Towards Building Knowledge Graphs from Natural Language Text with Open Relation Extraction and Semantic Disambiguation

Dag Vegard Kollstrøm Johnsen

Master's Thesis

Department of Information Science and Media Studies
University of Bergen

March 1, 2019

Abstract

Natural language text, from messages on social media to articles in newspapers, constitutes a significant portion of the content available on the Web. These texts are readable by humans, but cannot easily be used for advanced queries and reasoning by machines. Thus, the automated conversion of natural language text into a formal representation that is machine-readable is an important goal. The extraction of knowledge graphs from text is of particular importance in the context of the Semantic Web and Linked Open Data initiatives.

This thesis describes the exploratory, example-driven development of an approach to knowledge graph extraction from natural language texts through the use of Open Relation Extraction systems, which are capable of extracting facts from texts in the form of relational triples in an efficient, domain-independent manner. The intuition is that these triples can be disambiguated and converted into machine-readable statements. This approach is partially implemented and in turn qualitatively assessed on the text domain of the lead paragraphs of newspaper articles, which express facts about notable entities. Solutions are discussed for many of the problems discovered through the implementation and assessment. The results indicate that Open Relation Extraction shows promise as an underlying technique for knowledge graph extraction from natural language text.

Acknowledgements

I would like to thank Professor Andreas Lothe Opdahl for his thorough supervision. His sharp insight, patience, and positive outlook contributed enormously to this thesis.

Contents

Chapter 1

Introduction

The World Wide Web contains vast amounts of natural language text ranging from user-generated content like social media messages to articles by professionally organized newspapers. This content is readable by humans, but cannot easily be queried or reasoned over by a computer. The original vision of the Semantic Web project is the transformation of this old Web of documents into a Web of data that is machine-readable. Though focus has since shifted from making web pages themselves machine-readable through embedded metadata tags onto other uses of semantic technologies, such as Linked Open Data and knowledge graphs, the early efforts have resulted in many standards and large-scale knowledge bases useful for making sense of natural language. Because of this, the intersection of natural language processing and semantic technologies is today a vibrant research area, in which the automatic extraction of knowledge from natural language text into a formal representation is an important task. One such representation may be RDF knowledge graphs, which use the standard semantics frameworks of the Semantic Web.

This thesis presents an approach to building knowledge graphs from natural language texts through the use of Open Relation Extraction, a paradigm of extraction tools capable of processing text across domains without a training or preparation phase. Open relation extractors detect relations involving entities that are indicative of facts, and output them in the form of "subject; predicate; argument" triples that closely correspond to the triple building

blocks of the Semantic Web. However, these triple segments are textual, consisting of words and phrases that are not connected to any reference like a knowledge base.

There is as such a need for a disambiguation phase where the information held in these textual triples may be identified and linked to background knowledge bases. To solve this problem, the use of semantic annotation tools is proposed. These tools can disambiguate textual fragments such as words and phrases in natural language texts through tasks such as the identification and classification of named entities, and the linking of both named entities and non-entity words and phrases to their entries in knowledge bases. Like Open Relation Extraction systems, these types of annotation tools tend to be largely independent of text domain.

These triples, after being converted from ambiguous text into disambiguated resources that can be dereferenced, will constitute a knowledge graph representation of a textual output. The intuition is then that the relations expressed in the input become assertions that can be used by machines for reasoning. The construction of such knowledge graphs is in and of itself a worthwhile pursuit, with numerous applications such as information retrieval and question answering, event detection and monitoring, and many other tasks that may utilize knowledge graphs automatically extracted from text.

Specifically, I have in this thesis investigated the following two research questions:

1. Is knowledge graph extraction from natural language text attainable through combining semantic annotation and existing Open Relation Extraction systems?

2. Is Open Relation Extraction a promising research direction for knowledge graph extraction from natural language text?

1.1 Research method

In order to investigate and attempt to answer these research questions, I have proposed an approach to knowledge graph extraction using the technologies of interest. I have also developed a system that can be considered a partial implementation of this approach, and identified and discussed many of the challenges encountered while doing so. For some of these challenges I have proposed solutions, while others have proved to be difficult or impossible to solve with the technologies considered as part of this approach.

Both the development of the system and the discussion of challenges has been an exploratory, example-driven process. I have worked with a specific domain of input text: the lead paragraphs ("leads") of newspaper RSS feeds, specifically from Reuters and the BBC, which was chosen for several reasons. Newspapers are an important source of new and changing facts about notable entities, such as famous people and organizations that are likely to have corresponding entries in existing knowledge bases. Also, it is assumed that, while newspapers do have a particular style of prose, the writing style is not so idiomatic that results are not relatively generalizable to other domains of texts. In addition, newspapers and RSS feeds are a domain of interest for the News Angler project at the University of Bergen [1], which aims to assist journalists by harvesting information from big data and social media sources. As such, investigating the performance of the techniques my approach uses on this domain was interesting.

Leads were selected for use as example inputs based on the following criteria:

1. The lead must contain a relation that expresses some fact about an event in the world.
2. The lead must contain a verb phrase with a main verb that cannot also serve as an auxiliary verb, such as "is" or "has".
3. Either the subject or the argument (or both) of the relation must correspond to a named entity, though it need not be notable.

[1] `https://www.uib.no/en/rg/ssis/114224/discovering-unexpected-connections-news`

4. The lead must be an independently coherent piece of text that does not significantly rely upon the title or the rest of the article for context through the use of anaphora such as pronouns.

5. Leads with highly non-standard styles are disregarded.

Criterion 1 was selected to limit leads to the type of texts this approach is applicable to, and to exclude irregular leads such as meta-announcements about the newspaper itself and its readers. Criterion 2 was selected because, as will be discussed, verbs that can also be auxiliary verbs cannot be disambiguated with the tools considered in this thesis. As explained, this thesis is oriented towards extracting knowledge concerning entities, which is why criterion 3 was selected. Since the disambiguation tools this approach relies upon require context, criterion 4 was important to ensure that there is a minimum of context missing from the lead itself. Finally, criterion 5 was necessary as newspaper RSS feeds sometimes contain leads with a very non-standard style, an example being "Two white teenagers. One seaside town. Millions of views ... this is the story of YouTube's most unlikely beef." Leads like this, where they may e.g. contain ungrammatical sentences without some sort of predicate, are excluded.

Based on these criteria, a total of 21 leads were used as input examples for this approach. 16 of these were used during development as test leads and as a way of both estimating the performance of the tools I worked with and to uncover barriers standing in the way of conversion into knowledge graphs. The other 5 were selected near the end of the work process as control leads to attempt to control for any possible unconscious bias in selecting the development sentences.

This exploratory research method was a natural choice because it was important that the approach I proposed was implementable. In addition, evaluating the performance of both open relation extractors and annotation tools on leads was important, and this example-driven work process was a straight-forward way of accomplishing that. Lastly, as I will explain in the following section, I initially believed my approach to be somewhat more novel than it is. As such the initial development started off in a quite experimental manner, and more traditional research methods driven by e.g. iterative prototype development seemed less appropriate to this thesis given that I could

not place the approach in a existing class of systems at the time.

1.2 Topic progression

This thesis is presented somewhat ideally in terms of chronological progression in order to be easier to read, but the scope and ambition has in fact changed throughout the work process. I started with the goal of detecting and disambiguating relations between notable entities in Twitter messages in order to aid emergency situation detection and monitoring. While investigating existing techniques for extracting relations, I first discovered the Slot Filling task, then (plain) Relation Extraction, and finally Open Relation Extraction (ORE). Unfortunately, ORE systems proved to have reduced performance on tweets, but the assessment of Zouaq et al. (2017) where they suggest that the extraction of structured knowledge for the Semantic Web is an exciting prospect for ORE intrigued me.

As such I shifted my focus from tweets and emergencies onto the extraction and disambiguation of relations (in the context of the lead paragraphs of newspaper articles), and started work on the implementation, ending my literature search. At this point, I was thinking more along the lines of "relation extraction and disambiguation" than knowledge graph extraction, in the sense that the textual triples of from ORE systems could be directly converted into RDF triples, resulting in very simple graphs. I soon discovered that several of my assumptions were naive, for example that ORE predicates could be mapped to RDF properties without issues, and that in order to make my system output RDF triples, a more advanced representation was necessary.

As such I conducted a second literature review investigating how natural language had been converted to/represented as RDF graphs in previous work, as I was uncertain about how the solutions I was considering compared to the state-of-the-art. Aftter much reading, I came across Martínez-Rodríguez et al. (2018), who had just presented (their paper was available online 10. July 2018, halfway through my work process; I discovered it a month later) a system based on an approach very similar to my own, and who cite Kumar Dutta (2014), who was working on the disambiguation of ORE extractions

much earlier. This challenged the novelty of my approach, though the similarity of both the tools chosen and several of the solutions proposed also strengthen the validity of my conclusions. Unfortunately, due to the time lost from both initially changing the focus of my thesis and the second literature review, I ended up not being able to complete my implementation because of time constraints. In particular, as will be detailed in chapter 3, the final step of converting annotated relation extractions to a valid RDF knowledge graph was not completed, and is as such instead discussed at the theoretical level of the overarching approach.

Chapter 2

Literature Review

In this chapter the research areas that are most important to the thesis will be presented, and the specific technologies used will be described. Afterwards, adjacent approaches to knowledge graph extraction from natural language will be described.

2.1 The Semantic Web

The Semantic Web is important to this thesis, as it is the source of technology that several of the annotation tools used rely upon. It is also part of the motivation for the thesis, although this knowledge graph extraction from natural language texts is in and of itself a meaningful pursuit with a myriad of applications. The Semantic Web is a project that started with the goal of transforming the "old" World Wide Web of text and hyperlinks into a Semantic Web, where all the data can be processed by machines, or "intelligent agents" (Berners-Lee, 1999). The use of Semantic Web standards allows for many advantages, such as the integration of data across heterogeneous syntaxes and structures, and more intelligent processing (particularly processing that exploits relationships) allowing for improved searching and question answering. Knowledge can be used and re-used through various dictionaries and knowledge bases where information can be interlinked, and

low level data (e.g. sensor data) can be abstracted into higher level symbolic representations to better aid decision making when data volumes can be very large (Sheth and Thirunarayan, 2012, pp. 6-8).

The most important standard of the Semantic Web is the Resource Description Framework (RDF), a data model based around subject-property-object resource triples. Each resource takes the form of an IRI that can be accessed for a description. In this way things (from concrete entities to abstract concepts) can be described, and even interlinked in these RDF triple statements. A collection of these statements are often visualized (and described as) as a (directed) (knowledge) graph, where subjects and objects are represented as nodes and properties as edges connecting them. Many notable RDF knowledge graphs are interlinked, forming the Linked Open Data (LOD) cloud consisting of many billions of triples. Since the use of IRIs can take up a lot of space, RDF graphs are typically serialized with namespaces and identifiers, where the namespace refers to a Web repository of resources, and the identifier locates the specific resource (Allemang and Hendler, 2008). For example, the resource `http://dbpedia.org/page/Dog` could be substituted by `dbr:Dog`.

Only a small vocabulary of pre-defined classes are considered part of RDF itself, most importantly the `rdf:type` property which expresses that the subject is an instance of the object. The RDF vocabulary is extended by RDF Schema (RDFS), and the Web Ontology Language (OWL). RDFS provides more inference capabilities through a vocabulary that allows for, amongst other features, the creation of hierarchies of classes and properties (through properties like e.g. `rdfs:subClassOf`). OWL is available in several subsets with increasing levels of expressiveness, but can generally be said to provide more powerful reasoning through new constraints that can be placed between classes and properties in an ontology. One example is equivalence, which can be asserted between classes with the `owl:equivalentClass` property, and between individuals with the `owl:sameAs` property (Allemang and Hendler, 2008).

2.2 Information Extraction Tasks

Though "Information Extraction" (IE) is a commonly used term, it is in defined in two different ways in various papers. Niklaus et al. (2018) for example state that IE is to extract "unstructured information expressed in natural language text into ... relational tuples consisting of a set of arguments and a phrase denoting a semantic relation between them". This is a narrow definition that would largely be synonymous with Relation Extraction (RE), which is defined as "the task of recognizing the assertion of a particular relationship between two or more entities in text" by Banko and Etzioni (2008). Other papers, such as Daiber et al. (2013), (implicitly) consider IE to a broader term than RE, which in addition encompasses other tasks such as named entity recognition (NER) that do not involve (textual) relations. If we take a broad view of what IE is, such as the automatic extraction of any sort of structured information from unstructured text, we may even include related tasks such as entity linking (EL), and even word sense disambiguation (WSD). The NER, EL, and WSD tasks have in common that they annotate words or phrases with information that can be said to disambiguate input text in some manner, which is why they are grouped together as "annotation tools" in this thesis.

NER is the task of recognizing named entities in texts, and classifying them into some pre-defined class schema. As will be discussed, a "named entity" is a somewhat nebulous concept, but for now the definition "a NE is a phrase that uniquely refers to an object by its proper name, acronym, nick-name or abbreviation" (Simon, 2013) will suffice. A wide variety of tools offer NER (and EL tools tend to rely upon some form of underlying NER), but for this project the only tool used for NER annotation was the Stanford NER classifier, which is part of the Stanford CoreNLP toolkit (described in Manning et al. (2014)). Like many other types of IE systems, Stanford NER relies upon a form of machine learning classification, specifically Conditional Random Field sequence models (described in Rose Finkel et al. (2005)).

EL (also known as named entity disambiguation, among other terms) also involves the recognition of named entities, but instead of classification they are linked to some entry in a knowledge base. Two popular entity linking tools are DBpedia Spotlight (Daiber et al., 2013) and Babelfy (Moro et al.,

2014). DBpedia Spotlight takes a textual input, and annotates words and phrases that are mentions of resources present in DBpedia (described in Bizer et al. (2009), a large RDF knowledge base created by extracting information from Wikipedia, that is a prominent part of the LOD cloud). DBpedia Spotlight operates with a two-step procedure, involving first spotting mentions using various language-dependent means, including a NER model and Part of Speech (POS) tagging, and then disambiguation using a generative probabilistic model.

Babely on the other hand is a unified method to perform both EL, and WSD ("...assigning meanings to single-word and multi-word occurrences within text"). Moro et al. (2014) consider the difference between EL and WSD to be that the former task uses encyclopedic knowledge bases while the latter uses lexical knowledge bases, and that EL unlike WSD may annotate partial mentions (e.g. a first name mention being linked to an entity with a longer name). They use a graph-based disambiguation strategy with two main steps. They start by using a random walks with restart to create candidate semantic graphs that connect textual mentions (both entities and non-entities) by exploiting connections in BabelNet (described in Navigli and Ponzetto (2012)), a multilingual semantic network available in RDF that integrates both lexical and encyclopedic knowledge bases, most notably Wikipedia and WordNet (a prominent lexical database described in Miller (1995)). After a graph of connected candidate meanings is obtained, they use a densest subgraph heuristic algorithm to determine the most likely candidates.

2.3 Open Relation Extraction

Traditional relation extraction requires supervision in the form of handcrafted extraction patterns or training data specific to a finite amount of relations. Open Relation Extraction (ORE) (or Open Information Extraction (OpenIE) [1]) was introduced by Banko, Cafarella, et al. (2007) along with TextRunner, the first system of the paradigm. OREs aim to extract relations

[1]The terms ORE and OpenIE are often used synonymously, although relation extraction is strictly speaking a subtask of information extraction

from the Web, where "Corpora are massive and heterogeneous, the relations of interest are unanticipated, and their number can be large" (Banko, Cafarella, et al., 2007). The output of an ORE is typically triple extractions consisting of a subject, predicate, and an argument. There may be multiple arguments if the ORE can perform n-ary RE, although binary RE has been most studied (Zouaq et al., 2017).

In a recent survey, Niklaus et al. (2018) divide the approaches to ORE into learning-, rule-, and clause-based systems. Examples of learning-based OREs include TextRunner (Banko, Cafarella, et al., 2007), which uses a self-supervised Naive Bayes classifier that is trained on data labeled by heuristic constraints over POS tagged sentences, and Ollie (Mausam et al., 2012), which uses a training set from a predecessor ORE, ReVerb, to learn extraction patterns over dependencies. ReVerb uses shallow syntactic patterns to extract relations, and is an example of a rule-based ORE.

Clause-based OREs include Stanford OpenIE, ClausIE, and MinIE. ClausIE (Corro and Gemulla, 2013) uses dependency parsing in order detect clauses in an input text, where clauses are found for combinations of subject and verb dependencies, as well as relative pronouns, and appositions and possessives (implicit "is" and "has" relations). The clause type (Subject-Verb-Object-Complement etc.) is determined by exploiting certain grammatical rules of the English language, in what can be envisioned as an if/then decision tree). Though ClausIE was found to have state-of-the-art performance in both their own evaluation and in an assessment by (Zouaq et al., 2017), it has been criticized for giving overly specific extractions (Niklaus et al., 2018).

That was the motivation for MinIE (Gashteovski et al., 2017), which uses ClausIE for relation extraction but minimizes (i.e. removes words from) the extractions through a rule-based strategy. In particular, words are shifted from the argument segment into the relation, and words are dropped from extractions through rule sets based on dependencies, as well as POS and NER tags, with several modes of varying aggression being available. In addition, MinIE annotates extractions with attributes like polarity (truth value), attribution, and modality by detecting specific words and phrases from a domain-independent (but English) lexicon.

Stanford OpenIE (Angeli et al., 2015) extracts relations by recursively

traversing dependency trees. A distantly supervised multinomial logistic regression classifier is used to determine which edges (dependencies) yield independent clauses. These clauses are then shortened into smaller extractions using natural logic to infer where words can be removed without damaging or changing the meaning of the clause.

Several articles, including Gashteovski et al. (2017) and Angeli et al. (2015), note that smaller extractions are more useful for downstream semantic applications. In their assessment of OREs, Zouaq et al. (2017) stress that while the ORE field has seen significant efforts since its inception, it cannot be considered a mature research area in part because OREs have not been used successfully in different application scenarios (in other words, downstream applications), and that the textual extractions produced by OREs are by themselves of limited use for machine reasoning. They note however that binary OREs produce triple extractions while the Semantic Web operates on RDF triples, as well as that ORE extractions require some sort of disambiguation or generalization to become useful. They therefore propose that a promising application of OREs may be to extract triple facts for the growth of the Semantic Web, while the Semantic Web may symbiotically provide knowledge bases for the disambiguation of extractions. This specific observation was the inspiration for this thesis.

