Journal of Universal Computer Science, vol. 24, no. 1 (2018), 2-24
submitted: 23/2/17, accepted: 26/1/18, appeared: 28/1/18 © J.UCS

GerIE - An Open Information Extraction System for the German Language

Akim Bassa
(Unycom GmbH, Graz, Austria
akim.bassa@unycom.com)

Mark Kröll
(Know-Center, Graz, Austria
mkroell@know-center.at)

Roman Kern
(Graz University of Technology, Graz, Austria
Know-Center, Graz, Austria
rkern@tugraz.at)

Abstract: Open Information Extraction (OIE) allows to extract relations from a text without the need of domain-specific training data. To date, most of the research on OIE has been focused to the English language and little or no research has been conducted on other languages, including German. To tackle this problem, we developed GerIE, an OIE system for the German language. We surveyed the literature on OIE in order to identify concepts that may apply to the German language. Our system is based on the output of a German dependency parser and a number of handcrafted rules to extract the propositions. To evaluate the system, we created two dedicated datasets: one derived from news articles and the other devised from texts from an encyclopedia. Our system achieves F-measures of up to 0.89 for correctly-preprocessed sentences.

Key Words: open information extraction, fact extraction, German language
Category: I.2.7, I.7.m

1 Introduction

Traditional *Information Extraction* (IE) predominantly follows a supervised approach, which requires the desired relationships to be specified in advance via a large number of training examples. This involves human annotators and the manual labour scales linearly with the number of specified relations. Since this is not scalable to a large and heterogeneous corpus as the web, [Banko et al., 2007] introduced the concept of *Open Information Extraction* (OIE), which aims to enable domain- and relation-independent discovery of relations. The idea is to learn how relations are expressed in general in written text, using unlexicalised features, e.g. part-of-speech tags or dependency relations. However, these general features are still language specific. Generally, the input of an OIE system is a single sentence for which the OIE system produces a set of relations: relational tuples often termed propositions and representing facts. For example, the sentence "*A. Einstein, who was born in Ulm, has won the Nobel Prize*" may yield tuples like ("*A. Einstein*", "*has won*", "*the Nobel Prize*"), ("*A. Einstein*", "*was born*") and ("*A. Einstein*", "*was born in*", "*Ulm*"). OIE has many useful applications, including question answering, opinion mining, fact checking and semantic full-text search.

Due to its potential usefulness, OIE has received increasing attention in recent years, and continuous research has improved performance of OIE systems constantly. Nearly all of the work done so far has focused on the English language. Although German is a major language with about 90 million native speakers, only little research effort has been dedicated to building German OIE systems. Since OIE systems use language specific features, English systems are not directly applicable for German. Due to a smaller target audience in comparison with English, fewer resources and tools are available for German. To fill this gap, we developed an OIE system for the German language that is not based on existing systems. Additionally we designed our system in such a way to allow to analyse its own behaviour.

To that end, we first surveyed existing OIE systems and examined if and to what extent these systems can be applied to the German language. This included the investigation of: i) if the existing methods can be applied on the German grammar, and ii) if similar performance can theoretically be expected, and iii) which (German) preprocessing tools are expected to deliver the satisfying results. We used the gained insights in order to develop a German OIE system, which is not based on existing systems, which we named GerIE. In addition we took care to handle special cases, which may cause problems in succeeding processing steps. For example, we excluded propositions derived from direct speech by default since such information should not be treated identical to other propositions in many use-case scenarios. To evaluate GerIE's performance, we created a German OIE evaluation dataset based on two domains, news and encyclopedia. The published results and datasets should provide a benchmark for future systems.

2 Background

In order to develop a OIE system for the German language we placed an emphasis on the initial in-depth analysis of the existing literature. We focused on the body of research published on Open Information Extraction, as well as related research to acquire a good understanding of the various algorithmic approaches, their key characteristics, their performance, and the potential to be applied on the German language.

[Piskorski and Yangarber, 2013] define *Information Extraction* (IE) as follows: "*The task of Information Extraction is to identify instances of a particular* *pre-specified class of entities, relationships and events in natural language texts, and the extraction of the relevant properties (arguments) of the identified entities, relationships or events.*" Many of the existing systems approach the task of IE as a supervised machine learning task, rather than manual rules, heuristics and unsupervised methods. With that regard, (ground truth) data sets have to be first established and later used for training and testing. This is a taxing job, since the annotation process has to be conducted manually by experts. Moreover, in order for the supervised machine learning algorithms to produce good results, the data sets have to exceed a certain size. Another drawback of this approach is that for each new domain and type of entity/relation this process must be repeated. Therefore, transferring a traditional IE model to a new domain requires much effort.

Open Information Extraction (OIE) was originally introduced in 2007 by [Banko et al., 2007] to tackle the challenge of scaling up IE to the web. The web has several properties that make traditional IE inapplicable. First, it contains all possible kinds of domains and article types, whereas most of IE work is concentrated on specific domains. Secondly, the set of relations of interest on the web is often unknown and their number is high, which also makes the use of IE with its predefined relations impractical. Lastly, the web contains billions of documents meaning that a highly scalable extraction techniques must be applied.

