{|\{\text{Relevant resources}\} \cap \{\text{Retrieved resources}\}|}{|\{\text{Retrieved resources}\}|} \tag{3}$$

Recall is the fraction of relevant resources that have been retrieved over the total amount of relevant resources.

$$recall = \frac{|\{\text{Relevant resources}\} \cap \{\text{Retrieved resources}\}|}{|\{\text{Relevant resources}\}|} \tag{4}$$

F-Measure or F-Score is the harmonic mean of precision and recall. (Equation 5).

$$F = 2 \times \frac{precision \times recall}{precision + recall} \tag{5}$$

Datasets We rely on three different question answering datasets namely SimpleQuestion dataset for Wikidata [229], WebQSP-WD [141] and LC-QuAD 2.0 [230]. The SimpleQuestion dataset contains 5,622 test questions which are answerable using Wikidata as underlying KG. WebQSP-WD contains 1639 test questions, and LC-QUAD 2.0 contains 6046 test questions. SimpleQuestion and LC-QuaD 2.0 provide the annotated gold standard for entity and relations, whereas WebQSP-WD only provides annotated gold standard for entities. Hence, we evaluated entity linking performance on three datasets and relation linking performance on two datasets. Also, SimpleQuestion and WebQSP-WD contain questions with a single entity and relation, whereas LC-QuAD 2.0 contains mostly complex questions (i.e., more than one entity and relation).

Baselines **OpenTapioca** [139]: is available as a web API; it provides Wikidata URIs for entities. We run OpenTapioca API on all the three datasets.
Variable Context Granularity model (VCG) [141]: is a unifying network that models contexts of variable granularity to extract features for mention detection and entity disambiguation. We were unable to reproduce VCG using the publicly available source code. Hence, we only report its performance on WebQSP-WD from the original paper [141] as we are unable to run the model on the other two datasets for entity linking. For the completion of the approach, we also report the other two baselines provided by the authors, namely **Heuristic Baseline** and **Simplified VCG**.
S-Mart [140]: was initially proposed to link entities in the tweets and later adapted for question answering. The system is not open source, and we adapt its result from [141] for WebQSP-WD dataset.
No Baseline for Relation Linking: To the best of our knowledge, there is no baseline for relation linking on Wikidata. One argument could be to run the existing DBpedia based relation linking tool on Wikidata and compare it with our performance. We contest this solely because Wikidata is extremely noisy. For example, in "What is the longest National Highway in the world?" the entity surface form "National Highway" matches four(4) different entities in Wikidata that share the same entity label (i.e., "National Highway"). In comparison, 2,055 other entities contain the full mention in their labels for the surface form "National Highway". However, in DBpedia, there exists only one unique label for "National Highway". Hence, any entity linking tool or relation linking tool tailored for DBpedia will face issues on Wikidata (cf. table 4.5). Therefore, instead of reporting the bias and under-performance, we did not evaluate their performance for a fair comparison. Hence, we report `Falcon 2.0` relation linking performance only to establish new baselines on two datasets: SimpleQuestion and LC-QuAD 2.0.

Table 4.3: Entity linking evaluation results on LC-QuAD 2.0 & SimpleQuestion datasets. Best values are in bold.

Approach	**Dataset**	**P**	**R**	**F**
OpenTapioca [139]	LC-QuAD 2.0	0.29	0.42	0.35
`Falcon 2.0`	LC-QuAD 2.0	**0.50**	**0.56**	**0.53**
OpenTapioca [139]	SimpleQuestion	0.01	0.02	0.01
`Falcon 2.0`	SimpleQuestion	**0.56**	**0.64**	**0.60**
OpenTapioca [139]	SimpleQuestion Uppercase Entities	0.16	0.28	0.20
`Falcon 2.0`	SimpleQuestion Uppercase Entities	**0.66**	**0.75**	**0.70**

Table 4.4: Entity linking evaluation results on the WEBQSP test dataset. Best values are in bold.

Approach	**P**	**R**	**F**
S-MART [140]	0.66	0.77	0.72
Heuristic baseline [141]	0.30	0.61	0.40
Simplified VCG [141]	**0.84**	0.62	0.71
VCG [141]	0.83	0.65	0.73
OpenTapioca [139]	0.01	0.02	0.02
`Falcon 2.0`	0.80	**0.84**	**0.82**

Experimental Details `Falcon 2.0` is extremely lightweight from an implementation point of view. A laptop machine, with eight cores and 16GB RAM running Ubuntu 18.04 is used for implementing and evaluating `Falcon 2.0`. We deployed its web API on a server with 723GB RAM, 96 cores (Intel(R) Xeon(R) Platinum 8160CPU with 2.10GHz) running Ubuntu 18.04. This publicly available API is used to calculate the standard evaluation metrics, namely Precision, Recall, and F-score.

4.4.1 Experimental Results

Experimental Results 1 In the first experiment described in Table 4.3, we compare entity linking performance of `Falcon 2.0` on SimpleQuestion and LC-QuAD 2.0 datasets. We first evaluate the performance on the SimpleQuestion dataset. Surprisingly, we observe that for the OpenTapioca baseline, the values are approximately 0.0 for Precision, Recall, and F-score. We analyzed the source of errors and found that out of 5,622 questions, only 246 have entity labels in uppercase letters. Opentapioca fails to recognize and link entity mentions written in lowercase letters. Case sensitivity is a common issue for entity linking tools over short text, as reported by Singh et al. [138, 211] in a detailed analysis. From the remaining 246 questions, only 70 are answered correctly by OpenTapioca. Given that OpenTapioca finds limitation in linking lowercase entity surface forms. We evaluated `Falcon 2.0` and OpenTapioca on the 246 questions of SimpleQuestion to provide a fair evaluation for the baseline (reported as SimpleQuestion uppercase entities in table 4.3). OpenTapioca reports F-score 0.20 on this subset of SimpleQuestion. On the other hand, `Falcon 2.0` reports F-score 0.70 on the same dataset (cf. Table 4.3). For LC-QuAD 2.0, OpenTapioca reports F-score 0.35 against `Falcon 2.0` with F-score 0.53 reported in Table 4.3.

Experimental Results 2 We report performance of `Falcon 2.0` on WebQSP-WD dataset in Table 4.4. `Falcon 2.0` clearly outperforms all other baselines with highest F-score value 0.82. OpenTapioca demonstrates a low performance on this dataset as well. Experiment results 1 & 2 answer our first research question (RQ1).

Ablation Study for Entity Linking and Recommendations For the second research question (RQ2), we evaluate the impact of Wikidata's specific background knowledge on the entity linking performance. We evaluated Falcon on the WebQSP-WD dataset against `Falcon 2.0`. We linked Falcon predicted DBpedia IRIs with corresponding Wikidata IDs using owl:sameAs. We can see in the Table 4.5 that `Falcon 2.0` significantly outperforms Falcon despite using the same linguistic driven approach. The jump in `Falcon 2.0` performance comes from Wikidata's specific local background knowledge, which we created by expanding Wikidata entities and relations with associated aliases. It also validates the novelty of `Falcon 2.0` when compared to Falcon for the Wikidata entity linking.

We observe an indifferent phenomenon in our performance for three datasets, and the performance for `Falcon 2.0` differs a lot per dataset. For instance, on WebQSP-WD, our F-score is 0.82, whereas, on LC-QuAD 2.0, the F-Score drops to 0.57. The first source of error is the dataset(s) itself. In both the datasets (SimpleQuestion and LC-QuAD 2.0), many questions are grammatically incorrect. To validate our claim more robustly, we asked two native English speakers to check the grammar of 200 random questions on LC-QuAD 2.0. Annotators reported that 42 out of 200 questions are grammatically incorrect. Many questions have erroneous spellings of the entity names. For example, `"Who is the country for head of state of Mahmoud Abbas?"` and `"Tell me about position held of Malcolm Fraser and elected in?"` are two grammatically incorrect questions in LC-QuAD 2.0. Similarly, many questions in the SimpleQuestion dataset are also grammatically incorrect. `"where was hank cochran birthed"` is one such example in the SimpleQuestion dataset. `Falcon 2.0` resorts to fundamental principles of the English morphology and finds limitation in recognizing entities in many grammatically incorrect

Table 4.5: Entity Linking Performance of Falcon vs Falcon 2.0 on WEBQSP-WD. Best values are in bold.

Approach	**P**	**R**	**F**
Falcon [68]	0.47	0.45	0.46
`Falcon 2.0`	**0.80**	**0.84**	**0.82**

questions.

We also recognize that the performance of `Falcon 2.0` on sentences with minimal context is limited. For example, in the question "when did annie open?" from the WebQSP-WD dataset, the sentential context is shallow. Also, more than one instance of "Annie" exists in Wikidata, such as Wiki:Q566892 (correct one) and Wiki:Q181734. `Falcon 2.0` wrongly predicts the entity in this case. In another example, "which country is lamb from?", the correct entity is Wiki:Q6481017 with label "lamb" in Wikidata. However, `Falcon 2.0` returns Wiki:13553878, which also has a label "lamb". In such cases, additional knowledge graph context shall prove to be useful. Approaches such as [231] introduced a concept of feeding "entity descriptions" as an additional context in an entity linking model over Wikipedia. Suppose the extra context in the form of entity description (1985 English drama film directed by Colin Gregg) for the entity Wiki:13553878 is provided. In that case, a model may correctly predict the correct entity "lamb." Based on our observations, we propose the following recommendations for the community to improve the entity linking task over Wikidata:

- Wikidata has inherited challenges of vandalism and noisy entities due to crowd-authored entities [232]. We expect the research community to come up with more robust short text datasets for the Wikidata entity linking without spelling and grammatical errors.

- Rule-based approaches come with its limitations when the sentential context is minimal. However, such methods are beneficial for the nonavailability of training data. We recommend a two-step process to target questions with minimal sentential context: 1) work towards a clean and large Wikidata dataset for entity linking of short text. This will allow more robust machine learning approaches to evolve 2) use of entity descriptions from knowledge graphs to improve the linking process (same as [231]).

Experimental Results 3: In the third experiment (for RQ3), we evaluate the relation linking performance of `Falcon 2.0`. We are not aware of any other model

Table 4.6: Relation linking evaluation results on LC-QuAD 2.0 & SimpleQuestion datasets.

Approach	**Dataset**	**P**	**R**	**F**
`Falcon 2.0`	LC-QuAD 2.0	**0.44**	**0.37**	**0.40**
`Falcon 2.0`	SimpleQuestion	**0.35**	**0.44**	**0.39**

Table 4.7: **Sample Questions from LC-QuAD 2.0 datset**. The table shows five sample questions and associated gold standard relations. These sentences do not include standard sentential relations in the English language. Considering Wikidata is largely authored by the crowd, the crowd often creates such uncommon relations. `Falcon 2.0` finds limitation in linking such relations, and most results are empty.

Question	**Gold Standard IDs**	**Gold Standard Labels**	**Predicted IDs**	**Predicted Labels**
Which is the global-warming potential of dichlorodifluoromethane?	P2565	global warming potential	[]	_
What is the AMCA Radiocommunications Licence ID for Qantas?	P2472	ACMA Radiocommunications Client Number	P275	copyright license
What is ITIS TSN for Sphyraena?	P815	ITIS TSN	[]	_
What is the ARICNS for Fomalhaut?	P999	ARICNS	[]	_
Which is CIQUAL 2017 ID for cheddar?	P4696	CIQUAL2017 ID	[]	_

for relation linking over Wikidata. Table 4.6 summarizes relation linking performance. With this, we established new baselines over two datasets for relation linking on Wikidata.

Ablation Study for Relation Linking and Recommendations Falcon reported an F-score of 0.43 on LC-QuAD over DBpedia in Table4.2 whereas `Falcon 2.0` reports a comparable relation linking F-score 0.40 on LC-QuAD 2.0 for Wikidata (cf. Table 4.6). The wrong identification of the entities does affect the relation linking performance, and it is the major source of error in our case for relation linking. Table 4.7 summarizes a sample case study for relation linking on five LC-QuAD 2.0 questions. We observe that the relations present in the questions are highly uncommon and nonstandard, and it is a peculiar property of Wikidata. `Falcon 2.0` finds limitations in linking such relations. We recommend the following:

- Wikidata challenges relation linking approaches by posing a new challenge: user-created nonstandard relations such as in Table 4.7. A rule-based approach like ours faces a clear limitation in linking such relations. Linking user-created relations in crowd-authored Wikidata is an open question for the research community.

4.5 Impact

In August 2019, Wikidata became the first Wikimedia project that crossed 1 billion edits, and over 20,000 active Wikidata editors[14]. A large subset of the information extraction community has extensively relied on its research around DBpedia and Wikidata targeting different research problems such as KG completion, question answering, entity linking, and data quality assessments [168, 233, 234]. Furthermore, entity and relation linking tasks have been studied well beyond information extraction research, especially NLP and Semantic Web. Despite Wikidata being hugely popular, there are limited resources for reusing and aligning unstructured text to Wikidata mentions. However, when it comes to a short text, the performance of existing baselines are limited. We believe the availability of Falcon and `Falcon 2.0` as a web API along with open source access to its code will provide researchers an easy and reusable way to annotate unstructured text against Wikidata. We also believe that a rule-based approach, such as ours that does not require any training data, is beneficial for low resource languages (considering Wikidata is multilingual [15]).

4.6 Adoption and Reusability

Falcon and `Falcon 2.0` are open source. The source code is available in our public GitHub[16] for reusability and reproducibility. Both approaches are easily accessible via a simple CURL request or using our web interface. Detailed instructions are provided on our GitHub. It is currently available for the English language. However, there is no assumption in the approach or while building the background knowledge base that restricts its adaptation or extensibility to other languages. The background knowledge of Falcon and `Falcon 2.0` is available for the community and can be easily reused to generate candidates for entity linking [235] or in question answering approaches such as [236]. The background knowledge consists of 48,042,867 alignments of Wikidata entities and 15,645 alignments for Wikidata predicates. MIT License allows for the free distribution and re-usage of `Falcon 2.0`. We hope the research community and industry practitioners will use `Falcon 2.0` resources for various usages such as linking entities and relations to Wikidata, annotating an unstructured text, developing new low language resources, and others.

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

[15] https://www.wikidata.org/wiki/Help:Wikimedia_language_codes/lists/all

[16] https://github.com/SDM-TIB/Falcon2.0

4.7 Maintenance and Sustainability

`Falcon 2.0` is a publicly available resource offering of the Scientific Data Management(SDM) group at TIB, Hannover[17]. TIB is one of the largest libraries for Science and Technology in the world [18]. It actively promotes open access to scientific artifacts, e.g., research data, scientific literature, non-textual material, and software. Similar to other publicly maintained repositories of SDM, `Falcon 2.0` will be preserved and regularly updated to fix bugs and include new features[19]. The `Falcon 2.0` API will be sustained on the TIB servers to allow for unrestricted free access.

4.8 Summary

In this chapter, we presented Falcon and Falcon2.0, an approach for recognizing and linking named-entities and relations in short text to corresponding Knowledge Graph entities. The proposed approach adopts two novel concepts. First we demonstrated how a fused KG comprising several complimentary semantic and linguistic resources can be employed as background knowledge. Secondly, we devised a linguistic understanding based method for processing the text, that leverages the extended background KG for knowledge extraction tasks. Our comprehensive empirical evaluations provide evidence that the approach outperforms the state-of-the-art on several benchmarks. Additionally, Falcon and Falcon2.0 are offered as online tools as well as APIs. Our approach provides considerable benefits over machine learning based approaches for short text. While the proposed approach achieves better results, it does not require training data and is easily adaptable to new domains. This work has highlighted the importance of background knowledge available in fused KGs as well as the linguistic understanding of the text.

[17] https://www.tib.eu/en/research-development/scientific-data-management/

[18] https://www.tib.eu/en/tib/profile/

[19] `https://github.com/SDM-TIB`

Chapter 5

Neuro-symbolic approach for Named-Entity Recognition and Disambiguation

In the era of digitization, data availability has exponentially grown in recent years, and a similar growth rate is expected in the next decade. Although a large volume of data is presented in structured formats (e.g., relational tables, Knowledge Graphs"KGs"/Bases"KBs"), a vast amount of data is still present in an unstructured format. The problem of bridging the gap between unstructured text and (semi) structured text has attracted the interest of a vast research community [16, 17, 18]. Named-entity Disambiguation (NED) [237] is an essential task in this direction that maps a given entity mentioned in a sentence to the most likely KG/KB entities. It is a well-studied topic in scientific literature and finds applicability in specific domains such as question answering, social media, knowledge base construction, etc [5, 6]. One of the key challenges of entity linking is to resolve the ambiguity of a given text [143]. The problem gets alleviated in specific domains, such as biomedical (aka. medical), due to the necessity of domain knowledge in resolving such ambiguity [151]. We have reported the content of this chapter in this publication [68] and the research work that is under review. The results of this chapter provide an answer to the following research questions:

RQ1: What is the impact of the linguistics rules of a natural language on the tasks of knowledge extraction?

To answer this research question, we present *Noreen*, a neuro-symbolic approach for biomedical entity linking. Current domain-specific entity linking requires the availability of large labeled-training data and face challenges of retraining to each new dataset/setting. *Noreen* combines symbolic and sub-symbolic components to overcome the limitation of available training data. The symbolic component consists of rule-based entity extraction and linking approach backed by a catalog of linguistics and domain-specific rules. The sub-symbolic component increases the accuracy of the symbolic components by employing the knowledge encoded in the definitions of the medical entities. Since the sub-symbolic component only depends on the definitions of the medical entities, our approach does not need any further training data. We have empirically studied the performance of *Noreen* on various benchmarks related to the biomedical domain. The experiments show that our approach improves the state-of-the-art entity linking accuracy over biomedical benchmarks (e.g., from **60.9** (best baseline) to **75.3** (ours) on unseen data accuracy). Furthermore, proposed approach works equally well on sentences and keywords.

RQ2: How does the knowledge represented in community-maintained KGs and domain-specific KBs can be utilized for knowledge extraction tasks?

To answer this research question, we propose a deductive database (background knowledge) built on top of community-maintained KGs and domain-specific KBs. We devise an alignment representation of the knowledge represented in the community-maintained KGs and domain-specific KBs. We exploit different properties of the resources in the used knowledge sources (e.g., labels, semantic types, ...). The deductive database consists of extensional and intensional databases. Furthermore, we evaluate the quality of the extracted knowledge by applying an ablation study where we use the different community-maintained KGs or domain-specific KBs individually or combined. The results show that the knowledge represented in community-maintained KGs and domain-specific KBs empowers knowledge extraction approaches by the knowledge encoded in these knowledge sources. Moreover, the results suggest that the choice of the underlying knowledge source depends on the domain of the studied unstructured data (e.g., the biomedical domain). The devised background knowledge is also utilized in the proposed neuro-symbolic approach used for NER and NED. The sub-symbolic component of the neuro-symbolic approach resorts to the proposed background knowledge to predict the semantic type of an entity in order to improve named-entity disambiguation.

In essence, this chapter makes the following contributions to the problems of NER and NED:

- *Noreen*, the first neuro-symbolic approach for entity linking in biomedical domain (works well both for sentences and keywords). We study the effectiveness of our approach on various benchmarks related to the biomedical domain and significantly outperform state-of-the-art entity linking methods (e.g., from **60.9** (best baseline) to **75.3** (ours) on unseen data accuracy) - Contribution 5 in Chapter 1.

- A semantic-based model able to encode the meaning of an entity encoded in a short text. We define a background knowledge to describe entities from target KGs; it is implemented as a deductive database of safe Horn clauses. [238].
- A rule-based approach able to encode the domain-specific knowledge required for recognizing and linking entities related to a specific domain (e.g., UMLS). We define rules to indicate the linguistic properties and domain characteristics of entities (defined approach in Chapter 3).
- A background knowledge (deductive database) that encodes knowledge from community-maintained KGs and domain-specific KBs (Contribution 2 in Chapter 1).
- Besides source-code, we release public API endpoints for wider use of our approach.

This chapter's structure is as follows: the next Section motivate the work proposed in this chapter. Section 5.2 describes the problem of NER and NED statement and presents in details the components of the proposed neuro-symbolic approach. Section 5.4 explains the experimental settings followed by the observed results. The observed results allow us to answer the research questions **RQ1** and **RQ2**. Finally, we present the closing remarks of this chapter in Section 5.5.

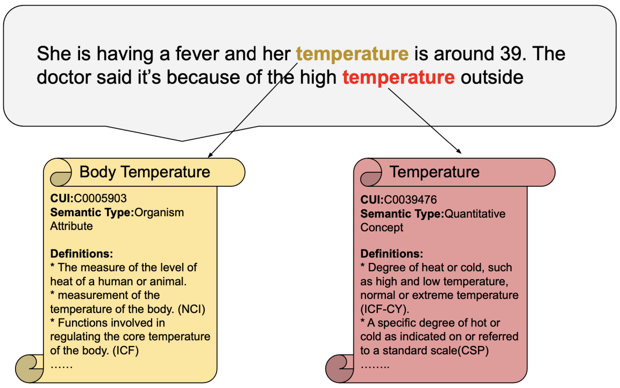

She is having a fever and her temperature is around 39. The doctor said it's because of the high temperature outside
Body Temperature
CUI:C0005903
Semantic Type:Organism Attribute
Definitions:
* The measure of the level of heat of a human or animal.
* measurement of the temperature of the body. (NCI)
* Functions involved in regulating the core temperature of the body. (ICF)
......
Temperature
CUI:C0039476
Semantic Type:Quantitative Concept
Definitions:
* Degree of heat or cold, such as high and low temperature, normal or extreme temperature (ICF-CY).
* A specific degree of hot or cold as indicated on or referred to a standard scale(CSP)
....

Figure 5.1: Exemplar sentence illustrating Biomedical Entity Linking challenges. The surface form "temperature" is used in two different contexts, and with the same label in UMLS, disambiguation will require additional domain-specific knowledge, such as entity descriptions.

5.1 Motivating

Figure 5.1 highlights the ambiguity challenge for NED task that we tackle in this chapter. Here, the surface form "temperature" could be linked to two different concepts that share the same label in the Unified Medical Language System (UMLS) knowledge base [239]; *Body Temperature* and *Temperature* represent two different concepts while sharing the same preferred label (temperature). However, both concepts have different semantic types; *Organism Attribute* for *Body Temperature* and *Quantitative Concept* for *Temperature.* Thus, predicting the semantic type of the surface forms *Body Temperature* and *Temperature* w.r.t to their definitions and the context of the motivating example sentence can solve the described ambiguity challenge.

5.2 Problem Statement

The *Noreen* approach maps the surface forms (aka mentions) within a text into the textual representation of entities in a knowledge graph (KG). This mapping follows a particular strategy which is formalized in the following. Formally, a given short text is a set of tokens $\mathcal{T} = \{t_1, ..., t_n\}$. A KG consists of a set of resources U. $\mathcal{P}(\mathcal{T})$ and $\mathcal{P}(\mathcal{U})$ are, respectively, the power set of tokens and the power set of resources. Mapping the set of tokens in a text to the power set of resources $\mathcal{P}(\mathcal{U})$ is defined by the partial function ($p : \mathcal{P}(\mathcal{T}) \rightarrow \mathcal{P}(\mathcal{U})$). Let ϕ be a function that represents an oracle that correctly associates a set of tokens with resources/entities; $\phi : \mathcal{P}(\mathcal{T}) \rightarrow \mathcal{P}(\mathcal{U})$. Let $acc(.,.)$ be a utility function that assigns a value of accuracy to the linking assigned by $p(.)$ in comparison to the one produced by the oracle $\phi(.)$. The function $p(.)$ should solve the following optimization conditions.
Accuracy of the mappings is maximized

$$\arg \max_{token \in \mathcal{P}(\mathcal{T})} acc(\phi(token), p(token))$$

Maximizing a mapping's number of tokens

$$\arg \max_{\substack{token \in \mathcal{P}(\mathcal{T}) \\ entities \in \mathcal{P}(\mathcal{U}) \\ p(token) = entities}} |token|$$

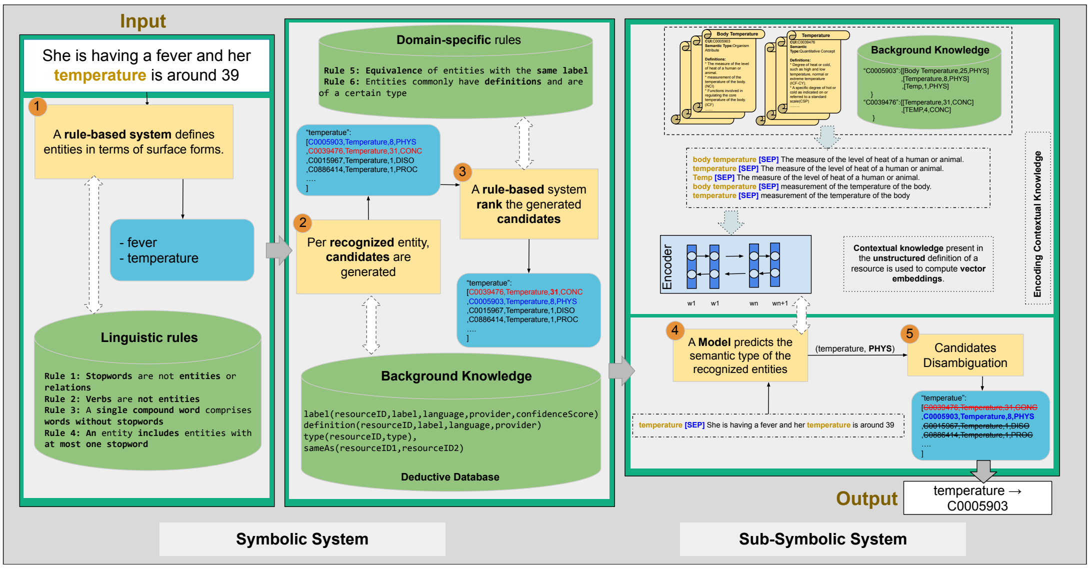

Input
She is having a fever and her temperature is around 39
1
A rule-based system defines entities in terms of surface forms.
- fever
- temperature
Linguistic rules
Rule 1: Stopwords are not entities or relations
Rule 2: Verbs are not entities
Rule 3: A single compound word comprises words without stopwords
Rule 4: An entity includes entities with at most one stopword
Domain-specific rules
Rule 5: Equivalence of entities with the same label
Rule 6: Entities commonly have definitions and are of a certain type
"temperatue":
[C0005903,Temperature,8,PHYS
,C0039476,Temperature,31,CONC
,C0015967,Temperature,1,DISO
,C0886414,Temperature,1,PROC
....
]
2
Per recognized entity, candidates are generated
3
A rule-based system rank the generated candidates
"temperatue":
[C0039476,Temperature,31,CONC
,C0005903,Temperature,8,PHYS
,C0015967,Temperature,1,DISO
,C0886414,Temperature,1,PROC
....
]
Background Knowledge
label(resourceID,label,language,provider,confidenceScore)
definition(resourceID,label,language,provider)
type(resourceID,type),
sameAs(resourceID1,resourceID2)
Deductive Database
Body Temperature
CUI:C0005903
Semantic Type:Organism Attribute
Definitions:
* The measure of the level of heat of a human or animal.
* measurement of the temperature of the body. (NCI)
* Functions involved in regulating the core temperature of the body. (ICF)
......
Temperature
CUI:C0039476
Semantic Type:Quantitative Concept
Definitions:
* Degree of heat or cold, such as high and low temperature, normal or extreme temperature (ICF-CY).
* A specific degree of hot or cold as indicated on or referred to a standard scale(CSP)
....
Background Knowledge
"C0005903":{[Body Temperature,25,PHYS]
,[Temperature,8,PHYS]
,[Temp,1,PHYS]
}
"C0039476":{[Temperature,31,CONC]
,[TEMP,4,CONC]
}
body temperature [SEP] The measure of the level of heat of a human or animal.
temperature [SEP] The measure of the level of heat of a human or animal.
Temp [SEP] The measure of the level of heat of a human or animal.
body temperature [SEP] measurement of the temperature of the body.
temperature [SEP] measurement of the temperature of the body
Encoder
w1
w1
wn
wn+1
Contextual knowledge present in the unstructured definition of a resource is used to compute vector embeddings.
Encoding Contextual Knowledge
4
A Model predicts the semantic type of the recognized entities
(temperature, PHYS)
5
Candidates Disambiguation
temperature [SEP] She is having a fever and her temperature is around 39
"temperatue":
[~~C0039476,Temperature,31,CONC~~
,C0005903,Temperature,8,PHYS
,~~C0015967,Temperature,1,DISO~~
,~~C0886414,Temperature,1,PROC~~
....
]
Output
temperature → C0005903
Symbolic System
Sub-Symbolic System

Figure 5.2: The Noreen architecture with running example. The first component is the symbolic component. With the given sentence, Noreen first extracts the entities' surface form (in-case surface forms are present, first step can be skipped). The next step generates the candidates from the background knowledge (deductive database) from the surface forms. Domain-specific rules decide the rank of the candidates at this step. Now, a pre-trained sub-symbolic component (pre)trained using entity definitions component predicts the entity type. Using entity type, the final ranking of the candidate list generated by the symbolic component is decided for the correct output.

Minimizing a mapping's number of entities

$$\operatorname*{arg\,min}_{\substack{token \in \mathcal{P}(\mathcal{T}) \\ entities \in \mathcal{P}(\mathcal{U}) \\ p(token) = entities}} |entities|$$

5.3 Proposed Solution

The proposed solution is a neuro-symbolic approach that combines symbolic and sub-symbolic components. As a proof of concept, the proposed solution is implemented for the biomedical domain targeting UMLS knowledge base for linking. The symbolic components are the background knowledge built on top of the UMLS knowledge base and the rule-based system used to extract mentions of entities in a text and link them to the target KG. The rule-based system relies on a catalog of linguists and domain-specific rules. The sub-symbolic components are the contextual knowledge encoded in the definitions of biomedical entities extracted from UMLS; an encoder is used to encode the definitions of the biomedical entities to get a vectorial representation of the definitions, and a semantic type prediction model pretrained on top of the vectorial representation of the definitions.

5.3.1 Symbolic System

Background Knowledge: Each resource (entity) in the background knowledge is described by its resourceID, labels, language, semantic type, sameAs link to other KGs, and a heuristic confidence score that counts the number of the shared labels of a resource among different sources or KGs. The sameAs link empowers the background knowledge by integrating knowledge from other KGs different from the target KG (e.g., more synonyms). The heuristic confidence score supports the neuro-symbolic approach in solving ambiguity by assigning a higher score to the resource that shares the same label among more providers (e.g., step 3 in Figure 5.2). A deductive database defines the background knowledge. The background knowledge consists of extensional and intensional databases. The following ground predicates compose the extensional database; they state the main properties of the resources collected in the background knowledge. These properties include labels, definitions, semantic types, and provider.

```
label(resourceID,label,language,provider,confidenceScore).
definition(resourceID,label,language,provider).
type(resourceID,type).
sameAs(resourceID1,resourceID2).
```

The intensional database inductively defines new properties of the resources. It relies on the Leibniz Inference Rule [66] to align the properties of the equivalent resources (i.e., resources connected via sameAs). Rules correspond to Horn clauses where all the variables in the head of a rule, are part of the body (i.e., the rules are safe).

```
sameAs(resourceID1,resourceID2),
label(resourceID1,label,language,provider,confidenceScore)=>
label(resourceID2,label,language,provider,confidenceScore)
```

Intensional rules also model the sameAs properties of reflexivity, associativity, and transitivity.

A Rule-based system defines entities in terms of surface forms (e.g., as illustrated in step 1 in Figure 5.2.). The rule-based system resorts to the catalog of linguistic and domain-specific rules for extracting mentions from the input short text in order to be linked to the target KG (e.g., UMLS). The rule-based system also composes Horn clauses; they are included in Appendix B.1.4.

Linguistic rules state the criteria to recognize entities in a sentence of a particular language. These rules are inspired from [68, 143] As a proof of concept, we have defined the following rules for the English language. Appendix B.1.2 presents the corresponding formalization using Horn clauses.

```
Rule 1:  Stopwords are not entities or relations
Rule 2:  Verbs are not entities
Rule 3:  A single compound word comprises words without stopwords
Rule 4:  Entities with only stopwords between them are one entity
```

Domain-specific rules define what is an entity in a particular domain. As a proof of concept, we resort to the properties of the resources in a knowledge graph, to define entities. Thus, our rules are based on the assumption that entities have labels, definitions, and semantic types. Since these knowledge graphs can be community-maintained, the same resource may have several values of the same property (e.g., various labels or definitions). Additionally, a resource may have equivalent resources. The following rules encode these characteristics commonly present in knowledge graphs like DBpedia, Wikidata, and UMLS. Appendix B.1.3 presents the corresponding formalization using Horn clauses.

```
Rule 5:  Equivalence of entities with the same label
Rule 6:  Entities commonly have definitions and are of a certain type
```

5.3.2 Candidates Generation

For a given sentence and an annotated surface form, this step aims to retrieve all the possible candidates from the background knowledge (deductive database). Sakor et al. [68] proposed an approach to index entity candidates expanded with entity aliases. For example, in Wikidata, the entity Q33[1] has the main label

[1] https://www.wikidata.org/wiki/Q34

"Sweden", and it can be enriched with aliases from Wikidata such as "Kingdom of Sweden", "Konungariket Sverige", "Sverige", and "SE". We adopted this approach and indexed a local KG to generate entity candidates per entity mention for UMLS KB. The local KG (aka background knowledge) has a querying mechanism using BM25[†] algorithm and is ranked by the calculated score. In summary, each index contains information for an entity in terms of its label, semantic type, score, and definition. Once the candidates (top N candidates) are retrieved, we apply domain-specific rules to re-rank them. Now the intermediate-ranked candidate list during inference is passed to the (pretrained) Sub-Symbolic component.