2.4 Knowledge Graph Extraction

Knowledge graph (KG) extraction is a wide term that has been used to describe many different tasks. KG extraction from natural language is sometimes equated with the tasks of machine-reading and natural language understanding (e.g. by Gangemi et al. (2017)), in which case we can say that some form of KG extraction or building has been pursued since the early days of AI research[2]. Two important distinctions are ontology learning versus population, and input domain. Ontology learning is the task of extracting ontological knowledge (i.e. classes and properties, or T-box knowledge), while ontology population extracts facts about individuals (A-box knowledge). While some approaches to KG extraction perform both tasks in uni-

[2] `https://en.wikipedia.org/wiki/Conceptual_dependency_theory`

son, it is assertions about individuals that are the subject of interest in this thesis. Input domain can also vary immensely, as KG extraction can be used to describe extraction from structured or semi-structured data sources like medical records and wikis rather than natural language. Even within natural language, domain can be very important, since e.g. instruction manuals or biomedical articles may differ significantly from e.g. newspaper articles or social media messages.

Due to how wide this topic is, only the most closely related approaches will be considered in this review. These approaches can be divided into two types: methods based on OREs, and methods based on Frame Semantics, which is an influential linguistic theory that relates verbs/relations to "semantic frames", which are events involving a variable number of arguments with different roles (Fillmore (1976), according to Martínez-Rodríguez et al. (2018)). The task of assigning these roles to words/phrases in sentences is called Semantic Role Labeling (SRL). FrameNet[3] is a lexical database that formalizes many such semantic frames. PropBank[4] is a project with many similarities, though unlike FrameNet it is only oriented towards verbs, and it has a more lightweight semantics.

OREs have not been extensively used for KG extraction previously, although REs have been used in similar but smaller scope approaches to disambiguating/generalizing relations (examples include Verburg et al. (2015) and Schutz and Buitelaar (2005), both working with very narrow domains with few relation types). Kumar Dutta (2014) and Dutta et al. (2015) presented an approach to disambiguating relations extracted by ReVerb, and NELL[5], a "never-ending language learner" that is generally not considered an ORE due to operating with a smaller set of relations. In contrast to the approach of this thesis, they do not use annotation services to disambiguate extraction segments, but rather create hypothetical `sameAs` links between subject/argument segments and DBpedia resources based on, to my understanding, the intuition that Wikipedia articles which contain the segment in their titles are more likely to correspond to the segment the more outgoing links the article has. They then filter hypotheses by using a Markov Logic Network with a knowledge base that has both the uncertain hypotheses, and

[3] `https://framenet.icsi.berkeley.edu/fndrupal/`

[4] `http://verbs.colorado.edu/~mpalmer/index.html`

[5] `URL`

certain axioms extracted from DBpedia. In order to disambiguate predicate segments, they test several different workflows based around using Markov Clustering to group similar predicates together and comparing them to DBpedia (ontology) properties based on similarity measures, primarily whether the predicates and properties occur between the same entities. Unlike this thesis, Kumar Dutta (2014) and Dutta et al. (2015) work with datasets of previously extracted facts, which is a more narrow domain.

While my thesis was ongoing, Martínez-Rodríguez et al. (2018) presented a system based on an approach very similar to the one presented in this thesis, both in terms of the specific tools used and the form of knowledge graph produced. Their system uses ClausIE for relation extraction, and an ensemble of entity linking tools (DBpedia Spotlight, Babelfy, and TagMe[6]) for annotation. The subjects and arguments of extractions are connected to annotations based on POS noun phrase tags. Their approach can be considered a hybrid of ORE- and semantic frame-based KG extraction, because they use SRL to determine whether the subjects and arguments of extractions correspond to the Agent or Patient roles, as well as to disambiguate (verbal) relations by linking them to their verb sense in PropBank. In their RDF KG output, they use an n-ary relation representation where the (verbal) relation is represented as an instance (individual) of the verb sense resource it is mapped to, that is connected to the subject and argument using Agent and Patient properties. Since subjects and arguments may be compound and correspond to multiple resources (an example they use is "cancer patient"), they use automatically generated resources to represent them (e.g. `cvst:cancer_patient`, and connect them to the resource(s) they are annotated with using local `:isPartOf` properties.

There have been several KG extraction from natural language systems developed within the paradigms of Frame Semantics, or Discourse Representation Theory (a framework that can represent natural language in Discourse Representation Structures (DRSs), a First Order Logic-like form), or both. A state-of-the-art example is FRED (Gangemi et al., 2017), which has been used for several downstream applications. FRED uses the tool Boxer to produce DRSs, which are then labeled with semantic frames and roles. A comprehensive "heuristic-based triplification" is then run to convert the DRSs

[6] `https://tagme.d4science.org/tagme/`

to a RDF/OWL representation, where (verbal) relations are represented as subject resources that are instances of a frame class (i.e. an n-ary representation). The graph is then enriched with `owl:sameAs` connections to named entity and word sense resources identified by TagMe and UKB[7]. Other systems include PIKES[8], which use SRL but not DRSs to extract KGs that use a more strongly Frame Semantics-oriented semantics than the previously mentioned systems, and LODifier[9] which extracts KGs with a representation that closely corresponds to DRSs, but which does not use SRL.

It is also worth mentioning that various forms of both relation extraction and KG extraction have been pursued using Artifical Neural Networks and Word Embeddings, a recent example being Sorokin and Gurevych (2017). While this is a vibrant paradigm that is related to the research area of this thesis, it is generally not included under the broad banner of "Open Information Extraction", and the type of knowledge produced does not, to my knowledge, tend to correspond closely to the type of KGs produced by e.g. FRED. As such, this type of approach has not been considered in this thesis.

2.5 On terminology in this thesis

There are several terms for relational "triples" used in this thesis that closely correspond to each other but differ subtly and can cause confusion. I have tried to be consistent by using "subject", "property", and "object" when discussing RDF triples; "subject", "predicate", and "argument" when discussing ORE output directly (adopting ClausIE terms); and "subject", "verb", and "object" when discussing syntactic entities in language. Since the "relations" extracted by OREs are in fact triples and not just the predicate segment, I call these output triples "extractions", and limit the use of "relation" to the more general notion of the term (i.e. the meaning expressed rather than surface mentions themselves). I refer to parts (i.e. the subject, argument or predicate) of extractions as "segments".

[7] `http://ixa2.si.ehu.es/ukb/`
[8] `http://pikes.fbk.eu/`
[9] `http://www.aifb.kit.edu/web/LODifier/en`

"Annotation tools/services" is used in this thesis as a useful shorthand for tasks such as NER, EL, and WSD as they all associate some fragment of text with a resource or type. This is however not an established term in the literature. Also, though ORE stands for "Open Relation Extraction" rather than "Open Relation Extractor", I also use "ORE" (with a preceding article) and "OREs" to refer to the family of systems.

Lastly, some terminology related to English linguistic phenomena, particularly grammatical categories such as tense, could not be avoided in this thesis as it deals directly with natural language. I have generally not used academic references for this, as the terms are for the most part common knowledge and not from e.g. more advanced linguistic theories.

Chapter 3

Grounding Binary Relations

In this chapter the proposed approach towards grounding binary relations will be presented. Afterwards, the system developed to explore the approach will be detailed: first developmental choices of technology and limitations will be explained, and then the final system will be described in terms of its modules and output.

3.1 Proposed Approach

The origin of the approach proposed in this thesis came from the intuition that a. there is available a class of systems, Open Relation Extractors, that can extract textual triples from natural language text, b. there exists a variety of tools that can link words (not limited to named entities) to knowledge bases, and c. the Semantic Web operates with RDF triples. I therefore wanted to investigate whether or not combining these classes of tools could allow for the creation of what we might call grounded extractions or knowledge graphs: the output of an ORE substituted by its Semantic Web IRI equivalents.

The approach for knowledge graph extraction I propose can be divided into a step-based procedure:

1. Take a natural language text input.
2. Create textual triples using the Open Relation Extraction paradigm.
3. Annotate the input with Semantic Web resources using a collection of tools performing named entity recognition, entity linking, and word sense disambiguation.
4. Map the annotations to the textual triples.
5. Convert the textual triples into Semantic Web RDF triples.

Step 1: In this step, the natural language text input should not be limited to any specific domain of text, because this would run contrary to a goal that is broadly shared both by the ORE and the Semantic Web projects; that of domain independence. This is in fact made an explicit characteristic of the entire OpenIE paradigm in the seminal paper by Banko, Cafarella, et al. (2007), who coined the term. The Semantic Web project on the other hand was originally envisioned as an extension of the World Wide Web where *all* the data is machine-readable (Berners-Lee, 1999), and semantic interoperability and the integration of heterogeneous data are important elements of the project. Furthermore, the capacity to process unstructured text across domains is in and of itself more desirable than domain-dependent extraction.

Step 2: This step will vary in terms of accurate performance as well as the quality of extractions (the size of segments, how much information is retained etc.) depending on the system used for ORE, and also whether the ORE extracts binary or n-ary relations. For this thesis, the used ORE was limited to binary relation extraction. The processing of unary predicates (i.e. Subject-Verb sentence patterns, e.g. "He died" or "Marvin slept") and n-ary relations (where $n > 2$) (e.g. "The new treatment, gave, some of the patients, better sleep") is therefore beyond the scope of this approach as presented here. This delimitation was made because the RDF data model does not with its common usage (i.e. modeling nouns as individuals or classes, and predicates as properties) support n-ary relations [1], and because the ORE paradigm has

[1] It is however possible to represent n-ary relations in RDF by modeling predicates as individuals rather than properties

thus far mainly focused on binary relation extraction (Zouaq et al., 2017; Niklaus et al., 2018). To be clear, this delimitation to binary relations only concerns the ORE part of the approach, not the conversion into RDF. In the case of unary predicates these are simply disregarded entirely, while binary OREs tend to process sentences that can be interpreted as ternary relations as one or more binary relations (e.g. "The new treatment, gave, some of the patients better sleep"). In many cases this type of extraction tends to have overly long predicates or arguments that lead to vague (i.e. the segment does not correspond to one "thing" and requires further decomposition) or overly specific meanings (i.e. the segment is so specific that it is unlikely to correspond to anything in an existing knowledge base) which complicate the grounding process however, as will be discussed in more detail in section 4.1.

Step 3: This step involves a notable necessity of this approach to knowledge graph extraction in that the disambiguation of words should not simply be performed on the textual triples themselves. This is because, as explained in the previous chapter, these techniques rely on as much context as possible in order to disambiguate between different possible meanings of a given word or phrase. Since the textual triples produced by ORE systems are often smaller parts of larger texts (for example a dependent clause of a sentence, or a sentence with one or more prepositional phrases missing), the triples typically bear less context than their source input. Therefore it is this input text that has to be linked to semantic resources, and then the subject, predicate and argument of the textual triple must be aligned with the annotated words in the source text.

Another, albeit minor, issue with step 3 is that while some of the available tools, such as DBpedia Spotlight and Babelfy, operate with Semantic Web resources, others, such as Stanford NER or IBM Watson Natural Language Understanding (formerly AlchemyAPI), do not. Given that the goal of this approach is triples that are grounded through the use of RDF resources, there is then the question of whether or not the non-RDF semantic types can be mapped to their Semantic Web equivalents. Assuming these types are not particularly specific this should be an easy task, as semantic interoperability is one of the main principles of the Semantic Web. For example Stanford NER's Person class and the resource given by `http://xmlns.com/foaf/spec/#term_Person` might be aligned as long as the class descriptions match. In

the case of `http://xmlns.com/foaf/spec/#term_Person`, it is described as "people . . . alive, dead, real or imaginary", while the Stanford NER Person class has, to my knowledge, no available description, but is juxtaposed with the Organization and Location classes and can as such probably safely be assumed to refer to the same type of "person" (as opposed to e.g. a "legal person" that may be an organization).

Step 4: This step can roughly be divided into two separate issues: the purely programmatic issue of mapping the annotated text to the textual triples extracted by the ORE, and the broader problems stemming from the fact that segments of the textual triples may consist of multiple words in a phrase, where potentially only a subset (possibly none) of these will be linked to a Semantic Web resource. The first of these issues seems trivial at first glance, however the fact that OREs typically produce partially duplicate extractions (effectively multiple interpretations of a singular relation occurring in the input text), produces some complexity in terms of how the issue should be handled. This is especially the case if fringe cases of homonyms are taken into consideration, as will be discussed in some detail near the end of section 3.5. The latter issue is heavily intertwined with step 5, the conversion from textual extractions into RDF triples, as it greatly complicates that process.

Step 5: Because this step was, as explained in the introduction (section 1.2), not fully implemented, and because it requires more discussion than the previous steps, it is considered in the following section.

3.2 RDF conversion challenges

Finding a way of representing natural language (the extractions produced by OREs often retain enough structure from the input text that they are by themselves valid sentences) in a machine-readable format such as RDF is a major part of the challenge of this approach, and indeed knowledge graph extraction from natural language text in general. This conversion from textual triples into RDF triples is by no means trivial, and nor is the handling of this step obvious from the intuition underlying the approach

either. Only a brief description of the problem and the proposed solution will be presented in this section, as the many sub-problems and possible solutions will be discussed in more detail in section 4.1 in the following chapter. In particular, it will here be assumed that the subjects and arguments of an extraction are limited to noun phrases, and that the entirety of the phrases is annotated. The complications introduced by the various forms of clauses that can also occur as subjects and arguments, as well as partial annotation is left for later discussion.

With a very simple clause or sentence of the form "entity1 predicate entity2" (for example, "James Cameron directed 'Titanic'"), it is trivial to represent it through RDF, for example as `dbr:James-Cameron schema:director dbr:Titanic-(1997-film) .` Although it is not uncommon for relations in sentences to have one or more segments that can be directly substituted by an URI in this way (indeed Dutta et al. (2015) appear to work exclusively with relations where all segments fulfill this criteria), segments that consist of phrases that do not correspond to any single URI are so plentiful that determining how they should be represented in RDF must be addressed. An example could be the noun phrase "Kremlin critic Alexei Navalny", which we can imagine as the subject or argument of an extraction. Clearly, this is a segment that corresponds to three different IRIs: `dbr:Alexei_Navalny`, `dbr:Government_Of_Russia`, and `dbr:Critic`.

If we were choosing between one of these IRIs as a representation of the noun phrase in a grounded relation, the obvious choice would be the head of the phrase, `dbr:Alexei_Navalny`, and with this example, where the head is a proper noun, it would be an accurate representation, even though context (i.e. the fact that Alexei Navalny is a Kremlin critic) would be lost. On the other hand, many relations have noun phrases where the head is a common noun as subjects or arguments, for example the noun phrase "A contentious piece of literature", where the head "piece" by itself bears little of the meaning carried by the entire noun phrase. Depending on the annotation tool , "piece of literature" instead of only "literature" might be mapped to e.g. `dbr:Literature` or `dbr:Book` (this is not the case with DBpedia Spotlight). In any case it is clear that the meaning inherent in relation segments consisting of multiple words can generally not be represented by a single IRI.

This problem, that the subject, predicate, and argument segments of an extraction may be compound phrases that cannot be mapped to just a single IRI, requires two main adaptations from the common usage way of representing relations in RDF in a single triple. Firstly, as the predicate (i.e. the verb phrase) may be compound (e.g. "has been charged"), it cannot be represented with a single IRI, which necessitates an n-ary relation representation where the predicate is represented as an individual (RDF node) rather than a property (RDF edge). This allows the head of the verb phrase (e.g "charged") to be described with one IRI, and the rest of the phrase to be described with further IRIs connected to this "head node" through properties.

Secondly, compound phrases in subjects and arguments (that do not correspond to a single IRI, like e.g. "The United States of America" should) also require several connected IRIs in order for the full meaning of the phrase to be described. Ideally, the head of the phrase should be a "main node" which is attached to nodes representing e.g. adjectives and determiners through properties. For example, the phrase "California bar shooting" might receive an RDF representation like (using a placeholder namespace and resources):

```
local:California_bar_shooting1 rdf:type ex:shooting(crime);
                               ex:location local:California_bar1 .
local:California_bar1 rdf:type ex:Bar;
                      ex:location ex:California .
```

Unfortunately, such a representation is not possible with the tools used in this approach, as the level of analysis required to determine that the non-head parts of a phrase indicate location (as opposed to any other possible description, e.g. size, origin or purpose) is not present. Instead of such a representation with context-specific properties like `ex:location`, all that can be determined is that the phrase is "associated" with various IRIs. This gives us a representation like:

```
local:California_bar_shooting1 rdf:type ex:shooting(crime);
                               loc:associatedWith ex:California,
                                                  ex:Bar .
```

Note that unlike in the ideal example, it cannot be assumed that "Califor-

nia bar" is a connected (sub-)phrase as this may not be the case generally (it is not in e.g. "tragic bar shooting"), at least without a grammatical analysis such as constituency- or dependency-based parsing.

3.3 Ideal solution

For the sake of illustration, the step-based approach will here be applied to an example sentence. As will be discussed later, the implementation I developed is incomplete and there are also limitations to the underlying tools available. Therefore, this example will be done manually (as opposed to programmatically), using example RDF resources rather than those returned by an annotation API. In other words, this is an ideal knowledge graph extraction with this approach that does not take into consideration the performance-related limitations of current technology (such as partial or erroneous annotation). The following sentence can be imagined as the input:

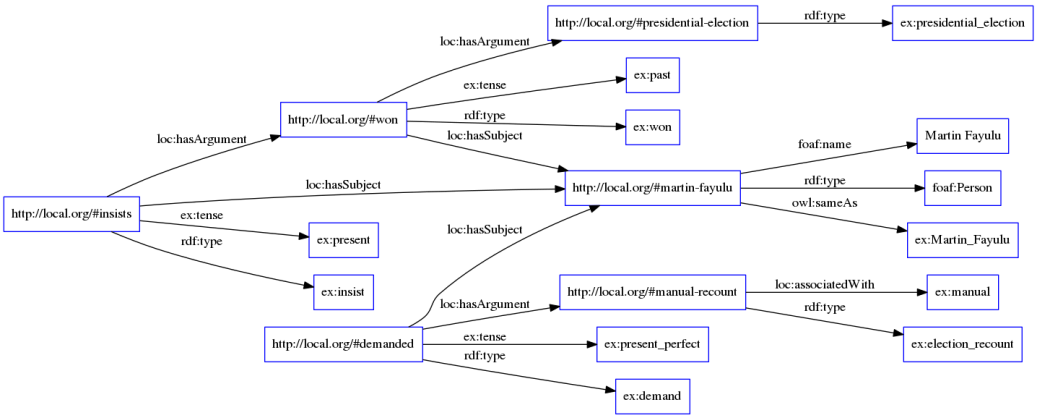
> Martin Fayulu insists he won the presidential election and has demanded a manual recount.