We surveyed algorithms for OIE, which are briefly described in the following section. An overview of the covered systems is provided in Table 1. Most of the proposed systems target the English language, and only few systems have been developed for other languages. The majority of systems make use of grammatical dependencies, and *Part-of-Speech* (POS) is the next popular input. Some systems additionally take *noun phrases* (NP) produced by a chunking parser, as input. *Named entity recognition* (NER) was also considered. Only one system bases its analysis on the results of a *Semantic Role Labelling* (SRL), which is typically associated with a high runtime complexity.

2.1 English OIE Systems

[Banko et al., 2007] proposed the *TextRunner* system, which used a Naive Bayes classifier to train a model based on shallow features and could then extract triples in a single pass over a corpus. *Wanderlust* [Akbik and Broß, 2009] was the first to utilise deep syntactic parsing in the form of link grammar [Sleator and Temperley, 1995]. This system automatically learned 46 patterns from an annotated corpus of 10,000 sentences.

[Wu and Weld, 2010] proposed the systems WOE^{pos} and WOE^{parse}. The former one works with shallow features in combination with Conditional Random Fields, and the latter one uses features from dependency parse trees in combination with a pattern learner to establish whether the shortest path between

System	**Year**	**Input**	**Pattern Creation**	**Trained**	**Languages**
TextRunner	2007	PoS, NP-chunks	Naive Bayes classifier	✓	English
StatSnowball	2009	PoS	Markov logic networks	✓	English
Wanderlust	2009	link grammar	pattern learner	✓	English
WOE^{parse}	2010	dependencies	pattern learner	✓	English
WOE^{pos}	2010	PoS, NP-chunks	CRF	✓	English
SRL-IE	2010	SRL	rule-based conversion		English
ReVerb	2011	PoS, NP-chunks	syntactic and lexical constraints + logistic regression classifier	✓	English
DepOE	2012	dependencies	hand-crafted rules		English, Romance languages
Kraken	2012	dependencies	hand-crafted rules		English
OLLIE	2012	dependencies	Open Pattern Learning		English
Patty	2012	dependencies	frequent itemset mining	✓	English
ClausIE	2013	dependencies, constituents	hand-crafted rules		English
LSOE	2013	PoS	Qualia Structure Based Patterns		English
CSD-IE	2013	constituents	hand-crafted rules		English
TK	2013	dependencies	SVM tree kernels	✓	English
ExtrHech	2013	PoS	hand-crafted rules		Spanish
ReNoun	2014	dependencies, NP-chunks, NER	pattern learner		English
SCOERE	2014	dependencies, constituents, NER	CRF	✓	Chinese
BoostingOIE	2014	PoS, NP-chunks	hand-crafted rules		English
ArgOE	2015	dependencies	hand-crafted rules		English, Portugese, Spanish
PropsDE	2016	dependencies	(transferred) hand-crafted rules		German
GerIE	2018	dependencies	hand-crafted rules		German

Table 1: Overview of existing OIE systems together with their characteristics, i.e. the expected input, how the patterns are created, whether they require annotated training data as well as the designated language - Non-English parsers are in italics.

two noun phrases expresses a relation. Unlike *TextRunner*, they have a high-quality training corpus obtained from Wikipedia by automatically matching the infobox attribute values to the corresponding sentences. By directly comparing WOE^{parse} and WOE^{pos} they showed that, unlike shallow features, dependency parse features improve the precision and recall.

StatSnowball [Zhu et al., 2009] used shallow parsing techniques since they are less expensive and more robust. They viewed the pattern selection as a problem of structure learning in Markov logic networks [Kok and Domingos, 2005].

[Fader et al., 2011] proposed *ReVerb*, the successor of TextRunner that aims to prevent TextRunner's frequent errors, i.e. incoherent and uninformative extractions. To that end, they articulated syntactic and lexical constraints on binary, verb-based relation phrases, which yielded more informative relations.

[Christensen et al., 2010] observed that semantically labelled arguments in a sentence often match the arguments in the OIE extractions, and the verbs often correspond to the OIE relations. Based on this observation, they proposed a system (*SRL-IE*) that converts the output of a Semantic Role Labelling system to OIE facts. The downside of this approach is low precision for highly redundant text (as expected for web content).

[Akbik and Löser, 2012] developed *Kraken* using their previous work Wanderlust demonstrating that a limited number of patterns can suffice for deep syntactic parsed sentences. As a result, Kraken applies dependency parsing in combination with hand-crafted rules.

Following the trend of using dependency-parse features, [Schmitz et al., 2012] created *OLLIE*, the successor of ReVerb. OLLIE uses high precision tuples of ReVerb to bootstrap a training set for its pattern learner. In contrast to previous OIE systems, it also extracts relations mediated by nouns or adjectives. Additionally, it includes essential contextual information in the extractions (e.g. when the relation is within a belief or conditional context). [Xu et al., 2013] adapted a SVM dependency tree kernel model [Moschitti, 2006] for their system (*TK*), achieving results superior to OLLIE and ReVerb.

[Nakashole et al., 2012] applied OIE to their system (*Patty*) in order to organise the extracted relations into synsets and finally create a taxonomy, similar to WordNet [Fellbaum, 1998]. They used dependency parsing and named entity recognition to extract a relation and assign it a to pattern synsets, such as: `<Politician> politician from <State>`

[Del Corro and Gemulla, 2013] introduced the clause-based approach implemented in *ClausIE*, separating the detection and generation of facts. They worked with hand-crafted rules utilising the dependency structure of a sentence. Furthermore, they identified the type of clauses according to the grammatical function of its constituents and exploited this knowledge to generate multiple propositions out of a single clause.