5.3.3 Sub-Symbolic System

Contextual knowledge encoded in the unstructured definition of a resource is used to compute vector embeddings. Per resource, vector embeddings are computed for labels, definitions, and types; all of them are inferred using the extensional and intensional databases in the background knowledge. Embeddings of the same category (i.e., labels, definitions, and types) are combined independently. The aggregated embeddings per category (i.e., labels, definitions, and types) are concatenated into one embedding. One vector embedding is generated per resource; it encodes the contextual knowledge deduced from extensional and intensional databases in the background knowledge.
A Model predicts the semantic type of the recognized entities from the input text; it consists of three encoding steps. First, *coarse-grained* context encoding is created for the input text, e.g., using BERT. Next, a *fine-grained* context encoding is created for the recognized entities; definitions and labels can be used to compute these embeddings. Finally, the coarse and fine-grained context encodings are combined to generate an aggregated embedding. The model compares the generated embeddings of the input short text with the embeddings of the contextual knowledge to predict the semantic types of the recognized entities. Predicting semantic types reduces the ambiguity among the generated candidates. Figure 5.2 illustrates, in step 5, the use of type prediction (by Step 4) to disambiguate the candidate entities (i.e., body and air temperature) based on their semantic types (i.e., Organism Attribute and Quantitative Concept).

5.3.4 Candidates Disambiguation Step

Once we retrieve the types from the sub-symbolic component, the intermediate ranked list from the candidate generation step is further enhanced with type information. As such, we discard all the candidates that do not match the predicted entity type. Hence, the remaining candidates sorted in the top K order are returned based on their highest score (descending order); step 5 in Figure 5.2.

5.4 Experiment and Results

Our goal is to assess the performance of our proposed approach compared to state-of-the-art entity linking methods on the same experiment settings as previous baselines [151]. Three datasets are included in the study (cf., Table 5.1); they are from the biomedical domain. We aim to answer the following research questions. **RQ1:** What is the efficiency of our neuro-symbolic approach for biomedical entity linking? **RQ2:** How do domain-specific rules affect the performance of *Noreen*? **RQ3**: What is the contribution of the sub-symbolic component in our approach?

5.4.1 Datasets

Our first dataset is MedMention [240] comprising 4392 abstracts annotated with 203,282 biomedical research articles. UMLS is the underlying knowledge base. Although UMLS has a small entity definition number overall, we still wanted to understand impact of domain knowledge in neuro-symbolic approach. This was also one of our reason for selecting MedMention as the primary dataset. The validation and test sets have both entities which are present in the training set as well as entities that are zero-shot (never seen at training time). We use the author-recommended ST21pv subset that contains over 42 percent of entities in test time as zero-shot. Please note as our model does not need any training with sentences and associated entities, we run our model on the test set to report the final results. We use seen and unseen settings for a fair comparison with baselines. Considering the dataset is also accompanied by gold entity types, we report results using 1) the type prediction model in the sub-symbolic component of *Noreen*, and 2) when we assume gold type is present. Another dataset is created by doctors from one of the European Medical Schools for Lung Cancer (EMS) using biomedical research abstract containing 323 entity mentions. The entities correspond to 20 types/classes. This dataset aims to understand our model's generalizability on unseen data without re-training sub-symbolic components but in an end-to-end setting (keywords only). We further experiment with another keyword-based dataset to understand the behavior of the *Noreen* approach when no context is present. Thus, we extracted all the 25,007 biomedical entities (common with UMLS) present in Wikidata using a SPARQL query (Appendix B.1.5). These entities have 628 associated entity types and we name the dataset as Wikidata-keyword (aka Wikidata).

5.4.2 Implementation

Noreen is implemented in Python3.7. A pre-trained BERT tuned for the biomedical domain[241] is utilized. The model acts as knowledge-source and is pre-trained

Table 5.1: **Dataset Statistics**. |Mentions| is the number of mentions. |Entities| is the number of unique entities in the labeled partition, not the total Knowledge Base size. For EMS and Wikidata datasets, each surface form correspond to a unique entity in the corresponding knowledge base and it is only a test set.

		MedMentions	**EMS**	**Wikidata**
|Mentions|	Train	120K	-	
	Dev	40K	-	
	Test	40K	323	25007
|Entities|	Train	19K	-	
	Dev	9K	-	
	Test	8K	323	25007

for 10 epochs, we adapt pretraining idea from [156]. The pretraining time is 131hr and 28min. The learning rate, batch size, and max length are 1e-2, 64, and 128, accordingly. The semantic type prediction model is trained using four NVIDIA GeForce RTX 3090 GPUs with 12GB each. The experiments are executed using a server with 723GB RAM, and 96 cores (Intel(R) Xeon(R) Platinum 8160CPU with 2.10GHz). For generating negative samples during training, we generated false samples by replacing the correct semantic type with a wrong semantic type. The wrong semantic type does not share any root or child with the correct semantic type in the hierarchy tree of their semantic groups.

5.4.3 Baselines for Comparison

We compare our results with competitive baselines in biomedical entity linking domain.

1. **TF-idf**: is the widely used candidate retrieval model [151, 242].
2. **BIO-SYN**: uses synonym marginalization technique for biomedical EL [243].
3. **SAPBERT**: presents a self-alignment pretraining for large language models and achieve better results than BioBERT [241].
4. **INDEPENDENT**: is a zero-shot entity linking model by [132] trained on entity descriptions from large knowledge bases.
5. **CLUSTERING-BASED**: enables biomedical entity linking predictions using novel clustering-based approach [151].
6. **SciSpacy**: is based on Spacy library for biomedical text progressing [121].
7. **DUAL**: incorporates mention to mention coreference relationship to assist the entity linking process [155].

Table 5.2: Accuracy on the MedMention dataset (test) using two settings. Baseline values are from [151, 155]. First, in the absence of gold entity types and second (right side) with gold types. The seen subset of test data is when entities are seen during baseline training with corresponding sentences of the document. The unseen test set is the zero-shot setting. For *Noreen* both unseen and seen are the same behavior because we do not train our model with corresponding sentences of the entities. *Noreen* substantially outperforms baselines.

	MedMentions			MedMentions w/ Gold Types		
	Overall	Acc. on		Overall	Acc. on	
	Acc.	Seen	Unseen	Acc.	Seen	Unseen
SciSpacy[121]	38.8	48.3	26.6	47.2	59.8	31
N-GRAM TF-IDF[242]	50.9	50.9	51	67.9	69	64
BIOSYN[243]	72.5	76.5	58.7	77	80.7	64.1
SAPBERT[244]	69.8	72.9	58.9	74.1	77	63.8
INDEPENDENT[132]	72.8	75.9	61.9	76.8	79.2	68.4
CLUSTERING-BASED[151]	74.1	77.3	62.9	79.1	81.5	70.5
DUAL[155]	75.7	79.9	60.9	NA	NA	NA
Noreen (**ours**)	**78.1**	**80.2**	**75.3**	**82.2**	**81.7**	**77.6**

5.4.4 Main Results

Table 5.2 presents our results against the baselines on the MedMention dataset. *Noreen* consistently outperforms all the baselines in both settings (seen/unseen and gold entity types). BERT-based models (SAPBERT and INDEPENDENT) show limited performance. In contrast with these models, which rely on training of a large language model with sentences and labeled entities, our employed BERT model in the sub-symbolic component is not trained with sample sentences and entity mentions. This is a key fundamental difference we bring in this domain against the baselines on how a large language model is pre-tained and later used as knowledge source at inference. *Noreen* performs significantly better than all baseline approaches on unseen data.

In another experiment, we follow another setting in which, with a given mention, the idea is to link the corresponding mentions without providing context. For the same, we use EMS and Wikidata-keyword datasets. For this experiment, we compare against SciSpacy (an end-to-end entity linker) for a fair comparison, as other baselines require training and do not support keyword linking without context. Table 5.3 presents the results where *Noreen* substantially outperforms the baseline. The results indicate the impact of the domain-specific rules on entity linking when no context is present. Similarly, the results reveal the importance

Table 5.3: Generalization of Noreen. In this experiment, we follow an end-to-end setting to link a given keyword to a background knowledge base.

	EMS	**Wikidata**
SciSpacy [121]	41.1	45.9
Noreen	**77.2**	**72.4**

of background knowledge when linking keywords without context. The richness of the background knowledge with various labels collected from different sources and extended by applying the intensional rules empowers the *Noreen* approach to successfully link mentions to the target knowledge graph without context. Furthermore, our approach shows a stable performance to what has been observed on the MedMention dataset. These two experiments successfully answer **RQ1** and **RQ2**.

Ablation Studies

Understanding Contributions of Various Components: For the same, we provide two configurations: 1) *Noreen* wo/ sub-symbolic component: does not have sub-symbolic component. 2) *Noreen* wo/ domain-specific rules: does not have domain-specific symbolic rules. As we observe in Table 5.4, both components complement each other in performance (successfully answering **RQ3**). It is interesting to observe that the sub-symbolic component significantly boosts the performance of *Noreen*. However, when we remove domain-specific rules, the performance also drops. It signifies that in the medical domain, knowledge incorporated in an ML model based on the inputs of subject matter experts matters justifying the symbolic component.

Understanding the Recall: In the second ablation study, we aim to understand *Noreen's* performance for top N candidates. Hence, we calculated recall values at two intervals. Table 5.6 summarizes the results. We observe that for the top 10 candidates, the recall value is 86.3, and it drops to 78.1 for the top1 candidates. In a case study below, we analyze the source of these errors.

Table 5.4: Ablation Study on MedMention Dataset. We clearly see that symbolic and sub-sybolic components complement each other's performance (accuracy).

	MedMentions
Noreen wo/ sub-symbolic component	65.3
Noreen wo/ domain-specific rules	72.6
Noreen	**78.1**

Understanding the end-to-end Accuracy: In this experiment, we analyzes the end-to-end performance of the *Noreen* approach. We discard the gold entity

Table 5.5: End-to-end entity linking performance on MedMention dataset.

	Precision @		Recall @		F-Score @	
	#1	#10	#1	#10	#1	#10
SciSpacy	28.7	36.3	38.8	49.1	32.3	40.8
Noreen	**60.1**	**68.3**	**66.4**	**74.2**	**63.1**	**71.1**

mentions and directly feed the sentence into our model. As baselines used in the paper besides SciSpacy do not perform end-to-end entity linking, we only consider SciSpacy. In Table 5.5, we observe that in spite *Noreen* performing better than baseline, overall results drop compared to table 5.2. We identified that for entity recognition, we relied on similar rules by [68] that caused most errors in recognizing biomedical entities. Other researchers have also pointed out the limitations of these rules for entity recognition as these rules do not consider coherence among the concepts [48]. As a next step, we plan to improve the entity recognition module of *Noreen* using Abstract Meaning Representation techniques that showed promising results on short texts [245].

Case Study: To understand the sources of errors, we randomly sampled 200 failed cases on MedMention. We identify that around 69% percent of failures were due to wrong semantic type prediction. For instance, for the exemplary sentence "Effect of primary health care reforms in Turkey on health service utilization and user satisfaction," our model predicted "Turkey" as a bird. Furthermore, 23 percent of errors were due to the wrong applicability of domain-specific rules, and eight percent were due to linguistic rules. The primary source of error due to domain-specific rules is that our model discards the whole document context and only considers sentence-by-sentence as it does not involves any training with sentences and gold entity mentions. However, we believe techniques proposed in [155] for incorporating coreference knowledge can be potentially used on top of our model. By doing this, our approach will have an additional sub-symbolic component besides the type prediction encoder. We leave this promising research direction to improve the sub-symbolic component in *Noreen* as future work.

Table 5.6: Recall values for Noreen for Entity Disambiguation with given entity mentions.

Recall@	**MedMentions**
1	78.1
10	84.4
20	86.3

5.5 Summary

This chapter presented a neuro-symbolic approach for entity linking, which works equally well for sentences and keywords. As a proof of concept, the proposed approach is implemented for the biomedical domain targeting UMLS KB for linking. The neuro-symbolic approach consists of symbolic and sub-symbolic components. From empirical observations, we conclude that the sub-symbolic component complements the performance of a symbolic system consisting of human-given rule templates, outperforming the black-box neural baselines. Our work successfully discards the necessity of labeled training data and can easily be extended to a completely different KB than UMLS (e.g., Wikidata). Furthermore, the interpretability of our approach allowed us to backtrack the errors at each step.

Chapter 6

Context-aware Framework for Unveiling Semantically related Posts in a Corpus

Capturing knowledge is of paramount relevance to support the new generation of data-driven digital technologies for improving quality of life [246], industrial competitiveness [247], and Web-based health data analysis [174]. Social networking channels have become an information dissemination media for personal discussions, as well as to report relevant scientific research results. In particular, it has become a common practice in the biomedical domain to announce public results of clinical studies on social media channels[1]. These outcomes are relevant for the scientific community and of interest to a broader audience besides the biomedical domain. Given the wealth of knowledge encoded in these announcements, users on social media search to uncover exciting insights, information, and novel findings of trending topics such as COVID-19 and new lung cancer treatments. Nevertheless, effective search and recommendation tools are demanded to support users in hunting the most informative and meaningful social media posts. There are several approaches in the literature for recommending related posts based on the hashtags [57], sentence similarities [60], and by extracting similar concepts or entities in the post [177]. Albeit effective, existing approaches cannot utilize the plethora of knowledge available in publicly available knowledge sources, either encyclopedic or domain-specific. Especially in the biomedical domain of rare diseases such as lung cancer or a newly emerged pandemic like COVID-19, creating and curating sizable labeled training data is extremely challenging to employ deep-learning-based approaches for recommending related posts. Further, while searching, users require

[1] https://twitter.com/WHO/status/1317032089951358977

knowledge about specific terms, such as the name of the drug used for COVID-19 or newly tested interventions for lung cancer. However, deep-learning approaches are not suitable to search in domains, like biomedical relevant content in social media, with scarce training data (Section 6.4). Nonetheless, methods employing the community curated knowledge in Knowledge Graphs (KGs) –e.g., DBpedia [77], Wikidata [14], and Unified Medical Language System (UMLS) [15]– may have a pivotal role for unveiling semantically related posts. We have reported the content of this chapter in this journal article [248]. The results of this chapter provide an answer to the following research question:

RQ3: How contextual knowledge can be used to enhance knowledge extraction over unstructured data?

To anser this research question, we presents PINYON, a knowledge-driven framework, that retrieves associated posts effectively. PINYON implements a two-fold pipeline. First, it encodes, in a graph, a CORPUS of posts and an input post; posts are annotated with entities for existing knowledge graphs and connected based on the similarity of their entities. In a decoding phase, the encoded graph is used to discover communities of related posts. We cast this problem into the Vertex Coloring Problem, where communities of similar posts include the posts annotated with entities colored with the same colors. Built on results reported in the graph theory, PINYON implements the decoding phase guided by a heuristic-based method that determines relatedness among posts based on contextual knowledge, and efficiently groups the most similar posts in the same communities. PINYON is empirically evaluated on various datasets and compared with state-of-the-art implementations of the decoding phase. The quality of the generated communities is also analyzed based on multiple metrics. The observed outcomes indicate that PINYON accurately identifies semantically related posts in different contexts. Moreover, the reported results put in perspective the impact of known properties about the optimality of existing heuristics for vertex graph coloring and their implications on PINYON scalability. This chapter makes the following contributions to the problem of recognizing and linking entities and relations:

- A new technique for discovering semantically related posts in a corpus by mapping the problem of context-aware post recommendation into the Vertex Coloring Problem (Contribution 6 in Chapter 1).
- A heuristic algorithm *PINYON-Context-Aware-Community-Detection(PINYON-CACD)* to efficiently identify highly related posts in various contexts.
- An empirical evaluation of our approach on two different topics demonstrating the effectiveness of our approach for solving the studied problem.

This chapter is structured as follows: Section 6.1 motivates our work by illustrating posts related to an announcement of a promising treatment for lung cancer; Section 6.2 addresses the problem statement of this work and describes formally the proposed solution; Section 6.3 details the implementation of our approach and describes the components of the proposed approach architecture; we present and discuss the outcomes of our empirical evaluations in Section 6.4. Finally, Section 6.5 presents the closing remarks of this chapter and conclude our findings.

6.1 Motivating Example

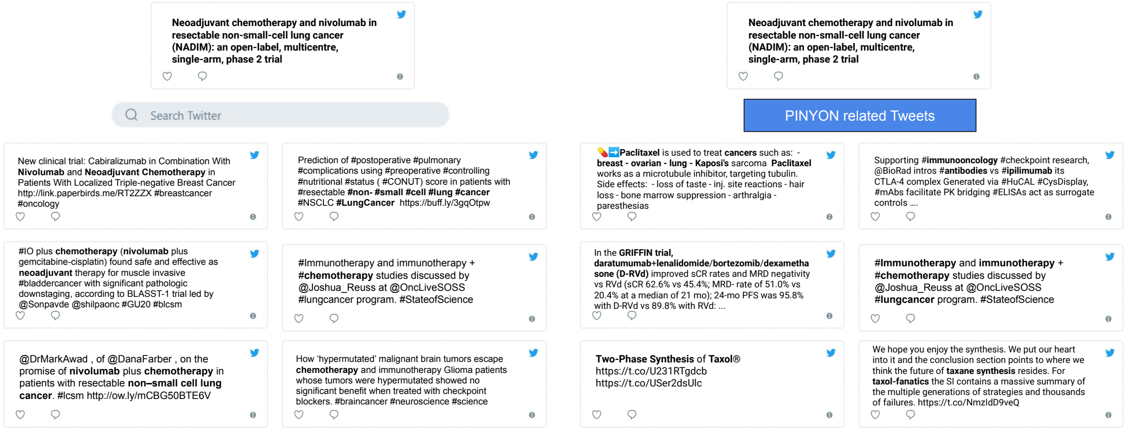
We motivate our work by presenting a post with revolutionary news about a novel treatment that can increase survival probability in patients with non-small-cell lung cancer (NSCLC) in stage IIIA. NSCLC is terminal in most patients with advanced stages of the disease. Effective interventions with the potential of increasing the median survival are celebrated by the patients, their families, and oncologists. The post refers to a scientific article by Provencio et al. [249] published in The Lancet Oncology[2], one of the most prestigious venues in medicine [3]. The post was announced on Twitter from the account of the first author of the work[4] on September 25th, 2020. Since then, it has captured the attention of the scientific community, resulting in 43 Retweets, 10 Quote Tweets, 91 Likes. The novelty of the announced treatment relies on the promising results of assessing antitumor activity when neoadjuvant therapies are applied. They combine chemotherapy drugs (Paclitaxel and Carboplatin) plus an immunological drug (Nivolumab) before surgery. Then, this treatment is followed by adjuvant intravenous monotherapy (also with Nivolumab) for one year. The evaluation was conducted in a cohort of 46 patients who received neoadjuvant therapy. 41 (89%) of 46 patients had surgery, and at the time of the publication, these patients were alive and free of recurrence, with a median follow-up of 24·0 months. The authors claim the novelty of assessing the effectiveness of neoadjuvant Chemoimmunotherapy in NSCLC patients in stage IIIA. More importantly, the referred results support the hypothesis about the efficacy of neoadjuvant Nivolumab and platinum-based Chemotherapies, and potentially represents a paradigm shift in treating lung cancer patients with advance stages of the disease.

Figure 6.1 presents the original tweet; given the relevance of the announced results, it is of great interest to retrieve posts that announce similar results either

[2] `https://www.thelancet.com/journals/lanonc/home`

[3] A follow-up version of this work have been presented in ASCO2022 `https://s3.amazonaws.com/files.oncologymeetings.org/prod/s3fs-public/2022-05/AM22-Lung-Cancer-Non-Small-Cell-Local-Regional-Small-Cell-Thoracic-Cancers.pdf?null`

[4] `https://twitter.com/MARIANOPROVENCI/status/1309355589676535810`



Neoadjuvant chemotherapy and nivolumab in resectable non-small-cell lung cancer (NADIM): an open-label, multicentre, single-arm, phase 2 trial

New clinical trial: Cabiralizumab in Combination With **Nivolumab** and **Neoadjuvant Chemotherapy** in Patients With Localized Triple-negative Breast Cancer http://link.paperbirds.me/RT2ZZX #breastcancer #oncology

Prediction of #postoperative #pulmonary #complications using #preoperative #controlling #nutritional #status (#CONUT) score in patients with #resectable **#non- #small #cell #lung #cancer** #NSCLC **#LungCancer** https://buff.ly/3gqOtpw

#IO plus **chemotherapy** (**nivolumab** plus gemcitabine-cisplatin) found safe and effective as **neoadjuvant** therapy for muscle invasive #bladdercancer with significant pathologic downstaging, according to BLASST-1 trial led by @Sonpavde @shilpaonc #GU20 #blcsm

#Immunotherapy and immunotherapy + **#chemotherapy** studies discussed by @Joshua_Reuss at @OncLiveSOSS #lungcancer program. #StateofScience

@DrMarkAwad , of @DanaFarber , on the promise of **nivolumab** plus **chemotherapy** in patients with resectable **non–small cell lung cancer**. #lcsm http://ow.ly/mCBG50BTE6V

How 'hypermutated' malignant brain tumors escape **chemotherapy** and immunotherapy Glioma patients whose tumors were hypermutated showed no significant benefit when treated with checkpoint blockers. #braincancer #neuroscience #science

(a) Twitter Search

Neoadjuvant chemotherapy and nivolumab in resectable non-small-cell lung cancer (NADIM): an open-label, multicentre, single-arm, phase 2 trial

💊➡️**Paclitaxel** is used to treat **cancers** such as: - **breast - ovarian - lung - Kaposi's** sarcoma **Paclitaxel** works as a microtubule inhibitor, targeting tubulin. Side effects: - loss of taste - inj. site reactions - hair loss - bone marrow suppression - arthralgia - paresthesias

Supporting **#immunooncology** #checkpoint research, @BioRad intros **#antibodies** vs **#ipilimumab** its CTLA-4 complex Generated via #HuCAL #CysDisplay, #mAbs facilitate PK bridging #ELISAs act as surrogate controls ....

In the **GRIFFIN trial, daratumumab+lenalidomide/bortezomib/dexamethasone (D-RVd)** improved sCR rates and MRD negativity vs RVd (sCR 62.6% vs 45.4%; MRD- rate of 51.0% vs 20.4% at a median of 21 mo); 24-mo PFS was 95.8% with D-RVd vs 89.8% with RVd: ...

#Immunotherapy and **immunotherapy** + **#chemotherapy** studies discussed by @Joshua_Reuss at @OncLiveSOSS **#lungcancer** program. #StateofScience

Two-Phase Synthesis of **Taxol®**
https://t.co/U231RTgdcb
https://t.co/USer2dsUlc

We hope you enjoy the synthesis. We put our heart into it and the conclusion section points to where we think the future of **taxane synthesis** resides. For **taxol-fanatics** the SI contains a massive summary of the multiple generations of strategies and thousands of failures. https://t.co/NmzldD9veQ

(b) PINYON related Tweets

Figure 6.1: Motivating Example. A tweet announcing the promising results of combining Neoadjuvant chemotherapy and Nivolumab in patients with Non-small-cell lung cancer in stage IIIA. a) Related tweets resulting from searching using the Twitter API the criteria *Neoadjuvant chemotherapy and Nivolumab* and *Resectable non-small-cell lung cancer*. b) Results of semantically related tweets based on PINYON. Tweets retrieved from Twitter include at least one of the keywords in the input post. Tweets identified by PINYON are semantically related to the input post based on the shared context.

in lung cancer or in other types of cancers. Social media platforms (e.g., Twitter) provide searching capabilities by indexing relevant entities in the post or using user annotations like hashtags. Figure 6.1a presents six tweets identified by Twitter API for two search criteria, *"Neoadjuvant chemotherapy and nivolumab"* and *"resectable non-small-cell lung cancer"*. As expected, the output includes 1. tweets discussing the prescription of chemotherapy and nivolumab in other cancers (e.g., breast, bladder, or brain tumors), or 2. outcomes of non-small-cell lung cancer treatments. Despite these results' relevance, they only include posts whose text comprises at least one of the keywords in the search criteria because the search depends on the hashtags and keywords (annotation-based) mentioned in the tweet. Following such search strategy affects the richness of the knowledge encoded in the retrieved posts because it is only limited within the keywords and hashtags without considering any semantics of the entities mentioned in a post. Contrary, PINYON exploits knowledge encoded in the context of the input post. This knowledge composes PINYON background knowledge. It is extracted from existing encyclopedic knowledge graphs like DBpedia or Wikidata, and domain-specific ones like Unified Medical Language System (UMLS). Furthermore, PINYON employs a graph coloring algorithm to identify semantically similar communities of posts for a relevant context. As a result, PINYON can identify the tweets that exactly match the terms in the input text (Figure 6.1b). More importantly, it outputs tweets that refer to treatments that combine other novel oncological treatments (e.g., GRIFFIN trials, synthesis of Taxol, or immunoncology) and are used to treat other various diseases (e.g., sarcoma and breast, ovarian, and lung cancer). Given the wealth of information present in posts announced in social media, the possibility of retrieving the posts semantically related to a given announcement, represents a fundamental change in post recommendation towards more informative and meaningful outcomes.

6.2 Problem Statement and Solution

This section formally describes the problem statement and the approach implemented in PINYON. Please refer to Table 6.1 for better understanding of the used annotation.

6.2.1 Preliminaries

The Vertex Coloring Problem

The vertex coloring problem corresponds to the coloring of the vertices in a graph *G=(V,J)* with the minimal number of colors such that adjacent vertices are colored with distinct colors; where V is the set of vertices (or nodes) of the graph and J is the set of edges (or links) of the graph. Formally, let $\mathcal{SC}$ be a set of colors and $\mu(.)$ is a mapping from V to $\mathcal{SC}$. The function $\mu(.)$ is a solution to the vertex coloring problem for G if $\mu(.)$ is defined as follows:

- Adjacent vertices are in distinct colors, i.e., if v_i and $v_j \in V$ are adjacent then, $\mu(v_i) \neq \mu(v_j)$.

- Number of colors in $\mu(.)$ is minimized, i.e., the optimization objective is formally defined as follows
  $\arg\min_{\mathcal{USC} \subseteq \mathcal{SC}} \mid \{\mu(n_i)/n_i \in V \wedge \mu(n_i) \in \mathcal{USC}\} \mid$

Lemma 6.2.1. *Let G=(V,J) be a graph. Let $\mathcal{SC}$ be a set of colors available to color G=(V,J) and $\mu(.)$ be a mapping from V to $\mathcal{SC}$ that corresponds to a solution to the vertex coloring problem for G=(V,J). Let v, $deg_{G(v)}$, and $\mu(v)$ be a vertex in J, the degree of v in G, and $\mu(v)$ the color of v, respectively. The node v is the only vertex in V with the color $\mu(v)$ if and only if $deg_{G(v)}$ is equal to $|J|$-1.*

Proof of 6.2.1 is presented in Appendix C.1.1.

Table 6.1: Summary of PINYON Notation

Notation	**Explanation**
$BP=(V_1 \cup V_2,E)$	Bipartite Graph, $V_1 \cap V_2=\emptyset$, $E \subseteq V_1 \times V_2$
$LP(G)=(F,T)$	$LP(G)$ is a graph, and it is the *line graph* of $G=(V,J)$, $\|J\|=\|F\|$, $\|T\|=\|\{(e_q,e_k)/e_q,e_k \in J \wedge (e_q = (v_i,v_z) \wedge e_k = (v_z,v_j)) \vee (e_q = (v_z,v_i) \wedge e_k = (v_z,v_j))\}\|$
$Comp(G)=(V,K)$	$Comp(G)$ is a graph. It is the complement graph of $G=(V,J)$ iff $K=((V\times V)-J)$
$\mathcal{SC}$	Colors available to color a graph $G=(V,J)$ using a function $\mu(.)$ from V to $\mathcal{SC}$. $\mathcal{USC}$ subset of $\mathcal{SC}$ used in $\mu(.)$
$\mathcal{P}$	An input post expressed in terms of entities (words or tokens) annotated with terms from various contexts
$\mathcal{R}$	A database of posts p also described with a set $p_e=\{en_1,\ldots,en_m\}$ of entities
$\mathcal{C}$	A set of contexts c modeled as contextual knowledge graphs (KGs)
$PostRelated(\mathcal{P},\mathcal{R}',c)$	Metric for the semantic relatedness of the posts in $\mathcal{R}'$ and $\mathcal{P}$ in the context c
$\mathcal{P}_c(\mathcal{R})$	Partition of $\mathcal{R}$, i.e., grouping of the elements into non-empty subsets, where every element is in exactly one subset
$PostRelated(\mathcal{P},\mathcal{P}_c(\mathcal{R}),c)$	Overall value of $PostRelated(\mathcal{P},\mathcal{R}',c)$ for all the $\mathcal{R}'$ in $\mathcal{P}_c(\mathcal{R})$ by using a triangular norm
$\delta(p,e_i,c)$	For entity e_i in p_e and a context c, returns the set of terms in c
$e_{i,j}=(t_i,t_j)$	t_i and t_j and terms, and $e_{i,j}$ is an edge in $BP=(V_1 \cup V_2,E)$ and a vertex in $LP(BP)=(F,T)$
$\rho_\gamma(e_{i,j},e_{z,q})$	Quantifies similarity of terms t_i,t_j,t_z, and t_q based on a similarity measure γ
$\mathcal{SP}_c(\mathcal{R})$	All the possible partitions of $\mathcal{R}$
$ComS(.)$	Overall relatedness among the terms in the vertices colored with the same color
$ColoredSimilarity(\mu(.))$	Aggregates $ComS(.)$ for all colors in $\mu(.)$
Degree of similarity $\upsilon(e_{i,j})$	Aggregates values of the similarity of $e_{i,j}$ with the rest of other vertices
SCom	*PINYON-CACD* communities
CAC_c	A context-aware community for a context c
$GComS(.)$	The Global community similarity is defined using a triangular norm among the community similary of the communities in CAC

Line, Complement, and Bipartite Graphs

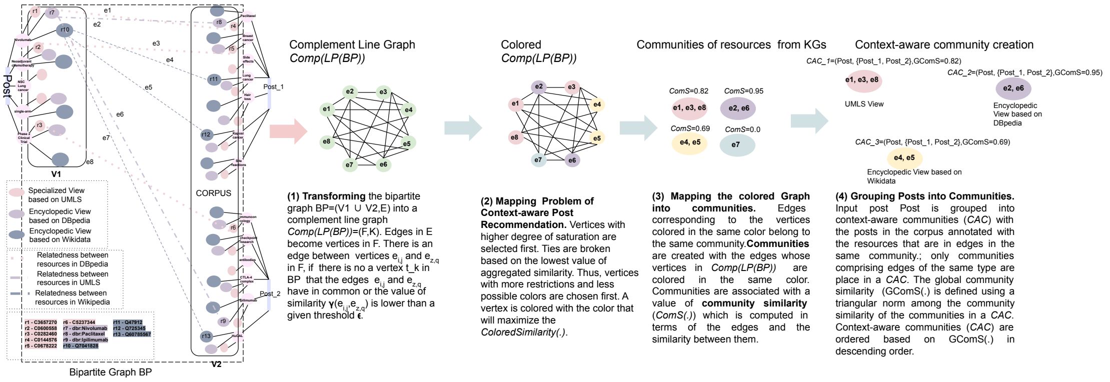
Line Graph: Given a graph *G=(V,J)* such that $J \subseteq V \times V$, the *line graph LP(G)=(F,T)* of G comprises a) a vertex f_{e_q} in F per each edge e_q in J, and b) an edge (f_{e_q},f_{e_k}) in T if e_q and $e_k \in J$ and share a vertex in common, i.e., the following edges belong to J: e_q=(v_i, v_z) and e_k=(v_j, v_z), or e_q=(v_z, v_i) and e_k=(v_z, v_j).
Complement Graph: Given a graph *G=(V,J)*, the *complement graph* of G is a graph *Comp(G)=(V,K)*, where vertices of G and *Comp(G)* are the same, and K is the complement of J, i.e., *K=((V× V)-J)*.
Bipartite Graph: A *bipartite graph BP=(*$V_1 \cup V_2$*,E)* comprises vertices in $V_1 \cup V_2$ and edges are in $E \subseteq V_1 \times V_2$; the intersection of V_1 and V_2 is empty, i.e., $V_1 \cap V_2 = \emptyset$.