This sentence has begins with a simple noun followed by a verb phrase and then an independently valid clause, with an omitted "that" conjunction. In terms of relation extraction there are not many different ways the sentence can be interpreted correctly, although either "Martin Fayulu" or "he" (as a pronoun referring to the same person) could be the subject of the "has demanded" relation. Let us assume that the following relations[2] are extracted in step 2):

> He; won; the presidential election (1)
> Martin Fayulu; insists; [1]
> Martin Fayulu; has demanded; a manual recount

[2]Note that I use a ID-based notation to indicate a sort of reification here, see discussion in 4.2

Figure 3.1: Manual RDF graph



Step 3 will then be the annotation of the input sentence independently of the ORE extractions. It is the case currently that annotation tools, whether they perform named entity disambiguation, named entity recognition, or word sense disambiguation, are rarely able to annotate the entirety of sentences, even when combined. In particular certain classes of words such as determiners, particles, and auxiliary verbs are often missed or beyond the scope of these tools, but for this example it will be assumed that the entire sentence is annotated.

Step 4, the alignment of the annotated input text and the extractions, is a non-issue in this idealized example where there are no duplicate extractions and the entire input is annotated. Step 5 will then involve converting the annotations and extractions into an RDF graph with considerations given in the previous section. For the first extraction the RDF graph will then be the following graph shown in figure 3.1

This graph illustrates how the aforementioned n-ary representation might look. As can be seen, the relations are represented as nodes (insists, won, and demanded) that are connected to their subjects and arguments via `hasSubject` and `hasArgument` properties imagined as part of a local domain ontology (with the `loc` prefix) as opposed to the example annotation resources (like `ex:election_recount` or `ex:Martin_Fayulu`) from an external knowledge base. The graph also displays how the common case where one relation has another as its subject or argument can be represented. In

the input sentence, "Martin Fayulu 'insists' (that) he 'won' the presidential election", and in the output graph, the "insists" relation has the "won" relation as its argument. This is discussed in more detail in the following discussion chapter, as is the question of whether or not representing tense in this manner is appropriate.

3.4 Implementation choices

The system I developed for this project was intended to investigate the possibility of extending the ORE paradigm by linking each part of a relation to Semantic Web resources. True success or "completion" in this task would be represented by a tool that could create knowledge graphs from natural text. This was a lofty goal for a master's thesis, given that automatic conversion from natural language text to computer-readable logical form has been pursued within the fields of information extraction and semantic parsing for years, and yet this task remains open. Because I believed that this challenge meant that true completion was unlikely, I decided from the start that the system would be an exploratory "programmatic experiment", as opposed to an artifact developed within a research methodology such as design science, and intended for evaluation by end-users.

This choice shaped the development of the system in multiple ways, such as there being no need to develop with users in mind, which for example meant that creating a GUI or Web API was superfluous, and there was little need to handle rare IO exceptions gracefully. Most importantly however, it shaped the "philosophy" of development. Rather than starting with a low-fidelity prototype and a set of formal requirements to fulfill, I set out to explore how far towards automated knowledge graph extraction combining ORE with annotation tools would get me, and what potential challenges and limitations such an approach would have.

As the project's programming language I opted to use Java. I knew that several of the ORE systems I had become familiar with through the literature review were written in Java, and that Java APIs were available for several of the annotation tools I had in mind. It was also the language I was most familiar with.

3.4.1 ORE choices

Choosing a method of relation extraction was an important early step towards the goal of grounded relations. Initially I conducted a brief, informal comparison of three popular open relation extractors; Ollie, Stanford OpenIE, and ClausIE, in order to see which tool was best suited for the project. I processed 7 lead paragraphs (several consisting of a single sentence) taken from Reuters' RSS News feeds with each of these relation extractors, and compared the output given by each. Precision and recall were not the only criteria, partly because as pointed out by Zouaq et al. (2017), the lack of gold standards and a consensus among researchers of open relation extraction about what constitutes a well-formed relation means these values vary for any given ORE depending on who is performing the evaluation. More important, with the goal of grounding relations, was the form of the output (i.e. extractions) and how easily it could be converted into a more formal representation.

The resulting extractions have many issues, and can be be viewed in full in supplemental files. An example of the extractions given from the following lead is shown in table 3.1: "The diminutive actor who starred in the Austin Powers movies' as "Mini Me", Verne Troyer, died Saturday at a hospital in Los Angeles. He was 49." Taking all 7 leads into account, Stanford OpenIE has the largest amount of extractions, but many of them are erroneous or partially duplicate. The form is also not very close to the natural language of the input, with verbs being converted into present tense, and the determiner "her" being converted into the pronoun "she". These are major problems given that my approach for mapping annotations and extractions together relies upon connecting them to the input text. One advantage of the Stanford OpenIE extractions is that they for the most part have quite minimalistic subjects, relations, and arguments in the extractions, however as the aforementioned disadvantages could not be removed through changing settings (to my knowledge), Stanford OpenIE was excluded as an option.

The Ollie extractions have for the most part a quite good form with relatively short extractions, however there was an erroneous extraction for 6 of the 7 test sentences. By comparison ClausIE has far more extractions overall although many of them are partially duplicate, and most of the extractions

Table 3.1: ORE comparison example

Subject	Predicate	Object	ORE
Mini Me”	died Saturday at	a hospital	Ollie
Saturday	be died at	a hospital	Ollie
The diminutive actor	starred in	the Austin Powers movies	Ollie
Mini Me”	died	Saturday	Ollie
He	was	49	Ollie
Mini Me”	died Saturday in	a hospital	Ollie
Austin Powers movie	die at	hospital in Los Angeles	Stanford OIE
hospital	be in	Los Angeles	Stanford OIE
Austin Powers movie	die at	hospital	Stanford OIE
Austin Powers movie	die at_time	Saturday	Stanford OIE
He	be	49	Stanford OIE
The diminutive actor	starred	in the Austin Powers movies as Mini Me	ClausIE
The diminutive actor	starred	in the Austin Powers movies	ClausIE
Mini Me	is	Verne Troyer	ClausIE
The diminutive actor	died	Saturday	ClausIE
The diminutive actor	died	at a hospital in Los Angeles	ClausIE
The diminutive actor	died		ClausIE
He	was	49 Los Angeles	ClausIE

are overly long and specific. However, while ClausIE also has many erroneous extractions with these difficult lead sentences, it does extract relations that Ollie misses. For these reasons I ended up choosing ClausIE as the ORE. Though this comparison was shallow and subjective, my conclusion seems to be supported given that Zouaq et al. (2017) evaluate ClausIE to have the best performance alongside their own ORE and Reverb (which trades high precision for low recall), while Martínez-Rodríguez et al. (2018) also chose ClausIE after a small evaluation.

Near the end of the development process I discovered that the original authors and developers of ClausIE, Corro and Gemulla (2013), had recently made available a new ORE that can be viewed as an extension of ClausIE, MinIE (Gashteovski et al., 2017). MinIE specifically addresses an issue ClausIE has received criticism for, and which I had found to be among the greatest problems for my approach: overly-specific extractions. As it was interesting to see whether MinIE might solve this issue, I decided to fork the development of the system into two branches, one using ClausIE as the ORE, and the other utilizing MinIE instead.

3.4.2 Annotation tool choices

After I had picked an ORE for the relation extraction itself, a choice of annotation tools had to be made. There were in particular two alternative approaches to consider, the first being using an ensemble of existing tools for the same annotation tasks with the benefit of superior performance but with the requirement of a decision protocol when annotation tools provide conflicting annotations. The second option was to use different tools for the different sub-tasks of annotation: named entity recognition, entity linking, and word sense disambiguation. Reasoning that the tools I had in mind have different strengths despite some overlaps, I chose the latter alternative.

DBpedia Spotlight was a clear and convenient choice because of its straightforward API that is tightly integrated with DBpedia, and though I knew from testing that erroneous mappings are not infrequent, particularly in short texts with limited context such as leads, it could be tuned somewhat towards more confident candidate mappings, even if finding an optimum is difficult. How-

ever, although Spotlight does sometimes link verbs to resources (although typically with quite low confidence scores), these annotations link to DBpedia articles, not properties in the DBpedia Ontology or even resources representing the verb in question. As an example we can consider the paragraph below:

> Pop star Michael Jackson has died in Los Angeles, aged 50. Paramedics were called to the singer's Beverly Hills home at about midday on Thursday after he stopped breathing. He was pronounced dead two hours later at the UCLA medical centre. Jackson's brother, Jermaine, said he was believed to have suffered a cardiac arrest.

None of the verbs will be annotated by default, but after lowering confidence scores sufficiently, "died" will link to a resource about 'Death of Michael Jackson'[3], "called" will link to 'Vocation'[4], while "suffered" will link to 'Passion of Jesus'[5]. As these annotated verbs rarely if ever link to useful resources that describe the verbs themselves, DBpedia Spotlight is first and foremost useful for linking named entities rather than verbs/relations.

To solve the issue of linking verbs to their correct RDF resource, I determined that an API more focused on the linguistic level of word senses would be more suitable to the disambiguation of verbs. I therefore choose Babelfy as a dedicated word sense disambiguation API for its high performance (citepBabelfy). The result is that verbs will typically link to a BabelNet IRI that represents the synset of the verb and is in turn linked to its WordNet equivalent synset. There are some verbs that BabelNet appears to ignore during annotation, in particular auxiliary verbs, even when they serve as the main verb, due to their absence in BabelNet. As discussed in more detail in section 4.1, this is of course very problematic when it comes to representing a verb phrase in RDF. Although Babelfy does not limit its analysis to verbs and can perform entity linking just like DBpedia Spotlight, I only used it to disambiguate the word sense of verb phrases, leaving the implementation

[3] `http://DBpedia.org/page/Death_of_Michael_Jackson`

[4] `http://DBpedia.org/page/Vocation`

[5] `http://DBpedia.org/page/Passion_of_Jesus`

of an ensemble entity linking module exploiting multiple APIs to increase performance as future work.

As a last annotation service I used Stanford CoreNLP's NER module to perform named entity recognition. Stanford NER is able to annotate many entities in the input text with one of three classes: Person, Location, and Organization. As these annotations are not RDF, they are as discussed substituted by IRIs bearing the same meaning, namely `rdf:type` as the property, and `dbo:Person`, `dbo:Location`, and `dbo:Organization` as the objects. Technically DBpedia Spotlight can perform much of the same task as the resources it associates named entities with are in turn linked to various classes that are returned by the API. There is however, to my knowledge, no convenient way of filtering these classes by importance or relevance to avoid having a multitude of classes that may sometimes be similar or overlapping returned (e.g. a specialization hierarchy). For example, the word "Fiji" is associated with the following classes: `Wikidata:Q6256, Schema:Place, Schema:Country, DBpedia:PopulatedPlace, DBpedia:Place, DBpedia:Location,` and `DBpedia:Country`. Furthermore it was of interest to diversify the usage of annotation tools in order to increase performance by covering potential "blind spots" the different techniques underlying the tools might have. For instance, DBpedia Spotlight requires some context in order to link the mention of a person, such as the full name of a celebrity, while Stanford NER can classify entities that are not notable enough to be associated with an existing RDF resource, such as that "Dan" is a Person.

3.4.3 Input domain

The system was primarily developed with a specific domain of text used as input for testing and evaluative purposes: the lead paragraphs ("leads") of news articles taken from RSS feeds, primarily from Reuters. This is a rather challenging domain of text, as these leads, often comprised of one or two sentences, are usually a form of summary of an entire news article and frequently contain at least one rather long and complex sentence, potentially with multiple conjuncts, prepositional phrases, and various subordinate clauses. Depending on the newspaper, these leads may be a standalone text that can be read on its own (as is the case with Reuters), or dependent on

the headline for context through e.g. pronouns (as is the case with many BBC leads).

This lead domain was chosen for several reasons as stated in the introduction (section 1.1), an important one because it was a form of input text more suitable to the task of knowledge graph extraction with my approach than the initial domain of tweets, as leads both provide more context for annotation tools, increasing their performance, and also more reliable grammatical consistency which results in better performance from OREs. As mentioned, the News Angler project was also an important reason for using RSS feeds from newspapers. RSS feeds are also readily available and easy to switch between, although news article feeds were particularly interesting for their difficulty, and because newspapers often express new and changing facts about notable entities. These entities, such as businesses, governments, politicians, and entertainers, are relatively likely to have resources about them on the Semantic Web, and are for that reason convenient to work with when it comes to entity linking. The difficulty aspect on the other hand is convenient for evaluating how feasible the method is, because it is easy to make the method work properly on simple, example sentences without the same approach being viable for more realistic, complicated sentences in other domains. At the same time, the challenge of the lead domain made getting relatively poor results inevitable in many cases, as state-of-the-art performance in semantic annotation and Open Relation Extraction is not able to handle certain leads without errors.

3.5 The System

The system I developed is a standalone Java application that integrates several existing tools; ClausIE, Babelfy, Stanford NER, and DBpedia Spotlight; in a partial implementation of the method presented in section 3.1. It takes as its input one or more sentences of natural language text. As output two types of textual triples are produced: the triples that represent the relations detected by the system's open relation extractor, and a variable number of triples for each of the first type of triples that connect individual words or phrases in these triples to IRIs.

It is a partial implementation because the output is not in the form of valid RDF statements, or in other words it is not a knowledge graph. This was because finding a way with which to convert the textual triples and aligned IRIs into RDF triples was a challenge that required a secondary, time-intensive literature review to investigate the state-of-the-art in the formal representation of natural language. This did not leave enough time for to implement this last step of the investigated approach to knowledge graph building. This process of conversion into a knowledge graph and its challenges is explored at a theoretical level in the next chapter.

In lieu of a valid knowledge graph, the system outputs an intermediate stage between the textual triples and their grounded RDF equivalents. For each relation detected by ClausIE, the textual extraction is given, and then for each annotated word or phrase in said extraction, this annotation is given in its own "pseudo-RDF" triple. These triples have the word or phrase (corresponding to part of or the whole of a segment from the ClausIE extraction) as the subject, a RDF property that differs depending on the source annotation tool as the property, and the object is the target IRI given by the annotation tool. To be clear, these are not valid RDF triples that can be used for reasoning, but rather a means for illustrating which words or phrases in an extraction have been annotated, and a way to assess the performance of both the ORE and annotation tools used. Three examples can be viewed in table 3.2, where ClausIE extractions are marked in gray and the pseudo-RDF triples follow below.

Table 3.2: System output triples for 3 extractions

Denis Norden	wrote	some of radio and TV 's funniest scripts in a 60-year partnership with Frank Muir
Denis Norden	owl:sameAs	`http://dbpedia.org/resource/Denis_Norden`
radio	owl:sameAs	`http://dbpedia.org/resource/Radio`

Frank Muir	owl:sameAs	http://dbpedia.org/resource/Frank_Muir
Denis Norden	rdf:type	foaf:Person
Frank Muir	rdf:type	foaf:Person
wrote	skos:definition	http://babelnet.org/rdf/s00095847v
Britain 's Queen Elizabeth	will attend	a special concert
Queen Elizabeth	owl:sameAs	http://dbpedia.org/resource/Elizabeth_II
Britain	rdf:type	dul:Location
Elizabeth	rdf:type	foaf:Person
attend	skos:definition	http://babelnet.org/rdf/s00082907v
John McDonnell	can avoid	no-deal Brexit
John McDonnell	owl:sameAs	http://dbpedia.org/resource/John_McDonnell
Brexit	owl:sameAs	http://dbpedia.org/resource/Brexit
John McDonnell	rdf:type	foaf:Person
avoid	skos:definition	http://babelnet.org/rdf/s00085002v

The system can be divided into several modules, most of which connect to the various APIs used in this project for relation extraction and annotation. The most crucial is of course the "Relation Extraction" module, which depending on the system branch can be ClausIE or MinIE. The ClausIE build was run with largely the default configuration, although the processing of appositive and possessive relations, two types of non-verbal relation, was disabled. An example of a sentence with both kinds of relation is "Kate, the wife of Britain's Prince William, was admitted to hospital in the early stages of labour on Monday to give birth to the couple's third child.", where

ClausIE with the default configuration will extract "Kate; is; the wife of Britain 's Prince William" as a appositive relation. Two possessive relations will also be extracted: "Britain; has; Prince William", and "the couple; has; third child". Although this type of relation is useful for the task of knowledge graph extraction, they typically have simple, common verbs that are difficult to ground due to their ambiguity, and Babelfy ignores many of them as they are not covered by WordNet. As such they were excluded because this thesis has largely been focused on the issue of verbal relations.

With the alternative branch discussed in the last section where MinIE is used for relation extraction, appositive and possessive relations are not disabled as this choice is not offered with MinIE to my knowledge (in fact, MinIE produces additional forms of implicit relations based on named entity pattern matching). MinIE does however offer several modes with different levels of aggression for the minimization process. I opted to use the safe mode, which gives the output shown in the next section and in the supplemental files. This choice was made because during development I found aggressive mode to produce erroneous extractions for several leads, while dictionary mode tended to produce the same extractions as safe mode except with occasional shortened phrases that reduce the coherence of the extraction. These observations mirrors the results found in the evaluation of MinIE (Gashteovski et al., 2017), where aggressive mode was found to reduce precision somewhat while dictionary mode's precision was almost identical to safe mode.

MinIE produces "minimal" extractions where parts of the extraction that indicate polarity, modality, attribution and quantity may be removed from the extraction itself, and instead given through what Gashteovski et al. (2017) call "semantic annotations" (using a rather different notion than this thesis, where (semantic) annotation has been used to denote a span of text being linked to a knowledge base or schema, particularly Semantic Web ones). While this functionality is highly interesting and relevant to my research, it could not easily be integrated into the system at this stage, which was developed with ClausIE extractions in mind. As such, this experimental branch only uses the minimal extractions themselves and not the "semantic annotations". A consequence is that the extractions themselves may be erroneous, as e.g. negations like "not" are missing. Nevertheless it was interesting to see whether the minimal extractions would be more simple to convert into

an RDF knowledge graph.

There are three annotation modules that handle calling the annotation APIs used in the project (DBpedia Spotlight, Babelfy, and Stanford NER), and return the annotation input for the given inputs. In the case of Babelfy, it is implemented using the provided Java API. The Babelfy parameters are modified so that the "annotation resource" (i.e the knowledge base used to disambiguate words) is set to BabelNet (excluding WordNet and Wikipedia, the former because BabelNet is already linked to WordNet, and the latter because Wikipedia does not have many resources about verbs), the Most Common Sense back-off is strategy enabled, and to return only the top ranked word sense candidates for a text fragment (as opposed to all candidates). Babelfy is thus far only used on verbs (i.e. the relations in ClausIE/MinIE extractions) in order to retrieve their BabelNet word sense URL, which along with the character offset of the annotated text fragment is put into a HashMap in order to map the annotations onto the textual triples in the module described in the following section.