LSOE [Castella Xavier et al., 2013] was the first system using hand-crafted rules for POS-tagged texts, utilising the Qualia structure [Cimiano and Wenderoth, 2005], which provides additional information about the role of words in a sentence.

[Bast and Haussmann, 2013] applied a technique termed contextual sentence decomposition, to decompose a sentence into pieces that semantically belong together. They employ rules to convert the output of a constituency parser to a Sentence-Constituent-Identification tree and showed that the new representation allows for easy extraction of various types of relations for their system (*CSD-IE*).

While most of the OIE systems focused only on verb-mediated relations, [Xavier and de Lima, 2014] proposed a method (*BoostingOIE*) to enrich a text in such a way that common OIE systems could also extract noun compounds (`glass vase`) and adjective-noun pairs (`raw food`). The idea was to replace the phrase with a modified phrase containing a verb that has the same meaning. Finally, *ReNoun* [Yahya et al., 2014] focused entirely on the extraction of noun-mediated relations, accounting for the lack of work done in this area.

2.2 Other Languages

[Gamallo et al., 2012] showed that their system, *DepOE*, made OIE based on dependency trees suitable for languages other than English. They used a multilingual parser with a common output tagset for the supported languages (English and Romance languages).

The improved multilingual OIE system *ArgOE* [Gamallo and Garcia, 2015] attempted to be more open to various dependency parsers by using the CoNLL-X format. Due to the gap in performance of the multilingual parsers compared with the English parsers, these results were not as good as the previously published results.

[Falke, 2016] showed that an OIE system for the English language can be applied to German, when porting the rules from English to a new target language. They named their system *PropsDE.*

[Zhila and Gelbukh, 2013] described the Spanish system *ExtrHech*, working with POS-tagged input and semantic constraints, demonstrating that for Spanish this approach delivers similar results as for English. [Wang et al., 2014] applied OIE on Chinese articles (*SCOERE*), but chose to use a semi-supervised method and focused on a fixed set of entities (i.e. person, organisation, location and time).

2.3 Target Properties

Apart from the three main target properties of an OIE system (domain independence, automation and efficiency), there is a number of additional, noteworthy criteria mentioned in the literature.

2.3.1 Minimality

CSD-IE [Bast and Haussmann, 2013] aims to extract relations that are minimal. This means that a relation should not contain other relations. In the sentence "*President Barack Obama was born in the USA.*" two minimal extractions would be *[Barack Obama][is][President]* and *[Barack Obama][was born][in the USA]*. A non-minimal fact would be *[President Barack Obama][was born][in the USA]*, which should be avoided. Minimality is also required when the information of a separated fact cannot be excluded from another fact. In this case, the excluded fact may be referenced:

#1: *[Obama][said][that #2]*

#2: *[America][is not][a Christian nation]*

The authors provided two reasons why minimality should be incorporated: i) the use of extracted relations in semantic full-text search, and ii) easier transformation of OIE triples into disambiguated relations within a formal ontology.

2.3.2 Levels of Granularity

Levels of granularity describe how the extracted fact is stored or provided for further use. For example the reported tuples could consist only of the surface forms, or have a more rich representation also including the respective lemmas or grammatical information. Depending on the consumer of the OIE output such information might be highly or not at all relevant. [Gamallo et al., 2012] emphasise that "substantial postprocessing is needed to derive relevant linguistic information from the tuples", which is why it is not enough to output triples in textual form. To that end, DepOE [Gamallo et al., 2012] additionally provides syntax-based information, POS tags, lemmas and heads. The successor of DepOE, ArgOE [Gamallo and Garcia, 2015], also has this property.

2.3.3 Separation of Detection and Representation

OIE systems decoupling the process of detection of relations and their representation allow for a higher flexibility. This approach was first presented by [Del Corro and Gemulla, 2013] and implemented in their system ClausIE. For a detected clause *[Gandhi was born in India.]* different propositions may be generated, e.g. *[Gandhi][was born][]*, *[Gandhi][was][born in India]*, etc. CSD-IE uses a similar approach by decomposing sentences into their basic constituents and afterwards creating triples out of these constituents.

2.3.4 Separation of Relation Detection and Relation Extraction

[Xu et al., 2013] addressed the task of determining whether a relation between a pair of entities in the sentence exists before extracting the information. The authors pointed out that previous OIE systems ignored this question and reported conflicting results. This task is difficult since it is not always clear what a relation constitutes. For instance in the phrase "*Newton eats apple pie.*": is there a relation between *Newton* and *apple*?

2.4 Summary of OIE approaches

The majority of recent OIE systems rely on grammatical dependencies as their main features. Therefore, the usage of grammatical dependencies appears to be a promising starting point when developing an OIE for a new language. But this is limited to cases, where mature deep parsing tools are available for the new language, which is currently restricted to a small sub-set of languages. Fortunately, there are a number of parsing libraries available for the German language.