6.2.2 The DSATUR Algorithm

The Vertex Coloring Problem is NP-hard [191], and various approximate algorithms have been proposed to provide efficient solutions to tractable instances of the problem [250]. DSATUR [191] employs an algorithm that colors each vertex of the graph once, using a heuristic to select the colors. Assuming that we have a graph $G = (V, E)$, DSATUR dynamically orders the vertices in V depending on the number of different colors appointed to the adjacent vertices of each vertex in V, i.e., vertices are picked based on the degree of saturation on the partial coloring of the graph created so far; only adjacent vertices that are already colored are taken into account. Intuitively, choosing a vertex with the highest degree of saturation enables one to color first those vertices with more restrictions and smaller sets of colors available. Ties are broken depending on the maximum vertex degree of the tied vertices, i.e., the number of neighboring nodes colored or not; DSATUR has a time complexity of $O(|V^3|)$. Furthermore, the optimality requirements of the suggested algorithms have attracted attention in past years; features of the graphs that are difficult to color, in terms of time complexity, for each algorithm [251]. Thus, DSATUR colors most k-colorable graphs optimally, i.e., k is the number of ideal colors, G is k-colorable, and $UsedColors(G) \leq k$. The propositions described in Appendix C.1.2 list graphs for which DSATUR is optimum [252].

6.2.3 Problem Statement

Given an input post $\mathcal{P}$, a dataset $\mathcal{R}$ of posts, and a set of contexts $\mathcal{C}$, we tackle the problem of *identifying the minimal groups of posts* in $\mathcal{R}$ that maximize the *context-aware relatedness* to $\mathcal{P}$ according to each of the contexts c in $\mathcal{C}$. Formally, assume that given a subset $\mathcal{R}'$ of $\mathcal{R}$, and a context c from $\mathcal{C}$, *PostRelated(*$\mathcal{P}$*,*$\mathcal{R}'$*,c)* is a metric that quantifies the semantic relatedness of the posts in $\mathcal{R}'$ and $\mathcal{P}$



(1) Transforming the bipartite graph BP=(V1 ∪ V2,E) into a complement line graph *Comp(LP(BP))*=(F,K). Edges in E become vertices in F. There is an edge between vertices $e_{i,j}$ and $e_{z,q}$ in F, if there is no a vertex t_k in BP that the edges $e_{i,j}$ and $e_{z,q}$ have in common or the value of similarity $\boldsymbol{\gamma}(e_{i,j},e_{z,q})$ is lower than a given threshold $\boldsymbol{\epsilon}$.

(2) Mapping Problem of Context-aware Post Recommendation. Vertices with higher degree of saturation are selected first. Ties are broken based on the lowest value of aggregated similarity. Thus, vertices with more restrictions and less possible colors are chosen first. A vertex is colored with the color that will maximize the *ColoredSimilarity(.)*.

(3) Mapping the colored Graph into communities. Edges corresponding to the vertices colored in the same color belong to the same community.**Communities** are created with the edges whose vertices in *Comp(LP(BP))* are colored in the same color. Communities are associated with a value of **community similarity** (*ComS(.)*) which is computed in terms of the edges and the similarity between them.

(4) Grouping Posts into Communities. Input post Post is grouped into context-aware communities (*CAC*) with the posts in the corpus annotated with the resources that are in edges in the same community.; only communities comprising edges of the same type are place in a *CAC*. The global community similarity (GComS(.) is defined using a triangular norm among the community similarity of the communities in a *CAC*. Context-aware communities (*CAC*) are ordered based on GComS(.) in descending order.

Figure 6.2: **Running Example**. Mapping the Problem of Context-Aware Post Recommendation into the Vertex Coloring Problem. Entities in a post are annotated with terms in different contexts (e.g., domain-specific or encyclopedic), relatedness between the terms in the same context are represented by edges of a Bipartite Graph *BP=(V_1 ∪ V_2,E)*; a similarity measure quantifies relatedness of the terms in a context. (1) The complement of the line graph of BP is computed *Comp(LP(BP))=(F,K)*. (2) *Comp(LP(BP))=(F,K)* is colored with the minimal number of colors that maximize the value of *ColoredSimilarity(.)*, (3) Colored Graph in mapped into Communities. (4) Communities are used to create Context-Aware Communities of Posts.

in the context c. A solution to the *context-aware post recommendation* problem (aka *CWPR*) corresponds to a partition $\mathcal{P}_c(\mathcal{R})$ of $\mathcal{R}$ such that the overall value of *PostRelated*($\mathcal{P}$,$\mathcal{R}'$,c) over all the parts $\mathcal{R}'$ of $\mathcal{P}_c(\mathcal{R})$ is maximized while the number of parts in $\mathcal{P}_c(\mathcal{R})$ is minimized. Suppose *PostRelated*($\mathcal{P}$,$\mathcal{P}_c(\mathcal{R})$,$c$) corresponds to the overall value of relatedness and is computed by combining the values of *PostRelated*($\mathcal{P}$,$\mathcal{R}'$,c) for all the $\mathcal{R}'$ in $\mathcal{P}_c(\mathcal{R})$ by using a triangular norm (aka t-norm) $\mathcal{T}(.,.)$ [253]. Formally, if $\mathcal{SP}_c(\mathcal{R})$ represents all the possible partitions of $\mathcal{R}$, a solution for *CWPR* is a $\mathcal{P}_c(\mathcal{R})$ with minimal cardinality and satisfying the condition:

$$\arg\max_{\mathcal{P}_c(\mathcal{R})\subseteq\mathcal{SP}_c(\mathcal{R})} \mathit{PostRelated}(\mathcal{P}, \mathcal{P}_c(\mathcal{R}), c)$$

6.2.4 Proposed Solution

The proposed solution (PINYON) resorts to existing knowledge bases (e.g., DBpedia, Wikidata, and UMLS) to represent encyclopedic or domain-specific contexts. Moreover, our proposed solution assumes that the input post $\mathcal{P}$ and the posts in $\mathcal{R}$ are annotated with the terms of these knowledge bases to encode context-aware knowledge expressed in a post. Formally, a post p is described in terms of a set p_e

of words or tokens that represent entities, i.e., p_e=$\{en_1, \ldots, en_m\}$ indicates that p is expressed in terms of the m entities in p_e. Additionally, p is associated with a function $\delta(p, e_i, c)$ that, for each entity e_i in p_e and a context c, returns the set of terms in the knowledge base c that represent e_i in the context modeled by c. To illustrate, Figure 6.3 presents the five entities in p_e in the post in the motivating example Figure 6.1, and the terms from DBpedia, Wikidata, and UMLS that represent these entities. Similarly, the posts in $\mathcal{R}$ are expressed in terms of entities and terms from contextual knowledge bases. A bipartite graph *BP=(*$V_1 \cup V_2$*,E)* embodies relatedness between the terms that describe the input post $\mathcal{P}$ and the posts in $\mathcal{R}$. *BP* is defined as follows:

- V$_1$ represents the contextual description of $\mathcal{P}$, i.e., V$_1$ is equal to terms in the union of $\delta(\mathcal{P}, en_i, c)$, for all the entities en_i in $\mathcal{P}_e$=$\{en_1, \ldots, en_m\}$ and the knowledge bases c in $\mathcal{C}$.

- V$_2$ corresponds to the contextual description of posts in $\mathcal{R}$, i.e., V$_2$ comprises all the terms in $\delta(p, en_i, c)$ for all p in $\mathcal{R}$, its entities in p_e and contexts in $\mathcal{C}$.

- E encodes context-aware relatedness among $\mathcal{P}$ and $\mathcal{R}$ and edges in E meet the following conditions:

  - Associate terms in the same context, i.e., if (t_i, t_j) belongs to E then, t_i and t_j are terms of the same knowledge base.
  - Relate terms semantically similar, i.e., if (t_i, t_j) belongs to E then, for a given similarity measure γ, $\gamma(t_i, t_j)$ is equal or greater than a given threshold ϵ.

A partition $\mathcal{P}_c(\mathcal{R})$ is computed from a partition of the edges in *BP* formulated as a solution of the Vertex Coloring Problem (*VC*). The mapping of *CWPR* to *VC* is defined as follows.

Bipartite Graph Transformation: The bipartite graph *BP* for $\mathcal{P}$ is transformed into an undirected line graph *LP(BP)=(F,T)* as follows: 1. Edges representing context-aware relationships between terms in *BP* are modeled as vertices in *LP(BP)*, i.e., there exists a vertex $e_{i,j}$ in F iff there exists an edge $e_{i,j} = (t_i, t_j)$ in E. 2. Co-occurrence of a term in several posts modeled as terms that appear in several edges in *BP*. For each pair of different edges $e_{i,j} = (t_i, t_j)$ and $e_{z,q} = (t_z, t_q)$ in *BP*, such as t_i=t_z or t_j=t_q, there is an edge between the vertices $e_{i,j}$ and $e_{z,q}$ in *LP(BP)*.

The degree of a vertex $e_{i,j}$ in *LP(BP)* that represent the edge $e_{i,j} = (t_i, t_j)$ in *BP* is equal to $\binom{out(t_i)}{2} + \binom{in(t_j)}{2}$, where $out(t_i)$ and $in(t_j)$ represent the out-degree and in-degree of the vertices in *BP*, respectively.

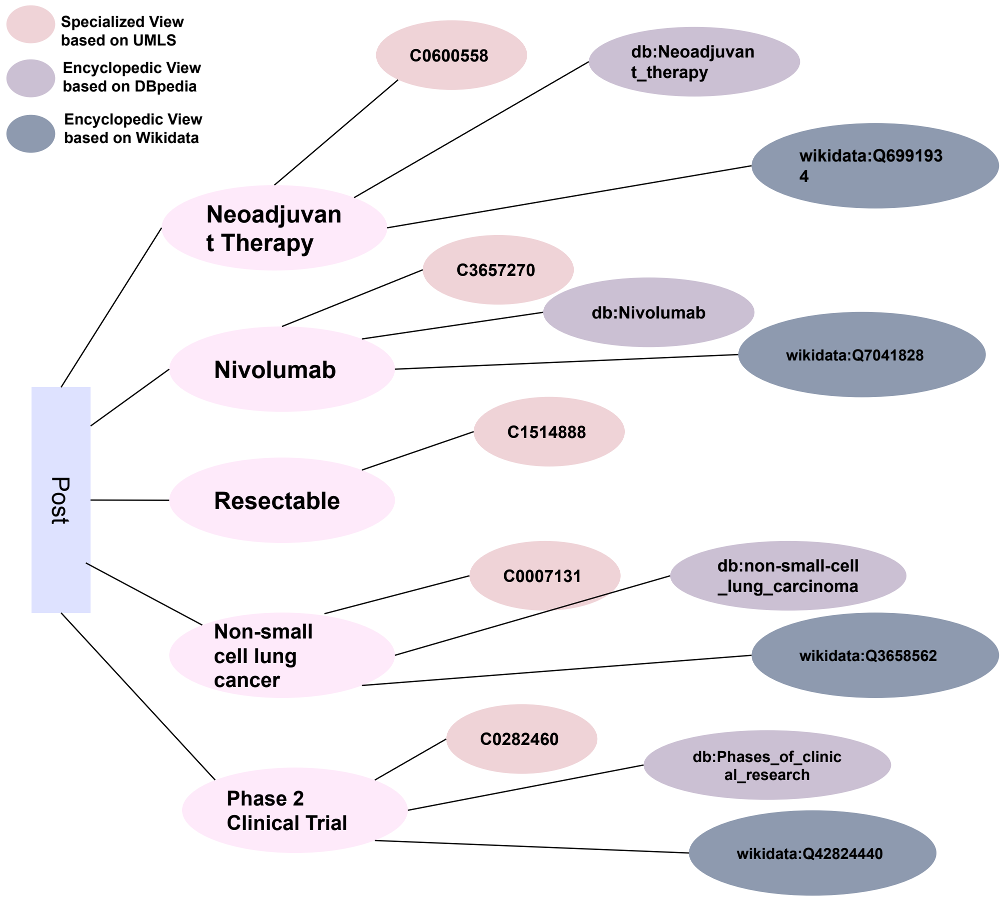


Figure 6.3: Running Example. Entities in a post are annotated with concepts in various contexts (e.g., domain-specific or encyclopedic) derived from the knowledge graphs.

Complement Graph Creation: The complement graph of $LP(BP)$=(F,T) corresponds to an undirected graph $Comp(LP(BP))$=(F,K) with the same vertices of $LP(BP)$ and with the complement edges of $LP(BP)$, i.e., K=$(F \times F) - T$. The degree of a vertex $e_{i,j}$ in $Comp(LP(BP))$ that represent the edge $e_{i,j} = (t_i, t_j)$ in BP is equal to $\binom{|V_2|-out(t_i)}{2} + \binom{|V_1|-in(t_j)}{2}$, where $|V|$ corresponds to the cardinality of a set V. Moreover, given the edges $e_{i,j} = (t_i, t_j)$ and $e_{z,q} = (t_z, t_q)$ in BP, a function $\rho_\gamma(e_{i,j}, e_{z,q})$ quantifies the similarity of the terms t_i,t_j,t_z, and t_q based on a similarity measure γ; $\rho_\gamma(e_{i,j}, e_{z,q})$ corresponds to the result of applying a t-norm $\mathcal{T}(\gamma(t_i, t_z), \gamma(t_j, t_q))$.

Finding Communities of Semantically Related Posts: The Vertex Coloring Problem is solved over $Comp(LP(BP))$. However, the concept of number of colors used during the coloring of the vertices of *Comp(LP(BP))* is redefined to ensure that the color assignment both minimizes the numbers of colors and the overall value of relatedness among the terms in the vertices colored with the same color (i.e., *ComS(.)*). Thus, the number of colors in $\mathcal{USC}$ for a solution $\mu(.)$ of the vertex coloring of *Comp(LP(BP))* corresponds to 1-*ColoredSimilarity($\mu(.)$)*, where *ColoredSimilarity($\mu(.)$)* is defined as follows:

- Let $\mathcal{P}(F)$ be the partition of the vertices of *Comp(LP(BP))* such that all the vertices in one part of $\mathcal{P}(F)$ are colored with the same color in $\mu(.)$, and each part is colored in a different color. We call each part *pa* a community, and the community similarity *ComS(pa)* corresponds to the results of applying a t-norm $\mathcal{T}$ over all the unordered pair pairs $(e_{i,j}, e_{z,q})$ of $\rho_\gamma(e_{i,j}, e_{z,q})$. Note that a pair $(e_{i,j}, e_{z,q})$ is considered unordered because $\rho_\gamma(.,.)$ is assumed to be symmetric.

- *ColoredSimilarity($\mu(.)$)* aggregates the values of the community similarity *ComS(.)* of all communities in $\mathcal{P}(F)$. A t-norm or the average can be used to compute this aggregated value.

The PINYON-CACD Solution: *PINYON-CACD* is used to color *Comp(LP(BP))*. It follows a heuristic to color the vertices and meet the condition of minimizing the number of used colors $\mathcal{USC}$ as defined previously; it resembles the coloring heuristic proposed by Palma et al. [183] and implemented by the DSATUR algorithm. In addition to the degree of saturation, each vertex is associated with a *degree of similarity*, which represents how much similar the vertex is with respect to the rest of the vertices in the graph. The *degree of similarity* of a vertex $e_{i,j}$, aka $\upsilon(e_{i,j})$, is computed as the aggregated value of the similarity of $e_{i,j}$ with the rest of the vertices; the aggregation function can be the average (e.g., arithmetic or geometric mean) or a triangular norm.

PINYON-CACD in A Nutshell: Intuitively, a vertex with a high degree of similarity is highly similar to many of the vertices in the graph. Vertices are ordered based on degree, and the one with the maximal degree is chosen first. In case of ties, i.e., at least two vertices have the same degree, the one with the lowest value of similarity is selected first. Thus, *PINYON-CACD* starts coloring the vertex *with more restrictions.* Then, *PINYON-CACD* iteratively traverses the list of ordered vertices and chooses the one which the highest degree of saturation and in case of ties, the vertex with the lowest *degree of similarity* is chosen. This decision also enables *PINYON-CACD* to select the vertex with fewer options to be colored. Once a vertex is chosen, a color is selected; it is among the suitable colors the one that maximizes the *ColoredSimilarity(.)* of the assignment $\mu(.)$ of colors created so far. *PINYON-CACD* finalizes when all the vertices are colored; it creates a community per used color. Each community comprises all the vertices colored with the same color and is described in terms of the community similarity *ComS(.)*.

Running Example-PINYON-CACD: Figure 6.2 illustrates in the steps (1), (2), and (3). **Step (1):** a bipartite graph *BP=(*$V_1 \cup V_2$*,E)* is transformed into a complement line graph *Comp(LP(BP))*. Thresholds of similarity values are utilized to decide when two entities are similar or not; entities collected from different contexts are not similar. In settings with unrelated entities, *Comp(LP(BP))* can include numerous edges, i.e., it may be a complete graph. **Step (2):** *Comp(LP(BP))* is colored. The color assignment optimality depends on the topology of the *Comp(LP(BP))*. Lemma 6.2.1 and propositions in subsection 6.2.2 state the graph topologies where the coloring is optimal. Thus, the resulting partition of the edges in *Comp(LP(BP))* maximizes the value of *ColoredSimilarity(.)*. **Step (3):** the colored *Comp(LP(BP))* is utilized to generate communities; a community only comprises edges connecting highly similar terms from the same context. Each community *pa* is associated with *ComS(pa)*; Figure 6.2 illustrates four partitions resulting from the execution of steps (1), (2), and (3). **Step (4):** posts are grouped into context-aware communities *(CACs)* composed of the input post $\mathcal{P}$ and the posts in $\mathcal{R}$ that are highly related to $\mathcal{P}$ in a given context c. Note that *PostRelated(*$\mathcal{P}$,$\mathcal{P}_c(\mathcal{R})$*,c)* corresponds to the result of applying an aggregation function over all the values of *ComS(pa)*, where *pa* is a community of edges associating terms of the context c.

Context-Aware Communities: A context-aware community for a context c, CAC_c, is defined inductively as follows:

Base Case. Simple Community $CAC_c(\{cc\})$

For $cc = \{e_1, ..., e_m\}$ in *SCom*, $CAC_c(\{cc\})$=($\mathcal{P}$,$\mathcal{SP}$,ComS(cc)), where $\mathcal{SP}$ corresponds to the posts in $\mathcal{R}$ annotated with at least one $t_{i,k}$ from an edge $e_i = (t_{i,j}, t_{i,k})$ in *cc*.

Inductive Case. Composed Community $CAC_c(C_{r,s})$

$CAC_c(C_{r,s})$=($\mathcal{P}$,$\mathcal{SP}$,GComS($C_{r,s}$)) is created from
$CAC_c(C_r)$=($\mathcal{P}$,$\mathcal{SP}$,GComS(C_r)) and
$CAC_c(C_s)$=($\mathcal{P}$,$\mathcal{SP}$,GComS(C_s)),
where $C_{r,s}$=$C_r \cup C_s$ and *GComS(*$C_{r,s}$*)* is the aggregated value of *GComS(*C_r*)* and *GComS(*C_s*)*.

Running Example-PINYON-CACD (Cont.): Figure 6.2 illustrates, in step (4), the context-aware communities created from the running example. Three communities are created, one per context, i.e., CAC_1 for UMLS, CAC_2 for DBpedia, and CAC_3 for Wikidata. Note that describing each community CAC_c, where $c \in$ {UMLS, DBPedia, Wikidata}, based on the communities created by *PINYON-CACD* enables the traceability of the whole process of post recommendation. This is a unique feature of *PINYON* that cannot be achieved with any of the baselines for post recommendation included in the empirical study.

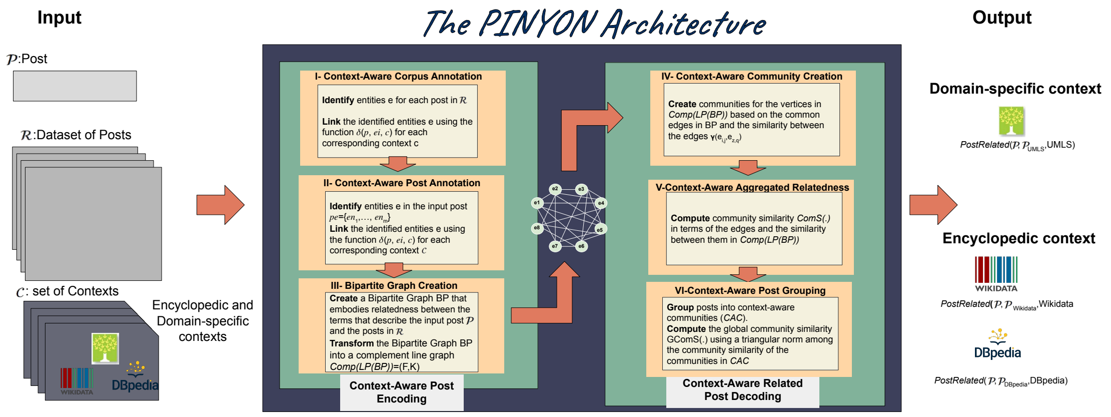

Input
The PINYON Architecture
Output
$\mathcal{P}$:Post
$\mathcal{R}$:Dataset of Posts
$\mathcal{C}$: set of Contexts
Encyclopedic and Domain-specific contexts
WIKIDATA
DBpedia
I- Context-Aware Corpus Annotation
Identify entities e for each post in $\mathcal{R}$
Link the identified entities e using the function $\delta(p, ei, c)$ for each corresponding context c
II- Context-Aware Post Annotation
Identify entities e in the input post $pe=\{en_1,\ldots, en_m\}$
Link the identified entities e using the function $\delta(p, ei, c)$ for each corresponding context c
III- Bipartite Graph Creation
Create a Bipartite Graph BP that embodies relatedness between the terms that describe the input post $\mathcal{P}$ and the posts in $\mathcal{R}$
Transform the Bipartite Graph BP into a complement line graph Comp(LP(BP))=(F,K)
Context-Aware Post Encoding
e1
e2
e3
e4
e5
e6
e7
e8
IV- Context-Aware Community Creation
Create communities for the vertices in Comp(LP(BP)) based on the common edges in BP and the similarity between the edges $\gamma(e_{i,j},e_{z,q})$
V-Context-Aware Aggregated Relatedness
Compute community similarity ComS(.) in terms of the edges and the similarity between them in Comp(LP(BP))
VI-Context-Aware Post Grouping
Group posts into context-aware communities (CAC).
Compute the global community similarity GComS(.) using a triangular norm among the community similarity of the communities in CAC
Context-Aware Related Post Decoding
Domain-specific context
PostRelated($\mathcal{P}$, $\mathcal{P}_{UMLS}$,UMLS)
Encyclopedic context
WIKIDATA
PostRelated($\mathcal{P}$, $\mathcal{P}_{Wikidata}$,Wikidata
DBpedia
PostRelated($\mathcal{P}$, $\mathcal{P}_{DBpedia}$,DBpedia)

Figure 6.4: **The PINYON approach architecture**. The pipeline receives an input post $\mathcal{P}$, a dataset $\mathcal{R}$ of posts, and a set of contexts $\mathcal{C}$ (e.g., DBpedia, Wikidata, UMLS) and outputs related posts in the corresponding contexts *PostRelated*($\mathcal{P}$,$\mathcal{P}_c(\mathcal{R})$,*c*). During the encoding phase, the input post $\mathcal{P}$ and the Corpus $\mathcal{R}$ are annotated with terms from the provided contexts $\mathcal{C}$. The annotations are used to build the Bipartite Graph BP; then the Bipartite Graph is transformed to Complement Line Graph *Comp(LP(BP))=(F,K)*. The complement line graph is utilized to create context-aware communities during the decoding phase. The posts are grouped into the created communities, and community similarity is calculated to determine the related posts.

6.3 The PINYON Approach

This section describes the techniques that implement the proposed solution reported in Section 6.2.4. Figure 6.4 depicts the components of the pipeline for retrieving semantically related posts. The pipeline comprises, first, the phase of encoding where the CORPUS and the input post are annotated and represented as a complement line graph *Comp(LP(BP))=(F,K)*. Second, communities of posts are detected to retrieve the related posts in the decoding phase.

6.3.1 Context-Aware Post Encoding

The step of Context-Aware Post Encoding comprises the components of Context-Aware Corpus Annotation, Context-Aware Post Annotation, and Bipartite Graph Creation. Given an input post $\mathcal{P}$, a dataset $\mathcal{R}$ of posts, and a set of contexts $\mathcal{C}$, annotations for the input post and the corpus are created. The created annotations are utilized to build the Bipartite Graph, which is transformed to a complement line graph.

Context-Aware Corpus Annotation

The dataset of posts is annotated by identifying first the entities for each post in $\mathcal{R}$. Any Named Entity Recognition & Linking tool could be used in this step (e.g., TagMe [136], Falcon [68], and DBpedia Spotlight [119]). However, the current PINYON implementation resorts to two versions of Falcon [68] for performing this task. This decision is supported on the experimental results reported in Chapter 4, which show that Falcon outperforms these state-of-the-art engines in the existing benchmarks. Falcon 2.0 [69] identifies entities in a short text and links the recognized entities to DBpedia and Wikidata knowledge graphs. BioFalcon[5] recognizes and links entities in a short text to UMLS. Once the annotation is completed, a dictionary of the recognized entities is created where the posts' ids for each entity's mention are stored; the dictionary is used during the last step of the PINYON approach to retrieve posts with a specific entity mention. The Context-Aware Corpus component is computed once-for-all, and there is no need to perform it again for new input posts. All the remaining components of the approach pipeline have to be computed again for each new input post.

Context-Aware Post Annotation

As in the previous step, the input post $\mathcal{P}$ is annotated by recognizing the entities in the input post then linking the recognized entities to each corresponding KG in $\mathcal{C}$. The same Named Entity Recognition & Linking tools are utilized.

Bipartite Graph Creation

During this step, a bipartite graph $BP{=}(V_1 \cup V_2{,}E)$ is created. The BP graph embodies relatedness between the terms that describe the input post $\mathcal{P}$ and the posts in $\mathcal{R}$. The BP graph is formed by creating edges between all the entities in $\mathcal{P}$ and each post's entities in $\mathcal{R}$ (Figure 6.2). The created bipartite graph is transformed into a Complement Line Graph $Comp(LP(BP))$=(F,K). Edges in the BP graph become vertices in the complement line graph and there is an edge between vertices $e_{i,j}$ and $e_{z,q}$ if there is no a vertex t_k in BP that the edges $e_{i,j}$ and $e_{z,q}$ have in common or the value of similarity $y(e_{i,j}, e_{z,q})$ is lower than a given threshold ϵ. The similarity between the edges is the average value of similarity between all the pairs of entities that form the edges. This process is similarity

[5] https://labs.tib.eu/sdm/biofalcon/

metric-agnostic. However, in the current PINYON implementation, the similarity between two entities is computed based on the cosine similarity between the vectors (embedding) representing the entities. Two embedding techniques are considered, RDF2Vec [167] and CUI2Vec [254]. RDF2Vec is utilized to retrieve embedding for entities in DBpedia and Wikidata KGs, while CUI2Vec is used to create entity embeddings for entities in UMLS.

Algorithm 1 PINYON-CACD

```
Input: graph G=(F,K)
Output: vertices color c($v$): $v \in$ C
Begin
C:=∅; U:=F; Compute $deg_{G(u)}$
For each $v \in$ F
    if $deg_{(v)}$ $==$ $|F|$-1
    then $c(v) = c'$ ; where $c'$ is unassigned color
select one uncolored vertex $v$ randomly with $\max_{v \in U}\{deg_G U(v)\}$; ties are broken based on the lowest value of aggregated similarity
$c(v) := 1; C := C \cup \{v\}; U := U \setminus \{v\}$
update $CACD_{G(C}$ and $deg_{G(U)}$
repeat
    find an uncolored vertex v with $max_{v \in U}\{CACD_{G(C)}(v)\}$
    if a subset $U'$ of multiple vertices with the same max degree of saturation is found
    then select one uncolored vertex $v$ randomly with $max_{v \in U'}\{deg_{G(U)}(v)\}$; ties are broken based on the lowest value of aggregated similarity
    find the least possible color $k$ that can color the selected vertex $v$
    $c(v) := k; C := C \cup \{v\}; U := U \setminus \{v\}$; update $CACD_{G(C)}$ and $deg_{G(U)}$
until U=∅
End
```

6.3.2 Context-Aware Post Decoding

Context-Aware Post Decoding comprises the components of Context-Aware Community Creation, Context-Aware Aggregated Relatedness, and Context-Aware Post Grouping. Given a complement line graph *Comp(LP(BP))*=(F,K), communities of edges are created using different community detection techniques. The posts are grouped into the created communities, and a community similarity is calculated to determine the related posts.

Context-Aware Community Creation

In the Context-Aware Community Creation step, communities of edges are created. This component could be implemented by any community detection approach, e.g., METIS [182] or SemEP [183]. However, PINYON maps the problem of context-aware community detection to the graph vertex coloring problem. Algorithm1 sketches the details of *PINYON-CACD*.

Creating a Reduced Complement Line Graph: *PINYON-CACD* receives *Comp(LP(BP))*=(F,K) and follows 6.2.1 to reduce the size of the graph to be colored. Thus, *PINYON-CACD* first identifies all the nodes that meet this condition, and assigns to each one a new color; no other node will be colored with these assigned colors. We have observed that numerous vertices meet this condition. As a result, a smaller portion of the original complement line graph is colored following the DSATUR heuristic. Thus, as shown in Algorithm1 Lines 5-7, for each vertex v in F, such as $deg_{G(v)}$ is equal to $|F|$-1, a new color $c(v)$ is assigned.

Furthermore, if this *reduced* complement line graph has one of the topologies presented in the propositions in Section 6.2.2, *PINYON-CACD* generates an optimal coloring. Thus, *PINYON-CACD* generates a mapping $\mu(.)$ that corresponds to a solution to the vertex coloring problem for *reduced* complement line graph. Vertices in the *reduced* complement line graph with a higher degree of saturation, but that have not been colored yet, are selected first.

Coloring a Reduced Complement Line Graph: *PINYON-CACD* orders the vertices in F, which have not been colored so far, dynamically based on the number of different colors assigned to the adjacent vertices of each vertex in F, i.e., the vertices are chosen based on the degree of saturation on the partial coloring of the graph built so far, and only colored adjacent vertices are considered. The degree of saturation represents the number of different colors used in the neighbor vertices of a vertex. Intuitively, selecting a vertex with the *maximum degree of saturation* allows for coloring first the vertices with more restrictions and for which there are a smaller number of available colors. Ties are broken based on the lowest value of aggregated similarity (Lines 8-9). Thus, vertices with more restrictions and less possible colors are chosen first. A vertex is colored with the color that maximizes the Colored Similarity. This process is repeated until all the vertices are colored (Lines 11-17).

Creating Communities: Once the graph coloring is done, the colored graph is mapped into communities. Edges corresponding to the vertices colored in the same color belong to the same community. Communities are created with the edges whose vertices in *Comp(LP(BP))* have the same color.

PINYON-CACD Time Complexity: The worse-case time complexity of *PINYON-CACD* is $O(|F'|^2)$, where F' is the subset of F without the nodes v with $deg_{G(v)}$ equal to $|F|$-1.

PINYON Agnostic-Solver Implementation: PINYON approach is also agnostic of the technique used for coloring the graph, i.e., any graph coloring algorithms could be utilized. For example, Welsh and Powell graph coloring algorithm [192]. In our ablation study (Section 6.4), we replace the *PINYON-CACD* technique with Welsh&Powel. Additionally, we use METIS to study the effect of the PINYON technique for computing Context-Aware Communities.

Context-Aware Aggregated Relatedness

In this step, a community similarity (*ComS(.)*) value is computed for each community from the previous step. The *ComS(.)* is computed in terms of the edges and their similarity. As in the Bipartite Graph Creation step, the entities' embedding is utilized to compute the similarity between two entities in order to compute the similarity between two edges. However, in this step, the community similarity is computed among all the pairs of edges inside the community.

Context-Aware Post Grouping

During the Context-Aware Post Grouping step, the input post is grouped into context-aware communities (CAC) with the posts in the corpus annotated with the resources that form edges in the same community. The dictionary created in the Context-Aware Corpus Annotation step is utilized to retrieve posts annotated with the entities that comprise the edges. The global community similarity (*GComS(.)*) is defined using a triangular norm among the community similarity of the communities in a CAC. CAC are ordered based on *GComS(.)* in descending order. The posts annotated with the highest number of entities that exist in a single community(considering ordering the communities based on the global community similarity) are selected first as related posts.

6.3.3 Agnostic of the PINYON architecture

The PINYON approach is agnostic of the studied social media platform; Twitter is just a use-case for our experiments; it is agnostic of the community detection technique. Any community detection technique can be used for our approach. The PINYON approach is agnostic of the post annotation tools. Any tool that ensures identifying entities in a short text and supports the input context can be used. The PINYON approach is also agnostic of the entities similarity metric. Any entities' embedding that supports the input contexts can be used.

6.4 Experimental Study

We study the following research questions: **RQ1)** How does knowledge represented in public KGs empower PINYON to outcome semantically related posts? **RQ2)** What is the effect of the contextual description on the accuracy of retrieving related posts? **RQ3)** How does the specificity of the entities in the input texts influence the accuracy of retrieving related posts?.

6.4.1 Experimental Configuration

Baselines

Our study includes various types of baselines; they depend on the particular to be analyzed. In all experiments, we pass the same input fed to our approach to all the baselines.