The DBpedia Spotlight and Stanford NER modules share the same class, which is simply responsible for sending a HTTP POST request to a server; the DBpedia Spotlight Web API in the former case, and a locally hosted Stanford CoreNLP server in the latter. For both modules a JSON response is returned. The Stanford NER module was initially implemented using the standalone Stanford NER/CRFClassifier Java API[6], however it (as well as the full Stanford CoreNLP toolkit Java API) proved to be incompatible with the use of ClausIE due to dependency conflicts, because of its use of an older version of the standalone Stanford Parser. As such using the Stanford CoreNLP server was the most practical solution, even though it introduces slight delay of up to a few seconds to the runtime in some cases. The server is configured to only use the 3 class model (Person, Location, and Organization), as many of entity classes offered by other models, such as numerical (e.g. Percent) or temporal classes (e.g. Time), were more abstract and less immediately useful for the task of describing a referent (i.e. the subject or argument of a relation).

The output triples of the system (i.e. both ClausIE extractions and the triples that link extraction segments or parts of segments to IRIs ("link-

[6] `https://nlp.stanford.edu/software/CRF-NER.shtml`

ing triples")) are represented as a simple model class, "OutputQuadruples", which has a data field for each segment of the triple, and a fourth field indicating which of the two kinds of triple it is. A list of these objects is created in the main class, which expands as more linking triples are added by successive calls to the annotation modules.

Handling the risk of homonyms

Another module is the "DuplicateLemmaHandler", which was made to address a somewhat strange challenge: the risk of fringe cases involving homonyms (i.e. multiple words in a sentence that have the same lemma but different senses). Though rare (as it risks making the text confusing for humans, it is mostly used as a form of word play), particularly in the domain of news article leads, it was something I wished to handle given the abstract ambition of creating a knowledge graph extractor that can transform the plain text on the Web into machine-readable structured content. This issue is also likely something that becomes a greater risk with larger text inputs, e.g. entire news articles or Wikipedia articles, where the greater number of words increase the possibility of homonyms being used without a rhetorical or comedic intent. Because in leads that typically consist of one or two sentences I deemed it highly unlikely that homonyms would be used to make noun phrases ambiguous, I concentrated on addressing homonymous verb phrases.

This problem is somewhat compounded by the nature of many OREs because, as mentioned, in a lot of cases they produce multiple alternative extractions per relation they find in text. ClausIE is no exception from this behaviour, particularly in how prepositions in arguments are handled. As an example (taken from the paper that presented Clausie (Corro and Gemulla, 2013)), the sentence "AE was awarded the NP in Sweden in 1921" will give the following three extractions with the default configuration: "AE; was awarded; the the NP in Sweden in 1921" ,"AE; was awarded; the NP in Sweden" and "AE; was awarded; the NP". The difference between these extractions is only the exclusion of prepositional phrases, but the result is that many sentences, certainly in the news article lead domain, give multiple extractions for some relations.

The problem then is that there may be duplicate words in the input text itself, and there may be duplicate words in the extractions. These two factors complicate the seemingly simple implementation step 4 of the method; the mapping of annotations onto the ORE extractions, at least if one takes into account the risk of these rare fringe cases. This complication might not be inherent to the method, as it would be trivialized if the ORE stores the offset in the input text of the words that it uses in its extractions (i.e. the source of the subject, predicate and arguments), such that the annotated input text could simply be mapped to the extractions by comparing offsets in the input text string, but unfortunately this was not the case with ClausIE.

Examples of the kind of sentence with these fringe case homonyms are hard to make without them being extremely contrived. Still, to illustrate the problem, a (fictitious) example might be: "After SpaceX launched their record-breaking spacecraft, their official fan store launched a new line of commemorative clothing." Here we have the verb "launch" used in one sentence with two senses: the first one is to "propel with force" and the second is to "establish" or "set up". Ideally this sentence should result in at least two extractions (as each verb functions as a relation), but the relations should be represented by different synsets (that is, WordNet collections of synonyms). Unfortunately, Babelfy's disambiguation performance for this specific verb is not strong enough to detect the second sense of the word and so always appears to return the former. However this is always a risk with annotation, as even human annotators are prone to make conflicting or erroneous judgments when tasked with word sense disambiguation (Rumshisky and Batiukova, 2008).

My solution to this issue was to first divide the problem into four possible cases:

1. There are no duplicate words in the input text nor the extractions
2. There are duplicates in the extractions but not in the input text
3. There are duplicate words in the input text, but not in the extractions
4. There are duplicate words in both the input text and the extractions

Cases 1 and 2 are trivial and can be handled by simply naively mapping

via string matching, as there are no potential homonyms in the text. Case 3 is, as far as I have determined, not possible to definitely solve (without perhaps modifying the source code of ClausIE itself), as there is no way to determine which duplicate word resulted in one of the segments of a ClausIE extraction. For example, a sentence might contain two senses of the verb "skip", which are each detected by the annotation tools, yet the ORE might only detect one "skip" relation, resulting in only one extraction. Which sense of the word, i.e. IRI, the relation should be mapped to is difficult to solve. It might be possible to solve this in a lot of scenarios by applying a heuristic based on the distance between words in the input text; if the subject and argument of the relation in question are closer to sense1 than sense2, then it is likely that sense1 should be chosen. However this logic does not apply in some other scenarios, such as if the extraction is the result of an implicit relation, a minimal extraction that omits many words, or an erroneous interpretation by the ORE. As such I did not solve case 3, and handle it like case 1 and 2.

For case 4 the different senses have to be mapped to their appropriate extractions, but two factors complicate the issue. Firstly, there may be equally many senses as extractions, but there is no guarantee that each sense is present in the extractions, as it may be that one sense results in multiple extractions while another does not result in a single one. Secondly, it is equally possible that one sense results in more extractions than another, for example giving multiple extractions while the other(s) only results in one sense.

My solution to this case was to keep track of how many times a relation has been "handled" (i.e. mapped to its surface form in the input, and by extension to the annotations), and to assign it to its n-th occurrence in the text based on how many n times it had been handled. The intuition here is based on the fact that ClausIE outputs extractions sorted by their occurrence in the input, and thus we can at least safely assume that the first extraction with a homonymous relation will correspond to the first sense in the input. If the number of homonymous relations in the input text and the extractions are equal, it is also likely that the mappings become correct with this solution, because the first factor mentioned above is quite unlikely to happen as ClausIE rarely completely misses relations in text.

The second factor is a far more likely risk given ClausIE's propensity to

produce multiple partially duplicate extractions, especially with prepositional phrases, and will result in there being more instances of the homonymous relation in the extractions than in the text. In these cases, my solution continues its default assignment until n becomes higher than the number of relations in the text, at which point all extractions are mapped to the last occurrence in the text. This can result in erroneous mappings in cases where partially duplicate extractions are produced for a homonymous relation in the text which is not the last occurrence, as the first extraction will be mapped correctly, but then the subsequent partially duplicate extractions will be mapped to the next sense(s).

It was difficult to test this solution because the problem only exists in rare fringe cases and is difficult to search for, and thus finding actual news article leads that contain these problematic homonyms was not something time permitted given that the issue itself is not inherent to the general approach but rather the specific tools used in the implementation. With constructed sentences it functions as a heuristic that reduces the chance of such errors, particularly as the system was developed to process leads that typically have no more than two (verbal) relations. It cannot however be relied upon to solve the problem. In addition, as seen in the example discussed previously, annotation tools do not always have the performance needed to differentiate between different senses of homonymous relations. Furthermore these homonymous relations are generally quite rare in the domain of leads (although they may potentially be a greater risk in newspapers that allow for less formal styles, which can perhaps be seen in e.g. some tabloids and British newspapers). Given these issues, there is a question of how much practical use the solution was.

3.6 Output assessment

In this section complete examples of the system's output for lead inputs will be shown and assessed. As mentioned, the system is incomplete in the sense that it does not output knowledge graphs, but these examples still illustrate strengths and challenges in the approach in terms of ORE and annotation performance. A total of 21 leads were used for the development and testing of the implementation. They were selected based on the criteria

established in the introductory section 1.1, with 16 being selected throughout development, and 5 being control leads taken from sequential leads from an RSS feed near the end of the project. These control leads were selected in order to attempt to uncover any unconscious bias I may have had while selecting development examples. The performance between the development and control leads turned out to be very similar.

Because displaying the output requires a lot of space, only 5 examples will be considered here, with a slight bias towards shorter leads with fewer extractions. Furthermore, the 5 examples considered here are all from the 16 leads used during development, because it was the development leads that I used to uncover and explore the challenges I discuss in the next chapter, not the control leads. The full output is available in the the file "full_output.txt", and explanations about the leads and their origin is available in the "leads_and_sources.txt" file. An editor like Notepad++ or Sublime Text 3 is recommended for proper formatting. After these 5 examples have been addressed, the output of the same leads with the alternative MinIE-based branch of the system will be considered.

3.6.1 Main branch output

Example 1 "Korean Church..." Table 3.3 is the output of the lead "A Korean church hiding from looming 'global famine' in Fiji is facing growing allegations of abuse." This is an example of a relatively good result, both in terms of ClausIE extractions and annotations. ClausIE only makes two rather concise extractions (the grey rows of the table), and they are both correct interpretations that represent the only two relations present in the lead. One limitation here is that the lead has put quotes around the expression "global famine" to indicate that the idea is dubious and/or asserted by the church rather than the journalist. These quotation marks are not retained by ClausIE in the extractions, but even if they were this is an issue that requires more discussion, which is provided in chapter 5 section 4.8.

The annotations cover a large portion of this lead, although several of them are slightly inaccurate. Most obviously, in both extractions "Korean" is mapped to `dbr:Korean_language` which is not quite the correct interpre-

Table 3.3: System output example 1

A Korean church	be hiding	from looming global famine in Fiji
Korean	owl:sameAs	http://dbpedia.org/resource/Korean_language
church	owl:sameAs	http://dbpedia.org/resource/Catholic_Church
famine	owl:sameAs	http://dbpedia.org/resource/Famine
Fiji	owl:sameAs	http://dbpedia.org/resource/Fiji
Fiji	rdf:type	dul:Location
hiding	skos:definition	http://babelnet.org/rdf/s00089332v
A Korean church hiding from looming global famine in Fiji	is facing	growing allegations of abuse
Korean	owl:sameAs	http://dbpedia.org/resource/Korean_language
church	owl:sameAs	http://dbpedia.org/resource/Catholic_Church
famine	owl:sameAs	http://dbpedia.org/resource/Famine
Fiji	owl:sameAs	http://dbpedia.org/resource/Fiji
Fiji	rdf:type	dul:Location
facing	skos:definition	http://babelnet.org/rdf/s00087942v

tation as the meaning of "Korean church" is that the members of the church are of Korean nationality, not that the church uses the Korean language. A similar mistake is that "church" is mapped to `dbr:Catholic_Church` instead of `dbr:Local_church`. The former resource represents an article about the Catholic Church as a worldwide denomination, which this Korean church is not necessarily even a part of.

Another minor issue of accuracy is "facing" being assigned the sense of `http://babelnet.org/rdf/page/s00087942v`, defined as "Be oriented in a certain direction, often with respect to another reference point; be opposite to". This is the literal sense of "to face" as a verb, but the figurative sense of `http://babelnet.org/rdf/page/s00085597v`, defined as "Present somebody with something, usually to accuse or criticize", would be more accurate here. Other than these errors which all are characterized by imprecision rather than complete irrelevance, the rest of the annotations are correct.

Table 3.4: System output example 2

Kate	was admitted	to hospital in the early stages of labor to give birth to the couple 's third child
Kate	owl:sameAs	`http://dbpedia.org/resource/Kate_Ramsay`
Kate	rdf:type	foaf:Person
admitted	skos:definition	`http://babelnet.org/rdf/s00082267v`
Kate	was admitted	to hospital in the early stages of labor to give birth
Kate	owl:sameAs	`http://dbpedia.org/resource/Kate_Ramsay`
Kate	rdf:type	foaf:Person

admitted	skos:definition	http://babelnet.org/rdf/s00082267v
Kate	was admitted	on Monday to give birth to the couple 's third child
Kate	owl:sameAs	http://dbpedia.org/resource/Kate_Ramsay
Kate	rdf:type	foaf:Person
admitted	skos:definition	http://babelnet.org/rdf/s00082267v
Kate	was admitted	on Monday to give birth
Kate	owl:sameAs	http://dbpedia.org/resource/Kate_Ramsay
Kate	rdf:type	foaf:Person
admitted	skos:definition	http://babelnet.org/rdf/s00082267v
Kate	was admitted	to give birth to the couple 's third child
Kate	owl:sameAs	http://dbpedia.org/resource/Kate_Ramsay
Kate	rdf:type	foaf:Person
admitted	skos:definition	http://babelnet.org/rdf/s00082267v
Kate	was admitted	to give birth
Kate	owl:sameAs	http://dbpedia.org/resource/Kate_Ramsay
Kate	rdf:type	foaf:Person
admitted	skos:definition	http://babelnet.org/rdf/s00082267v

Example 2 "Kate..." Table 3.4 shows the output of the lead "Kate, the wife of Britain's Prince William, was admitted to hospital in the early stages of labour on Monday to give birth to the couple's third child." This is an example of a lead where ClausIE gives multiple partially duplicate extractions, which is frequently the case with leads. All the 6 ClausIE extractions here (grey rows) are really the same relation, "was admitted" in the text, that is given by ClausIE with the various prepositional phrases included or excluded. In this case, it is questionable whether ClausIE has inferred correctly which prepositional phrases are optional. "On Monday", "in the early stages of labor", and "to the couple's third child" can be safely omitted, but "to hospital" is perhaps not optional as "where" one is being admitted is an important part of the relation. If however we assume that "Kate ; was admitted ; to give birth" is a fully coherent extraction, then all of these partially duplicate extractions are valid. The fact that ClausIE outputs all of these overlapping extractions may seem messy, however it is in fact beneficial and a way of splitting up long arguments from an ORE perspective, as discussed by Zouaq et al. (2017).

Even so, the output would be considerably cleaner if not for a possible issue with ClausIE's configuration. For this output the minimum and maximum optional arguments parameters were set to respectively zero and one, which one might expect should restrict prepositional phrases to at most one per extraction (setting the maximum to zero would with this lead only output the last extraction). However as can be seen in the table, long arguments with multiple prepositional phrases (the first three extractions) are present. This does not seem to be a consequence of 'nested prepositional phrases' or dependencies in the underlying Stanford parse, and is present in the examples given in Corro and Gemulla (2013).

There are only three annotations for this lead, which are present for every extracted relation. Only the assignment of "Kate" into the Person class by Stanford NER (in turn substituted by the `foaf:Person` resource) is correct here. DBpedia Spotlight annotates the "Kate" mention with the `http://dbpedia.org/resource/Kate_Ramsay` resource, when "Kate" actually refers to Kate Middleton, described by `http://dbpedia.org/page/Catherine,_Duchess_of_Cambridge`. As with the previous output example, the Babelfy sense assignment lacks precision, although in this case it is semantically invalid in addition to not being the most appropriate sense out of several

alternatives. The sense http://babelnet.org/rdf/s00082267v assigned to the "admitted" relation, defined as "Declare to be true or admit the existence or reality or truth of", is not a coherent interpretation when applied to the input sentence. More appropriate alternatives would have been http://babelnet.org/rdf/page/s00082369v, defined as "Allow to enter; grant entry to", or perhaps even http://babelnet.org/rdf/page/s00082201v, "Admit into a group or community".

Table 3.5: System output example 3

Japan 's Naomi Osaka	beats	Dominika Cibulkova in straight sets at the Pan Pacific Open her first match since winning the US Open
Japan	owl:sameAs	http://dbpedia.org/resource/Japan
Naomi Osaka	owl:sameAs	http://dbpedia.org/resource/Naomi_Osaka
Dominika Cibulkova	owl:sameAs	http://dbpedia.org/resource/Dominika_Cibulkov%C3%A1
Pacific	owl:sameAs	http://dbpedia.org/resource/Pacific_Ocean
US Open	owl:sameAs	http://dbpedia.org/resource/US_Open_(tennis)
Japan	rdf:type	dul:Location
Naomi Osaka	rdf:type	foaf:Person
Dominika Cibulkova	rdf:type	foaf:Person
her	rdf:type	foaf:Person
beats	skos:definition	http://babelnet.org/rdf/s00083247v

Japan 's Naomi Osaka	beats	Dominika Cibulkova in straight sets her first match since winning the US Open
Japan	owl:sameAs	http://dbpedia.org/resource/Japan
Naomi Osaka	owl:sameAs	http://dbpedia.org/resource/Naomi_Osaka
Dominika Cibulkova	owl:sameAs	http://dbpedia.org/resource/Dominika_Cibulkov%C3%A1
US Open	owl:sameAs	http://dbpedia.org/resource/US_Open_(tennis)
Japan	rdf:type	dul:Location
Naomi Osaka	rdf:type	foaf:Person
Dominika Cibulkova	rdf:type	foaf:Person
her	rdf:type	foaf:Person
beats	skos:definition	http://babelnet.org/rdf/s00083247v

Example 3 "Naomi Osaka..." The lead used for the output shown in 3.5 is "Japan's Naomi Osaka beats Dominika Cibulkova in straight sets at the Pan Pacific Open, her first match since winning the US Open." As in the previous table, the extractions here are partially duplicate with the difference between the two being the inclusion of the prepositional phrase. An issue both share is that the embedded clause introduced by the comma in the input sentence is kept in the argument, yet the comma is removed. This makes the extractions incoherent, though they are otherwise correct.

In terms of annotations, the main clause of the sentence ("Japan's Naomi Osaka beats Dominika Cibulkova in straight sets at the Pan Pacific Open") is an example of the kind of sentence that is most suitable to this approach to knowledge graph extraction. Aside from the prepositional phrases, it consists of a verb phrase between two noun phrases with proper nouns that

correspond to noteworthy entities (two famous tennis players) with existing Semantic Web resources. As such the annotations for this lead are largely good. The NER classes are correct, as are the DBpedia Spotlight annotations with one arguable exception. The word "Pacific" from "Pan Pacific Open" is linked to the resource `http://dbpedia.org/resource/Pacific_Ocean`. While it is strictly speaking the case that the Pan Pacific Open, a tennis tournament, takes its name from the Pacific Ocean (and takes place on an island in the Pacific Ocean), this is only a partial disambiguation. At best all that can be asserted based on this annotation is that the Pan Pacific Open is somehow associated with the Pacific Ocean. This type of annotation issue is not uncommon, as many compound words have the same characteristic. Lastly, with this lead the Babelfy sense assignment is accurate unlike in the previous two examples, as "beats" is mapped to `http://babelnet.org/rdf/s00083247v`, defined as "Come out better in a competition, race, or conflict".