In terms of pattern creation there is a higher degree of diversity, which generally can be divided into two main categories: hand crafted rules and machine learning techniques. Both approaches are associated with their distinctive advantages and disadvantages. While machine learning approaches typically offer good performance, the learnt models tend to be difficult to comprehend. Hand crafted rules are a preferred choice, if the design goal is interpreting the result of the algorithm by human experts. Therefore, if an OIE system is developed for a new language, hand crafted rules allow for a more in-depth analysis of the results.

2.5 German vs. English

Although English and German share many characteristics, they have a number of key differences. In fact, some of these differences prevent English OIE systems from being directly applied to a German text.

Alphabet

In addition to 26 Latin based letters in English, German has ß (ligature of s and z) and *Umlaute* (ä, ö, ü). This should generally pose no problems since OIE system typically operate on a word based level.

Capitalisation

Another obvious difference is the capitalisation of words in sentences. While in English only proper nouns start with an upper case letter, in German this is also the case for common nouns. This is expected to have an impact on the Part-of-Speech tagging methods of the parser component and thus may have an influence on the quality of the OIE system.

Gender

Unlike English nouns, German nouns are either masculine, feminine or neutral. The article depends on the gender of the noun, e.g. die/eine Sonne (the/a sun), der/ein Mond (the/a moon), das/ein Haus (the/a house). This does affect the textual representation of relations.

Cases

German has four cases to describe a word's function in the sentence: nominative, accusative, dative and genitive. They are required to give the correct meaning to a sentence, because the word order is not as fixed as in English. The article changes depending on the case as well [Hentschel and Weydt, 2003, pp. 167-190].

Word order

English generally adheres to the subject-verb-object order, while in German there are few rules pertaining to the word order. The four cases complement the missing rules and provide the information needed to a sentence to be comprehensible. The high flexibility on how word might be shuffled within sentence has implications on the performance of German OIE systems, since it poses a challenge to the task of dependency parsing.

Subjunctive and noun semantics

Among the noteworthy grammatical differences between German and English is the usage of the subjunctive mood, where in German exist the so called present subjunctive and the past subjunctive. The German subjunctive allows for encoding hypothetical situations, where in English one would use "would" or phrases like "as if". Another difference is the so called *Funktionsverbgefüge*, where the semantics are in part transferred from the verb to the noun. For example, the literal translation for the German phrase "eine Frage stellen" would be "to put a question". These differences may have an impact on the quality of the German parsing components and thus indirectly influence the performance of the German OIE system.

Tools and Resources

Although this criterion is not different for the two languages, it may have profound effects on the performance of OIE systems. The available tools for preprocessing and parsing for the English language, as well as the available corpora, outnumber those for the German language. Another noteworthy difference is that the tagsets for English and German are not identical, and neither are the grammatical dependencies used. Table 2 provides a comparison of selected POS tags for the German TIGER corpus and their counterparts in the universal tagset[1].

Potential for OIE

Due to the differences between English and German one can identify promising key features for building a German OIE system. In particular, due to the more flexible word order in the German language, the usage of dependency parsing appears to be more promising than constituency parsing.

[1] http://universaldependencies.org

TIGER	Universal	Description
NN	NOUN	Noun
NE	PROPN	Proper noun
PIS	PRON	Pronoun
PPOSAT	PRON	Possessive pronoun
PRELAT	PRON	Relative pronoun
APPR	ADP	Preposition

Table 2: Comparison of two tagsets for the most imported tags, as used by GerIE. German tags from the TIGER corpus are compared to the Universal POS tag set. Since the TIGER postags are more fine grained, there are cases of multiple TIGER tags matching to a single Universal tag.

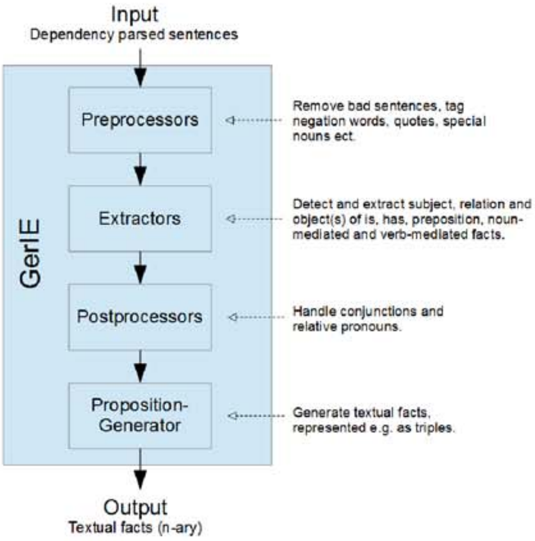

Input
Dependency parsed sentences
GerIE
Preprocessors
Remove bad sentences, tag negation words, quotes, special nouns ect.
Extractors
Detect and extract subject, relation and object(s) of is, has, preposition, noun-mediated and verb-mediated facts.
Postprocessors
Handle conjunctions and relative pronouns.
Proposition-Generator
Generate textual facts, represented e.g. as triples.
Output
Textual facts (n-ary)

Figure 1: The architecture of our GerIE system: The processing is arranged as a pipeline offering a flexible addition of modules.