Baseline for Recommendating Posts. We include various BERT models in the study; they are used to discover the posts in a CORPUS related to an input post. Our rationale to include language-model based baselines is: these language models are trained on large corpus and consists of domain-specific contextual knowledge when fine-tuned later on specific data. Therefore, they become natural choice to compare efficacy of our approach in recommending semantically similar posts. The various baseline models include: Sentence-BERT [60], BERTweet [171], COVID-Twitter-BERT [172], BioBERT-NLI [241], and CovidBERT-NLI[6]. These models are used as follows. Posts in the CORPUS and the input post, are short sentences and BERT models, especially Sentence-BERT, determine semantic textual similarity across these sentences [60]. Sentence-BERT is a modification of the pretrained BERT network that uses siamese and triplet network structures to derive semantically meaningful sentence embeddings that can be compared using cosine-similarity [60].

The other baselines are BERT-based methods trained over tweets, tweets related to COVID-19, or literature in the biomedical domain. Since the training data is similar to our corpus, we also compare with BERTweet, COVID-Twitter-BERT, BioBERT-NLI, and CovidBERT-NLI.

BERTweet is a large-scale language model pre-trained for English Tweets. COVID-Twitter-BERT is a transformer-based model pre-trained on a large corpus of Twitter messages on the topic of COVID-19. BioBERT-NLI is a BioBERT model fine-tuned on the Stanford natural language inference corpus [255] and the MultiNLI [256] datasets using the sentence-transformers library [60] to produce universal sentence embeddings. CovidBERT-NLI is the model CovidBERT trained by DeepSet

[6] `https://huggingface.co/gsarti/covidbert-nli`

on AllenAI's CORD19 Dataset [7] of scientific articles about COVID-19. CovidBERT-NLI uses the original BERT vocabulary and was subsequently fine-tuned on the SNLI and the MultiNLI datasets using the sentence-transformers library.
In addition to the BERT models, Twitter Search API [257] is studied. The comparison with the results of the Twitter Search API allow us to assess the accuracy of post retrieval whenever exact keywords or hashtags of the input query.
Baseline for Community Detection. The communities detected by *PINYON-CACD* are compared with the results generated by METIS, "one of the state-of-the-art community detection approaches for large networks [258]".
Baseline for the Vertex Coloring Problem. The quality and efficiency of the graph coloring generated by *PINYON-CACD* are compared to high-performance method Welsh and Powell [192]. In the experiments, the Welsh and Powell graph coloring algorithm implements the CACD graph coloring of the Context-Aware Related Post Decoding step.

Benchmarks

PINYON and the baselines are studied over two benchmarks of datasets comprising texts (sentences) of tweets.
Posts about COVID-19. A dataset of tweets (TweetsCOV19) [259] is employed in experiment 1. It contains tweets in the English language for May 2020 [259], which is the month when the initial scientific studies related to COVID-19 analysis started coming out [260]. The total number of tweets utilized is 1,922,405 tweets; they contain both scientific and general tweets, while both are related to COVID-19. For each tweet, we extract, link, and retrieve the corresponding embedding vectors of KG entities using RDF2Vec [167]. However, when UMLS is used as the underlying KG, CUI2Vec [254] is used to create entity embeddings because UMLS is not an RDF triplestore. The total number of extracted entities is 252,245. A considerable amount of tweets do not include entities that can be linked to existing KGs. Thus, the number of entities is lower than the number of tweets.
Posts about the World Cup Final. The second experiment focuses on generalizability. The second experiment employs a dataset of tweets [261]; it contains a random collection of 521,802 tweets starting from the 16th round until the World Cup Final that took place on July 15th, 2018. Each record in the dataset contains a tweet about the 2018 World Cup Final, including football players and their teams' names. The football players' names are considered as entities. We use the state-of-the-art entity recognition and linking tools – e.g., DBpedia Spotlight [119], TagMe [136], and Falcon 2.0 [69] – to recognize entities from the tweets and link them to DBpedia and Wikidata KGs. The primary task in this experiment is to retrieve

[7] `https://pages.semanticscholar.org/coronavirus-research`

tweets related to *La Liga* football players' from the tweets in the FIFA dataset by providing a tweet related to *La Liga* football players as an input. We choose *La Liga* topic because during the time frame of the tweets in the FIFA dataset there was much interest about *La Liga* topic [8]. To build the gold standard for the second experiment, we collect all the *La Liga* football players' names from different sources [9], then link the players' names to their corresponding URIs in DBpedia and Wikidata using the previously mentioned entity recognition and linking tools. In our experiments, we consider the FIFA dataset of tweets as the gold standard of relevant tweets, which contains mentions of *La Liga* football players.

Metrics.

We measure the performance of PINYON performance in terms of the accuracy of recommended related posts, the quality of the communities, and the solver execution time.
Measuring Recommendation Performance. We report the performance using the standard metrics of **Precision (P)**, **Recall (R)**, and **F-Score (F)** when the gold standard is available. On the other hand, **Accuracy** is utilized when we lack information about which tweets are relevant for an input tweet.

- **Precision (P)** is the fraction of *relevant posts* among the *retrieved posts.*
- **Recall (R)** is the fraction of *relevant posts* that have been retrieved over the *total amount of relevant posts.*
- **F-Score** is the *harmonic mean* of **P** and **R**.
- **Accuracy** is the fraction of *related posts* among all the top k *studied posts.*

Quality of the Detected Communities. For studying the quality of the computed communities, we utilize the metrics defined by the research community to measure the quality of a network community [262, 263, 264]. The metrics are **Conductance**, **Coverage**, **Modularity**, **Performance**, and **Total Cut**. Let $Q = \{C_1, \dots C_n\}$ be the set of communities obtained by PINYON:

- **Conductance**: measures relatedness of entities in a community and how different they are to entities outside the community [265]. It is computed as the ratio between the number of edges inside a community and the number of edges leaving the community. Thus, conductance is considered the simplest notion of a community quality, based on the intuition that a good network

[8] shorturl.at/qFPW1

[9] https://en.wikipedia.org/wiki/Category:La_Liga_players

community comprises nodes that have better internal than external connectivity. Conductance is a lower is better metric, and we report the inverse of the conductance $1 - \mathit{Conductance}(S)$.

- **Coverage**: compares the fraction of intra-community similarities among entities to the sum of all similarities among entities [265].
- **Modularity**: is the value of the intra-community similarities among the entities divided by the sum of all the similarities among the entities, minus the sum of the similarities among the entities in different communities, in the case they were randomly distributed in the communities [266]. It measures the strength of a community partition according to the degree distribution. The value of the modularity lies in the range $[-0.5, 1]$, which can be scaled to $[0, 1]$ by computing $\frac{\mathit{Modularity}(Q)+0.5}{1.5}$.
- **Performance**: sums the number of intra-community relationships, plus the number of non-existent relationships between communities [265]. A large value of modularity indicates a good community structure.
- **Total Cut**: sums all similarities among entities in different communities [267]. Values of total cut are normalized by dividing by the sum of the similarities among the entities; inverse values are reported, i.e., $1 - \mathit{NormTotalCut}(Q)$.

Performance of the Solvers. *Execution time* is defined as the elapsed time required to perform the Context-Aware Related Post Decoding step. It is measured as the absolute wall-clock system time, as reported by the `time` command of the Linux operating system.

Implementation

PINYON is implemented in Python 3.6. We run experiments on a server equipped with 96 cores and 900GB RAM running Ubuntu 18.04. All the resources used in the reported experimental study are publicly available[10].

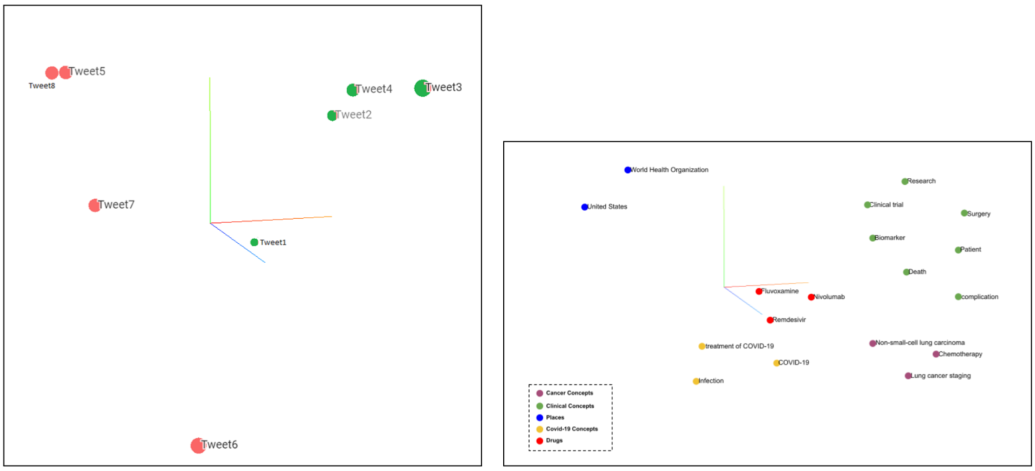
[10] `https://github.com/SDM-TIB/PINYON`

Table 6.2: Overview of the tweets used for Experiment 1

#	Tweet	Topic
T1	Neoadjuvant chemotherapy and nivolumab in resectable non-small-cell lung cancer (NADIM): an open-label, multicentre, single-arm, phase 2 trial	Lung Cancer
T2	#oncoalert Exciting data from NADIM for neoadjuvant chemo-IO in resectable NSCLC. All stage IIIA (74% N2) with 57% pathCR and 77.1% 2yr PFS in mod-ITT pop. Highly anticipate ph3 trial results and utility of pathCR (not just MPR!!) as predictive surrogate biomarker for survival!	Lung Cancer
T3	Interesting neoadjuvant trial of chemo with #immunotherapy that enrolled 46 resectable stage IIIA NSCLC patients. At 24 months, PFS was 77·1% (95% CI 59·9–87·7). Approach was not associated with surgery delays	Lung Cancer
T4	This is quite remarkable, look forward to seeing randomized neoadjuvant chemo-IO results. Definitely the most promising area for resectable NSCLC right now!	Lung Cancer
T5	Remdesivir, the only antiviral drug authorized for treatment of COVID-19 in the United States, fails to prevent deaths among patients, according to a study of more than 11,000 people in 30 countries sponsored by the World Health Organization.	COVID-19
T6	We now have evidence that an inexpensive drug, #Fluvoxamine, may be effective in preventing patients with mild #COVID19 cases from developing severe complications. Learn more and help us continue the research:	COVID-19
T7	Greater access to Remdesivir through the @NHFJamaica for the treatment of Covid-19 infection. However, please note that clinical trials for this drug are still ongoing in other countries.	COVID-19
T8	A new study from the World Health Organization has found that remdesivir — one of the anti-viral drugs that has been touted as a potential treatment for COVID-19 since — has “little or no effect” on COVID patients’ chances of survival	COVID-19

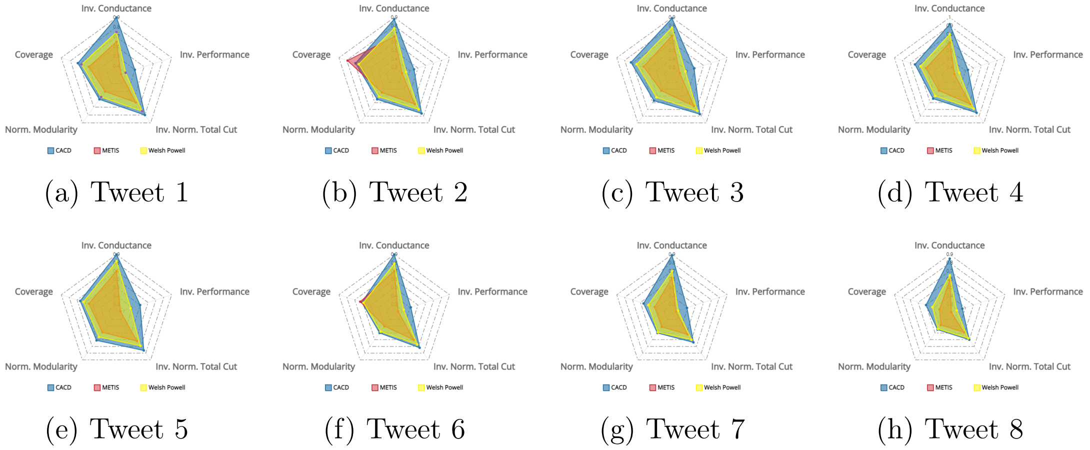
Table 6.3: **Size of the graphs *(*Comp(LP(BP))*)*** generated during the Context-Aware Post Encoding for each of the Tweets in Experiment 1

Tweets	Nodes	Edges	Nodes Pruned by PINYON-CACD	Pruning rate
T1	197895	20,162,431,025	5,645,480,687	72%
T2	205934	26,408,812,356	6,602,203,089	75%
T3	162389	15,370,187,321	3,074,037,464	80%
T4	140392	11,709,913,664	3,630,073,235	69%
T5	173628	21,146,682,384	5,498,137,419	74%
T6	184937	24,201,693,969	5,808,406,552	76%
T7	172865	18,882,308,225	5,475,869,385	71%
T8	187246	23,061,064,516	7,379,540,645	68%



(a) Tweets Embedding Visualization (b) Entities Embedding Visualization

Figure 6.5: Comparison of Text- and Entity-based Embeddings of Biomedical Tweets.



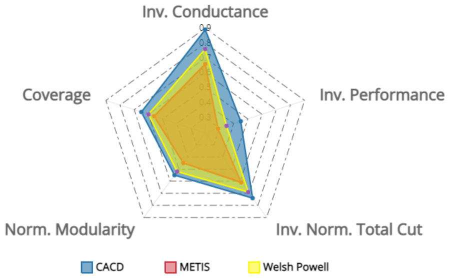
(a) Tweet 1 (b) Tweet 2 (c) Tweet 3 (d) Tweet 4

(e) Tweet 5 (f) Tweet 6 (g) Tweet 7 (h) Tweet 8

Figure 6.6: Quality of the computed communities. Communities evaluated in terms of five metrics (higher values are better); Communities for the eight tweets in 6.4.2 are reported. *PINYON-CACD* exhibits the best performance.

6.4.2 Experiment 1- Medical Domain

This experiment studies PINYON performance on a large dataset of tweets[259] related to the medical domain.

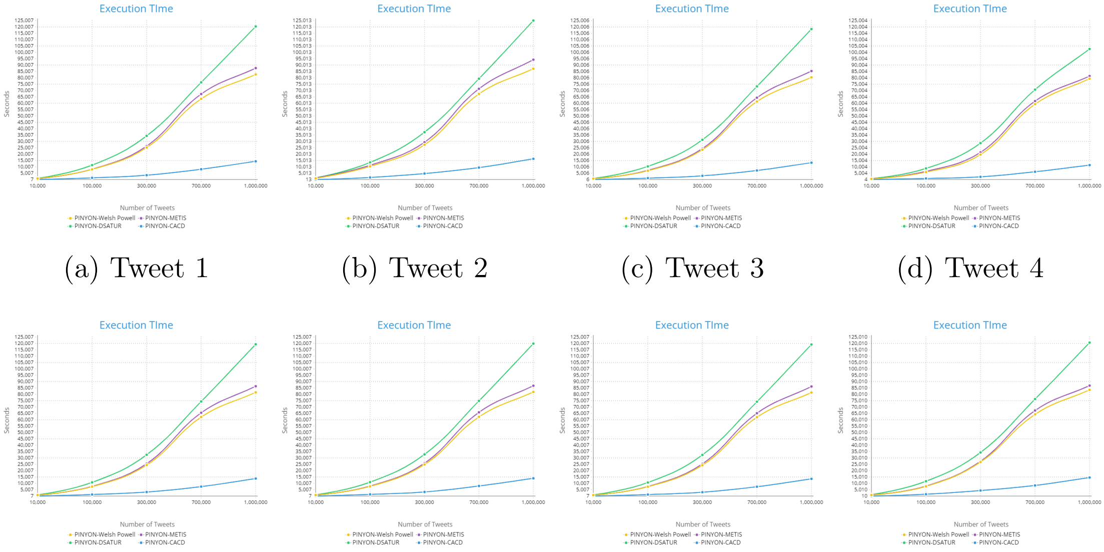


(a) Aggregated values for the communities quality metrics (average values for each metric) across all the tweets.

	Solver					
	DSATUR		METIS		Welsh Powell	
Metric	Avg	SD	Avg	SD	Avg	SD
Inv. Conductance	0.89	0.02	0.66	0.04	0.76	0.06
Inv. Performance	0.45	0.06	0.29	0.05	0.35	0.05
Inv. Norm. Total Cut	0.74	0.08	0.61	0.07	0.69	0.07
Norm. Modularity	0.55	0.06	0.45	0.05	0.52	0.05
Coverage	0.65	0.09	0.56	0.14	0.60	0.09

(b) Aggregated values for the community quality metrics. The average (Avg) and standard deviation (SD) are computed for each metric across all the tweets for the three solvers

Figure 6.7: Report of the Aggregated values for the community quality metrics.



(a) Tweet 1 (b) Tweet 2 (c) Tweet 3 (d) Tweet 4

(e) Tweet 5 (f) Tweet 6 (g) Tweet 7 (h) Tweet 8

Figure 6.8: Context-Aware Related Post Decoding execution time for each tweet in Experiment 1

Experiment Setup

PINYON is executed against three different configurations of the background knowledge (i.e., DBpedia, Wikidata, and UMLS). Moreover, an ablation study is conducted by utilizing METIS and Welsh Powell solvers. We fix a threshold ($\epsilon = 0.50$ from Section 6.2) to build the communities. The threshold is determined by running the 10-fold cross-validation to choose the best setting.

Input Posts. Eight different tweets related to two topics (Lung Cancer and

COVID-19) are used as testbeds for the experiment. Building testbeds for tweet relatedness is not our contribution, and we inherit the testbed settings from [55, 178, 179]. The eight tweets are described in Table 6.2. These tweets are selected based on the following parameters: three Medical doctors were asked to provide 24 (each doctor with eight tweets) tweets related to COVID-19 and Lung Cancer. A fourth medical doctor specializing in Lung Cancer recommended four tweets related to the topic. A fifth doctor specializing in general Medicine recommended four tweets related to COVID-19 to be part of the testbed.

Gold Standard. Due to the lack of test datasets for tweets in the medical domain, we asked six medical doctors (with at least the degree of general Medicine) to evaluate the experiment's performance. The medical doctors were asked to determine if the retrieved tweets are relevant with respect to the input tweet; medical doctors received tweets after anonymizing the approach name. These Medical doctors are not the same as the ones who selected the tweets in the testbed. *A tweet should be marked as relevant by Medical doctors if it satisfies the following criteria*: (a) fits the topic of the input tweet, (b) confirms the information in the input tweet or, (c) contains entities that are relevant to the entities in the input tweet, d) if a drug is mentioned in the input tweet, then all the occurrences of similar drugs appeared in the related tweets should be used for the same medical prescription. It is important to note that although the number of tweets in a testbed is eight, the corresponding tweets in the dataset are in millions (one-to-many mapping). Hence, in the ideal case, the solution implemented in PINYON for the vertex coloring problem should remove a lot of noise (irrelevant graph nodes representing the tweets in the communities) before finalizing the related communities of posts.

Performance of the solvers for discovering communities

Size of the Complement Line Graph per Input Tweet. The eight input tweets induce large and complex complement line graphs, which impose challenges for the whole process of related post recommendation. Table 6.3 reports the size of the complement line graph *Comp(LP(BP))* generated during the Context-Aware Post Encoding for each of the tweets presented in Table 6.2. As we can see in Table 6.3 the numbers of the generated nodes and edges are huge; an average of 178,160 nodes and 20,117,886,682 edges. However, built upon Lemma 6.2.1, *PINYON-CACD* executes the Context-Aware Related Post Decoding step efficiently; it can identify the colors assigned to exclusively one vertex. Thus, *PINYON-CACD* prunes the complement line graph before starting coloring, and the number of nodes and edges are reduced considerably (68-80%). As a result, *PINYON-CACD* minimizes the execution time needed for the Context-Aware Related Post Decoding compared to the other solvers (i.e., METIS, Welsh Powell) Figure 6.8. The pruning rate is related to how much noise is presented in the

tweet's text. For example, Tweet2 and Tweet3 look very similar utilizing the mentioned entities. However, the entities *months* and *surgery* in Tweet3 add more noise than Tweet2 making the pruning rate of Tweet3 higher than Tweet2.
Vector Representation of Input Posts. To have a more precise understanding of the properties of our approach (mapping the problem of Context-Aware Post Recommendation into the Vertex Coloring Problem) and the BERT models baselines, we visualize the embedding for each tweet using Sentence-BERT (Figure 6.5a) and the embedding for each entity mentioned in a tweet using CUI2Vec (Figure 6.5b). As we can observe in Figure 6.5a, Tweet6 and Tweet7 are far in the embedding space from Tweet5 and Tweet8 even though they share the same entities and are related to the same topic (COVID-19); the same observation is applied between Tweet1 and the remaining Lung Cancer related tweets. On the other hand, we can observe in Figure 6.5b that the entities mentioned in the tweets are grouped together in the embedding space powering the *PINYON-CACD* approach finding semantically related posts based on the similarity between entities mentioned in a tweet without being affected by the noise presented in the tweet's text as the *PINYON-CACD* remove such noise while pruning the complement line graph.
Time Performance of the Context-Aware Related Post Decoding Step. We also studied the time required for the Context-Aware Related Post Decoding by executing Experiment1 but with a different number of tweets in the studied corpus; we exclude the time required for The Context-Aware Post Encoding since it is similar for all the solvers. As we can observe in Figure 6.8, the execution time for PINYON-METIS and PINYON-Welsh-Powell increases considerably w.r.t the size of the studied corpus (840-86k seconds). In contrast, *PINYON-CACD* reports a much lower execution time (7-14k seconds) over the same graph because of the graph pruning step. PINYON-DSATUR (the *PINYON-CACD* approach without the graph pruning step) reports the highest execution time since the size of the graph to be colored is much higher than the pruned graph in the case of *PINYON-CACD* (68-80% difference).

Quality of the discovered communities

Communities' Quality. Figure 6.6 shows the quality of the generated communities for the tweets in Table 6.2. Communities for each tweet are computed using three solvers; CACD, METIS, and Welsh Powell. The communities generated by PINYON include closely related posts in all tweets. However, *PINYON-CACD* exhibits higher quality in terms of the five community-based metrics. These results corroborate our hypothesis that *PINYON-CACD* is able to group together entities into communities that are highly related and provide an explanation for the results reported in Table 6.4. Further, we can observe that the communities for the tweets related to Lung Cancer(Tweet 1-4) are of higher quality than those related to COVID-19 (Tweet 5-8). The reason behind such higher communities' quality is the richness of the BK for terms related to Lung Cancer compared to COVID-19. Tweet5 and Tweet8 have in common several entities that have different types. But, the mention of `United States` in Tweet5 enables to recognize the resource `COVID-19 pandemic in the United States,Q83873577` which provides contextual knowledge to facilitate the discovery of highly related tweets in the corpus. As a result, the values of performance, modularity, and coverage in Tweet5 are higher.

Average Communities' Quality. For understanding better the differences between the solvers, we computed the average and standard deviation (SD) for each metric used to measure the quality of the communities across all the tweets. From Figure 6.7b and Figure 6.7a We can observe that the aggregated values for the *PINYON-CACD* and Welsh Powell solvers are higher than METIS solver, which confirms the results reported in Table 6.4.

Performance of PINYON

The Accuracy metric is used to measure the performance of the task of retrieving semantically related tweets(top 40 tweets). The results of this experiment are reported in Table 6.4. A 2-fold cross-validation method was implemented while evaluating the results, and a majority voting to solve any disagreements. The results in Table 6.4 propose that PINYON-CACD-UMLS employing only the knowledge in UMLS KG as background knowledge for the approach outperforms all the baselines and the other configuration of PINYON. The reason behind the good performance is the richness of UMLS KG in the medical domain (14,608,809 different entities in the medical domain), which empowers the PINYON approach to retrieve related tweets in the studied topics (Lung Cancer and COVID-19). Additionally, *PINYON-CACD* have been customized to select the communities following heuristics that ensure that very related entities are put together in the same community (answering **RQ2** & **RQ3**). Twitter Search API reports poor accuracy and performs only exact string matching. Therefore, we passed the entities mentioned in the input tweet to the search API instead of the entire tweet content to get better results. Nonetheless, the reported accuracy of the API is not improved because of the strategy of matching hashtags and keywords without considering any semantics[257]. Note that METIS and Welsh Powell exhibit a competitive behavior, but as off-the-shelf components their algorithms are not customized to ensure that the overall relatedness among all the entities included in the same community is maximized. The reported accuracy for the baseline Sentence-BERT suggests that the specific domains as the medical domain represents a challenge even for pre-trained BERT models. BioBERT-NLI and CovidBERT-NLI

Table 6.4: **PINYON Performance compared to the baselines for Experiment 1.** TweetKB COV19 is the input dataset. The Accuracy metric is reported. Best results are in bold. The columns in Orange are Tweets related to Lung Cancer. The columns in Yellow are tweets related to COVID-19. Three different solvers are utilized for the ablation study; METIS for graph partitioning and community detection, CACD and Welsh Powell for graph coloring. Three different BKs are considered for this experiment.

Baselines		**T1**	**T2**	**T3**	**T4**	**T5**	**T6**	**T7**	**T8**
Sentence-BERT		0.58	0.62	0.50	0.41	0.42	0.45	0.39	0.47
BERTweet		0.75	0.69	0.73	0.77	0.62	0.66	0.60	0.63
COVID-Twitter-BERT		0.59	0.57	0.58	0.59	0.73	0.70	0.75	0.71
BioBERT-NLI		0.65	0.68	0.60	0.61	0.50	0.56	0.52	0.56
CovidBERT-NLI		0.54	0.52	0.49	0.55	0.63	0.65	0.60	0.61
Twitter Search API		0.18	0.21	0.15	0.23	0.20	0.13	0.23	0.16
PINYON									
BK	**Engine**								
UMLS	CACD	**0.85**	**0.83**	**0.84**	**0.86**	**0.85**	**0.82**	**0.81**	**0.79**
	METIS	0.82	0.79	0.80	0.82	0.81	0.81	0.76	0.74
	Welsh Powell	0.83	0.81	**0.84**	0.84	0.83	0.80	**0.81**	0.78
DBpedia	CACD	0.79	0.81	0.78	0.82	0.80	0.77	0.79	0.74
	METIS	0.72	0.75	0.70	0.76	0.71	0.69	0.72	0.68
	Welsh Powell	0.75	0.78	0.73	0.80	0.75	0.72	0.74	0.70
Wikidata	CACD	0.82	**0.83**	0.80	0.83	0.82	0.78	**0.81**	**0.79**
	METIS	0.75	0.73	0.71	0.69	0.74	0.67	0.70	0.69
	Welsh Powell	0.79	0.75	0.76	0.78	0.80	0.72	0.79	0.73

Table 6.5: **PINYON Performance compared to the Baselines for Experiment 2**. FIFA dataset is the input dataset. The reported metrics are Precision, Recall, and F-Score. Best results are in bold. Three different solvers are utilized for the ablation study; METIS for graph partitioning and community detection, CACD and Welsh Powell for graph coloring. Two different BKs are considered.

		Threshold=0.50			**Threshold=0.65**		
Baselines		**P**	**R**	**F**	**P**	**R**	**F**
Sentence-BERT		0.60	0.75	0.67	0.71	0.68	0.69
BERTweet		0.76	0.80	0.78	0.80	0.74	0.77
PINYON							
BK	**Solver**						
Wikidata	CACD	**0.85**	**0.88**	**0.86**	**0.89**	**0.81**	**0.85**
	METIS	0.79	0.82	0.80	0.85	0.77	0.81
	Welsh Powell	0.83	0.87	0.85	0.87	0.79	0.83
DBpedia	CACD	0.79	0.83	0.81	0.82	0.76	0.79
	METIS	0.72	0.76	0.74	0.78	0.72	0.75
	Welsh Powell	0.77	0.81	0.79	0.80	0.75	0.77

report higher accuracy since they are fine-tuned over medical corpus. BERTweet is trained over 850M English Tweets(5M Tweets related to COVID-19). Still, we can observe a drop in the accuracy for the COVID-19 tweets because of its inability to capture the underlying semantic meaning of entities, which PINYON successfully does. COVID-Twitter-BERT is trained over 97M tweets related to the topic COVID-19. Thus, COVID-Twitter-BERT reports the second-highest accuracy for COVID-19 tweets. A noticeable drop in accuracy for the baselines is observed for the tweets associated with the COVID-19 topic. This drop is the lack of labeled training data for the studied topic (COVID-19), which is always the case in new pandemics. Contrary, the richness of the PINYON background knowledge in the medical domain using drugs and diseases mentions prevents the drop in the accuracy of our approach for both topics (answering **RQ1** successfully).

6.4.3 Experiment 2- Sport Domain

This experiment aims to evaluate PINYON generalization. The experiment aims to accommodate understanding of PINYON's behavior in a domain that is different from the medical domain (Sport domain).

Tweets related to La Liga championship- Gold Standard This experiment aims to study the performance of PINYON on a tweets dataset from the sports domain to validate general domain performance. The dataset of tweets from the 2018 World Cup Final (Section 6.4.1) is utilized. Considering the domain of this experiment is general, we choose Sentence-BERT and BERTweet as the baselines. Additionally, we study the effect of the different features of the utilized knowledge graphs in the background knowledge. PINYON is executed against two different configurations of the background knowledge (i.e., DBpedia, Wikidata). Furthermore, two different thresholds for building the communities are used (i.e., 0.50 and 0.65); the same configuration from the previous experiment for the ablation study is applied.

PINYON Performance Average results of five runs are reported in Table 6.5 (different random input post from the gold standard); the best results are in bold; the average number of nodes, edges, and pruning rate of the complement line graph are 135,984, 12,294,158 and 68% accordingly; the average execution time for *PINYON-CACD* is 45,367 seconds. The accuracy of the retrieved relevant tweets is measured by precision (P), recall (R), and F-Score (F). Table 6.5 reports the results. We observe an increase in the performance of the baselines compared to experiment 1. The explanation for this increase is because of the generalizability of the studied benchmark, which matches the data used to train these models. PINYON-CACD-Wikidata benefits from the knowledge collected in a community-maintained knowledge graph, where predicates are proposed and then get accepted only if these additions meet specific criteria. Contrary, DBpedia data is automatically extracted from publicly available Wikipedia dumps, which may affect data quality as, in consequence, negatively impact on the studied solvers. Furthermore, the results reveal the importance of the value threshold that represents when two entities can be considered similar. The thresholds of 0.5 and 0.65 correspond to the percentiles 45 and 88 in the distribution of the values of similarities between the entities in the La Liga Tweets. As a result, when the threshold ϵ=0.65 is evaluated, 88% of the combinations of entities are considered not similar, enabling all the approaches to increase the precision. Particularly, in the case of *PINYON-CACD*, the value of ϵ=0.65 prevents that all this pair of entities are never placed together in the same community. Since the recall is not improved as the precision, a more precise method for tuning this threshold is required. The results enable us to answer **RQ1** and **RQ2** in the sports domain. We conclude that the richness of the contextual description encoded in the background knowledge and the features of the entities' mentions in the tweets affect the quality of the solutions.

6.4.4 Discussion

PINYON's configurations outperform the baselines across all the experimental settings. These results are grounded on the strategy followed by PINYON, where the information encoded in the contextual description is enough to determine the semantically related tweets. Moreover, integrating various KGs into the background knowledge empowers PINYON to capture knowledge from different domains and facilitate the accurate computation of entity and post-relatedness. It is also important to note that PINYON exhibits better performance (Experiment 1) because of the detailed description of the biomedical concepts present in UMLS. Moreover, it is worthy of mentioning that Wikidata is a community-maintained knowledge graph where predicates are proposed and then get accepted only if these additions meet specific criteria. Contrary, DBpedia data is automatically extracted from publicly available Wikipedia dumps, which may affect data quality. Albeit empowering PINYON with the knowledge encoded on Wikipedia, the number of entities compared to Wikidata affects the performance of PINYON-DBpedia as reported in Table 6.4 and Table 6.5. Our proposed approach and evaluation provide key insights: 1) Our approach is explainable and interpretable. Mapping the problem of Context-Aware Post Recommendation into the Vertex Coloring Problem permits us to explain the results generated by PINYON. Starting from the last step in PINYON (i.e., context-aware community creation), one can trace back in which of the computed communities a particular post appears, (Figure 6.2). 2) Moreover, PINYON can also find out, based on context description, why a particular post belongs in a community considering the community creation depends on the entities present in the posts. As a result, PINYON corresponds to a white box framework, which allows error tracing while recommending a particular post for a given community.

Limitations: PINYON suffers from the following limitations. First is the constraint to recognize dark entities. Dark entities are entities that do not exist in any knowledge graph [69]. Since PINYON resorts to various knowledge graphs to build its background knowledge, PINYON is not able to extract any knowledge (i.e., contextual description) about these dark entities. Further, the specificity of the entities in the corresponding KGs affects the accuracy of PINYON. Thus, if an entity is miss-linked to a KG class, the embeddings' quality will also be affected. it is important to mention that this limitation is a limitation of the studied corpus and the entity linking tools, and it is not a limitation in our algorithm design. New entities are added regularly to the used community-maintained KGs, enabling the entity linking tools to recognize the newly added entities. So adding more knowledge to the used KGs can overcome these types of limitations. Moreover, the solution presented in this work resorts to external components like Falcon2.0, FALCON, TagMe, and DBpedia Spotlight for entity linking. Hence, inheriting the pitfalls of such a tool. For instance, if a tool fails to extract and link a set of entities to the corresponding knowledge graph, PINYON will also be negatively affected. We also ignore edge features (relationship between entities within a tweet), and one possible extension is to study the effect of edge features on our approach.