Table 3.6: System output example 4

Barbara Bush	was remembered	at her funeral on Saturday Former first lady but caring figure
Barbara Bush	owl:sameAs	`http://dbpedia.org/resource/Barbara_Bush`
Barbara Bush	rdf:type	foaf:Person
her	rdf:type	foaf:Person
remembered	skos:definition	`http://babelnet.org/rdf/s00084413v`
Barbara Bush	was remembered	as a formidable Former first lady but caring figure
Barbara Bush	owl:sameAs	`http://dbpedia.org/resource/Barbara_Bush`
formidable	owl:sameAs	`http://dbpedia.org/resource/Formidable-class_frigate`

Barbara Bush remembered	rdf:type skos:definition	foaf:Person http://babelnet.org/rdf/s00084413v
Barbara Bush	was remembered	Former first lady but caring figure
Barbara Bush Barbara Bush remembered	owl:sameAs rdf:type skos:definition	http://dbpedia.org/resource/Barbara_Bush foaf:Person http://babelnet.org/rdf/s00084413v
figure devotion to her family	was matched	only by her commitment
her her matched	rdf:type rdf:type skos:definition	foaf:Person foaf:Person http://babelnet.org/rdf/s00082477v
figure devotion to her family	was matched	only to public service
her matched	rdf:type skos:definition	foaf:Person http://babelnet.org/rdf/s00082477v
figure devotion to her family	was matched	only
her matched	rdf:type skos:definition	foaf:Person http://babelnet.org/rdf/s00082477v

Example 4 "Barbara Bush..." Table 3.6 shows the output that can be seen as something close to a worst case for the system, where the majority of the extractions are erroneous and annotations disambiguate little of the input text. The lead used is "Former first lady Barbara Bush was remembered at her funeral on Saturday as a formidable but caring figure whose devotion to

her family was matched only by her commitment to public service". This is a long (32 word) sentence that expresses a very complex relation, both in terms of semantic meaning and grammatical construction. Out of the six extractions (grey rows), only the second is without issues: in the first, the argument is not grammatically valid; in the third the combination of the predicate and the argument is not valid; and in the remaining extractions the subject is both grammatically invalid and semantically incoherent. In addition, the combination of the predicate and argument is incoherent in both the fifth and sixth extractions.

The annotations are for the most correct, with the exception that the adjective "formidable" is mapped by DBpedia Spotlight to to `http://dbpedia.org/page/Formidable-class_frigate`. This phenomenon, a "common word"[7] being interpreted as a "proper word", is always a risk with DBpedia Spotlight, although it does not happen often with the default confidence threshold. The risk of errors such as these would likely be significantly lessened with an ensemble annotation approach, because erroneous mappings like this one could be challenged by alternative mappings with potentially higher certainty from other annotation tools. Although the remaining annotations are correct, it is a problem that there are only two to three annotations per extraction, which leaves large portions of the extractions without disambiguation.

Table 3.7: System output example 5

Martin Fayulu	insists	he won the presidential election and has demanded a manual recount
Martin	owl:sameAs	`http://dbpedia.org/resource/List_of_recurring_The_Simpsons_characters`

[7] As seen in this example, they are not necessarily nouns

presidential election	owl:sameAs	http://dbpedia.org/resource/United_States_presidential_election
Martin Fayulu	rdf:type	foaf:Person
he	rdf:type	foaf:Person
insists	skos:definition	http://babelnet.org/rdf/s00089629v
he	won	the presidential election
presidential election	owl:sameAs	http://dbpedia.org/resource/United_States_presidential_election
he	rdf:type	foaf:Person
won	skos:definition	http://babelnet.org/rdf/s00095777v
he	has demanded	a manual recount
he	rdf:type	foaf:Person
demanded	skos:definition	http://babelnet.org/rdf/s00082822v

Example 5 "Martin Fayulu..." As a final example, the output of the lead used to demonstrate the ideal case knowledge graph extraction in section 3.3 is displayed in table 3.7. The full sentence is "Martin Fayulu insists he won the presidential election and has demanded a manual recount." The extractions given by ClausIE are without issues, and differ little from the ones I manually determined were appropriate, although there is of course no "reification" to indicate that the argument of the first extraction is the combination of the second and third. Another fine detail is that ClausIE interprets the implicit subject of the "has demanded; a recount" relation to be the preceding "he" noun phrase, while I consider "Martin Fayulu" to be a more straight-forward interpretation.

Unfortunately the annotations are worse than average for this sentence,

in part due to there only being one proper noun, and in part because of some erroneous annotations. Clearly, "Martin Fayulu" is not a "The Simpsons" character. Given that it is a full name of a "notable entity", DBpedia Spotlight would normally be able to map this kind of name accurately, but Martin Fayulu (a Congolese businessman and lawmaker) appears to only recently have gained worldwide fame, as his Wikipedia article was created only in November 2018 and is not yet represented in DBpedia. '"residential election" is also mapped to a resource for US presidential elections, which is not accurate since this lead is about the Congolese election. This sort of "bias", where a general phenomenon is particularly strongly associated with or popular in a single context (such as a country, e.g. US presidential affairs, the UK royal family), is probably not uncommon with entity linking. This is similar to how, in the first example lead considered in this chapter, "church" was mapped to `dbr:Catholic_Church` rather than `dbr:Local_church`, presumably because the Catholic Church resource is much more well-connected and prominent. A highly related problem is that non-notable people often have their first name or surname annotated with an famous entity bearing the same name, which is not seen in the examples shown here, but is present in the full output. The Babelfy annotations for the verbs are on the other hand without issues. Compared to the ideal processing presented in section 3.3, the biggest issue are the erroneous and incomplete annotation of the sentence, as well as resolving the antecedents of pronouns.

3.6.2 MinIE branch output

The output of all three leads processed by the MinIE branch can be seen in table 3.8, with the "annotation triples" removed. This was because they are identical to those of the previous tables in this section due to being based on the same leads (with the exception of a few new Babelfy NED annotations, present due to words outside of the verb phrase being included in some MinIE relations). As mentioned in section 3, MinIE is not fully utilized as the MinIE annotations (polarity, attribution etc.) are not shown, and the branch was made to investigate whether or not more minimal extractions are easier to ground than the often rather long ones produced by ClausIE. These extractions were produced with MinIE's safe mode, because as mentioned I found during development that aggressive mode, while it may further

minimize extractions, increased the amount of errors.

Table 3.8: MinIE branch output

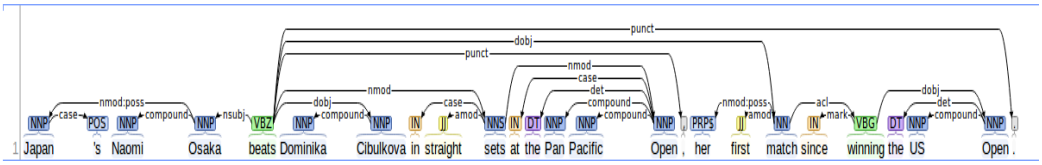
Japan 's Naomi Osaka	beats Dominika Cibulkova in sets at	Pan Pacific Open
Japan 's Naomi Osaka	beats	Dominika Cibulkova
Korean church hiding from looming global famine in Fiji	is facing growing allegations of	abuse
Kate	was admitted to hospital to give	birth to couple 's third child
Kate	was admitted to hospital to give	birth
Kate	was admitted in early stages of labour to give	birth to couple 's third child
Kate	was admitted in early stages of labour to give	birth
Kate	was admitted on Monday to give	birth to couple 's third child
Kate	was admitted on Monday to give	birth
Kate	was admitted to give	birth to couple 's third child
Kate	was admitted to give	birth
Barbara Bush	is	lady
first barbara Bush	was remembered at funeral on Saturday as	formidable figure
first Barbara Bush	was remembered at funeral on Saturday as	caring figure
first Barbara Bush	was remembered at funeral on	Saturday
her	has funeral on	Saturday
formidable figure	has	devotion to family
caring figure	has	devotion to family

her	has	family
formidable figure devotion to family	was matched by	only commitment to public service
caring figure devotion to family	was matched by	only commitment to public service
her	has	commitment to public service
Martin Fayulu	insists	has demanded manual recount
he	won	presidential election
he	has demanded	manual recount

Looking past the possessive relations (the extractions with a "has" relation for the "Barbara Bush" input), which were as mentioned disabled in ClausIE, there is a small but noticeable performance reduction between MinIE and ClausIE). Specifically, ClausIE extracts a "be hiding" relation from "hiding from looming global famine in Fiji" for the subject "A Korean church" that MinIE misses. Similarly, the MinIE extraction "Martin Fayulu; insists; has demanded manual recount" is both incoherent and grammatically invalid. Some of the minimizations also reduce coherence, as is the case with the relation "beats Dominika Cibulkova in sets at", where "straight" is omitted from the Tennis expression "straight sets", or when "lady" is removed from the subject "first Barbara Bush". The only case where MinIE arguably outperforms ClausIE is for the "Barbara Bush" sentence, where, with the exception of the aforementioned subject issue, the extractions are somewhat more coherent though there are still several of the same issues present. Furthermore, there are even greater performance differences in several of the leads not included among these examples, making these leads relatively charitable in terms of comparing MinIE and ClausIE.

In both the ClausIE and MinIE extractions, the subjects are for the most part the same, and the most obvious difference is that while the MinIE extractions have significantly shorter arguments, it is typically at the cost of the text being moved into the relations. Zouaq et al. (2017) discuss the length of relations in ORE extractions, and argue that while longer relations are valid, they may not be useful in the context of some kind of semantic analysis task due to their specificity and statistical infrequency. As such they

Figure 3.2: Stanford Dependency parse example



propose a heuristic that limits the number of dependency links that must be traversed from the head word of the subject to the head word of the argument to at most three. Applying this heuristic to the extractions of table 3.8, using Stanford CoreNLP dependency parsing, all of these relations are, somewhat surprisingly, valid. For example, "Japan's Naomi Osaka; beats Dominika Cibulkova in sets at; Pan Pacific Open" has three dependency links (Osaka → beats → sets → Open, see figure 3.2) and is as such valid. The same is the case for the fifth and sixth extractions, "Kate; was admitted in early stages of labour to give; birth (to couple's third child)", as the dependency links can be traversed in 3 steps (Kate → admitted → give → birth).

While these longer relations may not be inherently problematic for the ORE paradigm[8], unlike ClausIE many OREs only extract nominal arguments, putting e.g. prepositions into relations. For example, ClausIE extracts "The diminutive actor; starred; in the Austin Powers movies", while Ollie extracts "The diminutive actor; starred in; the Austin Powers movies", they are not beneficial to this approach towards knowledge graph extraction. In comparison to the short relations of ClausIE, which are usually a verb phrase comprised of a main verb and possibly auxiliary verbs, longer relations are more difficult to ground because they are composed of more elements. As such, a fragmented representation consisting of a sub-graph of multiple linked RDF nodes would be necessary to represent these longer relations, similar to what this approach has already proposed for compound verb phrases except that the sub-graph would be larger. At this point the difference between having the long, overly specific arguments of ClausIE and the long, overly specific relations of MinIE (at least using MinIE's safe mode) would not be very significant, and largely come down to whether the RDF nodes representing the text in question are connected to the "relation head node" or the "argument head node" in the RDF graph. This being the case,

[8]In fact, as noted by Zouaq et al. (2017)

MinIE does not represent an improvement over ClausIE as an ORE for this approach, at least without increasing minimization aggression, which comes at the price of reduced precision (Gashteovski et al., 2017).

3.6.3 Output conclusion

These output triples represent an intermediate stage rather than the final output of a completed implementation, but they still communicate some of the challenges, and also opportunities of the approach. There are significant issues present due to only parts of these extractions being annotated, in large part because DBpedia Spotlight mainly detects high-confidence annotation candidates for proper nouns and adjectives (e.g. Korean or Naomi Osaka), unless confidence thresholds are lowered. This however reduces precision, which is already an issue with the default threshold of 0.5, which was used to generate the discussed output. Interestingly, the use of MinIE which extracts longer relations demonstrated that Babelfy (used in the system to disambiguate the relation, i.e. verb phrase when using ClausIE) may in fact be superior to DBpedia Spotlight in the task of disambiguating subject and arguments. Unlike DBpedia Spotlight, Babelfy performs both WSD and NED, and as such annotates many common nouns and adjectives that DBpedia Spotlight misses by linking them to WordNet. These Babelfy annotation triples were omitted from table 3.8, but can be viewed along with the remaining output in the supplemental files.

This issue of partial annotation underlines the need for an ensemble of annotation tools rather than relying on Stanford NER and DBpedia Spotlight by themselves to annotate extraction subjects and arguments. By combining WSD and NED tools however, it seems likely that it is with current technology possible to disambiguate most of the words in a given sentence (at least within the domain of leads) with reasonably high performance. The remainder of the sentence that is difficult to disambiguate is largely words that have little individual semantic meaning but rather have grammatical functions in a sentence, such as determiners, particles, and auxiliary verbs. Some of these can and should probably not be annotated and linked to IRIs, but rather be processed in a lower-level grammatical analysis. This is a diverse class of words that cannot be used in a single way for knowledge graph extraction.

The ClausIE extractions shown in this section do not have many errors, and aside from the worst case example, they can be considered slightly above average in terms of validity for ClausIE extractions on the input domain of leads. Even so, ClausIE's performance on leads is quite high, and the major challenge is neither erroneous extractions nor relations in the input text being missed, but rather the long arguments (or relations in the case of MinIE) which complicate the process of creating an RDF representation for each segment of the extraction. The issue of how these arguments can be processed with this approach is discussed in section 4.1 in the following chapter. Overall the ORE extractions shown in this section show that there is promise in using OREs to extract important facts from newspapers articles, even though the output form which is close or identical to natural language makes the disambiguation task challenging.

Chapter 4

Discussion

In this chapter many of the problems discovered during the implementation process and in the assessment will be taken up and discussed, with solutions proposed where possible. The different problems are loosely organized by decreasing importance and increasing abstractness. An exception is the last section, which is concerned with a more thorough comparison to two of the most adjacent approaches to knowledge graph extraction from natural language text than chapter 2.

4.1 Representing textual relations in RDF

As has been alluded to multiple times, a significant challenge for this approach to automated knowledge graph extraction is finding appropriate ways of translating the extractions and associated annotations into valid RDF triples. As discussed in the literature review, this issue has been handled by various previous approaches to knowledge graph extraction such as FRED (Gangemi et al., 2017) or Martínez-Rodríguez et al. (2018), but the task is still open and there is no consensus in the literature about what an ideal representation of natural language in RDF looks like. This section will not discuss the advantages and disadvantages of prominent options available, but rather highlight some of the challenges involved in the task, and present a

general, exploratory approach to conversion, as an extension to the considerations mentioned in chapter 3 section 3.1.

This is the biggest issue faced during this master thesis and can be further divided into several smaller problems; namely 1. representing subject/argument phrases that correspond to multiple Semantic Web IRIs in RDF, 2. representing phrases where the annotation tools only maps parts of the phrase to Semantic Web IRIs, and 3. representing compound relational verb phrases as RDF. As mentioned, some discussion of English grammar is necessary in order to address these problems. However, this thesis will not go into great detail about linguistics, and it must be noted that the visited issues may have been studied much more rigorously in the humanities.

4.1.1 Embedded clauses

Starting with multi-word phrases that serve as subjects or arguments in extractions, a further distinction has to be drawn between noun phrases and other syntactic categories that can take the same place, specifically a variety of subordinate clauses. These may be embedded clauses that are not independently coherent (e.g. "They; are working; [to come to an agreement]" or "Martha; nodded; [slowly as if she understood]"), or clauses that stand on their own as valid sentences, e.g. "The police; claimed; [(that)[1] forensic tests were needed to identify the deceased]" (Carnie, 2013, Chapter 7).A solution to independently coherent clauses is some form of reification, as will be discussed in the next section.

The representation of embedded clauses that are not independently coherent, such as various non-finite clauses (e.g. "To sing in the rain", "For them to meet now", and "Having fun playing football") is more problematic. These clauses can become quite long, which makes the use of a single IRI impossible but at the same time they don't really correspond to the Subject-Property-Object form of RDF triples. A kind of triple relational representation might be envisioned for many of these clauses, such as in(sing, rain), meet(them, now) or fun(play, football). However, these examples do not retain all the

[1]Here, as in many cases when it used to introduce a subordinate clause, "that" is optional and can be omitted

information present in the clauses ("fun(play, football)" for example resembles an assertion stating that playing football is always fun, rather than the activity of "Having fun playing football" itself) and furthermore a triple representation is not possible for all clauses of this type, e.g. "Playing football" where there are not enough words in the phrase to make a triple (which is important in the context of extracting triples from natural language). Due to these challenges, as well as the fact that many of these clauses, if they can be said to express relations at all, express them through prepositions such as "in" or "for" as opposed to the verbal relations, a solution to this problem is beyond the scope of this thesis and is likely to require other techniques than those discussed here, such as a method of decomposition not offered by OREs. In addition, it should be noted that, to my knowledge, most previous work within ORE, whether "plain" ORE such as the assessment of Zouaq et al. (2017), or adjacent attempts at disambiguating extractions like Martínez-Rodríguez et al. (2018) limit their focus to relations where the subject and argument are noun phrases or noun phrases preceded by a preposition (e.g. "in the river"), and as such do not consider these challenging types of clauses.

4.1.2 Noun phrases

With compound noun phrases (NP) representation is more straight-forward, although there are still challenges with retaining information in some cases. Starting with NPs where the entire phrase is annotated, there are some compound NP that are not problematic at all, such as many proper nouns or open compound common nouns (e.g. "Barack Hussein Obama" and "prime minister"), as annotation tools will typically annotate the entire phrase (assuming the phrase is annotated at all). In other cases, the different elements of a compound NP might be annotated with different IRIs, for instance the phrase "British soil" might be mapped to the IRIs `dbr:British_isles` and "dbr:Soil". My intuition in such a situation is that the entity or concept described by the compound NP is itself fundamentally compound in the sense that it is best described with multiple IRIs. That is not to say that it is impossible for a single IRI to represent the phrase, but it is impractical in many cases because the specificity possible with phrases makes providing ontological coverage (i.e. IRIs) for any given phrase unfeasible. In a parallel approach to knowledge graph extraction, Martínez-Rodríguez et al. (2018)

solve the issue in a rather similar way, by representing NP-entities as automatically generated IRIs in a local context, that are in turn associated with multiple IRIs through a local property.