3 System Architecture

This section describes GerIE, our proposed OIE system for the German language, which takes dependency-parsed sentences as input. For the parsing of sentences, we selected Mate Tools[2], which can carry out lemmatisation, part-of-speech tagging, morphological tagging and dependency parsing. The provided models were trained on the full German TIGER corpus[3] meaning that all extraction patterns

[2] `https://code.google.com/archive/p/mate-tools/` (version: anna 3.61)

[3] `http://www.ims.uni-stuttgart.de/forschung/ressourcen/korpora/tiger.en.html`

Dependency	Interpretation
AG	Genitive adjunct
APP	Apposition
MNR	PP adjuncts (in noun phrases)
NK	Noun kernel modifier
SB	Subject
PD	Predicate
PG	Pseudo-genitive

Table 3: Overview of selected grammatical dependencies as produced by the Mate tools and models.

Rule	Pattern
i_1	$NN \xleftarrow{NK} \mathrm{N}$
i_2	$NN \xleftrightarrow{APP} \mathrm{N}$
i_3	$NN \xrightarrow{AG} \mathrm{PIS} \xleftrightarrow{APP} N$
i_4	$NN \xleftrightarrow{SB\|PD} \mathrm{V} \xleftrightarrow{SB\|PD} N$
i_5	$N \xleftrightarrow{APP} \mathrm{NN} \xrightarrow{PG} APPR \xrightarrow{NK} N$
i_6	$* \xleftarrow{SB} \mathrm{V} \xrightarrow{O} *$

Table 4: List of patterns representing the “Is-Fact Rules”, as applied on the dependency tree.

were specifically created for the tagsets used in this corpus[4]. Therefore, any parser trained on the TIGER corpus can be used by GerIE. Table 3 provides relevant grammatical dependencies together with their interpretation, selected by the one being used by GerIE.

We adopted *minimality* since it improves the quality of the extracted facts and *separation of detection and representation* since this helps to enforce minimality and allows for easier changes and customisation. *Levels of granularity* were disregarded since no substantial post processing is currently available. If necessary, additional output information could still be added at a later time with minor effort. *Separation of relation detection and extraction* was also discarded since its two parts are tightly bound; the moment a proper relation pattern is detected in a sentence, it is extracted.

Once the sentence is parsed, our proposed OIE system is applied to the dependency tree generating propositions in plain text form. The system consists of a pipeline architecture comprising four main stages, as depicted in Figure 1.

[4] http://www.ims.uni-stuttgart.de/forschung/ressourcen/korpora/TIGERCorpus/annotation/tiger_introduction.pdf

	Rule Pattern
h_1	$NN \xleftarrow{AG}$N
h_2	$NN \xleftarrow{NK} PPOSAT$
h_3	$(RC)NN \xleftarrow{AG} PRELAT$
h_4	$NN \xleftarrow{PG}$APPR$\xleftarrow{NK} N$
h_5	Compound Word

Table 5: List of patterns used to detect “Has-Fact” relations.

3.1 Preprocessors

The first step is preprocessing of the received dependency trees by adding additional annotations to the tree. To that end, interrogative clauses are removed from further processing since they are likely to not contain any facts, e.g. *Does alien life exist?* Additionally, sentences that do not contain a verb as a root element in their dependency tree structure are considered malformed or uninformative and are removed, e.g. *On the contrary.*

Next, quote symbols and POS tags are used to detect direct speech with the aim to prevent relations to be extracted from these. The reason is the observation that direct speech often represents a personal opinion that may not be a fact, e.g. *Kevin says: “Alien life exists”.* Although the sentence is still processed, the extraction of facts in between the quotes is omitted.

Nouns are further analysed and assigned to more specific categories. Nouns that are numbers (e.g. thousand), quantities (e.g. handful) or units (e.g. meter) are additionally annotated to facilitate the lookup of words in a sentence with a gazetteer list containing the names of units, numbers, etc[5].

Words that may negate a fact, such as *not, no, nobody*, are marked as well with the help of a list of known negation words[6]. These words are essential in a proposition since they can completely change its meaning.

3.2 Extractors

In this step the rules to extract facts are executed. We identified five types of relations associated with a set of rules. Tables 4 to 6 and Figures 2 and 4 provide a graphical notation, which can be interpreted as following: Nodes represent POS tags and edges are dependency labels. A rule matches, if there is a match in the

[5] Based on `https://de.wikipedia.org/wiki/Liste_physikalischer_Gr\%C3\%B6\%C3\%9Fen` and `http://www.canoo.net/services/GermanSpelling/Regeln/Gross-klein/Zahlen.html` (29.09.2016)

[6] `http://www.canoo.net/services/OnlineGrammar/Satz/Negation/Negationswort/index.html` (29.09.2016)

Rule Pattern	
p_1	$N \xleftarrow{MNR} \text{APPR} \xleftarrow{NK} N$
p_2	$N \xleftarrow{OP} \text{APPR} \xleftarrow{NK} N$

Table 6: The two "Preposition-Fact" rules.

POS tags together with a match of the relation type. If there is more than one relation type, the rule matches if any of the relation types match. The direction of the arrows shows which end may assume the role of the head. Thus the rule only matches the dependency graph if there is also a match in that direction. A relation is detected when the path from left to right matches a part of the dependency tree of the sentence.