Success cases: PINYON overcomes the common problem of lacking training data by depending on the knowledge encoded in its background knowledge to report a high accuracy. The variety of the knowledge graphs used to enrich the approach's background knowledge enables PINYON to perform semantically related posts retrieving with high accuracy, as observed in Section 6.4. PINYON is agnostic of similarity metrics (calculated using RDF2Vec or CUI2Vec). The lack of training data is observed in the current pandemic of COVID-19. The researchers cannot train the state-of-the-art models for tasks related to health data on the Web. PINYON is a good starting point in such cases, and can also be utilized to build datasets in a specific domain.

6.5 Summary

This chapter presents PINYON, a knowledge-driven approach for unveiling semantically related posts in a corpus. Our approach adopts two novel concepts. First, various KGs are employed as background knowledge; second, a vertex coloring algorithm leverages the extended background knowledge for creating the communities of related posts. Our empirical evaluations provide evidence that the approach outperforms the baselines on several benchmarks in two domains. More importantly, this work has highlighted the importance of capturing background knowledge encoded in existing KGs and the impact that this knowledge has on the tackled problem. Thus, our work broadens the repertoire of knowledge-driven tools for supporting the new generation of data-driven digital technologies.

Chapter 7

Applications of Knowledge Extraction

Knowledge extraction techniques have several applications in various domains (e.g., biomedical or energy domains). For example, knowledge extraction techniques contribute to knowledge graph creation pipelines by providing a semantic way to link concepts (e.g., entity alignment and entity linking). Moreover, applying knowledge extraction techniques allow analyzing natural language text (unstructured data) to support experts (e.g., medical doctors) in having a better insight into the studied data and making decisions (e.g., treatments of patients). Thus, knowledge extraction methods are essential in analyzing unstructured data in different areas. We have reported the content of this chapter in these publications [64, 70, 71, 72, 73, 74] and the research work that is under review. The results of this chapter provide an answer to the following research question:

> **RQ4:** How does the knowledge extracted from unstructured data can be used for real-world applications?

To answer this research question, we explore the applications of all the contributions of this thesis that are used in use cases of real-world projects. The proposed frameworks are used for knowledge extraction over unstructured data. The rule-based approaches [68, 69] Falcon[1] and Falcon2[2] are being used by the community for the tasks of NER, NED, RR, and RL using their public APIs. Both APIs are being heavily used; 4,119,179 hits since April 2019 for Falcon and 5,664,204 hits since February 2020 for Falcon2. Both approaches were used in life science projects

[1] `https://labs.tib.eu/falcon/`

[2] `https://labs.tib.eu/falcon/falcon2/`

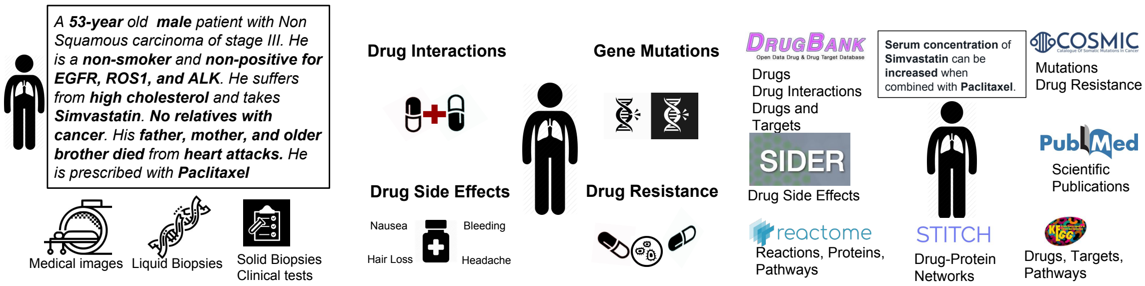
(e.g., iASiS, BigMedilytics, P4-Lucat, CLARIFY, ImProVIT, Knowledge4Hubris, and K4COVID [70]) to extract mentions of medical terms from a natural language text then link them to a domain-specific knowledge base like UMLS (e.g., drugs and comorbidities from doctors notes describing patients treatments). The rule-based approach and the background knowledge proposed in this thesis are also implemented for a specific domain (energy domain)[3] where the goal is to identify entities in a natural language text and link them to a specific semantic data model described by experts in the EU project PLATOON [71, 72]. In the CoyPu project, the proposed frameworks of this thesis are also used to extract knowledge from various datasets related to diverse domains. In the project Knowledge4Hubris, the proposed framework for unveiling semantically related posts in a corpus is utilized to analyze tweets related to people who suffer from hubris disease. The experts in this project are interested in identifying people who suffer from hubris based on their way of writing posts and found it challenging to analyze hundreds of millions of tweets. The proposed methods of this thesis are also used as components of knowledge graph creation and Semantic data integration pipelines [64, 73, 74].

This chapter is structured as follows: the next section motivates the applications of knowledge extraction techniques and how they could be used for data integration tasks. Section 7.2 describe the projects where the contributions of the thesis are applied. In Section 7.3, we present the different use cases of proposed knowledge extraction methods. The evaluation of the applied use cases is described in Section7.4. Finally, we present the closing remarks of this chapter in Section 7.5.

7.1 Motivation

The tasks of knowledge extraction are essential for integrating data from heterogeneous sources of knowledge. For example, Figure 7.1 presents a set of myriad sources of knowledge about the condition of a lung cancer patient, as well as typical integration problems caused as a result of well-known data complexity issues, e.g., variety, volume, and veracity. Electronic health records (EHRs) (Figure 7.1a) preserve the knowledge about the conditions of a patient that need to be considered in order for effective diagnosis and treatment prescriptions. Albeit informative, EHRs usually preserve patient information in an unstructured way, e.g., textual notes, images, or genome sequencing. Furthermore, EHRs may include incomplete and ambiguous statements about the whole medical history of a patient. In consequence, knowledge extraction techniques are required to mine and curate relevant information for an integral analysis of a patient, e.g., age, gender, life habits, mutations, diagnostics, treatments, and familial antecedents. In addition to evaluating

[3] `https://labs.tib.eu/sdm/efalcon/`



*A **53-year** old **male** patient with Non Squamous carcinoma of stage III. He is a **non-smoker** and **non-positive for EGFR, ROS1, and ALK**. He suffers from **high cholesterol** and takes **Simvastatin**. **No relatives with cancer**. His **father, mother, and older brother died** from **heart attacks.** He is prescribed with **Paclitaxel***

Serum concentration of **Simvastatin** can be **increased** when combined with **Paclitaxel**.

(a) Electronic Health Records (b) Impacts in Treatment Effectiveness (c) Biomedical Data Sources

Figure 7.1: **Motivating Example**. Heterogeneous sources of knowledge. (a) Unstructured data sources, e.g., clinical notes, medical images, and clinical tests, encode invaluable knowledge about a patient medical condition. (b) Factors impact on the effectiveness of a treatment; they need to be identified to increase a patient survival time. (c) Various biomedical repositories maintain knowledge collected by the scientific community about facts that can contribute to the prescription of effective treatments. Data sources range from structured (e.g, COSMIC), to unstructured (e.g., PubMed); and short texts in structured data sources may encode also relevant knowledge (e.g., drug interactions). Heterogeneity problems across sources need to be solved for extracting the required knowledge [62].

information in EHRs, physicians depend on their experience or available sources of knowledge to predict potential adverse outcomes, e.g., drug interactions, side-effect or resistance (Figure 7.1b). Diverse repositories and databases make available crucial knowledge for the complete description of a patient condition and the potential outcome (Figure 7.1c). Nevertheless, sources are autonomous and utilized diverse formats that range from unstructured scientific publications in PubMed[4] to dumps of structured data about cancer related mutations in COSMIC [5]. To illustrate, the effect of the interactions between two drugs is reported in DrugBank like short text, e.g., the effect of the interactions between Simvastatin and Paclitaxel. In order to detect the facts that can impact on the effectiveness of a particular treatment, e.g., Paclitaxel, the physician will have to search through these diverse data sources and identify the potential adverse events and interactions. Data complexity issues like data volume and diversity impede an efficient integration of the knowledge required to predict the outcomes of a treatment.

The proposed contributions of this thesis resort to techniques of knowledge extraction and representation to create a knowledge graph where data from dis-

[4] `https://www.ncbi.nlm.nih.gov/pubmed/`

[5] `https://cancer.sanger.ac.uk/cosmic`

parate data sources is integrated. A knowledge graph represents entities and their relations, and ontologies and controlled vocabularies are utilized to describe the meaning of relations, as well as for annotating entities in a uniform way in the knowledge graph. The unified Medical Language System (UMLS), the Human Phenotype Ontology (HPO), and the Gene Ontology (GO) are exemplar ontologies. Furthermore, entity linking techniques are part of the framework to allow for the linking of entities in the knowledge graph, e.g., the drug Paclitaxel, to equivalent entities in existing knowledge graphs, e.g., in DBpedia[6] and in Bio2RDF[7]. The linked knowledge graphs compose a federation, and a federated query is able to execute queries against the various knowledge graphs. Finally, (un)supervised techniques are built on top of the knowledge graphs for the support of conscientious diagnosis and personalized treatments.

7.2 Projects

In this section, we provide a brief description of the projects where the contributions of this thesis are applied in order to give a better overview of the use cases that utilize our methods and show the diversity of the projects' domains.

iASiS[8] is an EU-funded project that seeks to pave the way for precision medicine approaches by utilizing insights from patient data. It aims to combine information from medical records, imaging databases, and genomics data to enable more personalized diagnosis and treatment approaches in two disease areas – lung cancer and Alzheimer's disease. We contributed to this project by providing knowledge extraction methods that map the knowledge encoded in the textual representation of the projects' unstructured data to DrugBank[9] and UMLS.

BigMedilytics[10] (Big Data for Medical Analytics) is the largest EU-funded initiative to transform the region's healthcare sector by using state-of-the-art big data technologies to achieve breakthrough productivity in the sector by reducing cost, improving patient outcomes and delivering better access to healthcare facilities simultaneously. We contributed to this project by providing knowledge extraction methods that assist in mapping the encoded knowledge in the unstructured data of the project to existing knowledge sources. Moreover, our contributions are part of the knowledge graph creation pipeline of the project.

P4-Lucat[11] (Personalized Medicine For Lung Cancer Treatment) is created

[6] `http://dbpedia.org/resource/Paclitaxel`

[7] `http://bio2rdf.org/drugbank:DB01229`

[8] `https://project-iasis.eu/`

[9] `https://go.drugbank.com/`

[10] `https://www.bigmedilytics.eu/`

[11] `https://p4-lucat.eu/`

to develop a platform to be used by oncologists in the pursuit of helping them in the decision-making process to determine the best treatment for a lung cancer patient. To make this decision, the data that will be considered will be data from previous patients extracted from EHR, open data, and scientific publications. We contributed to this project by providing knowledge extraction methods that help medical doctors to annotate medical concepts related to Lung Cancer to the UMLS knowledge base.

CLARIFY[12] identify the risk factors for deterioration in a patient at the end of oncological treatment. Specifically, it will collect data about survivors of breast, lung, and lymphoma cancer (the most prevalent types) from hospitals in Spain. Using big data and artificial intelligence techniques, it will integrate all data with relevant publicly available biomedical information, as well as information from wearable devices used after the treatment. The data will be analyzed to predict the patient-specific risk of developing secondary effects and toxicities from their cancer treatments. We contributed to this project by providing knowledge extraction methods that map the knowledge encoded in the textual representation of the projects' unstructured data to DrugBank[13] and UMLS knowledge bases.

ImProVIT[14] aims at developing a knowledge-driven framework able to combine heterogeneous data sets coming from diverse analytical methods. Specifically, ImProVIT will integrate data from conventional FACS-based immunmonitoring with information from cutting-edge technologies such as chip cytometry, T cell and B cell receptor repertoire analysis, single-cell sequencing, cytokine arrays, transcriptomics, and whole genome sequencing. Additionally, the knowledge encoded in biomedical ontologies, e.g., the Human Phenotype Ontology (HPO), and open-source data databases such as Online Mendelian Inheritance in Man (OMIM) will be mined and represented in the knowledge graph. We contributed to this project by providing knowledge extraction methods that help medical doctors to annotate medical concepts from heterogeneous data sources for building a knowledge graph of the human immune system.

CoyPu[15] project addresses the complex economic challenges in crisis situations with an intelligent platform for the integration, structuring, networking, analysis, and evaluation of heterogeneous data from economic value networks as well as the industry environment and social context. Based on the cognitive modeling of data within a promoted system of networked knowledge graphs and flexibly configurable AI analysis tools, the CoyPu platform enables high-quality and up-to-the-minute insights into economic facts, trends, impact relationships, and forecasts. The crisis-

[12] https://www.clarify2020.eu/

[13] https://go.drugbank.com/

[14] https://www.tib.eu/en/research-development/project-overview/project-summary/improvit

[15] https://coypu.org/

relevant questions that can be answered in this way can concern individual value networks or concrete value chains, focus on different regions, industries or company sizes, or be located at the overall economic level. We contributed to this project by providing annotation methods (Falcon and Falcon2.0) to annotate concepts comprising the datasets of the project and linking these concepts to existing knowledge in community-maintained knowledge graphs (e.g., DBpedia and Wikidata).

PLATOON[16] project aims to digitalize the energy sector, enabling thus higher levels of operational excellence with the adoption of disrupting technologies. The project will reinforce the European efforts to modernize the European electricity grid, as it focuses on new smart grid services through data knowledge exploitation. Moreover, PLATOON will offer access to cheaper and sustainable energy for energy consumers and maximize social welfare. We contributed to this project by providing annotation methods (E-Falcon) that annotate concepts related to the energy domain to a semantic data model defined by experts in the energy domain.

Knowledge4Hubris[17] project aims is to create an integrated representation of all data relevant to the tenure of different forms of power. By applying graph theory to visualize and understand networks of interrelated information, the project expects to be able to identify determinants of behavior and personality change in powerful leaders. Moreover, this may lead to insights into the complex of attributes that characterize individuals holding power. The project partners are particularly interested in the differences between those who displayed hubristic behavior during their tenure of office and those who remained free from this damaging pattern of behavior. We contributed to this project by providing a framework that is able to retrieve semantically related posts in a corpus of social media posts in order to help the experts to identify the people who suffer from hubris based on their writing style in these social media posts.

Knowledge4COVID[18][70] is a framework that aims to showcase the power of integrating disparate sources of knowledge to discover adverse drug effects caused by drug-drug interactions among COVID-19 treatments and pre-existing condition drugs; it extracts relevant entities and predicates that enable the fine-grained description of COVID-19 treatments and the potential adverse events that may occur when these treatments are combined with treatments of common comorbidities, e.g., hypertension, diabetes, or asthma. We contributed to this project by providing knowledge extraction techniques that are able to integrate knowledge from different data sources related to the topic COVID-19.

[16] `https://platoon-project.eu/`

[17] `https://www.sgul.ac.uk/about/our-institutes/molecular-and-clinical-sciences/research-sections/neuroscience-research-section/knowledge-for-hubris`

[18] `https://github.com/SDM-TIB/Knowledge4COVID-19`

7.3 Use Cases

7.3.1 Knowledge Extraction from semi-structured data

Our proposed knowledge extraction methods are used for extracting knowledge from semi-structured data related to the biomedical domain, specifically for extracting knowledge encoded in the textual descriptions of Drug-Drug-Interactions (DDIS) in the DrugBank knowledge base. Falcon [68] is used to extract the Drug-Drug Interactions (DDIs) reported in DrugBank as short texts. We customize Falcon for analyzing the DDI text. Since the DDI text is related to the medical domain, UMLS is utilized as the background knowledge for Falcon. In this case, in addition to recognizing words that correspond to two drugs that interact, Falcon identifies the effect and impact of an interaction. Falcon resorts to the catalog of rules for extracting the previously mentioned types of entities; additionally, a background knowledge base is utilized to determine the semantic type of the extracted entities. Since most of the descriptions of the interactions share similar patterns, i.e., the structure of the sentences is very repetitive, only a few extra rules are required to be added to the catalog of rules. The rules were created by replacing each drug mention with a variable (DrugX, DrugY). Out of 1,273,052 drug-drug interactions collected from DrugBank, 320 patterns were recognized; Table 7.1 shows a sample of the extracted patterns.

Table 7.1: Overview of extracted DDI patterns. Drugs mentions are in bold. Effect is in Italic. Impact is underlined

DDI Patterns
DrugY may increase the *anticoagulant activities* of **DrugX**.
the *risk or severity of bleeding and hemorrhage* can be increased when **DrugX** is combined with **DrugY**.
the *risk or severity of gastrointestinal bleeding* can be increased when **DrugX** is combined with **DrugY**.
the *risk or severity of bleeding* can be increased when **DrugY** is combined with **DrugX**.
the *metabolism* of **DrugY** can be decreased when combined with **DrugX**.
DrugX may decrease the *vasoconstricting activities* of **DrugY**.
DrugX may decrease the *excretion rate* of **DrugY** which could result in a higher serum level.
DrugY may increase the *constipating activities* of **DrugX**.
the *risk or severity of gastrointestinal bleeding and gastrointestinal ulceration* can be increased when **DrugX** is combined with **DrugY**.

As a result of the knowledge extraction process executed by Falcon, the Scientific Open Data data ecosystem makes available fine-grained representations of DDIs. This representation enables the deduction of new drug-drug interactions implemented as a service of this data ecosystem. Moreover, these descriptions are also used to validate the prediction tasks implemented in the Scientific Publications.

7.3.2 Knowledge Extraction for Knowledge Graphs Creation Pipelines

The knowledge extraction methods proposed in this thesis are utilized as a part of knowledge graph creation pipelines. For example, Falcon and Falcon2.0 are parts of the Knowledge4COVID knowledge graph; they are used to extract relevant entities and predicates that enable the fine-grained description of COVID-19 treatments and the potential adverse events that may occur when these treatments are combined with treatments of common comorbidities. Falcon [68] recognizes the words corresponding to the drugs that interact and the effect and impact of these interactions. Additionally, the extracted words are linked to terms in UMLS. As illustrated in Figure 7.2, "Metformin" and "Chloroquine" correspond to the extracted entities from the short text collected from DrugBank. At the same time, "excretion rate" and "decrease" represent, respectively, the effect and impact of the interaction of "Metformin" and "Chloroquine". The UMLS identifiers C0025598 and C0020336 are linked to "Metformin" and "Hydroxychloroquine", while C2827741 and C0547047 are related to "excretion rate" and "decrease", respectively. Falcon2.0 [69] also connects "Metformin" and "Chloroquine" to their corresponding resources in DBpedia and Wikidata. Falcon [68] is also used to extract the Drug-Drug Interactions (DDIs) reported in DrugBank as short texts.

Knowledge extraction methods proposed in this thesis recognize biomedical entities from textual data and link them to UMLS, and to resources in DBpedia, Wikidata, Uniprot, and DrugBank. A total of 12,223,409 UMLS annotations have been extracted by Falcon. These annotations are used for solving entity alignment and semantic data integration of biomedical entities in the Knowledge4COVID-19 KG (e.g., drugs, phenotypes, side effects, and adverse events). Moreover, there are 3,739,445 links to DBpedia, 3,476,435 links to Wikidata, 5,248 links to the Uniprot RDF KG, and 3,427 links to DrugBank. Drug indications, side effects, and adverse events of drug-drug interactions are recognized by Falcon. The extracted entities are linked to equivalent resources in existing KGs (i.e., DBpedia, Wikidata, DrugBank, and Uniprot) and annotated using UMLS terms and relations; networks of drug-drug, drug-target, and drug-side effect interactions predicted using diverse methods are also merged. This makes the Knowledge4COVID-19 KG a complementary source of knowledge that can be connected to existing COVID-19 KGs (e.g., the one implemented by Reese et al.) using the linking techniques implemented by Falcon.

7.3.3 Knowledge Extraction for Entity Alignment

Entity alignment performs the NER and NED tasks. Jozashoori et al. [64] uses Falcon2.0 [69] for performing NER and NED tasks related to entity alignment.

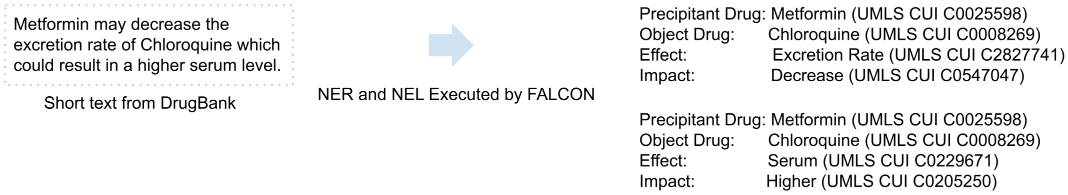
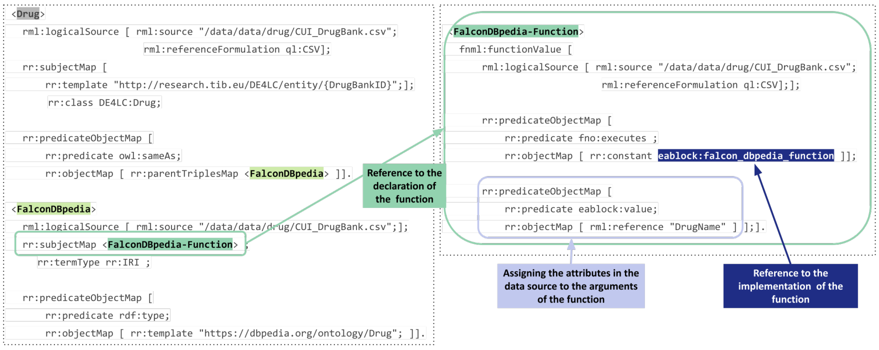

Metformin may decrease the excretion rate of Chloroquine which could result in a higher serum level.
Short text from DrugBank
NER and NEL Executed by FALCON
Precipitant Drug: Metformin (UMLS CUI C0025598)
Object Drug: Chloroquine (UMLS CUI C0008269)
Effect: Excretion Rate (UMLS CUI C2827741)
Impact: Decrease (UMLS CUI C0547047)
Precipitant Drug: Metformin (UMLS CUI C0025598)
Object Drug: Chloroquine (UMLS CUI C0008269)
Effect: Serum (UMLS CUI C0229671)
Impact: Higher (UMLS CUI C0205250)

Figure 7.2: FALCON Recognizes Relevant Entities and Predicates. As a result, a Fine-Grained Representation of Drug-Drug Interactions is part of the Knowledge4COVID-19 KG [70].

Figure 7.3 shows how Falcon is being used as part of the predefined functions of the *EABlock* approach. Falcon [68, 69] is empowered with a background knowledge that allows for the accurate recognition and linking of biomedical concepts. Falcon2.0 relies on background knowledge built from resources and their corresponding labels from diverse KGs (e.g., DBpedia, Wikidata, and UMLS). The labels in the background knowledge are the textual descriptions of the resources, which are connected using the `owl:sameAs` relation. The background knowledge utilized for Falcon API in *EABlock* is a subset of the one described in [69]. The background knowledge is filtered by omitting all the resources that are not related to the biomedical domain. The list that is utilized in the filtering process contains the following resource types: Chemicals & Drugs, Anatomy, Disorders, Living Beings, Organizations, Physiology, and Genes & Molecular Sequences. Applying this filtering to the background knowledge of Falcon reduces the ambiguity among the resources in the NED task and clears the noise that can be generated from unrelated resource types, e.g., street names.

7.3.4 Knowledge Extraction for Capturing Contextual Knowledge

The knowledge extraction methods presented in this thesis are used for capturing contextual knowledge in short text representing social media posts (tweets) in order to enhance the detection of eating disorders in social media posts. Benítez-Andrades et al. (under review) uses Falcon2.0 as one of the NED tools of the approach since the recognized entities by other approaches (Spacy2 [269]) were not always the same, and it was observed that a union of the entities obtained using



```
<Drug>
  rml:logicalSource [ rml:source "/data/data/drug/CUI_DrugBank.csv";
                      rml:referenceFormulation ql:CSV];
  rr:subjectMap [
      rr:template "http://research.tib.eu/DE4LC/entity/{DrugBankID}";];
       rr:class DE4LC:Drug;

  rr:predicateObjectMap [
      rr:predicate owl:sameAs;
      rr:objectMap [ rr:parentTriplesMap <FalconDBpedia> ]].

<FalconDBpedia>
  rml:logicalSource [ rml:source "/data/data/drug/CUI_DrugBank.csv";];
  rr:subjectMap <FalconDBpedia-Function> ;
     rr:termType rr:IRI ;

  rr:predicateObjectMap [
      rr:predicate rdf:type;
      rr:objectMap [ rr:template "https://dbpedia.org/ontology/Drug"; ]].
```

```
<FalconDBpedia-Function>
  fnml:functionValue [
      rml:logicalSource [ rml:source "/data/data/drug/CUI_DrugBank.csv";
                          rml:referenceFormulation ql:CSV];];

      rr:predicateObjectMap [
          rr:predicate fno:executes ;
          rr:objectMap [ rr:constant eablock:falcon_dbpedia_function ]];

      rr:predicateObjectMap [
          rr:predicate eablock:value;
          rr:objectMap [ rml:reference "DrugName" ] ];].
```

Reference to the declaration of the function
Assigning the attributes in the data source to the arguments of the function
Reference to the implementation of the function

Figure 7.3: An exemplary RML Mapping Rule calling one FnO function named FalconDBpedia-Function. This function performs NER and NEL to link drug names to resources in DBpedia [268].

both tools is more complete. A dataset of tweets is processed using Falcon2.0 and Spacy2, resulting in a total number of entities in the Wikipedia knowledge graph, as shown in Table 7.2.

Dataset	Tweets	Entities	Unique entities	Unique entities >2
Eating disorders	2,000	11,680	1,743	1,358

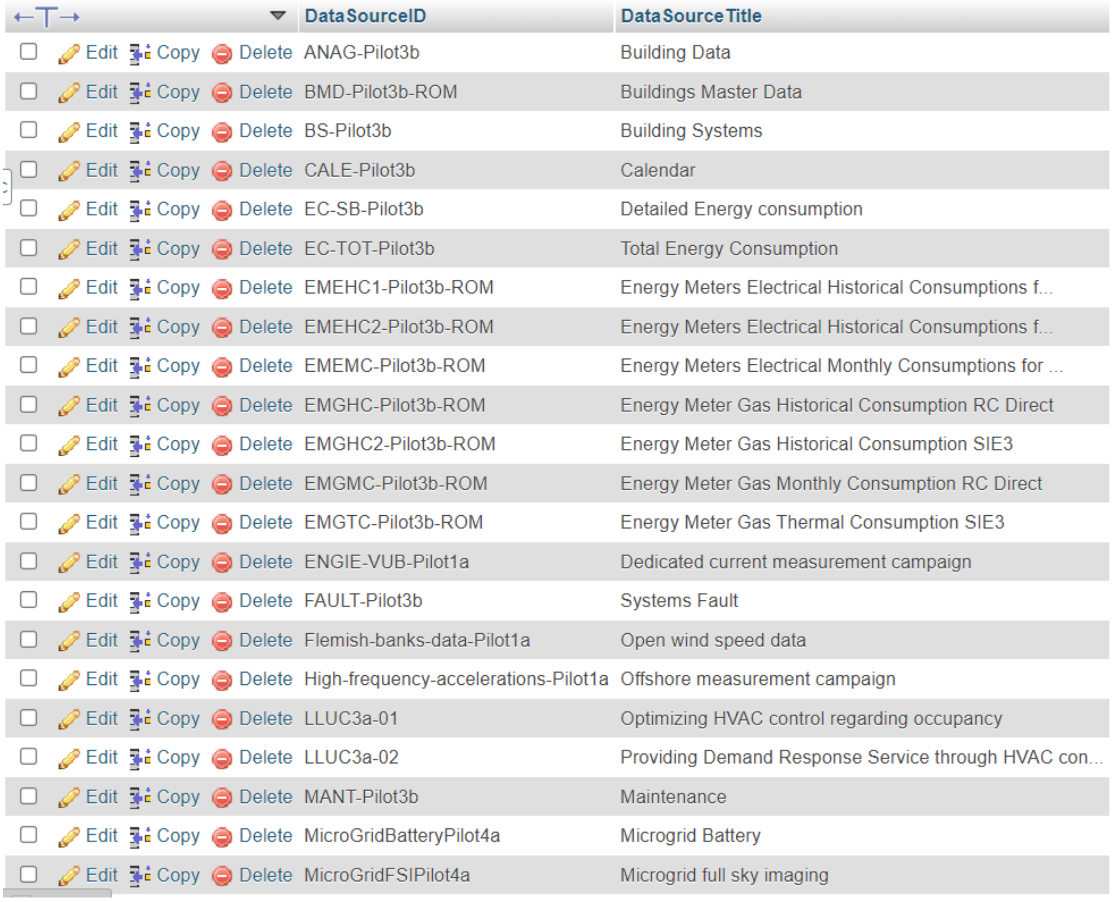
Table 7.2: Number of entities, unique entities, and unique entities appearing two or more times obtained from Wikidata for each of the datasets used.

7.3.5 Knowledge Extraction for annotating Customized Semantic Data Models

A customized version of Falcon [68] (E-Falcon[19]) is used to link entities recognized in a pilot datasets of a project related to the energy domain to concepts in a semantic data models. The datasets contain 40 different data sources with their corresponding titles (Figure 7.4). The annotation of the data sources allows to connect the knowledge in the semantic data models to existing knowledge in other vocabularies (e.g., Data Catalog Vocabulary[20]). Figure7.5 shows the online demo of E-Falcon.

[19] https://labs.tib.eu/sdm/efalcon/

[20] https://www.w3.org/TR/vocab-dcat-2/

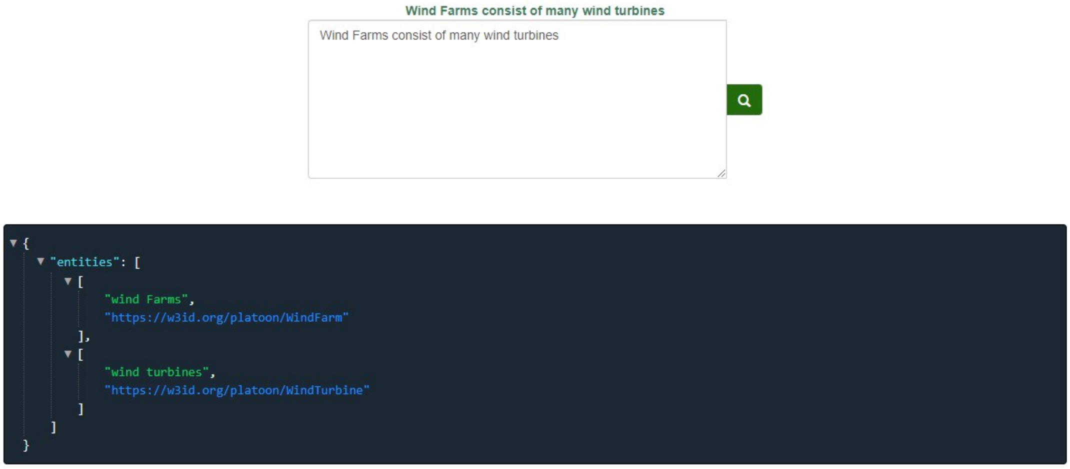


←T→			▼	DataSourceID	DataSourceTitle
□	Edit	Copy	Delete	ANAG-Pilot3b	Building Data
□	Edit	Copy	Delete	BMD-Pilot3b-ROM	Buildings Master Data
□	Edit	Copy	Delete	BS-Pilot3b	Building Systems
□	Edit	Copy	Delete	CALE-Pilot3b	Calendar
□	Edit	Copy	Delete	EC-SB-Pilot3b	Detailed Energy consumption
□	Edit	Copy	Delete	EC-TOT-Pilot3b	Total Energy Consumption
□	Edit	Copy	Delete	EMEHC1-Pilot3b-ROM	Energy Meters Electrical Historical Consumptions f...
□	Edit	Copy	Delete	EMEHC2-Pilot3b-ROM	Energy Meters Electrical Historical Consumptions f...
□	Edit	Copy	Delete	EMEMC-Pilot3b-ROM	Energy Meters Electrical Monthly Consumptions for ...
□	Edit	Copy	Delete	EMGHC-Pilot3b-ROM	Energy Meter Gas Historical Consumption RC Direct
□	Edit	Copy	Delete	EMGHC2-Pilot3b-ROM	Energy Meter Gas Historical Consumption SIE3
□	Edit	Copy	Delete	EMGMC-Pilot3b-ROM	Energy Meter Gas Monthly Consumption RC Direct
□	Edit	Copy	Delete	EMGTC-Pilot3b-ROM	Energy Meter Gas Thermal Consumption SIE3
□	Edit	Copy	Delete	ENGIE-VUB-Pilot1a	Dedicated current measurement campaign
□	Edit	Copy	Delete	FAULT-Pilot3b	Systems Fault
□	Edit	Copy	Delete	Flemish-banks-data-Pilot1a	Open wind speed data
□	Edit	Copy	Delete	High-frequency-accelerations-Pilot1a	Offshore measurement campaign
□	Edit	Copy	Delete	LLUC3a-01	Optimizing HVAC control regarding occupancy
□	Edit	Copy	Delete	LLUC3a-02	Providing Demand Response Service through HVAC con...
□	Edit	Copy	Delete	MANT-Pilot3b	Maintenance
□	Edit	Copy	Delete	MicroGridBatteryPilot4a	Microgrid Battery
□	Edit	Copy	Delete	MicroGridFSIPilot4a	Microgrid full sky imaging

Figure 7.4: A snippet of the data sources to be annotated by E-Falcon

7.3.6 Knowledge Extraction and Discovery from Social Media Posts

The knowledge extraction and discovery methods proposed in this thesis are employed in the project Knowledge4Hubris to analyze tweets related to people who suffer from hubris disease. Falcon and Falcon2.0 are used for annotating the corpus of tweets (15 million tweets). The PINYON approach is used to recommend related posts to the experts of the project based on predefined posts related to people who suffer from hubris disease. The experts in this project are interested in identifying people who suffer from hubris based on their way of writing posts and found it challenging to analyze hundreds of millions of tweets manually.