The more common situation is that annotation tools only cover parts of a NP. With many text domains, this is all but inevitable with many sentences considering the current performance offered by annotation tools, even if confidence thresholds are reduced (which introduces frequent erroneous mappings). It is especially common that words that are not proper nouns are left out, for example adjectives like "icy" or "tall", or common nouns like "deputy" or "man". The solution for wholly annotated NPs can be extended to cover these cases as well. The local IRI that represents the phrase itself might for example be `ex:the_diminutive_actor`, and this IRI might be linked only to the parts of the phrase that are identified, resulting in for example the triple `loc:the_diminutive_actor loc:associatedWith dbr:Diminutive .` if only the word "diminutive" was annotated. An alternative might be to represent non-annotated words with a local, automatically-generated "placeholder" IRI (e.g. in the example above, `ex:actor`) to avoid having parts of a phrase that are simply missing, unlike what would be the case if the entire phrase is annotated.

By themselves, these "placeholder" IRIs would serve little purpose as they cannot be dereferenced to identify the resource, and furthermore determining whether two or more successive, non-annotated words in the phrase are a unit that in fact corresponds to a single IRI (e.g. a compound noun) or are separate in terms of meaning introduces more complexity. However, this is where the Named Entity Recognition annotations from Stanford NER may be useful, as they can be used to partially disambiguate these placeholder IRIs by assigning them some class (e.g. `rdf:type foaf:person`). In this way, though a good resource like a DBpedia resource that is heavily interconnected in the LOD cloud is not available, at least something can be asserted about the resource. This is particularly useful when entities (for example people) that are not notable who do not have an associated resource in e.g DBpedia are encountered. While only three classes are used in the Stanford NER annotator in the current implementation, other NER annotators may provide more classes that are useful for this task, for example the IBM Natural Language Understanding API which provides a very extensive

hierarchy of types.[2]

In many cases, NPs are specified further by being the complements of prepositional phrases that may or may not be included in the subject/argument segments of extractions, depending on the ORE and input text. For some of these prepositional phrases it could be possible to make a relation from them, and to represent them in RDF by making a property representing the preposition (for example, we can perhaps imagine that the preposition "on" can be mapped to `dul:hasLocation` in some cases), and an object representing the rest of the prepositional phrase (if the rest of the phrase does not correspond to a single IRI, further triples may be necessary to describe it). These can then be linked to the node representing the "head" of the noun phrase. However, as with independently incoherent clauses, this type of non-verbal relation are not usually extracted by OREs and are therefore not solvable with the techniques discussed in this thesis.

4.1.3 Verb phrases

The most obvious and intuitive representation of verb phrases (relations) in RDF seems to be to represent them as object properties, considering that human languages express propositions with a subject, verb phrase, and object in some order, while RDF expresses facts in Subject-Property-Object triples. As seen in the "James Cameron directed 'Titanic'" example discussed in the previous chapter (section 3.1), this type of representation works well for simple facts, where the verb phrase consists of a single verb like "directed" or "divorced". Matters become muddled however when we take into consideration that many verb phrases are modified by auxiliary verbs and adverbs, for example in the verb phrases "would like", "may not go" or "has been accused". In these cases, the head verb could be mapped to a property in an ontology (although to the best of my knowledge, annotation tools that can perform such a task with high performance are not available, nor are ontologies dedicated to providing properties that can represent a wide coverage of

[2]Which can be viewed here: `https://cloud.ibm.com/docs/services/natural-language-understanding/entity-types-v1.html`

verbs [3]). Doing so however would by itself lose important information inherent in the verb phrase about tense[4], modality[5], aspect[6] and voice[7], unless the auxiliary verbs are also represented somehow.

Due to these two issues; the prevalence of multi-word verb phrases and verb-to-property ontologies not being available, the solution must involve modeling the verb phrase as an subject (node) rather than a property (edge). This is because the presence of these auxiliary verbs and adverbs modify the relation expressed by the main verb in important ways, even to the point of making the relation hypothetical or inverted (negated). Expressing these modifications (modality, tense etc.) along with a main verb in a single IRI would be highly impractical, which necessitates the representation of the relation as an individual that can be linked to other IRIs in order to represent these modifications.

There is an established design pattern for modeling n-ary relations involving the "individualization" of properties, namely the n-ary relation representation pattern suggested in a W3C report by Rector and Noy (2006) (which it must be noted is a work-in-progress, but does not appear to have been replaced). With this pattern, relations are represented as classes, and instances of these relations as instances of these classes (i.e. individuals). The intended use is to model n-ary relations as can be seen in the example of figure 4.1[8], but it would also be suitable for representing these modifications as RDF triples linked to the node representing the main verb.

There is then a matter of sophistication in this representation of modifications. A very simple and direct way of modeling could be to use properties like `ex:hasGrammaticalModifier` or similar along with an IRI representing the auxiliary verb or adverb. A complication here is that WordNet and related projects like BabelNet do not store auxiliary verbs in their knowledge bases. This introduces a problem of finding a suitable IRI, because alternative lexical knowledge bases with RDF representations seem scant. A

[3]WordNet, FrameNet, PropBank and related projects appear to solely represent verbs as individuals, not properties

[4]`https://glossary.sil.org/term/tense`

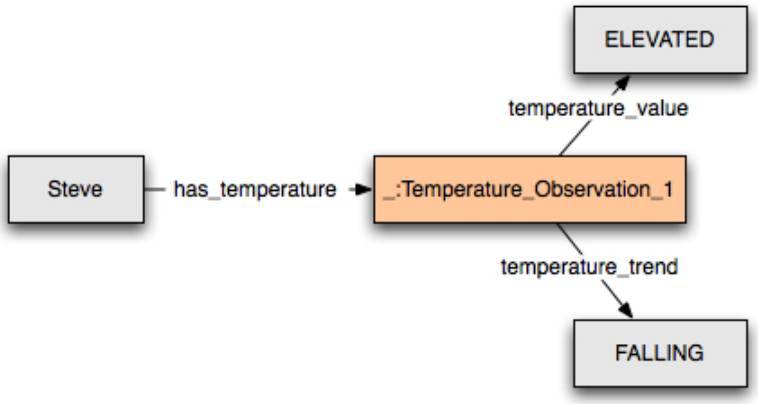
[5]`https://glossary.sil.org/term/mood-and-modality`

[6]`https://glossary.sil.org/term/aspect`

[7]`https://glossary.sil.org/term/voice`

[8]License: `https://www.w3.org/Consortium/Legal/2015/doc-license`

Figure 4.1: W3C *n*-ary relation example



weak alternative might be ConceptNet[9], a semantic graph which does include auxiliary verbs, however because its terms do not appear to be divided into senses on a per resource basis it lacks the precision of WordNet in that any IRI would link to a lemma (word form) with multiple possible interpretations rather than a single sense. Given that the auxiliary verbs (in English) are relatively few in number (though the exact number seems controversial, as the inclusion of e.g. "need" or "dare" varies (Carnie, 2013, p. 263)), a better solution might be to create a small domain ontology to represent them.

It is however questionable to which degree it is useful to ground these auxiliary verbs to indicate that they modify a main verb. A particular problem is that the same lemma of an auxiliary verb is not necessarily unambiguous. For example: "can" may indicate epistemic modality (that something is assessed as possible or likely) or deontic modality (that something is assessed as permissible or obligatory for the subject), while "will" can indicate epistemic modality, future tense or the habitual aspect (when combined with an adverbial phrase such as "typically" or "all the time"). In other words, the same auxiliary verb can modify a verb phrase in different ways, which means that a more sophisticated analysis than simply detecting the presence of an auxiliary verb is necessary. This would in turn necessitate a more sophisticated representation, such as IRIs that indicate the specific meaning contribution an auxiliary verb has in the context. Ontologies capable of describing these

[9] `http://conceptnet.io/`

meaning contributions do exist. An example is the Lexinfo ontology [10], which has both properties, classes and individuals that describe modifications like tense and mood. However, an automated way of determining the meaning contribution auxiliary verbs (and main verb inflections) have in a given input text would still be necessary, and is not something that any word sense disambiguation tool I am aware of can solve.

Frame Semantics provides some solutions to this problem, and some recent approaches within this paradigm represent several of these modifications present in multi-word verb phrases. Specifically, FRED (Gangemi et al., 2017) represents tense, modality and negation. For tense, they use a set of object properties inspired by Allen's interval Algebra relations [11], and classes and individuals representing tense, for example `fred:now_1` representing the present tense. To represent negation, they utilize the fact that their approach is oriented around the notion of events, which lets them give events negative truth values when negation is encountered. Interestingly, they make the claim "As for modality, OWL lacks formal constructs to allow the required expressivity.", and use only two individuals to represent modality: `boxing:Necessary` and "boxing:Possible". This representation distinguishes between epistemic (likelihood-oriented) and deontic (obligation-oriented) modality, but is not as expressive as natural language where there may be further distinctions between strong and weak modality. It is still adequate for representing the essential meaning carried by modal auxiliary verbs.

Martínez-Rodríguez et al. (2018) do not appear to consider tense, modality or negation from the examples they give, however they do provide a solution to the issue of voice. Through the semantic role labeling they perform, they are often given which part (subject or argument) of a relation is the "causer" ("agent" in the frame semantics vocabulary) and which is the "undergoer" ("patient"). In cases where this labeling is partial or missing, they use a heuristic to guess, with the default assumption that the agent and patient roles typically align with the subject and argument of ClausIE extractions. This essentially solves the problem of voice, as this default assumption is suitable for handling relations expressed by verb phrases in the active voice, while the detection of agents and patients accounts for passive

[10] `https://lexinfo.net/ontology/2.0/lexinfo.owl`

[11] `https://en.wikipedia.org/wiki/Allen%27s_interval_algebra`

voice verb phrases. Gangemi et al. (2017) accomplish similar results with FRED, but their work is less oriented around binary relations and they use a larger amount of roles from Frame Semantics, including "recipient" and "theme" etc.

These examples illustrate that a more elegant way of modeling auxiliary modifications to a relation is by taking them on a case by case basis, rather than rigidly representing them through a word sense resource for any occurring auxiliary verb or adverb in a verb phrase. While there is no large difference between, for example, the modality representations `ex:hasGrammaticalModifier ex:can_sense3(epistemic_modality)` versus `boxing:hasModality boxing:Possible`, it is unnecessary to represent voice in this manner when it should instead be used to determine the subject and argument of a relation. That is, if an input extraction used the passive voice, the subject and argument of that extraction should be inverted when represented in RDF. For example, even if ClausIE or another ORE produced the extraction "The pioneers; were attacked; by wolves", the "wolves" should be the subject and the "pioneers" the argument of the "attacked" relation in the resulting RDF graph. Similarly, representing tense and aspect in a unified manner is a better solution than separately, due to both simplicity and because these meanings are expressed not only through auxiliary verbs but also inflection of the main verb.

4.1.4 Definitiveness

Similar to how some auxiliary verbs should not be represented directly, definitiveness as expressed by articles ("the" and "a" in English) should ideally not be represented by itself (e.g. by using a "definitiveness" property), but rather be used to determine the appropriate way of representing the phrase following the article. If we consider the term "Catholic Church", the use of the definite article ("The Catholic Church") would normally refer to the worldwide religious denomination represented by the resource `http://dbpedia.org/page/Catholic_Church`, which would be an appropriate representation for the phrase by itself as a single RDF node. On the other hand, "a Catholic church" might refer to a non-identifiable entity (i.e. any local church community that happens to be Catholic), mak-

ing representing the phrase as a local or blank node that has properties linking it to "Catholic church" and "local church" appropriate. Although these two meanings expressed by articles are common in leads, this is by no means a full solution as to how definitiveness should be handled, as both the definite and indefinite article can have further meanings, particularly if more context is given (i.e. in longer phrases). For example, "The Catholic Church in Bucktown" might refer to the specific church represented by `http://dbpedia.org/page/St._Mary_of_the_Angels_(Chicago)` rather than `http://dbpedia.org/page/Catholic_Church`, while the indefinite article "a" can be used to identify something specific. An example might be "A famous church known for being the tallest in Iceland" which could be a description that serves as a kind of paraphrase for the specific church `http://dbpedia.org/page/Hallgr%C3%ADmskirkja`. In this case, assuming the annotation services can identify the entity referred to by the (para)phrase, it could be argued that the resource itself would suffice as a representation for the entire phrase, especially in this case as the relational information contained in the phrase is also implicit in the DBpedia resource (i.e. that the height of Hallgrímskirkja is the greatest of all Icelandic churches).

4.2 Relations between relations

The *n*-ary representation proposed in this thesis is not dissimilar to reification, although, as recommended by Rector and Noy (2006), the standard RDF reification vocabulary is not used due to the difference in semantics (in this case, between meta-statements about RDF triples, and disambiguating natural language). Aside from the representation of relations as nodes rather than edges however, there is an additional feature of reification that would be beneficial to this approach to knowledge graph extraction. With reification, a triple may itself be the subject or argument of another triple. At the same time, there is a class of sentence in natural language where a relational clause is used as the subject or object of the sentence.

An example could be the sentence "Air strikes by a Saudi-led military coalition killed at least 20 people attending a wedding in a village in northwestern Yemen late on Sunday, residents and medical sources said.". For this sentence, two extractions by an ORE might be: "Air strikes by a Saudi-led

military coalition; killed; at least 20 people attending a wedding in a village in northwestern Yemen late on Sunday" and "medical sources; said; Air strikes by a Saudi-led military coalition killed at least 20 people attending a wedding in a village in northwestern Yemen late on Sunday". Here, the first extraction is the argument of the second. These types of sentences are not uncommon, particularly in newspaper articles where information is often attributed to sources. An elegant way of representing such sentences would have the node representing the "killed" relation be the "argument node" of the node representing the "said" relation. Unlike with reification, these relation nodes are not quite the same thing as a reified triple (i.e. an entire statement), but the relation nodes still represent the events of the input text (i.e. a "said" relation about a "killed" relation).

A somewhat related problem is that sentences may contain adverbial clauses that modify relations, and sometimes these clauses themselves express relations. In other words, we have two relations with a connection, the meaning of which can vary depending on which subordinating conjunction (if any) is used to introduce the adverbial clause. For example, these connections may be temporal (e.g. "Before he rode hard for Texas, he saddled his horse") or purpose-oriented ("He bought a scratch ticket, so that he might win the lottery"). Clearly, these connections between relations should also be represented in an extracted knowledge graph. Unfortunately, this type of connection is not revealed through OREs like ClausIE (which typically interprets the adverbial clause as an optional argument), and therefore an automated way of detecting these connections requires some other technique. In terms of representation however, a straight-forward way would be to use properties from a small domain ontology capable of expressing the meaning of the various semantic types of adverbial clauses. These properties (which might e.g. be something like `loc:before` or `loc:possibleConsequence`) could be used to connect the relation nodes representing the two connected relations.

4.3 Limitations of binary relation extraction

As mentioned in the previous chapter, in my implementation ClausIE is set to only output binary relation extractions. This delimitation was made very

early in the project, and for what seemed to be sound reasons at the time: the majority of ORE work and assessment thus far has been focused upon binary relation extraction, ClausIE does not have n-ary extraction enabled by default or in the demo (perhaps suggesting that it is more experimental), and because the Semantic Web operates on triples (and at that time I had not realized that an ORE relation to RDF property mapping was impractical). However, with an n-ary RDF representation established as necessary, performing n-ary relation extraction becomes a boon rather than a hindrance.

This is particularly true for verbs that can take two objects. These ditransitive verbs, such as "gave" in the previously shown example extraction "The new treatment; gave; some of the patients better sleep", always give problematic arguments when extracted as binary relations. The two objects that make up the argument ("some of the patients" and "better sleep") cannot generally be decomposed either, as while the resulting extraction may be grammatically valid and semantically coherent, they may also be misleading or false. For example, "The new treatment; gave; better sleep" is misleading, as it may in fact be the case that some recipients got unchanging or worse sleep depending on the context of the input text. This problem is of course solved with an n-ary extraction such as "The new treatment; gave; some of the patients; better sleep", which removes the need for some other kind of processing of the problematic argument.

The n-ary relation extraction in ClausIE is not limited to these restrictive, ditransitive verbs however, as it is also used for sentences where there are "optional" arguments, typically prepositional phrases. For example, the sentence "Morocco inaugurated Africa's fastest train on Thursday" (where "inaugurate" is not ditransitive) will give the extraction "Morocco; inaugurated; Africa's fastest train; on Thursday". This is typically not an issue and in fact a good way to separate prepositional phrases from the object of the sentence. However, with ClausIE's division of prepositional phrases into mandatory and optional ones, it can sometimes result certain prepositional phrases being interpreted as parts of the primary object (argument) while they are in fact as optional as the others. For example, the "Dominika Cibulkova..." lead showed previously gives the following n-ary extraction with ClausIE "Japan 's Naomi Osaka; beats; Dominika Cibulkova in straight sets; at the Pan Pacific Open; her first match since winning the US Open". In this extraction, "in straight sets" is considered part of the first argument, in-

stead of an optional like "at the Pan Pacific Open". Despite this minor issue, n-ary relation extraction appears to be a solution to some of the previously discussed issues relating to long arguments, barring potential performance issues as the evaluation of n-ary relation extraction has been beyond the scope of this thesis.

4.4 Annotation performance issues

The performance of the implementation could likely be increased if the entire text of a newspaper article was annotated instead of simply the lead by itself. This would give more context to the annotation tools and should be expected to improve performance for both Babelfy (Moro et al. (2014) note that performance is lower on sentences compared to whole documents) and DBpedia Spotlight (Daiber et al. (2013) expect higher performance on larger text than on the short paragraphs used in their evaluation). Given that Babelfy and DBpedia Spotlight utilize fairly different techniques, this effect probably extends to many other disambiguation tools as well. However, this is not to say that disambiguation cannot be successful on shorter texts as well, as entity linking has been performed even on tweets (which may be shorter, and more importantly have a non-standard linguistic signature compared to newspaper leads) with promising results by, for example, Abel et al. (2012) (although it should be mentioned that they do not pursue full disambiguation of the entire text). Additionally, there is some suggestion that this effect may be limited to some extent, judging by a few informal tests where I used the entire newspaper article texts for annotation on a few of the input sentences that gave erroneous annotations (such as the Kate Middleton mention being mapped to "Kate Ramsay", or "Martin" in "Martin Fayulu" to a "The Simpsons" character). Expanding the annotation input did not resolve any of these errors, although much larger and more structured tests would be necessary to properly estimate the performance change.