3.2.1 Is-Fact Rules

These rules detect the "is" relation between entities: X is Y, where X is a common or proper noun, and Y is a common noun: *[Peter][is][human]*, *[lion][is][predator]*. In Table 4 the individual rules are listed.

3.2.2 Has-Fact Rules

These rules aim at detecting the "has" relations and are listed in Table 5. Since verbs used as nouns may lead to abstract facts we explicitly excluded these, e.g. *das Wohnen in Großstädten* (living in big cities) yields [Großstadt][hat][Wohnen] ([big city][has][living]). Many of the extracted facts are abstract and often describe general concepts. The rule named h_5 in Table 5 was introduced based on the observation that in the German language many compound words actually represent has-fact relationships, e.g. "Google-Chef" (Google has a boss).

3.2.3 Preposition-Fact Rules

This set of rules identifies relations mediated by prepositions, e.g. *[New York][in][America]*. Table 6 lists the two rules. Similarly to the has-fact rules, relations extracted via these patterns tend to have a more general nature.

3.2.4 Noun-Mediated-Fact Rules

This set of rules aims to detect relations mediated by a noun phrase, e.g. [New York][Stadt in][Amerika] ([New York][city in][America]). Since each common noun is a possible mediator between two entities every time the extractor module

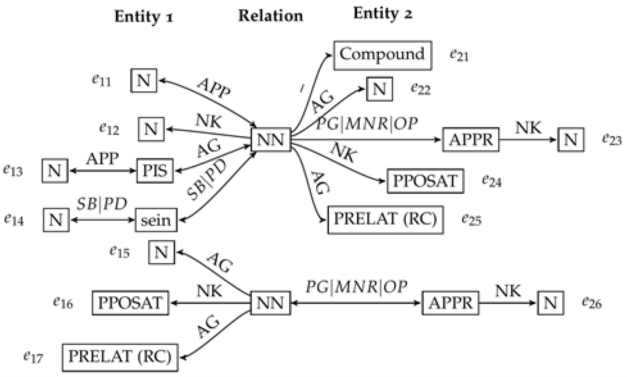


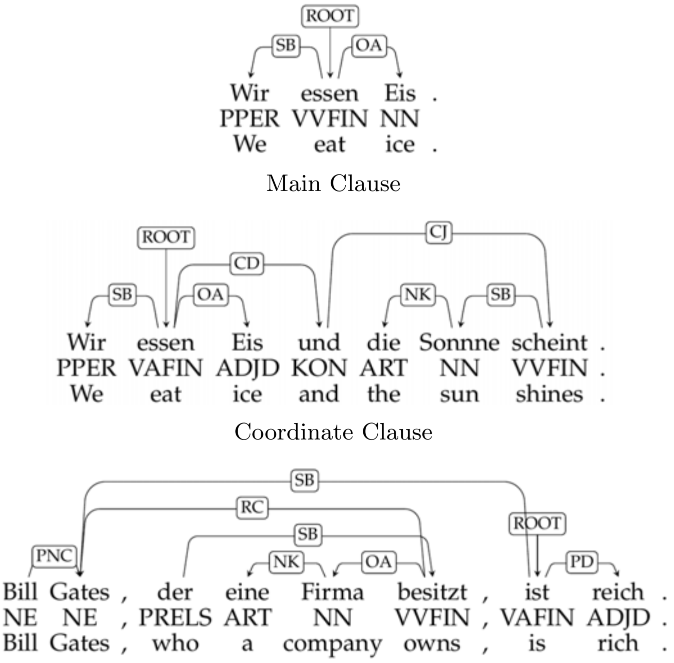
Figure 2: Overview of the rules associated with "Noun-Mediated-Facts". All patterns are combined into a single graph. The final extraction pattern can be constructed by following the path from left (*Entity 1*) to right (*Entity 2*), passing over the middle point NN (*Relation*).

encounters a common noun in the dependency tree, it tries to find a left entity and a right entity for it. The patterns for detecting these entities are shown in Figure 2. There are many possible combinations of patterns for the first and second entities mediated by the same common noun.

The extraction pattern e_{21} (Compound) is a special pattern that extracts entities from part of the relation. Here, the relation is a compound with a hyphen (Google-CEO). The first part is considered to be the entity and the second one is deemed to be the actual relation. e_{14} is another special pattern, which detects entities in relations that are explicitly written with a form of the word *sein* (to be), such as "*Obama ist Präsident von Amerika.*" (Obama is the President of America.)

3.2.5 Verb-Mediated-Fact Rules

A verb-mediated fact consists of one subject, one verb and an arbitrary number of objects. The dependency grammar offers a relatively easy way to obtain these components since the verb is always the root of a phrase in the tree and the subject is an explicitly labelled child. One of our requirements for facts was that they were minimal (as suggested by [Bast and Haussmann, 2013]). To achieve this, we iterated the node elements in the tree bottom-up and separated the extracted facts from the tree when they were expandable for the rest of the sentence. In this way, every time a verb with a subject appears in a sentence, the relation is extracted along with the inherent objects. Next, we decided if it is a stand-alone fact and may get separated from the dependency tree based on the clause type to which the verb belongs: i) main clause, consisting of the


ROOT
SB
OA
Wir essen Eis .
PPER VVFIN NN
We eat ice .
ROOT
CD
CJ
SB
OA
NK
SB
Wir essen Eis und die Sonnne scheint .
PPER VAFIN ADJD KON ART NN VVFIN .
We eat ice and the sun shines .
SB
RC
SB
ROOT
PNC
NK
OA
PD
Bill Gates , der eine Firma besitzt , ist reich .
NE NE , PRELS ART NN VVFIN , VAFIN ADJD .
Bill Gates , who a company owns , is rich .