Wind Farms consist of many wind turbines

Wind Farms consist of many wind turbines

```
▼{
  ▼"entities": [
    ▼[
        "wind Farms",
        "https://w3id.org/platoon/WindFarm"
     ],
    ▼[
        "wind turbines",
        "https://w3id.org/platoon/WindTurbine"
     ]
   ]
 }
```

Figure 7.5: A snippet of the data sources to be annotated by E-Falcon

7.4 Evaluation

7.4.1 Drug-Drug-Interactions Use Case

Data about drug-drug interactions is collected from DrugBank release 2022-01-04 with 1,273,052 entries composed of pairs of drugs and the textual description of the effects of each interaction. In order to evaluate the performance of Falcon in this use case, 1,198 DDI descriptions were manually annotated by twelve annotators; annotations correspond to CUIs from UMLS and constitute the gold standard of the evaluation. For example, for the DDI description: “The serum concentration of Lepirudin can be decreased when it is combined with Tipranavir”; Lepirudin and Tipranavir correspond to the extracted entities from the above record, while decrease and serum concentration represent, respectively, the effect and impact of the interaction of Tipranavir with Lepirudin. One of the annotators was a senior researcher, two were experts in the biomedical domain, and the rest were Computer Science Ph.D. students. Disagreements among the annotators were solved by majority voting. A 2-fold cross-validation was followed. The evaluation indicates a precision of 98%. The 2% where Falcon failed to extract and link the terms correctly are interactions that contain more than one interaction in the same sentence, where FALCON was only considering one interaction (Table 7.1 last pattern). All the drug-drug interactions that followed this pattern were corrected manually before integrating them into the Knowledge4COVID-19 KG [70].

7.4.2 Entity Alignment Use Case

The experiments setup in [64] use biomedical data. Accordingly, an API of Falcon[21] that provides a filtered subset of the background knowledge [69] omitting the resources that are not related to the biomedical domain is used. A list of related resource types is utilized for filtering the background knowledge. The list contains the following resource types: Chemicals & Drugs, Anatomy, Disorders, Living Beings, Organizations, Physiology, and Genes & Molecular Sequences. Applying this filtering to the background knowledge of Falcon reduces the ambiguity among the resources in the EL task and clears the noise that can be generated by irrelevant resources. In order to understand the experimental results reported in [64] related to our knowledge extraction methods, we describe in the following the Testbeds configuration used.

Testbeds: For each DBpedia, Wikidata, and UMLS, five testbeds are generated by manipulating the gold standard datasets considering frequent quality issues that may exist in datasets; character capitalization, elimination, insertion, and replacement. All created testbeds related to each DBpedia, Wikidata, and UMLS possess lower quality than the gold standard datasets due to the intentionall misspelling errors that we have generated in values of the records. The errors include: **a)** capitalizing all the characters of a record value; **b)** randomly eliminating one character from the value; **c)** randomly replacing one character with another randomly selected character; and **d)** inserting a randomly selected character to a random location in the record value. Accordingly, each of mentioned errors are introduced in 50% of the records in one of the testbeds, the other 50% of the records carry the same values as the gold standards. The last testbed created by including all four types of errors, has the lowest quality. In this dataset, each 20% out of 80% of the records involves exactly one of the four errors. Therefore, 20% of the records are error-free, in contrast to the other four test beds, in which 50% of all records are free from errors.

Table 7.3 demonstrates the results of running the Falcon's functions implemented in *EABlock* [64] over the five configurations of error types explained in *Testbeds.* The entity alignment engine (Falcon) reports relatively high performance. Cleaning the background knowledge of Falcon and filtering it to contain only resources related to the biomedical domain, reduces the ambiguity among the resources in the background knowledge; this improves the performance of Falcon significantly and plays a major role in having such high performance. Also, having the input of the entity alignment module as keywords without any noise, e.g., stopwords, helps the module recognize and link the labels precisely. Moreover, Table 7.3 suggests that the used entity alignment module is able to overcome the pro-

[21] https://labs.tib.eu/sdm/biofalcon/

Table 7.3: *Effectiveness.* The performance of *EABlock* is assessed in 15 datasets. The aligned entities are compared with the resources of the original labels. Misspelling errors are added with five types of transformations. Entity alignment is performed by the keyword-based functions over the keywords generated by the five transformations. The aligned entities are compared with the resources of the original labels. *EABlock* fails to recognize long labels generated from keywords with non-alphanumerical characters present in UMLS (e.g., n-((2r)-1-(((2r)-1-(((2r)-6-amino-1-(4-amino-4-carboxy-1-piperidinyl)-1-oxo-2-hexanyl)amino)-4-methyl-1-oxo-2-pentanyl)amino)-1-oxo-3-phenyl-2-propanyl)-d-phenylalaninamide).

UMLS			
Text Error Type	**Precision**	**Recall**	**F1 Score**
Capitalization of all characters	1.0	0.97	0.99
Elimination of a character	0.99	0.74	0.85
Replacement of a character	0.99	0.75	0.85
Insertion of a new character	0.99	0.50	0.66
Combination of all 4 errors	1.0	0.78	0.88
DBpedia			
Text Error Type	**Precision**	**Recall**	**F1 Score**
Capitalization of all characters	0.78	0.78	0.78
Elimination of a character	0.78	0.78	0.78
Replacement of a character	0.98	0.98	0.98
Insertion of a new character	0.78	0.78	0.78
Combination of all 4 errors	0.78	0.78	0.78
Wikidata			
Text Error Type	**Precision**	**Recall**	**F1 Score**
Capitalization of all characters	0.99	0.99	0.99
Elimination of a character	0.99	0.99	0.99
Replacement of a character	0.99	0.99	0.99
Insertion of a new character	0.99	0.99	0.99
Combination of all 4 errors	0.99	0.99	0.99

posed error types. There are records for which Falcon fails to return the expected linked entity based on the gold standard. These failures are mostly caused by data quality issues in the KGs. To clarify, we enumerate a couple of these examples with possible explanations: **a)** there are cases in which the linked entities retrieved by the entity alignment engine (Falcon) are correct, despite the fact that their identifiers differ from those available in the gold standard. For instance, for the keyword “malignant histiocytosis" *EABlock* using Falcon’s functions retrieves “Q164952"[22] from Wikidata, while, in the gold standard the Wikidata identifier for the same keyword appears to be “Q52962465"[23]. However, both identifiers lead to the same entry on Wikidata; for some keywords, more than one identifier exists in such KGs. **b)** Another example of unexpected retrieved linked entities, can be observed in case of having long combinations of keywords, such as “early infantile epileptic encephalopathy 19". In this case, the Falcon’s functions in *EABlock* link the first entity that can be recognized by the first couple of keywords; “Q61913448"[24] which belongs to the label “early infantile epileptic encephalopathy 37". The same failure case can be observed in retrieval from DBpedia as well; for the keyword “Chronic leukemia" *EABlock* retrieves a link to “B-cell_chronic_lymphocytic_leukemia"[25] while based on the gold standard the correct link is “Chronic_leukemia"[26].

7.4.3 Annotating Customized Semantic Data Model Use Case

Using the customized version of Falcon [68] (E-Falcon[27]) an evaluation study for annotating the data sources is performed. The titles are used as an input to E-Falcon. The goal of this study is to evaluate the performance of E-Falcon for linking concepts in a short text to data models. E-Falcon is able to annotate 25 data sources to classes in the data models. Thus, 62.5% of the data sources are annotated by E-Falcon. Figure 7.6 shows an overview of the annotated data sources with their corresponding resources in the data models. For the annotated data sources, E-Falcon is able to annotate the titles of the data sources with at least one annotation and at most two annotations. E-Falcon is not able to annotate 37.5% of the data sources since their titles contain entities that are not present in the data models. The reported results show the benefits of using Falcon for annotating the data sources without requiring any training data related to the energy domain (only the classes of the data models are used to build the background knowledge of the customized version of Falcon).

[22] https://www.wikidata.org/wiki/Q164952

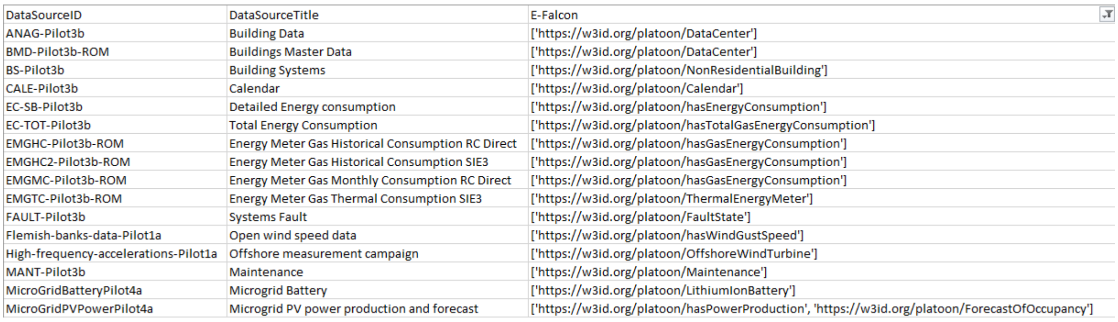
[23] http://www.wikidata.org/entity/Q52962465

[24] https://www.wikidata.org/wiki/Q61913448

[25] https://dbpedia.org/page/Chronic_lymphocytic_leukemia

[26] https://dbpedia.org/page/Chronic_leukemia

[27] https://labs.tib.eu/sdm/efalcon/



DataSourceID	DataSourceTitle	E-Falcon
ANAG-Pilot3b	Building Data	['https://w3id.org/platoon/DataCenter']
BMD-Pilot3b-ROM	Buildings Master Data	['https://w3id.org/platoon/DataCenter']
BS-Pilot3b	Building Systems	['https://w3id.org/platoon/NonResidentialBuilding']
CALE-Pilot3b	Calendar	['https://w3id.org/platoon/Calendar']
EC-SB-Pilot3b	Detailed Energy consumption	['https://w3id.org/platoon/hasEnergyConsumption']
EC-TOT-Pilot3b	Total Energy Consumption	['https://w3id.org/platoon/hasTotalGasEnergyConsumption']
EMGHC-Pilot3b-ROM	Energy Meter Gas Historical Consumption RC Direct	['https://w3id.org/platoon/hasGasEnergyConsumption']
EMGHC2-Pilot3b-ROM	Energy Meter Gas Historical Consumption SIE3	['https://w3id.org/platoon/hasGasEnergyConsumption']
EMGMC-Pilot3b-ROM	Energy Meter Gas Monthly Consumption RC Direct	['https://w3id.org/platoon/hasGasEnergyConsumption']
EMGTC-Pilot3b-ROM	Energy Meter Gas Thermal Consumption SIE3	['https://w3id.org/platoon/ThermalEnergyMeter']
FAULT-Pilot3b	Systems Fault	['https://w3id.org/platoon/FaultState']
Flemish-banks-data-Pilot1a	Open wind speed data	['https://w3id.org/platoon/hasWindGustSpeed']
High-frequency-accelerations-Pilot1a	Offshore measurement campaign	['https://w3id.org/platoon/OffshoreWindTurbine']
MANT-Pilot3b	Maintenance	['https://w3id.org/platoon/Maintenance']
MicroGridBatteryPilot4a	Microgrid Battery	['https://w3id.org/platoon/LithiumIonBattery']
MicroGridPVPowerPilot4a	Microgrid PV power production and forecast	['https://w3id.org/platoon/hasPowerProduction', 'https://w3id.org/platoon/ForecastOfOccupancy']

Figure 7.6: A snippet of the data sources to be annotated by E-Falcon

7.5 Summary

This chapter presents the applications of the proposed contributions of this thesis in real-life use cases related to projects. In our motivation, we presented a use case of integrating heterogeneous sources of knowledge and how knowledge extraction can enhance the performance of this task. Moreover, a use case of extracting knowledge from semi-structured data is presented; it shows how the proposed knowledge extraction techniques are applied effectively for transforming the textual description of Drug-Drug-Interactions into a structured representation that has been used in the Knowledge4Covid knowledge graph. Additionally, the application of knowledge extraction for entity alignment tasks and analysis of this application is described. A use case of employing the proposed knowledge extraction methods for annotating a domain-specific semantic data model and the related evaluation are presented. Thus, exploiting the different use cases of applying the proposed methods of this thesis in real-world scenarios allow us to answer the research question **RQ4**

Chapter 8

Conclusion and Future Directions

Daily, humans or computer systems generate a tremendous amount of data. The produced data are represented as unstructured, semi-structured, or structured data. The current growth of data demonstrated the necessity for a machine-readable representation of the data, where structured or semi-structured data formats should be used. Considering that most of the web-based data are unstructured data highlights the need for accurate knowledge extraction techniques that are able to capture knowledge from various data sources. Following the definition of the knowledge extraction task, the extracted knowledge should be mapped to existing knowledge (e.g., in a knowledge graph "KG"), where data is stored using a graph-structured data model. Knowledge extraction targeting existing knowledge in a KG comprises several tasks; mainly named-entity recognition (NER), named-entity disambiguation (NED), relation recognition (RR), and relation linking (RL). In this thesis, we study the problem of recognizing entities and relations in a natural language text. Furthermore, we address the problem of linking the recognized entities and relations to a KG, and define a rule-based AI approach that resorts to catalogs of linguistic and domain-specific rules, and deductive database to address the challenges related to the studied problem. Furthermore, we devise a neuro-symbolic approach consisting of symbolic and sub-symbolic components to enhance the performance of the defined rule-based AI approach for the same studied problem. Moreover, we investigate the problem of discovering patterns in the extracted knowledge, and develop a context-aware framework for unveiling semantically related posts in a corpus. Finally, we show how the defined contributions can be utilized in real-world applications for extracting knowledge from unstructured and semi-structured data in different domains and the advantages of extracting such knowledge. The outcomes provide an evidence to the effectiveness of combining the reasoning capacity of the symbolic frameworks with the power of pattern recognition and classification of sub-symbolic models.

8.1 Revising the Research Questions

RQ1: What is the impact of the linguistics rules of a natural language on the tasks of knowledge extraction?

Chapter 4 presents a rule-based AI approach for knowledge recognition and linking. The defined approach effectively identifies entities and relation in a natural language text and maps the recognized entities and relations within the text to their resources in a target knowledge graph or knowledge base. The defined approach relies on the defined catalog of linguistic rules to overcome the challenges of recognizing entities and relations in a natural language text. The defined approach performs joint entity and relation recognition by leveraging several fundamental principles of English morphology (e.g., compounding, headword identification). It uses the context of entities for finding relations and does not require training data. Additionally, it overcomes the challenges of linking entities and relations to a specific knowledge graph or knowledge base using a lightweight linguistic approach relying on the defined background knowledge. It is also important to highlight that the defined approach is agnostic of the studied natural language or the target knowledge graph or knowledge base. For example, it can be utilized for the German language by replacing the English linguistic rules with the corresponding German linguistic rules, and is implemented for different knowledge graphs (e.g., DBpededia, Wikidata) and knowledge bases (e.g., UMLS, DrugBank). Our empirical study using several standard benchmarks and datasets shows that the defined approach outperforms state-of-the-art knowledge recognition and linking approaches. Our work contributes to knowledge extraction and management methods and provides the basis for further development of knowledge extraction techniques over unstructured data for symbolic systems.

RQ2: How does the knowledge represented in community-maintained KGs and domain-specific KBs can be utilized for knowledge extraction tasks?

Chapter 5 describes a deductive database (background knowledge) created on top of community-maintained KGs and domain-specific KBs. We devise an alignment representation of the knowledge contained in the community-maintained KGs and domain-specific KBs. We exploit various properties of the resources in the used knowledge sources (e.g., labels, semantic types, etc.). Extensional and intensional databases comprise the deductive database. Furthermore, the interpretability of the defined approach allows for backtracking errors at each step. We evaluate the quality of the extracted knowledge by performing an ablation study in which we use the various community-maintained KGs or domain-specific KBs separately or in combination. The results show that the knowledge represented in community-maintained KGs and domain-specific KBs empowers knowledge extraction approaches by the knowledge encoded in these knowledge sources. In addition, the results indicate that the choice of the underlying knowledge source is dependent on the domain of the studied unstructured data (e.g., the biomedical domain). The devised background knowledge is also utilized in a neuro-symbolic approach used for NER and NED. The sub-symbolic component of the neuro-symbolic approach resorts to the defined background knowledge to predict the semantic type of an entity in order to improve named-entity disambiguation. The empirical evaluation of the neuro-symbolic approach concludes that the sub-symbolic component complements the performance of a symbolic system consisting of human-given rule templates, outperforming the black-box neural baselines.

RQ3: How contextual knowledge can be used to enhance knowledge extraction over unstructured data?

Chapter 6 presents a context-aware framework for unveiling semantically related posts in a corpus; it is a knowledge-driven framework that retrieves associated posts effectively. It implements a two-fold pipeline. First, it encodes a corpus of posts and an input post in a graph; To address the challenge of analyzing non-machine-readable data, posts are tagged with entities from existing KGs and linked based on the similarity of their entities. The encoded graph is utilized to discover communities of similar posts during the decoding phase. We formulate the challenge of uncovering semantically related posts in a corpus as the Vertex Coloring Problem, where communities of similar posts consist of posts annotated with entities of the same color. Utilizing this problem casting and the scalability of vertex coloring techniques enables the defined framework to overcome the scalability challenge. The defined framework performs the decoding phase driven by a heuristic-based method that identifies relatedness among posts based on contextual knowledge and effectively groups the most similar posts in the same communities using graph theory-derived results [67]. We empirically evaluated the defined framework over various datasets and compared it with state-of-the-art implementations of the decoding phase. Additionally, the quality of the produced communities is evaluated using a variety of metrics. The observed outcomes indicate that the defined framework accurately identifies semantically related posts in different contexts. Furthermore, the provided results contextualize the influence of known properties regarding the optimality of existing heuristics for vertex graph coloring and their implications for the scalability of the defined framework.

RQ4: How does the knowledge extracted from unstructured data can be used for real-world applications?

Chapter 7 presents the applications of all the presented contributions of this thesis use-cases of real-world projects. The presented frameworks are utilized to extract knowledge from unstructured data. The rule-based approaches [68, 69] Falcon[1] and Falcon2[2] are used by the research community for the tasks of NER, NED, RR, and RL using their public APIs. Both APIs are heavily utilized, with 4,119,179 hits for Falcon since April 2019 and 5,664,407 hits for Falcon2 since October 2020 (on November 2022). Both approaches are employed in life science projects (e.g., iASiS, BigMedilytics, P4-Lucat, CLARIFY, ImProVIT, Knowledge4Hubris, and K4COVID [70]) to extract mentions of medical terms from a natural language text then link them to a domain-specific knowledge base like UMLS (e.g., drugs and their side effects from medical notes describing patients treatments). The defined rule-based approach and background knowledge are also applied for a particular area (e.g., energy domain)[3] where the objective is to detect entities in a natural language text and link them to a predefined semantic data model described by experts in the EU project PLATOON [71, 72]. The frameworks described in this thesis are also applied in the CoyPu project to extract knowledge from diverse datasets linked to various topics. In the project Knowledge4Hubris, the defined approach for revealing semantically related posts in a corpus is employed to analyze tweets about hubris disease patients; it was challenging to evaluate hundreds of millions of tweets for the purpose of identifying people who suffer from hubris based on their writing style. In addition to the mentioned applications, the defined methods of this thesis are also used as components of knowledge graph creation and semantic data integration pipelines [64, 73, 74].

8.2 Limitations

Despite the fact that the overall achieved research objectives have been met, we acknowledge that there are limitations of the described research work which have not been covered in the scope of the thesis. First, the catalog of rules in the defined rule-based approach has limitations in processing complete paragraphs since it does not consider coherence among the concepts and fits better for short

[1] `https://labs.tib.eu/falcon/`

[2] `https://labs.tib.eu/falcon/falcon2/`

[3] `https://labs.tib.eu/sdm/efalcon/`

text (e.g., questions, facts). Secondly, our defined method for lightweight neuro-symbolic named-entity disambiguation is promising work in this direction. However, substantial work is needed collectively in the research community to develop strong theoretical foundations, solving scaling issues until neuro-symbolic methods are widely adapted against supervised models. For example, there could be a dataset or underlying KG where our approach finds limited performance. Here, one potential source of errors could be domain-specific rules, as our approach heavily relies on such human-made rules. Furthermore, as observed in the end-to-end entity linking experiment in Chapter 5, human-curated rules adversely affect entity recognition performance, resulting in less value for end-to-end settings than when entity mention is given. However, it is a trade-off either to create a few handcrafted rules versus a large corpus of labeled training data. Creating this human template is laborious work and requires domain experts, which is a limitation of this work. Also, we do not have an effective solution for improving entity type prediction without gold-labeled training samples describing entities (e.g., definitions). Finally, our defined framework for unveiling semantically related posts suffers from the following limitations. First is the constraint to recognize dark entities. Dark entities are entities that do not exist in any knowledge graph [69]. Since our defined approach resorts to various knowledge graphs to build its background knowledge, it is not able to extract any knowledge (i.e., contextual description) about these dark entities. Further, the specificity of the entities in the corresponding KGs affects the accuracy of our framework. Thus, if an entity is miss-linked to a KG class, the embedding's quality will also be affected. It is important to mention that this limitation is a limitation of the studied corpus and the entity linking tools, and it is not a limitation in our algorithm design. New entities are added regularly to the used community-maintained KGs, enabling the entity linking tools to recognize the newly added entities. So adding more knowledge to the used KGs can overcome these types of limitations. Moreover, the solution presented in this thesis resorts to external components like Falcon2.0, FALCON, TagMe, and DBpedia Spotlight for entity linking. Hence, inheriting the pitfalls of such a tool. For instance, if a tool fails to extract and link a set of entities to the corresponding knowledge graph, our framework will also be negatively affected. We also ignore edge features (the relationship between entities within a tweet), and one possible extension is to study the effect of edge features on our approach. Therefore, the limitations of the methods defined in this thesis can are related to the state-of-the-art methods' performance and could be overcome.

8.3 Future Directions

Based on our findings and the contributions made in this thesis, we present to the scientific community the following next directions for this work:

- Exploiting researching techniques that enable adjusting the catalog of rules and the background knowledge to the frequent changes in community-maintained knowledge graphs.

- Improving semantic type prediction for entities without description.

- Extending the described symbolic approach of this thesis for other languages.

- Applying the defined framework for unveiling semantically related posts in a corpus in a different domain (e.g., scholarly networks).

8.4 Closing Remarks

In the era of digitization, data availability has exponentially grown in recent years, and a similar growth rate is expected in the next decade. This growth demands effective knowledge extraction approaches. These approaches should be able to extract the encoded knowledge in a natural language text and represent it as machine-readable data to facilitate various applications that can be performed over the extracted knowledge. Transforming unstructured data to semi-structured or structured data requires novel and precise techniques for enabling not only knowledge extraction but also effective knowledge representations, integration, and discovery. In this thesis, we have shown that resorting to human-predefined rules enhances the performance of knowledge extraction techniques. Moreover, combining predefined human rules with advanced machine learning models improves the overall performance of knowledge extraction methods. In addition, the described contributions for knowledge extraction support knowledge discovery approaches by allowing knowledge discovery methods to analyze machine-readable data instead of unstructured data. These results suggest that the techniques defined in this thesis provide effective solutions for supporting knowledge extraction, management, and discovery over unstructured data.

Bibliography

[1] Heiko Paulheim. "Knowledge graph refinement: A survey of approaches and evaluation methods". In: *Semantic web* 8.3 (2017), pp. 489–508.

[2] Wei Shen, Jianyong Wang, and Jiawei Han. "Entity linking with a knowledge base: Issues, techniques, and solutions". In: *IEEE Transactions on Knowledge and Data Engineering* 27.2 (2014), pp. 443–460.

[3] Behrang Mohit. "Named entity recognition". In: *Natural language processing of semitic languages*. Springer, 2014, pp. 221–245.

[4] Nguyen Bach and Sameer Badaskar. "A review of relation extraction". In: *Literature review for Language and Statistics II* 2 (2007), pp. 1–15.

[5] Özge Sevgili et al. "Neural entity linking: A survey of models based on deep learning". In: *Semantic Web* Preprint (2022), pp. 1–44.

[6] Dieter Fensel et al. "Why We Need Knowledge Graphs: Applications". In: *Knowledge Graphs*. Springer, 2020, pp. 95–112.

[7] Yao Zhao et al. "Implicit Relation Linking for Question Answering over Knowledge Graph". In: *Findings of the Association for Computational Linguistics: ACL 2022*. 2022, pp. 3956–3968.

[8] Makoto Miwa and Yutaka Sasaki. "Modeling Joint Entity and Relation Extraction with Table 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*. 2014, pp. 1858–1869. URL: http://aclweb.org/anthology/D/D14/D14-1200.pdf.

[9] Johannes Kirschnick, Holmer Hemsen, and Volker Markl. "JEDI: Joint Entity and Relation Detection using Type Inference". In: *Proceedings of ACL-2016 System Demonstrations, Berlin, Germany, August 7-12, 2016*. 2016, pp. 61–66. DOI: 10.18653/v1/P16-4011. URL: https://doi.org/10.18653/v1/P16-4011.

[10] Shaolei Wang et al. "Joint Extraction of Entities and Relations Based on a Novel Graph Scheme". In: *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, July 13-19, 2018, Stockholm, Sweden*. 2018, pp. 4461–4467. DOI: 10.24963/ijcai.2018/620. URL: https://doi.org/10.24963/ijcai.2018/620.

[11] Mohnish Dubey et al. "EARL: Joint Entity and Relation Linking for Question Answering over Knowledge Graphs". In: *The Semantic Web - ISWC 2018 - 17th International Semantic Web Conference, Monterey, CA, USA, October 8-12, 2018, Proceedings, Part I*. 2018, pp. 108–126. DOI: 10.1007/978-3-030-00671-6_7. URL: https://doi.org/10.1007/978-3-030-00671-6%5C_7.

[12] Kuldeep Singh et al. "Capturing Knowledge in Semantically-typed Relational Patterns to Enhance Relation Linking". In: *K-Cap 2017, to appear.* 2017.

[13] Jens Lehmann et al. "DBpedia–a large-scale, multilingual knowledge base extracted from Wikipedia". In: *Semantic Web* 6.2 (2015), pp. 167–195. DOI: `10.3233/SW-140134`.

[14] Denny Vrandecic. "Wikidata: a new platform for collaborative data collection". In: *Proceedings of the 21st World Wide Web Conference, WWW 2012, Lyon, France, April 16-20, 2012 (Companion Volume).* 2012, pp. 1063–1064.

[15] Olivier Bodenreider. "The Unified Medical Language System (UMLS): integrating biomedical terminology." In: *Nucleic Acids Research* 32.Database-Issue (2004), pp. 267–270. URL: `http://dblp.uni-trier.de/db/journals/nar/nar32.html#Bodenreider04`.

[16] Shanchan Wu, Kai Fan, and Qiong Zhang. "Improving Distantly Supervised Relation Extraction with Neural Noise Converter and Conditional Optimal Selector". In: *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019.* AAAI Press, 2019, pp. 7273–7280.

[17] Samuel Broscheit. "Investigating Entity Knowledge in BERT with Simple Neural End-To-End Entity Linking". In: *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL).* 2019, pp. 677–685.

[18] Shulin Cao et al. "KQA Pro: A Dataset with Explicit Compositional Programs for Complex Question Answering over Knowledge Base". In: *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers).* 2022, pp. 6101–6119.

[19] Nuno Silva, David Ribeiro, and Liliana Ferreira. "Information extraction from unstructured recipe data". In: *Proceedings of the 2019 5th international conference on computer and technology applications.* 2019, pp. 165–168.

[20] Kiran Adnan and Rehan Akbar. "Limitations of information extraction methods and techniques for heterogeneous unstructured big data". In: *International Journal of Engineering Business Management* 11 (2019), p. 1847979019890771.

[21] Sunil Kumar et al. "Semantic Information Extraction from Multi-Corpora Using Deep Learning". In: *Computers, Materials and Continua* (2021), pp. 1–17.

[22] Shushanta Pudasaini et al. "Application of NLP for Information Extraction from Unstructured Documents". In: *Expert Clouds and Applications.* Springer, 2022, pp. 695–704.

[23] Ellery Smith et al. "Lillie: Information extraction and database integration using linguistics and learning-based algorithms". In: *Information Systems* 105 (2022), p. 101938.

[24] Kalgi Gandhi and Nidhi Madia. "Information extraction from unstructured data using RDF". In: *2016 International Conference on ICT in Business Industry & Government (ICTBIG).* IEEE. 2016, pp. 1–6.

[25] Kiran Adnan and Rehan Akbar. "An analytical study of information extraction from unstructured and multidimensional big data". In: *Journal of Big Data* 6.1 (2019), pp. 1–38.

[26] Quratulain N Rajput and Sajjad Haider. "Use of Bayesian network in information extraction from unstructured data sources". In: *International Journal of Computer and Information Engineering* 3.4 (2009), pp. 950–956.

[27] Gohar Zaman et al. "Information extraction from semi and unstructured data sources: A systematic literature review". In: *ICIC Exp. Lett.* 14.6 (2020), pp. 593–603.

[28] Raghu Anantharangachar, Srinivasan Ramani, and S Rajagopalan. "Ontology guided information extraction from unstructured text". In: *arXiv preprint arXiv:1302.1335* (2013).

[29] Ravendar Lal et al. "Information Extraction of Security related entities and concepts from unstructured text." In: (2013).

[30] Xiao-Yuan Bao et al. "A customized method for information extraction from unstructured text data in the electronic medical records". In: *Beijing da xue xue bao. Yi xue ban= Journal of Peking University. Health Sciences* 50.2 (2018), pp. 256–263.

[31] Ana-Maria Popescu. *Information extraction from unstructured web text.* Vol. 68. 02. 2007.

[32] W. Shen, J. Wang, and J. Han. "Entity Linking with a Knowledge Base: Issues, Techniques, and Solutions". In: *IEEE Transactions on Knowledge and Data Engineering* 27.2 (2015), pp. 443–460.

[33] Krisztian Balog. *Entity-oriented search.* Springer Open, 2018.

[34] Heng Ji. *Entity Discovery and Linking and Wikification Reading List.* Jan. 2019. URL: http://nlp.cs.rpi.edu/kbp/2014/elreading.html.

[35] Wikipedia. *Wikipedia.* PediaPress, 2004.

[36] Jonathan Raphael Raiman and Olivier Michel Raiman. "DeepType: multilingual entity linking by neural type system evolution". In: *Thirty-Second AAAI Conference on Artificial Intelligence.* 2018.

[37] Farzaneh Mahdisoltani, Joanna Biega, and Fabian M. Suchanek. "YAGO3: A Knowledge Base from Multilingual Wikipedias". In: *CIDR 2015, Seventh Biennial Conference on Innovative Data Systems Research, Asilomar, CA, USA, January 4-7, 2015, Online Proceedings.* 2015.

[38] Kurt D. Bollacker, Robert P. Cook, and Patrick Tufts. "Freebase: A Shared Database of Structured General Human Knowledge". In: *AAAI 2007.* 2007.

[39] Nikolaos Kolitsas, Octavian-Eugen Ganea, and Thomas Hofmann. "End-to-End Neural Entity Linking". In: *Proceedings of the 22nd Conference on Computational Natural Language Learning.* 2018, pp. 519–529.

[40] Octavian-Eugen Ganea and Thomas Hofmann. "Deep Joint Entity Disambiguation with Local Neural Attention". In: *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017.* 2017, pp. 2619–2629.

[41] Dmitry Zelenko, Chinatsu Aone, and Anthony Richardella. "Kernel Methods for Relation Extraction". In: *J. Mach. Learn. Res.* 3 (Mar. 2003), pp. 1083–1106. ISSN: 1532-4435. URL: http://dl.acm.org/citation.cfm?id=944919.944964.

[42] Razvan C. Bunescu and Raymond J. Mooney. "A Shortest Path Dependency Kernel for Relation Extraction". In: *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing.* HLT '05. Vancouver, British Columbia, Canada: Association for Computational Linguistics, 2005, pp. 724–731. DOI: 10.3115/1220575.1220666. URL: https://doi.org/10.3115/1220575.1220666.

[43] Michele Banko and Oren Etzioni. “The Tradeoffs Between Open and Traditional Relation Extraction”. In: *Proceedings of ACL-08: HLT*. Columbus, Ohio: Association for Computational Linguistics, June 2008, pp. 28–36. URL: http://www.aclweb.org/anthology/P/P08/P08-1004.