It is also likely that significantly more of the extractions could be disambiguated if I used Babelfy for disambiguating the entirety of extractions rather than only the relation (verb phrase), as I inadvertently discovered because the MinIE branch shifts non-verb phrase text from the argument to the predicate. For the entity linking task it is unclear how strongly Babelfy

performs in comparison to DBpedia Spotlight, as different evaluations have conflicting results. Moro et al. (2014) find that while DBpedia Spotlight has a very high precision, the overall F1 scores on their evaluation datasets are roughly half of those achieved by their own Babelfy system due to low recall, which they note as being consistent with previous evaluations. Chabchoub et al. (2018) on the other hand found that DBpedia Spotlight had a relatively slight lead on Babelfy on most of their datasets. Some of this inconsistency may be attributed to the time elapsed between these evaluations, as DBpedia Spotlight has been in active development, and the different data sets used. Interestingly, both evaluations found that DBpedia Spotlight performed poorly at the KORE50[12] dataset, which consists of sentences not unlike the leads discussed in the previous chapter, although with a somewhat shorter average word length (14 versus 24 for the leads used for this thesis). Due to this uncertainty about which service is more trustworthy, it would be difficult to make a decision protocol for entity linking when DBpedia Spotlight and Babelfy came into conflicts with only two entity linking annotators instead of a larger ensemble, especially as their individual confidence scores cannot easily be equated (it is not uncommon to see erroneous annotations with high confidence scores in either service). However it is still likely that there would be frequent cases where one of the annotators provided an entity annotation that the other missed, and in that way the performance could have been increased, even with a naive decision protocol for conflicts (e.g. prioritizing one service over the other, or picking the annotation with the highest confidence score).

More importantly, the fact that Babelfy performs word sense disambiguation in addition to entity linking means that it can, as observed in the MinIE branch testing, at least partially solve the issue of disambiguating the previously mentioned "common words" ("tall","'deputy" etc.) that do not typically correspond to entities. There are many nouns and adjectives that it can link to word senses when entities are not found, such as adjectives like "early" or common nouns like "wife". On the other hand, Babelfy does not generally cover function words such as auxiliary verbs, determiners, conjunctions and so on, and as discussed the meanings of these words require some other form of analysis and representation.

[12]Available at `https://www.mpi-inf.mpg.de/departments/databases-and-information-systems/research/yago-naga/aida/downloads/`

All of these observations suggest that an ensemble of annotation services would not only benefit this approach to knowledge graph extraction, but in fact be a necessity in order to obtain the required annotation performance, both in terms of avoiding inaccurate annotations and avoiding partial annotation for extractions. With a more robust implementation of an ensemble annotation system using a greater amount of annotation services, more words in extractions could be disambiguated. With currently available services for both entity linking and word sense disambiguation, it does not seem outside the realm of possibility that even the entirety of words (except function words) in many leads could be annotated, with significantly fewer errors than in the current implementation. There are cases where erroneous, missing or partial annotations cannot be avoided however, for example when we have leads with entities like "Martin Fayulu" who do not have a resource available. In these cases the only options are to not annotate the entity, erroneously link it to the wrong resource, or to partially annotate it (e.g. assign "Martin Fayulu" the Person class).

4.5 Representation issues

An issue with this approach to knowledge graph extraction is whether or not the words in the input text become represented appropriately in an ontological sense when mapped to resources from annotation tools. In particular, many of these tools are oriented around the concept of named entities, which in and of itself is a hard to define and nebulous concept (Simon, 2013). A common definition of named entities is that they are "unique identifiers", but this seems to allow two broad categories of words and phrases; the proper names of things like people ("Barack Obama"), and expressions that signify some specific value in a system or scale, like measures or temporal expressions ("Apr 20, 1998" or "275 Kelvin"). Clearly, there is a significant difference between these two categories, which is reflected in that the second category is generally only used in the context of the named entity recognition task, and not in entity linking (although if the value expression is particularly noteworthy, as with e.g. "9/11" or "Fahrenheit 451", it might belong in both categories).

In my implementation I chose to disregard the second category of named

entities, by limiting Stanford NER to the Person, Location and Organization classes. In retrospect I think this was a valid choice to reduce scope somewhat, as the value expression classes are not useful for (partially) disambiguating words/phrases in the same manner as classes in the first category. These value expressions should ideally be represented using literals (e.g. dates can be represented using the datatype `xsd:dateTime`), and depending on the context of the input text (i.e. whether a value expression is used to describe something or is itself the subject or object of a clause), special properties like `:hasWeight` or `:onDate`. NER can however still be useful for the task of detecting these value expressions as they need to be identified in order to be represented, although NER may not necessarily be a state-of-the-art technique for this task.

Even if the use of "named entities" is restricted to only "proper names" of things, there still remains the issue of a semantic distinction between these named entities, and the word senses. While it is most often the case that "common" words are not disambiguated during the entity linking task, these words often do have an associated resource, but the candidates discovered by e.g. DBpedia Spotlight have such a low confidence that they are not included. For example, the common noun "man" in many sentences would not be given an annotation by DBpedia Spotlight, but if thresholds are reduced sufficiently it will be linked to `http://dbpedia.org/page/Man`. The same is the case for many other common nouns as well as adjectives. Further muddying the problem, WordNet has entries for many proper names, such as the meaning of of common names like "David", and notable people bearing it like the biblical King David (although these rarely seem to be used for annotation by Babelfy), and some word sense resources in BabelNet are connected to DBpedia resources (an example being "winter"). This kind of interconnectivity between lexical and encyclopedic resources is only likely to increase in the future with the expansion of Linked Open Data, which will for some words make the distinction between named entity and word sense almost irrelevant. As such, given that the confidence scores of e.g. DBpedia Spotlight and Babelfy are not very reliable, there is no hard line that distinguishes which words can be linked to "entities" versus word senses.

An example of where this question comes into play was shown in the previous chapter's discussion of output, in table 3.3. Here, the word "Korean" in "A Korean church" was mapped to a DBpedia resource about the Ko-

rean language, in other words an entity. Putting aside the inaccuracy of the annotation, this is problematic because the word "Korean" is in this input sentence used as an adjective, yet it is mapped to DBpedia as though it were a named entity (i.e. in the same way it would have been represented if it were a named entity and a proper noun). In terms of representing the meaning of the sentence, there does not seem to be any difference between associating "Korean" with an encyclopedic, entity-oriented knowledge base like DBpedia instead of a lexical, word sense-oriented knowledge base like BabelNet, assuming that either type of resource describes the meaning inherent in the word. At least this seems true given that, with the techniques considered in this approach to knowledge graph extraction, properties more specific than `loc:associatedWith` are generally not possible for adjectives. Because of this, the difference between a word like "Korean" being used a noun and an entity versus as an adjective comes down to whether it is the head word of a noun phrase (always a noun). If it is the head, the annotated resource that represents the meaning of the word (whether this is a word sense or entity resource) should become connected to the node representing the noun phrase with an `owl:sameAs` or `rdf:type` property. If it is not the head word, then regardless of whether or not the word is a noun or an adjective (and represented as a word sense or entity resource), it is connected to the "head node" of the noun phrase using the generic `loc:associatedWith` property.

This potential, unpredictable variability between using named entities and word senses to represent words does however introduce inconsistencies into the output graph. "Common words" will tend to be represented as word senses and "proper names" as entities, but as established this will only be a tendency since there may be many exceptions, depending on the implementation and the annotation services used. As such, there will be a difference in representation that does not in fact reliably signify an ontological difference (e.g. whether a word is used as a noun or an adjective), since whether a word is linked to a word sense or entity resource depends on the word, input text, and annotation service(s) in question and not necessarily the Part of Speech category (noun, adjective etc.) of the word. If this was found to be an important problem, some adaptations could be made to combat it. For example, Part of Speech tagging could be used to restrict valid annotations for adjectives to e.g. only adjectival word sense resources, and entity linking annotations could be restricted to proper nouns (although this might leave out many types of entities with "proper names" that are not noun phrases,

such as "9/11" or "I am Legend"). However, compared to problems like imprecise and partial annotation with state-of-the-art services, not necessarily having the most appropriate resources for all words seems like a relatively minor issue.

4.6 Determining the meaning of annotation mappings

In the pseudo-RDF triples output by the current implementation for evaluating ORE and annotation performance, DBpedia Spotlight annotations are linked to the words they are mapped to by the `owl:sameAs` property. In other words, entity linking annotations are seen as carrying the meaning that the annotated word represents the same entity as the resource it is mapped to. This is a valid interpretation in many instances, including most of the output discussed in chapter 3. As established however, it is not always the case that an entity linking annotation tool like DBpedia Spotlight only annotates named entities, and even among named entities there can be different levels of abstraction. For example, in one of the output examples shown in the previous chapter (shown in table 3.7) DBpedia Spotlight (erroneously) annotates "presidential election" with the `dbr:United_States_presidential_election` resource. This is perhaps not a named entity in the sense of being a "unique reference" as it is a class that may have many instances, for example `dbr:2016_United_States_presidential_election`. It is on the other hand less abstract than "presidential election" and is a kind of resource that an entity linking too like DBpedia Spotlight often uses in annotation. In this input sentence, where "presidential election" refers to the most recent instance as opposed to the class (albeit the recent Congolese election rather than the American), `owl:sameAs` does not appropriately represent the meaning of the annotation, which is in fact, in RDF terms, `rdf:type`.

This type of annotation is not uncommon, and is important to handle in order to provide proper representation for the head words of noun phrases in particular. However, it is not a straight-forward task to distinguish between whether a DBpedia resource is an "individual" (where an `owl:sameAs`

is likely to be appropriate), or a general class (where `rdf:type` is appropriate), because resources in DBpedia (that is, the DBpedia resources that are harvested from Wikipedia, not including the DBpedia ontology) are essentially represented as individuals rather than classes, as they typically have the `rdf:type owl:thing` (the set of all individuals). Consequently, there does not appear to generally be `rdf:type` or `rdfs:subClassOf` connections between different DBpedia resources in DBpedia itself (although it appears that linked knowledge bases like Wikidata and YAGO may have such hierarchies for at least some resources that are connected to DBpedia).

This means that determining whether an annotation signifies an `owl:sameAs` or `rdf:type` relationship to the word/phrase is not as simple as checking for the absence or presence of a property in DBpedia itself. A light-weight solution may be to query Wikidata and check for the existence of rdfs:subClassOf properties (based on the intuition that "true individuals" that are not classes should not have this property). This intuition might not hold for all resources however, and Wikidata may not necessarily have accurate or complete data for any given DBpedia resource. A more rigorous solution might then be to use a machine learning classifier with the feature set being the properties of the resource both in DBpedia, and in other knowledge bases it is linked to via `owl:sameAs` properties. This would be a complicated solution for what is ultimately a relatively minor issue, but it would be less vulnerable to possible absent resources, errors, or inconsistencies in specific knowledge bases (assuming the classifier is not overfitted to that knowledge base).

4.7 An alternative representation

An alternative to the n-ary RDF representation to handle compound verb phrases suggested in this thesis may be singleton properties as proposed by Nguyen et al. (2014). Their approach is to create a new unique property to represent any relation that is seen as an instantiation of a "generic" property. For example, `isMarriedTo#1` could be a specific (singleton) instance of the generic property `isMarriedTo`. Then this singleton instance may be described with further triples, unlike what is possible with a traditional, fixed property. They also discuss that alternatively, singleton properties could

be viewed as specializations rather than instantiations of generic properties. Their proposed representation is oriented around providing an alternative to traditional reification to represent metadata like triple provenance, but it could also be used to represent the kind of knowledge relevant to this thesis topic, such as tense and negation.

It is especially in regards to the possibility of creating specialized properties that this approach is appealing for representing ORE relations. As mentioned, some OREs other than ClausIE extract relations that are not only a compound verb phrase, by including prepositions, adverbs and even nouns sometimes. If we have a relation like "looked under", an elegant way of modelling it is to consider it a specialization of the generic property "look". Similarly, "looked under the bed of" could be a further specialization of "looked under". Assuming that these specialization properties were automatically generated, and inherently encapsulated all the information in the relation (tense, mood etc.), it might not even be necessary to use singleton properties (as instantiations of these specialized properties) because describing the relation further would be unnecessary.

On the other hand there are several major problems in the way of using this approach, which is why this thesis chose to consider n-ary representation instead. For one, while this approach might allow us to represent compound verb phrases as properties rather than subjects, it would still be necessary to model compound noun phrases as sub-graphs. Furthermore, human languages offer countless ways of expressing "relations", at least if our notion of an relation is expanded to be more than just the verb phrase. While certain relations would be statistically frequent in some input domain, others would be virtually unique and cause the generation of an enormous amount of specialized properties. For example, as shown in the previous chapter (table 3.8), MinIE extracts relations as specific as "beats Dominika Cibulkova in sets at". Most importantly however, as mentioned, neither ontologies nor annotation tools with a wide coverage for representing verbs as properties are available to the best of my knowledge. Workarounds based on heuristics and/or machine learning could perhaps overcome this problem by e.g. mapping arbitrary verbs to candidate LOD properties or converting WordNet verb resources to new properties, but such projects would be significant undertakings by themselves.

4.8 Interpreting quotation marks in leads

In the previous chapter, during the discussion of the first output lead (the first example of section 3.6.1), there was an issue with the ClausIE extractions where quotation marks were not retained after the processing of the input "A Korean church hiding from looming 'global famine' in Fiji is facing growing allegations of abuse." However, even if the quotes were retained they pose an interesting issue of interpretation. This seems to be related to the problem of polarity (i.e. the truth value of a proposition), but here these quotes do not indicate falsehood precisely, as it is the case that this Korean church is hiding in Fiji, though the reason, a looming "global famine", is suspect or subjective.

A simple solution could be to assign a `dul:hasQuality` `ex:IronicOrQuoted` to the node representing the quoted part of the text. More complex reasoning, such as determining who the quotation is attributed to, may be more elusive. This is because it is difficult to automatically differentiate between the different uses quotation marks can have, particularly in the context of leads, where they may indicate dubiousness/irony, or that the enclosed expression is attributed to a party mentioned in the text (in this case the Korean church) or a party outside of the text, whether directly or through some degree of paraphrase (which appears to be especially common in leads that are often short summaries). In this example, the quotes could even have two of these meanings at the same time; the quote is from the Korean church, and it is also dubious/ironic. Although there exists previous work on the issue of distinguishing between direct and indirect quotation based on the notion that different verbs preceding quotations (e.g "said" and "believes") indicate different levels of directness for attributions (Pareti et al., 2013), to my knowledge automatic means of distinguishing between different uses of quotation marks by themselves have not been developed.

4.9 Simple fact extraction versus full representation of natural language

A recurring problem in this thesis is that some nuance found in natural language is lost in translation in the conversion from input text to a knowledge graph. This seems to be an unavoidable consequence of the reliance of this approach upon external tools that were not developed for "full disambiguation" (i.e. discovering the meaning of every word in an input text). This is perhaps unavoidable with current technological limitations, as even richer forms of representation than the simple RDF representation suggested in this thesis, such as e.g. Discourse Representation Structures are not, to my understanding, expressive enough to model natural language without any loss of nuance at all.

This does raise the question of what the end goal of knowledge graph extraction is. Do we need to fully represent a natural language input in a machine-readable output in the pursuit of full natural language understanding, or is the extraction of "simple facts" (where it may be acceptable that some context is lost) is sufficient? Lost context can of course vary significantly in importance, from losing some nuance (e.g. through using a simpler model of modality, similar to the `boxing:Possible` and `boxing:Necessary` used by FRED (Gangemi et al., 2017)) to losing potentially important context (for example, in an ORE extraction, minimization such as what is performed by MinIE might make an extraction easier to ground but cause extractions to bear less of the meaning of the input text). As such, there is a sliding scale between "full natural language understanding" and "simple fact extraction".

Given that machine reading is still an open problem, the state of the art is not quite at the full natural language understanding end of the scale yet. This is especially the case for the technologies utilized in this approach, which are, by themselves, not well-suited to full representation of natural language. After all, as has been explored in this thesis, there are many kinds of phenomena in natural language text that an ORE like ClausIE is not capable of resolving because the triple extractions it outputs gives us too large chunks of text to work with in many cases. Along the same line, semantic annotators like DBpedia Spotlight are not able to disambiguate entire sentences, and are really only intended to annotate (notable) entities. Even a word

sense disambiguation tool like Babelfy cannot disambiguate all the words it encounters, and function words in particular are not addressed at all. As such, the combination of these types of tools are better suited to handling simple relational facts between notable entities. This is probably even more the case with many OREs other than ClausIE, which as established has a tendency to have less minimized extractions than most other alternatives like MinIE, Ollie, and Stanford OIE.

Even so, in this thesis I have for the most part considered the advantages and limitations of this approach to knowledge graph extraction from a very ambitious perspective, and essentially framed the discussion around what is missing from being able to fully convert natural language into a machine-readable form. This was a deliberate choice as I found quite early that, contrary to my expectations, natural language text, even within the context of newspaper leads, rarely conveys simple, relational facts in the form of "notable entity1; semantically meaningful verb; notable entity2" by themselves. On the contrary, it was far more common to see sentences with numerous prepositional phrases resulting in complicated arguments, and problematic clauses that do not convey relational information, such as those discussed in section 4.1. Due to working within this complex input domain, I found it more worthwhile to discuss these barriers that prevent full natural language representation in a knowledge graph, than to consider a simpler form of fact extraction where we accept that the assertions in our knowledge graph may not be certain, or at least lack context, because some of the meaning of the input text was left out.

4.10 Lessons learned

There are a few issues related to my personal assumptions and approach to the project work that I believe in retrospect were harmful to how much was accomplished on the implementation side. One such mistake was to limit ClausIE to binary relation extraction. As discussed, this was done in order to limit the thesis scope to the more well-studied binary relation extraction, but n-ary relation extraction would in fact have solved some of the issues related to overly specific arguments. Another early decision was to not use an ensemble implementation for the annotation task, in order to

limit the complexity of a challenging problem area. This proved to be a mistake, because as established an ensemble (depending on how extensive it was) would likely solve many of the performance issues related to partial or missing annotation.

A significant time sink was that the ultimate goal of the project was not determined early on. As mentioned in section 1.2, I initially worked with the basic assumption that disambiguating binary relations would essentially be the same as knowledge graph extraction, with the initial literature review being focused upon relation extraction and annotation. Most of the literature I found related to notions like relation-focused knowledge graph building/extraction and "ontologizing relations" were chiefly forms of ontology learning, and not directly relevant to the problem area I was working on. As the project work and implementation continued however, it became clear that ORE extractions could not directly be converted into RDF triples due to challenges like phrases with multiple words and the lack to verb-to-property annotation tools. This made it necessary to conduct a second, relatively time intensive literature review in order to investigate how to best represent extractions (that were sometimes quite close to the natural language input) in RDF. If on the other hand I had been more familiar with other forms of knowledge graph extraction from natural language before starting with my implementation, I might have had the time to get further towards outputting proper RDF knowledge graphs.

Lastly, it may have been a mistake to devote the time I did to the risk of homonymous relations. As discussed in section 3.5, this is a problem that, to my knowledge, cannot be definitively solved without modifying ClausIE itself. Furthermore, while it is in theory a potential risk, the likelihood of encountering the problem is unknown but probably quite low as I was unable to find an example of it in any of the leads I examined throughout the project. As such, this is probably an example of over-engineering with little utility, and the time spent on it could have been better spent on solving other problems.