Main Clause

Coordinate Clause

Subordinate Clause

Figure 4: Example of sentences for the three clause types, which contain verb-mediated facts and their dependency tree.

verb of the first main clause, ii) coordinate clause, if multiple main clauses are connected via a coordinating conjunction, iii) subordinate clause, for relative clauses introduced by a relative pronoun or subordinate conjunctions. Figure 4 lists examples for the three types of clauses.

3.3 Post-Processing

The individual facts extracted from the dependency tree are further processed and filtered.

3.3.1 Conjunction Post-Processor

To separate the tasks of our modules, we considered conjunctions decoupled from the extraction of facts. All coordinating conjunctions (und, sowie, wie, aber, doch) except disjunctive are taken into account this way. We did not process compound conjunctions, e.g. “weder ... noch” (either ... or) or “nicht nur ... sondern auch” (not only ... but also), since they are rather infrequent and more difficult to handle. The idea was to ensure minimality. Facts such as [Garfield][likes][Lasagne and Spaghetti] should be split into [Garfield][likes][Lasagne] and [Garfield][likes][Spaghetti].

3.3.2 Relative Pronoun Post-Processor

This post-processor handled facts extracted from relative clauses (RCs) to remove or modify pronouns. Three types of pronouns must be addressed differently:

- Substituting relative pronoun (PRELS): "*der Mann, der in Graz arbeitet*" → [in Graz][arbeitet][~~der~~ der Mann]
- Attributive relative pronoun (PRELAT): "*der Mann, dessen Freund in Graz arbeitet*" → [in Graz][arbeitet][~~dessen Freund~~ der Freund des Mann(s)]
- Adverbial interrogative or relative pronoun (PWAV): "*Das Haus, wo er wohnt, ist groß.*" → [er][wohnt][~~wo~~ im Haus]

3.4 Proposition Generator

A proposition generator is used to convert facts into a proper output format. Rather than of generating multiple propositions for a single fact, we decided to use an n-ary representation. This procedure has been chosen to preserve all textual information.

4 Evaluation

To evaluate of our OIE system, two dedicated test data sets were gathered and manually annotated by two human annotators (native speakers) establishing a gold standard.

4.1 Data Sets

4.1.1 GerNews

The first dataset was created by randomly selecting 150 sentences from a collection of German news articles, gathered around 2014 from 17 German news websites, such as `http://www.faz.net/`, `http://diepresse.com`, `http://german.ruvr.ru` or `http://europa.eu/`. The articles consisted of 603 words on average. The sentence length in GerNews ranged from 3 to 40 words, and the average number of words was about 16 (which is in line with the expectations, e.g. [Groeben and Vorderer, 1982]). Due to the domain, the dataset contained more direct or reported speech then usual and included some sentences that did not fit the classical subject-predicate-object structure.

4.1.2 GerBH

In order to estimate how an OIE system generalises to other types of text, we introduced a second data set from the domain of classical printed encyclopaedias. We assembled the second dataset out of Brockhaus[7] articles. Until the rise of online encyclopaedias, Brockhaus used to be the largest German language encyclopaedia. Its writing style is special because the goal was to accommodate as much information as possible in a small space, since every additional page would increase printing costs. We randomly selected articles until we obtained 100 sentences. Note that we skipped articles that were shorter than 10 words since they often simply referred to another article. The length of the sentences ranged from 1 to 59, with an average number of 21.4 words per sentence.

4.2 Annotation Procedure

Our goal was to annotate each sentence with all possible distinct facts that can be extracted from it. Since such a procedure involves a high degree of manual work it is limited to smaller evaluation corpora. While many recent systems [Bast and Haussmann, 2013, Del Corro and Gemulla, 2013] eschewed gold facts and labelled each extraction as correct or incorrect manually, we were interested in automating the evaluation process as much as possible. Gold facts make it possible to calculate the precision of the extracted tuples and the recall value. Moreover, they allow to repeat the experiments using alternative methods and arrive at comparable results.

To make gold facts as consistent as possible we defined several requirements:

Syntax: We decided to apply the form [subject][relation][object][relation$_2$] to each gold fact. Object and relation$_2$ can be empty (e.g. [Einstein][died][][]). Relation$_2$ optionally contains the second part of a relation, which can be used either as part of the relation or as part of the object: in the fact [Einstein][likes][in summer][to swim] "to swim" could be attached either after "likes" or before "in summer".

Type: A gold fact has to belong to one of the 5 types GerIE supports (is fact, has fact, preposition fact, noun-mediated fact, verb-mediated fact). Without this restriction, the number of possible gold facts would be unknown since there is no objective way to establish which parts of a sentence constitute a fact.