[44] Jun Zhu et al. “StatSnowball: A Statistical Approach to Extracting Entity Relationships”. In: *Proceedings of the 18th International Conference on World Wide Web*. WWW ’09. Madrid, Spain: ACM, 2009, pp. 101–110. ISBN: 978-1-60558-487-4. DOI: 10.1145/1526709.1526724. URL: http://doi.acm.org/10.1145/1526709.1526724.

[45] Katrin Fundel, Robert Küffner, and Ralf Zimmer. “RelEx—Relation extraction using dependency parse trees”. In: *Bioinformatics* 23.3 (2007), pp. 365–371. DOI: 10.1093/bioinformatics/btl616. eprint: /oup/backfile/content_public/journal/bioinformatics/23/3/10.1093/bioinformatics/btl616/2/btl616.pdf. URL: http://dx.doi.org/10.1093/bioinformatics/btl616.

[46] Isaiah Onando Mulang’, Kuldeep Singh, and Fabrizio Orlandi. “Matching Natural Language Relations to Knowledge Graph Properties for Question Answering”. In: *Semantics 2017*. 2017.

[47] Mohnish Dubey et al. “AskNow: A Framework for Natural Language Query Formalization in SPARQL”. In: *The Semantic Web. Latest Advances and New Domains - 13th International Conference, ESWC 2016, Heraklion, Crete, Greece, May 29 - June 2, 2016, Proceedings*. Springer, 2016, pp. 300–316.

[48] Xueling Lin, Lei Chen, and Chaorui Zhang. “TENET: Joint Entity and Relation Linking with Coherence Relaxation”. In: *Proceedings of the 2021 International Conference on Management of Data*. 2021, pp. 1142–1155.

[49] Yixin Cao et al. *Neural Collective Entity Linking*. 2018. arXiv: 1811.08603. URL: http://arxiv.org/abs/1811.08603.

[50] Emrah Inan and Oguz Dikenelli. “A Sequence Learning Method for Domain-Specific Entity Linking”. In: *Proceedings of the Seventh Named Entities Workshop*. Melbourne, Australia: Association for Computational Linguistics, 2018, pp. 14–21. URL: http://aclweb.org/anthology/W18-2403.

[51] Alberto Cetoli et al. “A Neural Approach to Entity Linking on Wikidata”. In: *European Conference on Information Retrieval*. Springer. 2019, pp. 78–86.

[52] Isaiah Onando Mulang et al. “Context-aware Entity Linking with Attentive Neural Networks on Wikidata Knowledge Graph”. In: *arXiv preprint arXiv:1912.06214* (2019).

[53] Fabian Abel et al. “Twitter-based user modeling for news recommendations”. In: *Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence*. 2013, pp. 2962–2966.

[54] A. Belhadi et al. “A Data-Driven Approach for Twitter Hashtag Recommendation”. In: *IEEE Access* 8 (2020), pp. 79182–79191.

[55] Ralf Krestel et al. “Tweet-recommender: Finding relevant tweets for news articles”. In: *Proceedings of the 24th International Conference on World Wide Web*. 2015, pp. 53–54.

[56] Deepak Kumar Jain, Akshi Kumar, and Vibhuti Sharma. “Tweet recommender model using adaptive neuro-fuzzy inference system”. In: *Future Generation Computer Systems* (2020), pp. 996–1009.

[57] Areej Alsini, Amitava Datta, and Du Q Huynh. “On Utilizing Communities Detected From Social Networks in Hashtag Recommendation”. In: *IEEE Transactions on Computational Social Systems* (2020), pp. 971–982.

[58] Abdullah Alshammari et al. “Twitter User Modeling Based on Indirect Explicit Relationships for Personalized Recommendations”. In: *International Conference on Computational Collective Intelligence*. Springer. 2019, pp. 93–105.

[59] Ryan Kiros et al. “Skip-thought vectors”. In: *Advances in neural information processing systems*. 2015, pp. 3294–3302.

[60] Nils Reimers and Iryna Gurevych. “Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks”. In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. ACL, 2019, pp. 3973–3983.

[61] Jacob Devlin et al. “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, Volume 1 (Long and Short Papers)*. 2019, pp. 4171–4186.

[62] Maria-Esther Vidal et al. “Transforming Heterogeneous Data into Knowledge for Personalized Treatments A Use Case”. In: *Datenbank-Spektrum* (), pp. 1–12. URL: `https://doi.org/10.1007/s13222-019-00312-z`.

[63] Ronald Fagin. “Horn clauses and database dependencies”. In: *Journal of the ACM (JACM)* 29.4 (1982), pp. 952–985.

[64] Samaneh Jozashoori et al. “EABlock: a declarative entity alignment block for knowledge graph creation pipelines”. In: *Proceedings of the 37th ACM/SIGAPP Symposium On Applied Computing*. 2022, pp. 1908–1916.

[65] David Owen and Jonathan Davidson. “Hubris syndrome: An acquired personality disorder? A study of US Presidents and UK Prime Ministers over the last 100 years”. In: *Brain* 132.5 (2009), pp. 1396–1406.

[66] George Tourlakis. *Mathematical logic*. John Wiley & Sons, 2011.

[67] Douglas Brent West et al. *Introduction to graph theory*. Vol. 2. Prentice hall Upper Saddle River, 2001.

[68] Ahmad Sakor et al. “Old is Gold: Linguistic Driven Approach for Entity and Relation Linking of Short Text”. In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019,Volume 1 (Long Papers)*. Association for Computational Linguistics, 2019, pp. 2336–2346.

[69] Ahmad Sakor et al. “Falcon 2.0: An entity and relation linking tool over wikidata”. In: *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*. 2020, pp. 3141–3148.

[70] Ahmad Sakor et al. “Knowledge4COVID-19: A semantic-based approach for constructing a COVID-19 related knowledge graph from various sources and analysing treatments’ toxicities”. In: *Journal of Web Semantics* (2022), p. 100760.

[71] Valentina Janev et al. “Responsible Knowledge Management in Energy Data Ecosystems”. In: *Energies* 15.11 (2022), p. 3973.

[72] Dušan Popadić et al. “Towards a Solution for an Energy Knowledge Graph”. In: *Proceedings of ISIC 2022*.

[73] Maria-Esther Vidal et al. “Transforming heterogeneous data into knowledge for personalized treatments—A use case”. In: *Datenbank-Spektrum* 19.2 (2019), pp. 95–106.

[74] Maria-Esther Vidal, Samaneh Jozashoori, and Ahmad Sakor. “Semantic data integration techniques for transforming big biomedical data into actionable knowledge”. In: *2019 IEEE 32nd International Symposium on Computer-Based Medical Systems (CBMS)*. IEEE. 2019, pp. 563–566.

[75] Tim Berners-Lee, James Hendler, and Ora Lassila. “The semantic web”. In: *Scientific american* 284.5 (2001), pp. 34–43.

[76] Ali Khalili. *A Semantics-based User Interface Model for Content Annotation, Authoring and Exploration*. Jan. 2015. DOI: `10.13140/RG.2.1.1951.5600`.

[77] Sören Auer et al. “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*. 2007.

[78] Ion Androutsopoulos, Graeme D Ritchie, and Peter Thanisch. “Natural language interfaces to databases–an introduction”. In: *Natural language engineering* 1.1 (1995), pp. 29–81.

[79] F. M. Suchanek, G. Kasneci, and G. Weikum. “Yago: a core of semantic knowledge”. In: *Proc. of the 16th Int. Conf. on World Wide Web*. 2007, pp. 697–706. DOI: `10.1145/1242572.1242667`. URL: `http://doi.acm.org/10.1145/1242572.1242667`.

[80] George A. Miller. “WordNet: A Lexical Database for English”. In: *Commun. ACM* 38.11 (1995), pp. 39–41. DOI: `10.1145/219717.219748`. URL: `http://doi.acm.org/10.1145/219717.219748`.

[81] Marc Wick. *Geonames Ontology*. 2015. URL: `http://www.geonames.org/about.html` (visited on 04/22/2015).

[82] Marcin Synak, Maciej Dabrowski, and Sebastian Ryszard Kruk. “Semantic Web and Ontologies”. In: *Semantic Digital Libraries* (2009), pp. 41–54. DOI: `10.1007/978-3-540-85434-0_3`.

[83] Eric Prud’hommeaux and Andy Seaborne. *SPARQL Query Language for RDF*. 2005. URL: `http://www.w3.org/TR/rdf-sparql-query/`.

[84] John W. Lloyd and Rodney W. Topor. “A basis for deductive database systems”. In: *The Journal of Logic Programming* 2.2 (1985), pp. 93–109.

[85] Shan Shan Huang, Todd Jeffrey Green, and Boon Thau Loo. “Datalog and emerging applications: an interactive tutorial”. In: *Proceedings of the 2011 ACM SIGMOD International Conference on Management of data*. 2011, pp. 1213–1216.

[86] Rajiv Bagai and Rajshekhar Sunderraman. “A paraconsistent relational data model”. In: *International Journal of Computer Mathematics* 55.1-2 (1995), pp. 39–55.

[87] Prakash M Nadkarni, Lucila Ohno-Machado, and Wendy W Chapman. “Natural language processing: an introduction”. In: *Journal of the American Medical Informatics Association* 18.5 (2011), pp. 544–551.

[88] Sunita Sarawagi et al. “Information extraction”. In: *Foundations and Trends® in Databases* 1.3 (2008), pp. 261–377.

[89] Zhu Zhang et al. “Joint model of entity recognition and relation extraction based on artificial neural network”. In: *Journal of Ambient Intelligence and Humanized Computing* (2020), pp. 1–9.

[90] Wikimedia Commons. *File:Entity Linking - Example of pipeline.svg — Wikimedia Commons, the free media repository.* [Online; accessed 11-November-2022]. 2022. URL: `https://commons.wikimedia.org/w/index.php?title=File:Entity_Linking_-_Example_of_pipeline.svg&oldid=628985143`.

[91] Tom M Mitchell and Tom M Mitchell. *Machine learning.* Vol. 1. 9. McGraw-hill New York, 1997.

[92] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep learning.* MIT press, 2016.

[93] James A Anderson. *An introduction to neural networks.* MIT press, 1995.

[94] Imran Khan and Arun Kulkarni. “Knowledge extraction from survey data using neural networks”. In: *Procedia Computer Science* 20 (2013), pp. 433–438.

[95] Vladimiro Miranda and Adriana Rosa Garcez Castro. “Improving the IEC table for transformer failure diagnosis with knowledge extraction from neural networks”. In: *IEEE transactions on power delivery* 20.4 (2005), pp. 2509–2516.

[96] Jaemin Yoo et al. “Knowledge extraction with no observable data”. In: *Advances in Neural Information Processing Systems* 32 (2019).

[97] *Basic Perceptron Neural Network.* 2022 (last accessed: November 2022). URL: `https://www.allaboutcircuits.com/technical-articles/how-to-train-a-basic-perceptron-neural-network/`.

[98] Kai Han et al. “Transformer in transformer”. In: *Advances in Neural Information Processing Systems* 34 (2021), pp. 15908–15919.

[99] Xiou Ge et al. “CORE: A knowledge graph entity type prediction method via complex space regression and embedding”. In: *Pattern Recognition Letters* 157 (2022), pp. 97–103.

[100] Kayo Yin. “Sign language translation with transformers”. In: *arXiv preprint arXiv:2004.00588* 2 (2020).

[101] F Fdez-Riverola and Juan M Corchado. “Forecasting red tides using an hybrid neurosymbolic system”. In: *AI Communications* 16.4 (2003), pp. 221–233.

[102] Pascal Hitzler et al. “Neuro-symbolic approaches in artificial intelligence”. In: *National Science Review* 9.6 (2022), nwac035.

[103] Jonathan Waring, Charlotta Lindvall, and Renato Umeton. “Automated machine learning: Review of the state-of-the-art and opportunities for healthcare”. In: *Artificial intelligence in medicine* 104 (2020), p. 101822.

[104] Edward A Silver et al. “A tutorial on heuristic methods”. In: *European Journal of Operational Research* 5.3 (1980), pp. 153–162.

[105] Sharon A Curtis. “The classification of greedy algorithms”. In: *Science of Computer Programming* 49.1-3 (2003), pp. 125–157.

[106] Tu Bao Ho. “Knowledge discovery”. In: *Knowledge Science* (2016), pp. 70–93.

[107] David Jensen. “Knowledge Discovery from Graphs”. In: *International Symposium on Graph Drawing.* Springer. 2000, pp. 170–170.

[108] Santo Fortunato. "Community detection in graphs". In: *Physics reports* 486.3-5 (2010), pp. 75–174.

[109] Punam Bedi and Chhavi Sharma. "Community detection in social networks". In: *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* 6.3 (2016), pp. 115–135.

[110] Clara Pizzuti. "Ga-net: A genetic algorithm for community detection in social networks". In: *International conference on parallel problem solving from nature.* Springer. 2008, pp. 1081–1090.

[111] Ling Wu et al. "Deep learning techniques for community detection in social networks". In: *IEEE Access* 8 (2020), pp. 96016–96026.

[112] Jay Bagga. "Old and new generalizations of line graphs". In: *International Journal of Mathematics and Mathematical Sciences* 2004.29 (2004), pp. 1509–1521.

[113] Hao Ma, Irwin King, and Michael R Lyu. "Mining web graphs for recommendations". In: *IEEE Transactions on Knowledge and Data Engineering* 24.6 (2011), pp. 1051–1064.

[114] E Sampathkumar and L Pushpalatha. "Complement of a graph: A generalization". In: *Graphs and Combinatorics* 14.4 (1998), pp. 377–392.

[115] David W Matula, George Marble, and Joel D Isaacson. "Graph coloring algorithms". In: *Graph theory and computing.* Elsevier, 1972, pp. 109–122.

[116] Jenny Rose Finkel, Trond Grenager, and Christopher D. Manning. "Incorporating Non-local Information into Information Extraction Systems by Gibbs Sampling". In: *ACL 2005, 43rd Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference, 25-30 June 2005, University of Michigan, USA.* The Association for Computer Linguistics, 2005, pp. 363–370.

[117] TextRazor. *TextRazor Ltd.* 2018. URL: `https://www.textrazor.com/`.

[118] Ronan Collobert et al. "Natural language processing (almost) from scratch". In: *Journal of machine learning research* 12.ARTICLE (2011), pp. 2493–2537.

[119] Pablo N. Mendes et al. "DBpedia spotlight: shedding light on the web of documents". In: *I-SEMANTICS.* 2011.

[120] Matthew Honnibal and Mark Johnson. "An Improved Non-monotonic Transition System for Dependency Parsing". In: *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015.* 2015, pp. 1373–1378. URL: `http://aclweb.org/anthology/D/D15/D15-1162.pdf`.

[121] Mark Neumann et al. "ScispaCy: Fast and Robust Models for Biomedical Natural Language Processing". In: *Proceedings of the 18th BioNLP Workshop and Shared Task.* Florence, Italy: Association for Computational Linguistics, Aug. 2019, pp. 319–327.

[122] Ndapandula Nakashole, Gerhard Weikum, and Fabian M. Suchanek. "PATTY: A Taxonomy of Relational Patterns with Semantic Types". In: *EMNLP-CoNLL.* 2012.

[123] Michael J Cafarella, Michele Banko, and Oren Etzioni. *Open information extraction from the Web.* US Patent 7,877,343. Jan. 2011.

[124] Mike Mintz et al. "Distant supervision for relation extraction without labeled data". In: *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP.* 2009, pp. 1003–1011.

[125] Ricardo Usbeck et al. "AGDISTIS - Graph-Based Disambiguation of Named Entities Using Linked Data". In: *The Semantic Web - ISWC 2014 - 13th International Semantic Web Conference, Riva del Garda, Italy, October 19-23, 2014. Proceedings, Part I.* Springer, 2014, pp. 457–471.

[126] Isaiah Onando Mulang et al. "Encoding Knowledge Graph Entity Aliases in an Attentive Neural Networks for Wikidata Entity Linking". In: *Web Information System and Engineering* (2020).

[127] Nicola De Cao et al. "Autoregressive Entity Retrieval". In: *International Conference on Learning Representations.* 2020.

[128] Tom Ayoola et al. "ReFinED: An Efficient Zero-shot-capable Approach to End-to-End Entity Linking". In: *arXiv preprint arXiv:2207.04108* (2022).

[129] Isaiah Onando Mulang et al. "Evaluating the Impact of Knowledge Graph Context on Entity Disambiguation Models". In: *CIKM* (2020).

[130] Phong Le and Ivan Titov. "Distant Learning for Entity Linking with Automatic Noise Detection". In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics.* 2019, pp. 4081–4090.

[131] Phong Le and Ivan Titov. "Boosting Entity Linking Performance by Leveraging Unlabeled Documents". In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics.* 2019, pp. 1935–1945.

[132] Lajanugen Logeswaran et al. "Zero-Shot Entity Linking by Reading Entity Descriptions". In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics.* 2019, pp. 3449–3460.

[133] Ledell Wu et al. "Scalable Zero-shot Entity Linking with Dense Entity Retrieval". In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP).* 2020, pp. 6397–6407.

[134] Ambiverse. *Ambiverse GmbH.* 2018. URL: `https://ambiverse.com`.

[135] Andrea Moro, Alessandro Raganato, and Roberto Navigli. "Entity Linking meets Word Sense Disambiguation: a Unified Approach". In: *TACL* 2 (2014), pp. 231–244. URL: `https://tacl2013.cs.columbia.edu/ojs/index.php/tacl/article/view/291`.

[136] Paolo Ferragina and Ugo Scaiella. "TAGME: on-the-fly annotation of short text fragments (by wikipedia entities)". In: *Proceedings of the 19th ACM Conference on Information and Knowledge Management, CIKM 2010, Toronto, Ontario, Canada, October 26-30, 2010.* 2010, pp. 1625–1628. DOI: `10.1145/1871437.1871689`. URL: `http://doi.acm.org/10.1145/1871437.1871689`.

[137] Johannes Hoffart et al. "Robust Disambiguation of Named Entities in Text". In: *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing, EMNLP 2011, 27-31 July 2011, John McIntyre Conference Centre, Edinburgh, UK, A meeting of SIGDAT, a Special Interest Group of the ACL.* 2011, pp. 782–792. URL: `http://www.aclweb.org/anthology/D11-1072`.

[138] Kuldeep Singh et al. "Why Reinvent the Wheel: Let's Build Question Answering Systems Together". In: *Proceedings of the 2018 World Wide Web Conference, WWW 2018, Lyon, France, April 23-27, 2018.* ACM, 2018, pp. 1247–1256. DOI: `10.1145/3178876.3186023`. URL: `http://doi.acm.org/10.1145/3178876.3186023`.

[139] Antonin Delpeuch. "OpenTapioca: Lightweight Entity Linking for Wikidata". In: *The 1st Wikidata Workshop co-located with International Semantic Web Conference 2020 (to appear)* (2020).

[140] Yi Yang and Ming-Wei Chang. "S-MART: Novel Tree-based Structured Learning Algorithms Applied to Tweet Entity Linking". In: *ACL- IJCNLP (Volume 1: Long Papers).* 2015, pp. 504–513.

[141] Daniil Sorokin and Iryna Gurevych. "Mixing Context Granularities for Improved Entity Linking on Question Answering Data across Entity Categories". In: *Proceedings of the Seventh Joint Conference on Lexical and Computational Semantics.* 2018, pp. 65–75.

[142] Leon Derczynski et al. "Analysis of named entity recognition and linking for tweets". In: *Inf. Process. Manage.* 51.2 (2015), pp. 32–49.

[143] Hang Jiang et al. "LNN-EL: A Neuro-Symbolic Approach to Short-text Entity Linking". In: *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers).* 2021, pp. 775–787.

[144] Hai Wang and Hoifung Poon. "Deep Probabilistic Logic: A Unifying Framework for Indirect Supervision". In: *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing.* 2018, pp. 1891–1902.

[145] Kexin Yi et al. "Neural-symbolic vqa: Disentangling reasoning from vision and language understanding". In: *Advances in neural information processing systems* 31 (2018).

[146] Jing Zhang et al. "Neural, symbolic and neural-symbolic reasoning on knowledge graphs". In: *AI Open* 2 (2021), pp. 14–35.

[147] Nut Limsopatham and Nigel Collier. "Normalising Medical Concepts in Social Media Texts by Learning Semantic Representation". In: *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers).* 2016, pp. 1014–1023.

[148] Rajarshi Bhowmik, Karl Stratos, and Gerard de Melo. "Fast and effective biomedical entity linking using a dual encoder". In: *arXiv preprint arXiv:2103.05028* (2021).

[149] Zheng Yuan et al. "CODER: Knowledge-infused cross-lingual medical term embedding for term normalization". In: *Journal of biomedical informatics* 126 (2022), p. 103983.

[150] Hongyi Yuan, Zheng Yuan, and Sheng Yu. "Generative Biomedical Entity Linking via Knowledge Base-Guided Pre-training and Synonyms-Aware Fine-tuning". In: *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL 2022, Seattle, WA, United States, July 10-15, 2022.* Association for Computational Linguistics, 2022, pp. 4038–4048.

[151] Rico Angell et al. "Clustering-based Inference for Biomedical Entity Linking". In: *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies.* 2021, pp. 2598–2608.

[152] Maya Varma et al. "Cross-Domain Data Integration for Named Entity Disambiguation in Biomedical Text". In: *Findings of the Association for Computational Linguistics: EMNLP 2021.* 2021, pp. 4566–4575.

[153] Sheng Zhang et al. "Knowledge-rich self-supervised entity linking". In: *arXiv preprint arXiv:2112.07887* (2021).

[154] Mujeen Sung et al. “Biomedical Entity Representations with Synonym Marginalization”. In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*. Association for Computational Linguistics, 2020, pp. 3641–3650.

[155] Dhruv Agarwal et al. “Entity Linking via Explicit Mention-Mention Coreference Modeling”. In: *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. 2022, pp. 4644–4658.

[156] Ledell Wu et al. “Zero-shot entity linking with dense entity retrieval”. In: *arXiv preprint arXiv:1911.03814* (2019).

[157] Kuldeep Singh et al. “Frankenstein: A Platform Enabling Reuse of Question Answering Components.” In: *ESWC*. Ed. by Aldo Gangemi et al. Vol. 10843. Lecture Notes in Computer Science. Springer, 2018, pp. 624–638. ISBN: 978-3-319-93417-4. URL: `http://dblp.uni-trier.de/db/conf/esws/eswc2018.html#SinghBRS18`.

[158] Ricardo Usbeck et al. “AGDISTIS - Agnostic Disambiguation of Named Entities Using Linked Open Data”. In: *European Conference on Artificial Intelligence*. 2014, p. 2. URL: `http://svn.aksw.org/papers/2014/ECAI_AGDISTIS/ECAI_short_accepted/public.pdf`.

[159] Christina Unger et al. “Template-based question answering over RDF data”. In: *Proceedings of the 21st World Wide Web Conference 2012, WWW 2012, Lyon, France, April 16-20, 2012*. ACM, 2012, pp. 639–648. DOI: `10.1145/2187836.2187923`.

[160] Kuldeep Singh et al. “Qanary - The Fast Track to Creating a Question Answering System with Linked Data Technology”. In: *The Semantic Web - ESWC 2016 Satellite Events, Heraklion, Crete, Greece, May 29 - June 2, 2016, Revised Selected Papers*. 2016, pp. 183–188. DOI: `10.1007/978-3-319-47602-5_36`.

[161] Andreas Both et al. “Qanary - A Methodology for Vocabulary-Driven Open Question Answering Systems”. In: *The Semantic Web. Latest Advances and New Domains - 13th International Conference, ESWC 2016, Heraklion, Crete, Greece, May 29 - June 2, 2016, Proceedings*. Springer, 2016, pp. 625–641.

[162] Andreas Both et al. “Rapid Engineering of QA Systems Using the Light-Weight Qanary Architecture”. In: *ICWE 2017*. 2017.

[163] Shikhar Vashishth et al. “Improving broad-coverage medical entity linking with semantic type prediction and large-scale datasets”. In: *Journal of biomedical informatics* 121 (2021), p. 103880.

[164] Shuang Chen et al. “Improving entity linking by modeling latent entity type information”. In: *Proceedings of the AAAI conference on artificial intelligence*. Vol. 34. 05. 2020, pp. 7529–7537.

[165] Xuhui Sui et al. “Improving Zero-Shot Entity Linking Candidate Generation with Ultra-Fine Entity Type Information”. In: *Proceedings of the 29th International Conference on Computational Linguistics*. 2022, pp. 2429–2437.

[166] Russa Biswas et al. “Entity Type Prediction Leveraging Graph Walks and Entity Descriptions”. In: *International Semantic Web Conference*. Springer. 2022, pp. 392–410.

[167] Gilles Vandewiele et al. “pyRDF2Vec: Python Implementation and Extension of RDF2Vec”. In: IDLab. 2020. URL: `https://github.com/IBCNServices/pyRDF2Vec`.

[168] Changsung Moon, Paul Jones, and Nagiza F Samatova. “Learning entity type embeddings for knowledge graph completion”. In: *Proceedings of the 2017 ACM on conference on information and knowledge management.* 2017, pp. 2215–2218.

[169] Daniel Cer et al. “Universal Sentence Encoder for English”. In: *Empirical Methods in Natural Language Processing: System Demonstrations.* 2018, pp. 169–174.

[170] Zhouhan Lin et al. “A structured self-attentive sentence embedding”. In: *arXiv preprint arXiv:1703.03130* (2017).

[171] Dat Quoc Nguyen, Thanh Vu, and Anh Tuan Nguyen. “BERTweet: A pre-trained language model for English Tweets”. In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations.* 2020, pp. 9–14.

[172] Martin Müller, Marcel Salathé, and Per E Kummervold. “COVID-Twitter-BERT: A Natural Language Processing Model to Analyse COVID-19 Content on Twitter”. In: *arXiv preprint arXiv:2005.07503* (2020).

[173] Lars Döhling and Ulf Leser. “EquatorNLP: Pattern-based Information Extraction for Disaster Response”. In: *CEUR Workshop Proceedings* 798 (Jan. 2011).

[174] Gabriel Stanovsky, Daniel Gruhl, and Pablo N. Mendes. “Recognizing Mentions of Adverse Drug Reaction in Social Media Using Knowledge-Infused Recurrent Models.” In: *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers.* ACL, 2017, pp. 142–151.

[175] Mark Stevenson and Mark A. Greenwood. “Learning Information Extraction Patterns using WordNet”. In: *Proceedings of the 5th Intl. Conf. on Language Resources and Evaluations, LREC 2006 22 - 28 May 2006.* 2006, pp. 95–102.

[176] Amal Zouaq, Dragan Gasevic, and Marek Hatala. “Linguistic Patterns for Information Extraction in OntoCmaps.” In: *WOP.* Vol. 929. CEUR Workshop Proceedings. CEUR-WS.org, 2012.

[177] Ganggao Zhu and Carlos Angel Iglesias. “Sematch: Semantic Entity Search from Knowledge Graph.” In: *SumPre-HSWI@ESWC.* Vol. 1556. CEUR Workshop Proceedings. CEUR-WS.org, 2015.

[178] João Paulo Carvalho, Hugo Rosa, and Fernando Batista. “Detecting relevant tweets in very large tweet collections: The London Riots case study”. In: *2017 IEEE International Conference on Fuzzy Systems, FUZZ-IEEE 2017, Naples, Italy, July 9-12, 2017.* IEEE, 2017, pp. 1–6. DOI: `10.1109/FUZZ-IEEE.2017.8015635`. URL: `https://doi.org/10.1109/FUZZ-IEEE.2017.8015635`.

[179] Tomoya Noro and Takehiro Tokuda. “Searching for Relevant Tweets Based on Topic-Related User Activities”. In: *J. Web Eng.* 15.3–4 (July 2016), pp. 249–276. ISSN: 1540-9589.

[180] Adam Lerer et al. “PyTorch-BigGraph: A Large-scale Graph Embedding System”. In: *arXiv preprint arXiv:1903.12287* (2019).

[181] Petar Ristoski et al. “RDF2Vec: RDF graph embeddings and their applications”. In: *Semantic Web* 10.4 (2019), pp. 721–752.

[182] George Karypis and Vipin Kumar. “METIS: A software package for partitioning unstructured graphs, partitioning meshes, and computing fill-reducing orderings of sparse matrices”. In: (1997).

[183] Guillermo Palma, Maria-Esther Vidal, and Louiqa Raschid. “Drug-Target Interaction Prediction Using Semantic Similarity and Edge Partitioning”. In: *The Semantic Web – ISWC 2014*. 2014, pp. 131–146.

[184] Mohammad Taghi Hajiaghayi and Tom Leighton. “On the max-flow min-cut ratio for directed multicommodity flows”. In: *Theor. Comput. Sci.* 352.1-3 (2006), pp. 318–321. DOI: 10.1016/j.tcs.2005.10.037.

[185] Daniel A. Spielman and Shang-Hua Teng. “Spectral Partitioning Works: Planar Graphs and Finite Element Meshes”. In: *37th Annual Symposium on Foundations of Computer Science, FOCS ’96, Burlington, Vermont, USA, 14-16 October, 1996*. IEEE Computer Society, 1996, pp. 96–105. DOI: 10.1109/SFCS.1996.548468.

[186] Reid Andersen, Fan R. K. Chung, and Kevin J. Lang. “Local Graph Partitioning using PageRank Vectors”. In: *47th Annual IEEE Symposium on Foundations of Computer Science (FOCS 2006), 21-24 October 2006, Berkeley, California, USA, Proceedings*. IEEE Computer Society, 2006, pp. 475–486. DOI: 10.1109/FOCS.2006.44.

[187] Jure Leskovec, Kevin J. Lang, and Michael W. Mahoney. “Empirical comparison of algorithms for network community detection”. In: *Proceedings of the 19th International Conference on World Wide Web, WWW 2010, Raleigh, North Carolina, USA, April 26-30, 2010*. Ed. by Michael Rappa et al. ACM, 2010, pp. 631–640. DOI: 10.1145/1772690.1772755. URL: https://doi.org/10.1145/1772690.1772755.

[188] Ignacio Traverso Ribón et al. “Considering Semantics on the Discovery of Relations in Knowledge Graphs”. In: *Knowledge Engineering and Knowledge Management - 20th International Conference, EKAW 2016, Bologna, Italy, November 19-23, 2016, Proceedings*. 2016, pp. 666–680. URL: https://doi.org/10.1007/978-3-319-49004-5%5C_43.

[189] Ariam Rivas et al. “Unveiling Relations in the Industry 4.0 Standards Landscape Based on Knowledge Graph Embeddings”. In: *Database and Expert Systems Applications - 31st International Conference, DEXA 2020, Bratislava, Slovakia, September 14-17, 2020, Proceedings, Part II*. 2020, pp. 179–194. URL: https://doi.org/10.1007/978-3-030-59051-2%5C_12.

[190] Sahar Vahdati et al. “Unveiling Scholarly Communities over Knowledge Graphs”. In: *Digital Libraries for Open Knowledge, 22nd International Conference on Theory and Practice of Digital Libraries, TPDL 2018, Porto, Portugal, September 10-13, 2018, Proceedings*. 2018, pp. 103–115. URL: https://doi.org/10.1007/978-3-030-00066-0%5C_9.

[191] Daniel Brélaz. “New Methods to Color the Vertices of a Graph”. In: *Association for Computing Machinery* 22.4 (Apr. 1979), pp. 251–256. ISSN: 0001-0782. DOI: 10.1145/359094.359101. URL: https://doi.org/10.1145/359094.359101.

[192] Dominic JA Welsh and Martin B Powell. “An upper bound for the chromatic number of a graph and its application to timetabling problems”. In: *The Computer Journal* 10.1 (1967), pp. 85–86.

[193] Piotr Formanowicz and Krzysztof Tanaś. “A survey of graph coloring-its types, methods and applications”. In: *Foundations of Computing and Decision Sciences* 37.3 (2012), p. 223.

[194] Frank Thomson Leighton. “A graph coloring algorithm for large scheduling problems”. In: *Journal of research of the national bureau of standards* 84.6 (1979), pp. 489–506.

[195] Andreas Gamst. “Some lower bounds for a class of frequency assignment problems”. In: *IEEE transactions on vehicular technology* 35.1 (1986), pp. 8–14.

[196] T-K Woo, Stanley YW Su, and Richard Newman-Wolfe. "Resource allocation in a dynamically partitionable bus network using a graph coloring algorithm". In: *IEEE Transactions on Communications* 39.12 (1991), pp. 1794–1801.

[197] Aris Anagnostopoulos et al. "Spectral Relaxations and Fair Densest Subgraphs". In: *Proceedings of the 29th ACM International Conference on Information & Knowledge Management.* 2020, pp. 35–44.

[198] Abhijin Adiga et al. "Inferring Probabilistic Contagion Models Over Networks Using Active Queries". In: *Proceedings of the 27th ACM International Conference on Information and Knowledge Management.* 2018, pp. 377–386.

[199] Chayant Tantipathananandh, Tanya Berger-Wolf, and David Kempe. "A framework for community identification in dynamic social networks". In: *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining.* 2007, pp. 717–726.

[200] Enrico Malaguti, Michele Monaci, and Paolo Toth. "An exact approach for the vertex coloring problem". In: *Discrete Optimization* 8.2 (2011), pp. 174–190.