4.11 Comparison to adjacent work

When comparing my approach to closely related previous work I can only do so at the level of my overarching approach, due to the fact that my system was only partially implemented and does not output valid knowledge graphs. The most related work is without a doubt the system of Martínez-Rodríguez et al. (2018), who as mentioned published around the midway point of the work process behind this thesis. They use many of the same technologies, and many of the solutions proposed in this thesis were also proposed by them. In particular, the type of knowledge graph they produce that has the relation represented as an individual is similar, as is the compound nature of noun phrases being represented through `:isPartOf` properties (which are very similar to the `loc:isAssociatedWith` properties proposed in this thesis).

There are however some significant differences as well, though they are not easy to determine due to Martínez-Rodríguez et al. (2018) not, as far as I have been able to tell, providing their source code or output data, and because they do not show many examples in the article itself. Most importantly, they use a narrower scope than the one used in this thesis, as they exclude sentences with a word length over 25 (which is more than the mean, 24, of the leads used here, but would still exclude several leads), as well as sentences that do not have a noun phrase - verb phrase - noun phrase structure. In contrast, more difficult sentences where different types of clauses serve as extraction arguments have been considered here, though certain types of independently incoherent clauses were determined to be unsolvable by this approach.

Their representation of compound noun phrases is also slightly simpler, as I have argued that head words signify a `owl:sameAs` or `rdf:type` relationship to the annotated resource. They use SRL to disambiguate predicates instead of Babelfy WSD like in this approach, and they have an ensemble implementation of entity linking tools that outperforms both Babelfy and DBpedia Spotlight according to their evaluation. They do not appear to use WSD for disambiguation at all (unless SRL is considered WSD of verbs), and while they perform NER as a preprocessing step, they do not appear to use it for the "partial annotation" of non-notable entities as proposed in this thesis. Also, while it is difficult to confirm, judging by the example they give it appears their knowledge graphs do (always) not retain all the information

in ClausIE extractions, as a prepositional phrase is unaccounted for.

As a representative of the Discourse Representation Theory and Frame Semantics approach to knowledge graph extraction, FRED (Gangemi et al., 2017) is not quite as closely related in terms of technology used, although the ambition and the form of the knowledge graphs produced is similar. Most significantly, they also use a n-ary relation representation, and they also use a growing suite of entity linking and WSD tools for disambiguation. The biggest difference is probably that they perform some ontology learning in addition to population, for example creating hierarchies of automatically generated resources by modeling compound nouns as specializations of the head word (e.g. `fred:First_Lady` being a `rdfs:subClassOf` of `fred:Lady`). Because of this, it seems their approach is less oriented towards disambiguating every part of an input text as some sense can be made of words without external knowledge bases. As mentioned they also solve many of the issues discussed previously about auxiliary verb meaning and negation. Furthermore, they support 48 languages through translation into English, though the resulting knowledge graphs have English labels (and presumably external disambiguation through e.g. WordNet and DBpedia is impacted). Ultimately, though their relations are discovered through SRL rather than ORE, the resulting graphs are not dissimilar.

Chapter 5

Conclusion

5.1 Future work

In terms of future work there are three main categories it can be divided into: problems that have not been considered because they are relatively speaking banal, problems that have been considered but also determined to be unsolvable without introducing new techniques to the approach, and problems that have not been considered because they required more research than time allowed. For the first category, the most important tasks are of course the practical implementation of the solutions I proposed in the previous chapter. These are specifically the knowledge graph representation solutions for noun phrases, verb phrases, and independently coherent clauses, as well as improvements to the annotation part of the system through the implementation of a more extensive ensemble of annotation tools, and the use of the entire newspaper article for context. This is not easy to sum up as it requires the implementation of many rule-based procedures, most of them fairly rudimentary.

An example is what might be called the canonization of extracted knowledge graphs. As has been discussed, ClausIE (and several other OREs) tend to give duplicate extractions, and in the case of ClausIE there is no indication given in the output as to which of the duplicates originate from the same

relation mention in the input text. As discussed in regards to the utility of my implemented solution to the risk of homonymous relations however, it seems to be a rare occurrence for there to be multiple identical predicates that do not correspond to the same relation mention in the text. As such, it would be a strong starting point to simply use string matching to combine duplicate relations, and to use the differences (which largely occur in the arguments, in the case of ClausIE) as a way of further decomposing the extraction segments. For example, for many inputs the smallest duplicate extraction corresponds to a "noun phrase - verb phrase - noun phrase" structure in the input sentence, while the remaining duplicates may successively add e.g. additional prepositional phrases to the argument.

For the second category, the most important problems are finding solutions for handling some of the discussed barriers to knowledge graph representation, primarily the disambiguation and representation of auxiliary verbs and main verb inflections, prepositional phrases, and various other challenging phenomena like non-finite clauses. For auxiliary verbs some possible solutions have already been considered, but it appears that SRL in particular may be a promising direction. Prepositional phrases are challenging as I am not aware of state-of-the-art techniques for decomposing them into some sort of standard,"relational" form, but as discussed it does appear like this is possible for at least a subset of them. Non-finite clauses however are a major challenge that I have no starting point for at this time.

The most important task of the third category may be the resolution of anaphora in text, particularly pronouns, which I have barely considered in this thesis. How significant and extensive this problem is has not been investigated, but a starting point to resolve anaphora may be the Stanford coreference resolution system[1], which is a module in the CoreNLP API I am already using for NER annotation. Other issues that I ran out of time to consider include nominalization in newspapers (an issue in that meaning is "transferred" from verbs (relations) into nouns), and the processing of larger texts where there may be more references between sentences. Lastly, a more thorough exploration of SRL and its utility as an alternative to ORE for the extraction (and disambiguation) of verbal relations would be useful.

[1] `https://nlp.stanford.edu/software/dcoref.shtml`

5.2 Summary

This thesis has explored an approach to knowledge graph extraction from natural language text through the use of Open Relation Extraction systems and semantic annotation tools, and presented a partial implementation of the approach. In chapter 2 the important research areas, utilized technologies, and similar approaches to knowledge graph extraction were presented. In chapter 3, the overarching approach was first presented as a step-based procedure, with special consideration given to challenges related to RDF representation. Then, an ideal, manual example of the form of knowledge graph extraction proposed by this approach was given. In the following sections, implementation choices such as which tools were used and why was explained, and the implemented system itself was described in terms of its most important modules and what modifications were made to the default options of the utilized tools. In the final section of that chapter, the system was assessed in a small, qualitative evaluation of the performance of both the OREs ClausIE and MinIE, and the annotation tools DBpedia Spotlight, Babelfy, and the Stanford NER classifier.

Finally, in chapter 4, a wide variety of issues were discussed, and possible solutions were proposed for some of them, while others were deemed unsolvable with the current techniques included in the approach. These issues particularly relate to problems that were not solved in the implementation, namely: RDF knowledge graph representation, which is considered in terms of clauses, noun phrases, verb phrases, and definitiveness (grammatical articles); relations that have as their argument/object another relation; the limitations of binary relation extraction; and improved annotation performance through an ensemble implementation. The discussion then turned to more abstract and less immediate problems: a possible alternative relation representation; the appropriateness of word senses versus entity resources for representing natural language text; the difficulty of determining the meaning of quotation marks; the sliding scale of ambition between simple fact extraction versus full natural language representation; and various early assumptions that proved detrimental to how much of the approach I had time to implement. Lastly, the approach was compared to two of the most adjacent approaches to knowledge graph extraction from text.

5.3 Conclusion

In the introduction of this thesis, I posed the following research questions:

1. Is knowledge graph extraction from natural language text attainable through combining semantic annotation and existing Open Relation Extraction systems?
2. Is Open Relation Extraction a promising research direction for knowledge graph extraction from natural language text?

In answer to the first question, I believe knowledge graph extraction to be attainable with this approach. My implementation did not get to that stage, however I believe the knowledge graph representation I proposed to be feasible and implementable with the tools already included in the approach. If these solutions were implemented however, there would still as explained remain several linguistic phenomena that are not accounted for, and which cannot be represented with the proposed approach as of this thesis. As such, the approach is by no means a complete solution to knowledge graph extraction, and far more work is needed to get closer to the ultimate goal of full natural language representation.

As for the second question, I also believe Open Relation Extraction to be a paradigm that shows promise for this type of knowledge graph extraction. However, as has been a recurring problem in this thesis, current state-of-the-art OREs have a tendency to produce extraction segments that are too large chunks of texts for them to easily be disambiguated. A consequence of this is that other techniques are required to further decompose and make sense of these overly large chunks. In addition, the performance itself remains a problem, as erroneous extractions are quite frequent on the input domain of leads. Even so, OREs have many strengths, like being domain-independent and having fast processing times, and though their extractions sometimes have issues, relations are rarely completely missed. Also, it is likely that the mentioned issues will be improved upon to some extent in future research endeavors within the ORE paradigm. I therefore believe that OREs show promise as an underlying technique for knowledge graph extraction from natural language texts, particularly for more shallow extraction that is closer

to the simple fact extraction end of the scale than full natural language representation.

Bibliography

Abel, Fabian et al. (2012). "Semantics + filtering + search = twitcident. exploring information in social web streams". In: *23rd ACM Conference on Hypertext and Social Media, HT '12, Milwaukee, WI, USA, June 25-28, 2012*, pp. 285–294. DOI: 10.1145/2309996.2310043. URL: https://doi.org/10.1145/2309996.2310043.

Allemang, Dean and James Hendler (2008). *Semantic Web for the Working Ontologist: Effective Modeling in RDFS and OWL*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc. ISBN: 0123735564, 9780123735560.

Angeli, Gabor, Melvin Johnson Premkumar, and Christopher D. Manning (2015). "Leveraging Linguistic Structure For Open Domain Information Extraction". In: *ACL*.

Banko, Michele, Michael J. Cafarella, et al. (2007). "Open Information Extraction from the Web". In: *IJCAI 2007, Proceedings of the 20th International Joint Conference on Artificial Intelligence, Hyderabad, India, January 6-12, 2007*, pp. 2670–2676. URL: http://ijcai.org/Proceedings/07/Papers/429.pdf.

Banko, Michele and Oren Etzioni (2008). "The Tradeoffs Between Open and Traditional Relation Extraction". In: *Proceedings of ACL-08: HLT*. Columbus, Ohio: Association for Computational Linguistics, pp. 28–36. URL: http://aclweb.org/anthology/P08-1004.

Berners-Lee, Tim (1999). *Weaving the Web: The Original Design and Ultimate Destiny of the World Wide Web by Its Inventor*. 1st ed. HarperCollins Publishers.

Bizer, Christian et al. (Sept. 2009). "DBpedia - A Crystallization Point for the Web of Data". In: *Web Semantics: Science, Services and Agents on the World Wide Web* 7, pp. 154–165. DOI: 10.1016/j.websem.2009.07.002.

Carnie, Andrew (2013). *Syntax : a generative introduction*. 3rd ed. Wiley-Blackwell.

Chabchoub, Mohamed, Michel Gagnon, and Amal Zouaq (2018). "FICLONE: Improving DBpedia Spotlight Using Named Entity Recognition and Collective Disambiguation". In: *OJSW* 5.1, pp. 12–28. URL: http://nbn-resolving.de/urn:nbn:de:101:1-201808051930147807663.

Corro, Luciano Del and Rainer Gemulla (2013). "ClausIE: clause-based open information extraction". In: *22nd International World Wide Web Conference, WWW '13, Rio de Janeiro, Brazil, May 13-17, 2013*, pp. 355–366. DOI: 10.1145/2488388.2488420. URL: http://doi.acm.org/10.1145/2488388.2488420.

Daiber, Joachim et al. (2013). "Improving efficiency and accuracy in multilingual entity extraction". In: *I-SEMANTICS 2013 - 9th International Conference on Semantic Systems, ISEM '13, Graz, Austria, September 4-6, 2013*, pp. 121–124. DOI: 10.1145/2506182.2506198. URL: https://doi.org/10.1145/2506182.2506198.

Dutta, Arnab, Christian Meilicke, and Heiner Stuckenschmidt (2015). "Enriching Structured Knowledge with Open Information". In: *Proceedings of the 24th International Conference on World Wide Web*. WWW '15. Florence, Italy: International World Wide Web Conferences Steering Committee, pp. 267–277. ISBN: 978-1-4503-3469-3. DOI: 10.1145/2736277.2741139. URL: https://doi.org/10.1145/2736277.2741139.

Fillmore, Charles J. (1976). "FRAME SEMANTICS AND THE NATURE OF LANGUAGE*". In: *Annals of the New York Academy of Sciences* 280.1, pp. 20–32. DOI: 10.1111/j.1749-6632.1976.tb25467.x. eprint: https://nyaspubs.onlinelibrary.wiley.com/doi/pdf/10.1111/j.1749-6632.1976.tb25467.x. URL: https://nyaspubs.onlinelibrary.wiley.com/doi/abs/10.1111/j.1749-6632.1976.tb25467.x.

Gangemi, Aldo et al. (2017). "Semantic Web Machine Reading with FRED". In: *Semantic Web* 8.6, pp. 873–893.

Gashteovski, Kiril, Rainer Gemulla, and Luciano Del Corro (2017). "MinIE: Minimizing Facts in Open Information Extraction". In: *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017*, pp. 2630–2640. URL: https://aclanthology.info/papers/D17-1278/d17-1278.

Kumar Dutta, Arnab (Feb. 2014). "Integration of large scale knowledge bases using probabilistic graphical models". In: pp. 643–648. DOI: 10.1145/2556195.2556202.

Manning, Christoper et al. (Jan. 2014). "The Stanford CoreNLP Natural Language Processing Toolkit". In: DOI: 10.3115/v1/P14-5010.

Martínez-Rodríguez, José-Lázaro, Ivan López-Arévalo, and Ana B. Rios-Alvarado (2018). "OpenIE-based approach for Knowledge Graph construction from text". In: *Expert Syst. Appl.* 113, pp. 339–355. DOI: 10.1016/j.eswa.2018.07.017. URL: https://doi.org/10.1016/j.eswa.2018.07.017.

Mausam et al. (2012). "Open Language Learning for Information Extraction". In: *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*. EMNLP-CoNLL '12. Jeju Island, Korea: Association for Computational Linguistics, pp. 523–534. URL: http://dl.acm.org/citation.cfm?id=2390948.2391009.

Miller, George A. (Nov. 1995). "WordNet: A Lexical Database for English". In: *Commun. ACM* 38.11, pp. 39–41. ISSN: 0001-0782. DOI: 10.1145/219717.219748. URL: http://doi.acm.org/10.1145/219717.219748.

Moro, Andrea, Alessandro Raganato, and Roberto Navigli (2014). "Entity Linking meets Word Sense Disambiguation: a Unified Approach". In: *TACL* 2, pp. 231–244. URL: https://tacl2013.cs.columbia.edu/ojs/index.php/tacl/article/view/291.

Navigli, Roberto and Simone Paolo Ponzetto (2012). "BabelNet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network". In: *Artif. Intell.* 193, pp. 217–250. DOI: 10.1016/j.artint.2012.07.001. URL: https://doi.org/10.1016/j.artint.2012.07.001.

Nguyen, Vinh, Olivier Bodenreider, and Amit Sheth (2014). "Don'T Like RDF Reification?: Making Statements About Statements Using Singleton Property". In: *Proceedings of the 23rd International Conference on World Wide Web*. WWW '14. Seoul, Korea: ACM, pp. 759–770. ISBN: 978-1-4503-2744-2. DOI: 10.1145/2566486.2567973. URL: http://doi.acm.org/10.1145/2566486.2567973.

Niklaus, Christina et al. (2018). "A Survey on Open Information Extraction". In: *CoRR* abs/1806.05599. arXiv: 1806.05599. URL: http://arxiv.org/abs/1806.05599.

Pareti, Silvia et al. (2013). "Automatically Detecting and Attributing Indirect Quotations." In: *EMNLP*. ACL, pp. 989–999. URL: http://dblp.uni-trier.de/db/conf/emnlp/emnlp2013.html#ParetiOKCK13.

Rector, Alan and Natasha Noy (Apr. 2006). *Defining N-ary Relations on the Semantic Web*. W3C Note. W3C. URL: http://www.w3.org/TR/2006/NOTE-swbp-n-aryRelations-20060412/.

Rose Finkel, Jenny, Trond Grenager, and Christoper Manning (Jan. 2005). "Incorporating Non-local Information into Information Extraction Systems by Gibbs Sampling". In: DOI: 10.3115/1219840.1219885.

Rumshisky, Anna and Olga Batiukova (Aug. 2008). "Polysemy in verbs: systematic relations between senses and their effect on annotation". In: *Coling 2008: Proceedings of the workshop on Human Judgements in Computational Linguistics,* pp. 33–41. URL: http://www.cs.brandeis.edu/~arum/publications/HJCL08Rumshisky.pdf.

Schutz, Alexander and Paul Buitelaar (2005). "RelExt: A Tool for Relation Extraction from Text in Ontology Extension". In: *International Semantic Web Conference*.

Sheth, Amit and Krishnaprasad Thirunarayan (2012). *Semantics Empowered Web 3.0: Managing Enterprise, Social, Sensor, and Cloud-based Data and Services for Advanced Applications*. Morgan & Claypool Publishers. ISBN: 1608457168, 9781608457168.

Simon, Eszter (2013). "Approaches to Hungarian Named Entity Recognition". PhD thesis. PhD School in Cognitive Sciences, Budapest University of Technology and Economics. Chap. 2. URL: http://www.cogsci.bme.hu/~ktkuser/PHD_iskola/dissertations/20131011_Simon_Eszter/ertekezes.pdf.

Sorokin, Daniil and Iryna Gurevych (Jan. 2017). "Context-Aware Representations for Knowledge Base Relation Extraction". In: pp. 1784–1789. DOI: 10.18653/v1/D17-1188.

Verburg, Jochem, Mena Badieh Habib, and Maurice van Keulen (Oct. 2015). "Handling uncertainty in relation extraction: a case study on tennis tournament results extraction from tweets". Undefined. In: *Proceedings of the 8th International Conference on Knowledge Capture (K-CAP 2015)*. 10.1145/2815833.2816960. United States: Association for Computing Machinery (ACM), Article No. 26. ISBN: 978-1-4503-3849-3. DOI: 10.1145/2815833.2816960.

Zouaq, Amal, Michel Gagnon, and Ludovic Jean-Louis (2017). "An assessment of open relation extraction systems for the semantic web". In: *Inf.*

Syst. 71, pp. 228–239. DOI: 10.1016/j.is.2017.08.008. URL: https://doi.org/10.1016/j.is.2017.08.008.