[201] Rhyd Lewis. *A guide to graph colouring.* Vol. 7. Springer, 2015.

[202] Silviu Cucerzan. "Large-Scale Named Entity Disambiguation Based on Wikipedia Data". In: *EMNLP-CoNLL 2007, Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, June 28-30, 2007, Prague, Czech Republic.* 2007, pp. 708–716. URL: http://www.aclweb.org/anthology/D07-1074.

[203] Krisztian Balog. "Entity Linking". In: *Entity-Oriented Search.* Cham: Springer International Publishing, 2018, pp. 147–188. ISBN: 978-3-319-93935-3. DOI: 10.1007/978-3-319-93935-3_5. URL: https://doi.org/10.1007/978-3-319-93935-3_5.

[204] Wei Shen, Jianyong Wang, and Jiawei Han. "Entity Linking with a Knowledge Base: Issues, Techniques, and Solutions". In: *IEEE Trans. Knowl. Data Eng.* 27.2 (2015), pp. 443–460. DOI: 10.1109/TKDE.2014.2327028. URL: https://doi.org/10.1109/TKDE.2014.2327028.

[205] Johannes Hoffart, Yasemin Altun, and Gerhard Weikum. "Discovering Emerging Entities with Ambiguous Names". In: *Proceedings of the 23rd International Conference on World Wide Web.* WWW '14. Seoul, Korea: ACM, 2014, pp. 385–396. ISBN: 978-1-4503-2744-2. DOI: 10.1145/2566486.2568003. URL: http://doi.acm.org/10.1145/2566486.2568003.

[206] Ahmad Sakor, Kuldeep Singh, and Maria-Esther Vidal. "FALCON: an entity and relation linking framework over dbpedia". In: *CEUR Workshop Proceedings 2456 (2019).* Vol. 2456. Aachen: RWTH. 2019, pp. 265–268.

[207] Ozge Sevgili et al. *Neural Entity Linking: A Survey of Models based on Deep Learning.* 2020. arXiv: 2006.00575 [cs.CL].

[208] Sidharth Mudgal et al. "Deep Learning for Entity Matching: A Design Space Exploration". In: *Proceedings of the 2018 International Conference on Management of Data, SIGMOD Conference 2018, Houston, TX, USA, June 10-15, 2018.* 2018, pp. 19–34.

[209] Paolo Ferragina and Ugo Scaiella. "Fast and Accurate Annotation of Short Texts with Wikipedia Pages". In: *IEEE Software* 29.1 (2012), pp. 70–75.

[210] Spazio Dati. *Dandelion Ltd.* 2018. URL: https://dandelion.eu/.

[211] Kuldeep Singh et al. “No One is Perfect: Analysing the Performance of Question Answering Components over the DBpedia Knowledge Graph”. In: *CoRR* abs/1809.10044 (2018). arXiv: 1809.10044. URL: http://arxiv.org/abs/1809.10044.

[212] Laurie Bauer and Bauer Laurie. *English word-formation.* Cambridge university press, 1983.

[213] Edwin Williams. “On the notions" Lexically related" and" Head of a word"”. In: *Linguistic inquiry* 12.2 (1981), pp. 245–274.

[214] Saeedeh Shekarpour et al. “RQUERY: Rewriting Natural Language Queries on Knowledge Graphs to Alleviate the Vocabulary Mismatch Problem”. In: *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA.* AAAI Press, 2017, pp. 3936–3943. URL: http://aaai.org/ocs/index.php/AAAI/AAAI17/paper/view/14638.

[215] José Esquivel et al. “On the Long-Tail Entities in News”. In: *Advances in Information Retrieval - 39th European Conference on IR Research, ECIR 2017, Aberdeen, UK, April 8-13, 2017, Proceedings.* 2017, pp. 691–697. DOI: 10.1007/978-3-319-56608-5_67. URL: https://doi.org/10.1007/978-3-319-56608-5%5C_67.

[216] Priyansh Trivedi et al. “LC-QuAD: A Corpus for Complex Question Answering over Knowledge Graphs”. In: *The Semantic Web - ISWC 2017 - 16th International Semantic Web Conference, Vienna, Austria, October 21-25, 2017, Proceedings, Part II.* Springer, 2017, pp. 210–218. DOI: 10.1007/978-3-319-68204-4_22.

[217] Christina Unger et al. “Question Answering over Linked Data (QALD-5)”. In: *Working Notes of CLEF 2015 - Conference and Labs of the Evaluation forum, Toulouse, France, September 8-11, 2015.* CEUR-WS.org, 2015. URL: http://ceur-ws.org/Vol-1391/173-CR.pdf.

[218] OED. *Oxford English Dictionary.* Second. Oxford University Press, 1989. URL: http://www.oed.com/.

[219] Christopher J. Fox. “A Stop List for General Text”. In: *SIGIR Forum* 24.1-2 (1990), pp. 19–35. DOI: 10.1145/378881.378888. URL: https://doi.org/10.1145/378881.378888.

[220] Saeedeh Shekarpour, Axel-Cyrille Ngonga Ngomo, and Sören Auer. “Question answering on interlinked data”. In: *22nd International World Wide Web Conference, WWW ’13, Rio de Janeiro, Brazil, May 13-17, 2013.* Ed. by Daniel Schwabe et al. International World Wide Web Conferences Steering Committee / ACM, 2013.

[221] Eric Brill, Susan T. Dumais, and Michele Banko. “An Analysis of the AskMSR Question-Answering System”. In: *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing, EMNLP 2002, Philadelphia, PA, USA, July 6-7, 2002.* 2002. URL: https://aclanthology.info/papers/W02-1033/w02-1033.

[222] Clinton Gormley and Zachary Tong. *Elasticsearch: The Definitive Guide: A Distributed Real-Time Search and Analytics Engine.* " O’Reilly Media, Inc.", 2015.

[223] Zhiheng Huang, Marcus Thint, and Asli Çelikyilmaz. “Investigation of Question Classifier in Question Answering”. In: *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing, EMNLP 2009, 6-7 August 2009, Singapore, A meeting of SIGDAT, a Special Interest Group of the ACL.* 2009.

[224] Ricardo Usbeck et al. "GERBIL: General Entity Annotator Benchmarking Framework". In: *Proceedings of the 24th International Conference on World Wide Web, WWW 2015, Florence, Italy, May 18-22, 2015.* 2015.

[225] Ricardo Usbeck et al. "7th Open Challenge on Question Answering over Linked Data (QALD-7)". In: *Semantic Web Challenges - 4th SemWebEval Challenge at ESWC 2017, Portoroz, Slovenia, May 28 - June 1, 2017, Revised Selected Papers.* 2017, pp. 59–69.

[226] Antoine Bordes et al. "Large-scale Simple Question Answering with Memory Networks". In: *CoRR* abs/1506.02075 (2015). arXiv: 1506.02075. URL: http://arxiv.org/abs/1506.02075.

[227] Jörg Waitelonis and Harald Sack. "Named Entity Linking in #Tweets with KEA". In: *Proceedings of the 6th Workshop on 'Making Sense of Microposts' co-located with the 25th International World Wide Web Conference (WWW 2016), Montréal, Canada, April 11, 2016.* 2016, pp. 61–63.

[228] René Speck and Axel-Cyrille Ngonga Ngomo. "Ensemble Learning for Named Entity Recognition". In: *The Semantic Web - ISWC 2014 - 13th International Semantic Web Conference, Riva del Garda, Italy, October 19-23, 2014. Proceedings, Part I.* 2014, pp. 519–534.

[229] Dennis Diefenbach et al. "Question answering benchmarks for wikidata". In: 2017.

[230] Mohnish Dubey et al. "Lc-quad 2.0: A large dataset for complex question answering over wikidata and dbpedia". In: *International Semantic Web Conference.* Springer. 2019, pp. 69–78.

[231] Xiyuan Yang et al. "Learning Dynamic Context Augmentation for Global Entity Linking". In: *EMNLP-IJCNLP 2019.* Ed. by Kentaro Inui et al. 2019, pp. 271–281.

[232] Stefan Heindorf et al. "Vandalism detection in wikidata". In: *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management.* 2016, pp. 327–336.

[233] Ridho Reinanda, Edgar Meij, and Maarten de Rijke. "Document Filtering for Long-tail Entities". In: *Proceedings of the 25th ACM International Conference on Information and Knowledge Management, CIKM 2016, Indianapolis, IN, USA, October 24-28, 2016.* ACM, 2016, pp. 771–780. DOI: 10.1145/2983323.2983728. URL: http://doi.acm.org/10.1145/2983323.2983728.

[234] Zi Yang et al. "Building optimal information systems automatically: configuration space exploration for biomedical information systems". In: *22nd ACM International Conference on Information and Knowledge Management, CIKM'13, San Francisco, CA, USA, October 27 - November 1, 2013.* ACM, 2013, pp. 1421–1430.

[235] Ikuya Yamada et al. "Joint Learning of the Embedding of Words and Entities for Named Entity Disambiguation". In: *CoNLL 2016.* Ed. by Yoav Goldberg and Stefan Riezler. ACL, 2016, pp. 250–259.

[236] Xinbo Zhang and Lei Zou. "IMPROVE-QA: An Interactive Mechanism for RDF Question/Answering Systems". In: *Proceedings of the 2018 International Conference on Management of Data, SIGMOD Conference 2018, Houston, TX, USA, June 10-15, 2018.* 2018, pp. 1753–1756. DOI: 10.1145/3183713.3193555. URL: http://doi.acm.org/10.1145/3183713.3193555.

[237] Yixin Cao et al. “Neural Collective Entity Linking”. In: *Proceedings of the 27th International Conference on Computational Linguistics.* 2018, pp. 675–686.

[238] Stefano Ceri, Georg Gottlob, and Letizia Tanca. “What you Always Wanted to Know About Datalog (And Never Dared to Ask)”. In: *IEEE Trans. Knowl. Data Eng.* 1.1 (1989), pp. 146–166.

[239] Olivier Bodenreider. “The unified medical language system (UMLS): integrating biomedical terminology”. In: *Nucleic acids research* 32.suppl_1 (2004), pp. D267–D270.

[240] Sunil Mohan and Donghui Li. “MedMentions: A Large Biomedical Corpus Annotated with UMLS Concepts”. In: *Automated Knowledge Base Construction (AKBC).* 2018.

[241] Jinhyuk Lee et al. “BioBERT: a pre-trained biomedical language representation model for biomedical text mining”. In: *Bioinformatics* 36.4 (2020), pp. 1234–1240.

[242] Manoj Prabhakar Kannan Ravi et al. “CHOLAN: A Modular Approach for Neural Entity Linking on Wikipedia and Wikidata”. In: *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume.* 2021, pp. 504–514.

[243] Mujeen Sung et al. “Biomedical Entity Representations with Synonym Marginalization”. In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics.* 2020, pp. 3641–3650.

[244] Fangyu Liu et al. “Self-Alignment Pretraining for Biomedical Entity Representations”. In: *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies.* 2021, pp. 4228–4238.

[245] Tahira Naseem et al. “A semantics-aware transformer model of relation linking for knowledge base question answering”. In: *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers).* 2021, pp. 256–262.

[246] Steven Munevar. “Unlocking Big Data for better health”. In: *Nature Biotechnology* (2017), pp. 684–686.

[247] Ray Y. Zhong et al. “Big Data for supply chain management in the service and manufacturing sectors: Challenges, opportunities, and future perspectives”. In: *Computers & Industrial Engineering* 101 (2016), pp. 572–591.

[248] Ahmad Sakor, Kuldeep Singh, and Maria-Esther Vidal. “Resorting to Context-aware Background Knowledge for Unveiling Semantically Related Posts”. In: *IEEE Access* (2022).

[249] Mariano Provencio et all. “Neoadjuvant chemotherapy and nivolumab in resectable non-small-cell lung cancer (NADIM): an open-label, multicentre, single-arm, phase 2 trial”. In: *The Lancet Oncology* (2020). DOI: `https://doi.org/10.1016/S1470-2045(20)30453-8`.

[250] Pablo San Segundo. “A new DSATUR-based algorithm for exact vertex coloring”. In: *Computers & OR* 39.7 (2012), pp. 1724–1733.

[251] R. Janczewski et al. “The smallest hard-to-color graph for algorithm DSATUR”. In: *Discrete Mathematics* 236.1 (2001). Graph Theory, pp. 151–165. ISSN: 0012-365X. DOI: `https://doi.org/10.1016/S0012-365X(00)00439-8`. URL: `https://www.sciencedirect.com/science/article/pii/S0012365X00004398`.

[252] Maria-Esther Vidal et al. “On the Selection of SPARQL Endpoints to Efficiently Execute Federated SPARQL Queries”. In: *Transactions on Large-Scale Data- and Knowledge-Centered Systems XXV*. Ed. by Abdelkader Hameurlain, Josef Küng, and Roland Wagner. Berlin, Heidelberg: Springer Berlin Heidelberg, 2016, pp. 109–149. ISBN: 978-3-662-49534-6. DOI: 10.1007/978-3-662-49534-6_4. URL: https://doi.org/10.1007/978-3-662-49534-6_4.

[253] Erich Peter Klement, Radko Mesiar, and Endre Pap. “Triangular norms: Basic notions and properties”. In: *Logical, Algebraic, Analytic and Probabilistic Aspects of Triangular Norms*. Elsevier, 2005, pp. 17–60.

[254] Andrew Beam et al. “Clinical Concept Embeddings Learned from Massive Sources of Multimodal Medical Data”. In: Jan. 2020, pp. 295–306. ISBN: 978-981-12-1562-9. DOI: 10.1142/9789811215636_0027.

[255] *SNLI*. 2021 (last accessed: july 2021). URL: https://nlp.stanford.edu/projects/snli/.

[256] *MultiSNLI*. 2021 (last accessed: july 2021). URL: https://www.nyu.edu/projects/bowman/multinli/.

[257] TwitterInc. *Twitter Search API*. 2022 -last accessed: April 2022. URL: https://developer.twitter.com/en/docs/twitter-api/v1/tweets/search/api-reference/get-search-tweets.

[258] Yiye Ruan, David Fuhry, and Srinivasan Parthasarathy. “Efficient Community Detection in Large Networks Using Content and Links”. In: *Proceedings of the 22nd International Conference on World Wide Web*. WWW ’13. Rio de Janeiro, Brazil: Association for Computing Machinery, 2013, pp. 1089–1098. ISBN: 9781450320351. DOI: 10.1145/2488388.2488483. URL: https://doi.org/10.1145/2488388.2488483.

[259] Dimitar Dimitrov et al. “TweetsCOV19 - A Knowledge Base of Semantically Annotated Tweets about the COVID-19 Pandemic”. In: *Proceedings of the 29th ACM International Conference on Information and Knowledge Management*. CIKM ’20. Virtual Event, Ireland: Association for Computing Machinery, 2020, pp. 2991–2998. ISBN: 9781450368599. DOI: 10.1145/3340531.3412765. URL: https://doi.org/10.1145/3340531.3412765.

[260] Ruiyun Li et al. “Substantial undocumented infection facilitates the rapid dissemination of novel coronavirus (SARS-CoV-2)”. In: *Science* 368.6490 (2020), pp. 489–493.

[261] Rituparna. *Kaggle Fifa 2018 Tweets*. https://www.kaggle.com/rgupta09/world-cup-2018-tweets. [Online; accessed 19-Oct-2021]. 2018.

[262] Himansu Sekhar Pattanayak, Harsh K Verma, and A. L. Sangal. “Community Detection Metrics and Algorithms in Social Networks”. In: *2018 First International Conference on Secure Cyber Computing and Communication (ICSCCC)*. 2018, pp. 483–489. DOI: 10.1109/ICSCCC.2018.8703215.

[263] Mingming Chen, Tommy Nguyen, and Boleslaw K. Szymanski. “On Measuring the Quality of a Network Community Structure”. In: *2013 International Conference on Social Computing*. 2013, pp. 122–127. DOI: 10.1109/SocialCom.2013.25.

[264] Suman Saha and Satya P. Ghrera. “Network Community Detection on Metric Space”. In: *Algorithms* 8.3 (2015), pp. 680–696. ISSN: 1999-4893. DOI: 10.3390/a8030680. URL: https://www.mdpi.com/1999-4893/8/3/680.

[265] Marco Gaertler. “Clustering”. In: *Network Analysis: Methodological Foundations.* Ed. by Ulrik Brandes and Thomas Erlebach. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 178–215. ISBN: 978-3-540-31955-9. DOI: 10.1007/978-3-540-31955-9_8. URL: https://doi.org/10.1007/978-3-540-31955-9_8.

[266] Mark EJ Newman. “Modularity and community structure in networks”. In: *Proceedings of the national academy of sciences* 103.23 (2006), pp. 8577–8582.

[267] Aydın Buluç et al. “Recent Advances in Graph Partitioning”. In: *Algorithm Engineering: Selected Results and Surveys.* Cham: Springer, 2016, pp. 117–158.

[268] Fotis Aisopos et al. “Knowledge Graphs for Enhancing Transparency in Health Data Ecosystems”. In: ().

[269] Matthew Honnibal and Ines Montani. “spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing”. To appear. 2017.

[270] Arash Einolghozati et al. “Sound Natural: Content Rephrasing in Dialog Systems”. In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP).* 2020, pp. 5101–5108.

Appendix A

List of Publications

This thesis is based on the following publications.

A.1 Peer-Reviewed International Journals

- José Alberto Benítez-Andrades, María Teresa García-Ordás, Mayra Russo, **Ahmad Sakor**, Luis Daniel Fernandes Rotger, Maria-Esther Vidal. *Empowering Machine Learning Models with Contextual Knowledge for Enhancing the Detection of Eating Disorders in Social Media Posts.* In: the Semantic Web journal (2022). ***Full research paper***

- Fotis Aisopos, Samaneh Jozashoori, Emetis Niazmand, Disha Purohit, Ariam Rivas,**Ahmad Sakor**, Enrique Iglesias, Dimitrios Vogiatzis, Ernestina Menasalvas, Alejandro Rodriguez Gonzalez, Guillermo Vigueras, Daniel Gomez-Bravo, Maria Torrente, Roberto Hernández López, Mariano Provencio Pulla, Athanasios Dalianis, Anna Triantafillou, Georgios Paliouras,Maria-Esther Vidal. *Knowledge Graphs for Enhancing Transparency in Health Data Ecosystems.* In: the Semantic Web journal (2022). ***Full research paper***

- **Ahmad Sakor**, Kuldeep Singh, Maria-Esther Vidal. *Resorting to Context-aware Background Knowledge for Unveiling Semantically Related Posts.* In: IEEE Access (2022). ***Full research paper***

- **Ahmad Sakor**, Samaneh Jozashoori, Emetis Niazmand, Ariam Rivas, Kostantinos Bougiatiotis, Fotis Aisopos, Enrique Iglesias, Philipp D Rohde, Trupti Padiya, Anastasia Krithara, Georgios Paliouras, Maria-Esther Vidal. *Knowledge4COVID-19: A semantic-based approach for constructing a COVID-19 related knowledge graph from various sources and analyzing treatments' toxicities.* In: Journal of Web Semantics (2022). ***Full research paper***

- Valentina Janev, Maria-Esther Vidal, Dea Pujić, Dušan Popadić, Enrique Iglesias, **Ahmad Sakor**, Andrej Čampa. *Responsible Knowledge Management in Energy Data Ecosystems.* In: Energies Journal (2022). ***Full research paper***

- Maria-Esther Vidal, Kemele M. Endris, Samaneh Jozashoori, **Ahmad Sakor**, Ariam Rivas. *Transforming Heterogeneous Data into Knowledge for Personalized Treatments—A Use Case.* In: Datenbank-Spektrum Journal (2019). ***Full research paper***

A.2 Papers in Proceedings of Peer-Reviewed Conferences

- Samaneh Jozashoori, **Ahmad Sakor**, Enrique Iglesias, Maria-Esther Vidal. *EABlock: a declarative entity alignment block for knowledge graph creation pipelines.* In: Proceedings of the 37th ACM/SIGAPP Symposium On Applied Computing (2022). ***Full research paper***

- Dušan Popadić, Enrique Iglesias, **Ahmad Sakor**, Valentina Janev, Maria-Esther Vidal. *Towards a Solution for an Energy Knowledge Graph.* In: ISIC 2022 International Semantic Intelligence Conference (2022). **Best paper award**. ***Full research paper***

- Maria-Esther Vidal, Samaneh Jozashoori, **Ahmad Sakor**. *Semantic Data Integration Techniques for Transforming Big Biomedical Data into Actionable Knowledge.* 2019 IEEE 32nd International Symposium on Computer-Based Medical Systems (CBMS). ***Short research paper***

- **Ahmad Sakor**, Kuldeep Singh, Maria-Esther Vidal. *Falcon: An entity and relation linking framework over dbpedia.* In: ISWC2019, Demo track, CEUR Workshop Proceedings (2019). ***Demo paper***

- **Ahmad Sakor**, Kuldeep Singh, Anery Patel, Maria-Esther Vidal. *Falcon 2.0: An Entity and Relation Linking Tool over Wikidata.* In: CIKM '20: Proceedings of the 29th ACM International Conference on Information & Knowledge Management. ***Resource paper***

- **Ahmad Sakor**, Isaiah Onando Mulang', Kuldeep Singh, Saeedeh Shekarpour, Maria-Esther Vidal, Jens Lehmann, Sören Auer. *Old is Gold: Linguistic Driven Approach for Entity and Relation Linking of Short Text.* In: NAACL 2019 Annual Conference of the North American Chapter of the Association for Computational Linguistics (2019). ***Full research paper***

A.3 Under-Review

- **Ahmad Sakor**, Kuldeep Singh, Maria-Esther Vidal. *A Neuro-Symbolic Approach for Light-Weight Biomedical Entity Linking.* In: Anonymous submission. ***Full research paper***

Appendix B

Appendix

B.1 Neuro-symbolic approach components

B.1.1 Background Knowledge

```
label(resourceID,label,language,provider, confidenceScore)
definition(resourceID,label,language,provider)
type(resourceID,type)
sameAs(resourceID1,resourceID2)
sameAs(resourceID1,resourceID2),
label(resourceID1,label,language,provider,confidenceScore)=>
label(resourceID2,label,language,provider,confidenceScore)
```

B.1.2 Catalog of linguistic rules

Rule 1: Stopwords are not entities or relations.

stopWord(P,L): is true if P is a stopword of the language L.

$$S=\{s_1,\ s_2,\ s_3,\ \ldots,s_n\}$$
$$T=T_{Initial}\ -\ S$$

T is the set of all the words s_i in $T_{Initial}$ such that NOT(stopWord(s_i,L))

Rule 2: Verbs are not entities.

Assume the following predicates:

verb(A,L): true; if the set of words in A corresponds to a verb in a language L.
entity(A): indicates if the set of words in A correspond to an entity label.
The following rules apply:
1) Verbs cannot be an entity:

```
entity(A,T):-belong(A,T),NOT(verb(A,L)).
relation(A,T):-belong(A,T),verb(A,L).
```

2) Verbs act as a division point of the sentence in case of two entities and we do not merge tokens from either side of the verb:

```
R={r$_1$, r$_2$, r$_3$,..,r$_n$}
entity(A,T):-belong(A,T),NOT(stopword(A,L))
entity(A,T):-belong(A,T), (NOT(relation(A,T))
 OR capitalized(A,T))
```

where r represents the tokens which has the POS tag verb ⇒ E= T - R.

3) Tokens other than the sentence headword that begin with a capital letter refer to entities: For r in R ⇒ If r is capitalized ⇒ R=R- r and E=E+r.

Rule 3: A single compound word comprises words without stopwords.

$T = (t_1, t_2), (t_3), ..t_n$; Assume the following predicates If t_1 and t_2 do not have any token between them in the original text: hasStopWord(A,B): is true if there is a stopword s_i (stopWord(s_i,L)) between the set of words in A and the set of words in B. hasStopWord(.,.) can be recursively defined (Base Case and the inductive case).

```
singleCompoundWord(C,T):- belong(A,T), belong(B,T),
NOT(hasStopWord(A,B)), CONCAT(A,B,C)
```

Rule 4: Entities with only stopwords between them are one entity.

If s is between e1 & e2 ⇒
$E = \{(e_1, e_2), e_3, .., e_n\}$ then:

```
singleCompoundWord(C,E):-entity(A,E),
entity(B,E),hasStopWord(A,B,E), CONCAT(A,B,C)
topURL(E,[(R,Score)|L1],K):- K1=K-1,
topURL(E,L1,K1), hasURL(E,L),
top((R,Score),L,K).
```

B.1.3 Catalog of domain-specific rules

Rule 5: Equivalence of entities with the same label.

```
confidenceScore(resource, label):-
COUNT(label(resourceID, label, _ , _ ).
```

Rule 6: Entities commonly have definitions and are of a certain type.

```
prefCandidate(text, candidate):-
List(candidates, candidate),hasDefinition(candidate).
```

B.1.4 Rule-based approach

Step 1: Tokenization of all the surface forms in the text, except stop words (**Rule 1**).

$\mathtt{T}_{Initial}$={$\mathtt{t}_1$, $\mathtt{t}_2$, $\mathtt{t}_3$, ...,$\mathtt{t}_n$} where

$T_{Initial}$ is the set of tokens.

Step 2: Apply compounding principle based on **Rule 3** and **Rule 4**

Step 3: Apply **Rule 2**.

Step 4: For each e in E and r in R retrieve the candidates URIs list

```
E={ "e_1":[e_1u_1, e_1u_2, e_1u_3,..,e_1u_n]}
R={ "r_1":[r_1u_1, r_1u_2, r_1u_3,..,r_1u_n]}
relation(R,T):- temporal_entity(R,T),predicateSearchIndex(R)
entity(E,T):- temporal_entity(R,T), NOT(relation(R,T))
```

Step 5: Generate triples by combining E and R and assign a weight of zero for all the triples

```
TR={(e_1,r_1,0), (e_1,r_2,0), .....(e_n,r_1,0),
(e_n,r_2,0),..(e_n, r_n,0)}
hasURL(E,[(R,Score)]):- searchIndex(E,R,Score,B).
hasURL(E,[(R,Score)|L]):- entity(E,T),relation(R,T)
,searchIndex(E,R,Score,B).hasURL(E,L).
hasURL(R,[(E,Score)|L]):- entity(E,T),relation(R,T),
searchIndex(E,R,Score,B),hasURL(E,L).
topURL(E,[(R,Score)],1):- hasURL(E,L),top((R,Score),L,1).
```

Step 6: Increase the weight of each triple found in the KG.

```
KG={(s_1,p_1,o_1), (s_2,p_2,o_2),(s_3,p_3,o_3)
,..(s_n,p_n,o_n)}.
If (e_1,r_1,0) in KG → (e_1,r_1,0+w_i)
where w is a value which represents the weight.
w value varies depending on the triple found
(s,p,o) or (o,p,s) or (s,p,o)(o,p2,x)
```

Step 7: Return the entities and relations in the highest weighted triples as the correct answer.

```
E_output= {Max_w(e_1,[e1_URIs]),...}
R_ouput= {Max_w(r_1,[r1_URIs]),...}
entityAnswered(E,T):- entity(E,T),relation(R,T),
triple(E,R,_),topURL(R,[(E,Score)],1)
relationAnswered(R,T):- entity(E,T),relation(R,T),
triple(E,R,_), topURL(E,[(R,Score)],1)
```

Step 8:

If all the triples weights are equal to zero. Apply N-Gram splitting for the longest combination in E according to Rule 2.

```
If (e_i,r_i,0) in KG → NGramSplit(e_i).
NGramSplit(e_i):- Split(Max(Length(e_i))).
```

B.1.5 Wikidata dataset SPARQL query

The Query is executed on 10.06.2022.

```
SELECT DISTINCT ?concept ?label ?cui
 WHERE
 {
 ?concept wdt:P2892 ?cui.
 ?concept rdfs:label ?label.
 filter(lang(?label) = 'en')
}
```

B.1.6 Implementation details

SciSpacy: "en_core_sci_sm" is the model used in the experiments.
Noreen: "BioBERT-NLI" is the base model used for the semantic type prediction task.

For creating the EMS dataset, five medical doctors analyzed the abstracts and extracted the medical terms. The validation techniques for the quality of dataset is adapted from [270]. Considering the dataset only focuses on the keywords, errors were removed by majority votes. The Wikidata dataset was a simple extraction using a SPARQL query. Hence, there was no issue of finding the gold ground truth as SPARQL have returned the correct results. We used popular metric Accuracy for main experiments, which is a percentage of successes out of the total number of query mentions. For ablations we use standard precision, recall, and f-scores.

Appendix C

Appendix

C.1 Proofs and Propositions

C.1.1 Lemma 4.1

Proof $\Rightarrow$
By contradiction, suppose v is the only vertex in V with the color $\mu(v)$ and $deg_{G(v)}$ is different to $|J|$-1. Consider the vertex u in V, such that u is different to v, and v and u are not adjacent vertices, i.e., (v, u) is not in J. However, this leads to a contraction because $\mu(v)$ is minimal, and v and u should be colored with the same color.
$\Leftarrow$
By Contradiction, suppose that $deg_{G(v)}$ is equal to $|J|$-1 and there is a vertex u in V, such that u is different to v and $\mu(v)$=$\mu(u)$. This leads to a contraction, since v and u are adjacent vertices and cannot be colored with the same color. □

C.1.2 DSATUR Propositions

Proposition 1-DSATUR Optimality [251]. Let $G = (V, E)$ be a graph, a core of CG named $CG' = (V', E')$, is a sub-graph of G, i.e., $V' \subseteq V$ and $E' \subseteq E$, and there is no vertex v in V' such that, $degree(v)$ is 1. DSATUR optimally colors G if the core of CG is as follows: a single vertex; a bipartite graph, i.e., a graph that can be partitioned into two set of vertices such that each vertex in each set is connected to a vertex in the other set; A wheel, i.e., a graph formed by connecting one vertex with all the vertices of a cycle; A complete multipartite graph, i.e., a graph in which vertices are adjacent if and only if they belong to different partitions of the graph; a cactus, i.e., a graph in which any pair of cycles has a vertex in common; and a necklace, i.e., a graph made up of r beads where each bead comprises one cycle of length k which is incident with a path of length

l.

Proposition 2-DSATUR Optimality [251].

Let $G = (V, E)$ be a graph that corresponds to a polygon tree, i.e., (i) G is a cycle (Base Case), or (ii) G comprises two polygon trees G' and G'' that share exactly one edge. DSATUR optimally colors G.

Appendix D

Curriculum Vitae

Personal Details

Name	Ahmad Sakor
Date of Birth	09.04.1992
Place of Birth	Daraa, Syria
Email	sakor@l3s.de
Family status	Single

Education

2019–present	PhD in Informatics, Leibniz Universität Hannover, Hannover, Germany.
2016–2018	MSc in Computer Science, Rheinische Friedrich-Wilhelms-Universität, Bonn, Germany.
2010–2015	BSc in Informatics Engineering, Arab International University, Damascus, Syria.
2007–2010	High school, School of Excellence, Daraa, Syria

Professional Experience

2018–present	Research Assistant at the joint lab of L3S & TIB, Hannover, Germany.
2020	Teaching assistant for Databases course (BSc) at LUH.
2018	Student Assistant at Fraunhofer IAIS, Sankt Augustin, Germany.
2017	Software developer at EOOOM GmbH (Student job), Bonn, Germany.
2017	Teaching assistant for Advanced Logic Programming at the University of Bonn, Bonn, Germany.
2016–2017	Server Admin at the University of Bonn, Bonn, Germany.
2016	for Advanced Logic Programming at the University of Bonn, Bonn, Germany.
2015–2016	Software developer at New Wave Technologies LLC, Damascus, Syria.
2013–2015	Software developer at iConsult (Student job), Damascus, Syria.

Professional Activities

2022	Program committee member of International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 22).
2021	Program committee member of International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 21).
2021	Program committee member of International Workshop on scientific Knowledge: Representation, Discovery, and Assessment (Sci-K 21).
2020	Program committee member of International Conference on Information and Knowledge Management (CIKM 20).
2020	Program committee member of Scientific Knowledge Graphs (SKG 20).

Research Interests

EU projects	iASiS, BigMedilytics, P4-Lucat, CLARIFY, ImProVIT, CoyPu, PLATOON.
Focus areas	Natural Language Processing, Semantic Web, Artificial Intelligence, Information Extraction, Question Answering.

Technical Skills

Proficient	Python, C#, PHP, SPARQL, SQL.
Skilled	C++, HTML5, CSS3, Git, LaTeX, Docker.
Familiar	React, Matlab, OpenGL, R, Swift, Java.

Awards and Grants

2022 Best paper award @ISIC 2022 International Semantic Intelligence Conference.

2020 SIGIR Student Travel Grant from the Special Interest Group on Information Retrieval @CIKM 2020, student travel grant.

2015 Second place in graduation class of the bachelor studies @The Arab International University.

2014 Bronze Medal from the Syrian Computer Society @The Syrian Collegiate Programming Contest “SCPC".

2014 Solid Programmer from the Syrian Computer Society @The Syrian Collegiate Programming Contest “SCPC".

2010 Scholarship for bachelor studies from the Ministry of High Education in Syria @The Arab International University.

Languages

Arabic	Mother tongue
English	Full professional proficiency
German	Intermediate proficiency