Minimality: To reduce the number of gold facts, we decided that a gold fact has to be minimal. This requirement only affects the main items in the subject, relation and object. For example *[He][buys][red and yellow apples]*

[7] `http://www.brockhaus.de/`

is minimal since the main item of the object is “apple”, while “red and yellow” are merely describing it and should not be separated. In contrast, *[He][buys][apples and bananas]* has two main items as objects and requires two gold facts (one for each object). We did not apply this rule to facts that would lose meaning or make no sense, if the conjunct phrases were separated: *[John and Tim][are][a team]* is a minimal fact. Additionally, a fact should not occur within another fact, but only if the other fact would lose its meaning without the first one.

Word form: Only words from the sentence can be a gold fact, and they must have exactly the same form. This can lead to grammatically false facts when words in the sentence and in the fact are in a different case (e.g. “John’s car is blue” → [John’s][has][car], because in German “John’s” would be Genitive and written as “Johns”) or has a different number (e.g. “John and Tim work in America” → [John][work][in America]). This is necessary because GerIE does also not alter any words, and facts will be checked to the exact equality to gold facts. An exception with that regard is the implicit “is” in facts, “has” in has facts and “of” in noun-mediated facts.

Word selection: It is possible to write several gold facts, e.g. each with a different combination of adjectives for the noun. Since it is difficult to establish which words are indeed essential to a fact, we simply used all occurring words which fit in the fact. From the phrase “America’s hard-working president Obama...” a gold fact *[Obama][hard-working president of][America][]* could be obtained. This obviates the need to create multiple similar gold facts and ensures that our facts remain as distinct as possible.

Distinct facts: We excluded facts that can be inferred from other facts from the gold facts. For example, from the sentence *[Obama][hard-working president of][America][]* the two facts ([Obama][is][hard-working president][] and [America][has][hard-working president][]) can be inferred. Hence they are not gold facts. This always applies when a noun-mediated gold fact exists.

Word order: The word order in a gold fact should be same to that in the sentence.

Implicit references: We did not include phrases that were implicitly referenced in the gold phrases. For example, from the sentence “He visited India, talked to president Mukherjee.” a human can identify that Mukherjee is president of India, but our dependency parser does allow to derive this information. Since such references cannot be detected, we only accepted gold facts that can be identified with help of our dependency parser.

We also aimed to establish how useful extracted facts were. Although OIE systems can extract large amounts of relations, their usefulness is not always apparent. We selected 4 categories, into which each gold fact can be assigned in a relatively uncomplicated way:

Not Useful: a category for all facts that were viewed as non-informative. This applies to overly specific facts ([Kofi Annan][required][from Goodluck Jonathan the assurance, that they will accept the election outcome][]), overly general facts since the context is often missing ([refugees][live][with rebels][]) and the relations that are too unspecific ([elections][are][in four weeks in Nigeria][]).

Abstract: a category for has-facts, preposition facts and noun mediated facts. If one of the two objects in the relation is abstract (no real-life object), the fact deemed abstract, e.g. [terrorist][has][hate][], [events][in][Mariupol][] or [Moscow][has][notion][]. As shown in these examples, it can also apply to a named entity.

Concrete Named Entity: a category for the most interesting kind of facts, that are typically targeted by traditional IE systems. A concrete fact generally contains specific information about a named entity, such as [Kofi Annan][is][Nobel peace laureate][] or [Nigeria][has][politicians][]. References to named entities were also accepted (e.g. [she][lives][in Berlin][]).

General Knowledge: a category for all facts that provide concrete knowledge about things in the world and are not only valid for a short time period, e.g. [Ukrainians][work][on Russian construction sites][], [UDID][serves][real-time tracking of iPhones][].

We collected this information to allow for a more in-depth analysis, for example if the algorithm achieves significantly higher recall in a specific category.

4.3 Gold Facts

Using our annotation procedure, we annotated a total of 506 gold facts for the GerNews data set. Each sentence contained 3.37 distinct facts on average. Although the GerBH dataset is smaller, we were able to identify 452 gold facts there, leading to a larger rate of 4.52 facts per sentence. Due to its distinctive writing style, GerBH contains proportionally twice as many facts as GerNews. Moreover, sentences in GerBH often do not contain verbs, provided that the reader is likely to comprehend the meaning, such as in: "*John B., secretary, etc*".

Measure	GerNews	GerBH
Sentences	150	100
Gold facts	506	452
Per sentence	3.37	4.52
Extracted facts	512	407
Per sentence	3.41	4.07
Correct, minimal, complete	364	184
Correct, minimal	14	9
Correct, complete	16	24
Incorrect	118	187
Precision	0.77	0.54
Recall	0.78	0.48
F_1	0.77	0.51
Gold facts from correctly parsed phrases	446	241
Extracted facts from correctly parsed phrases	434	249
Precision	0.91	0.88
Recall	0.88	0.90
F_1	0.89	0.89

Table 7: Overview of GerNews and GerBH data sets, including detailed evaluation results of the fact-extraction process.

4.4 Results

We applied our algorithm on the two data-sets and measured the results. Table 7 shows that GerIE extracted 394 correct facts in total from the GerNews dataset, resulting in a precision of 0.77 and a recall of 0.78. 16 of the correct facts were not complete, and 14 were not minimal. The number of extracted facts per sentence is on average 3.37, indicating that more than two correct facts per sentence were obtained. Since GerIE's performance depends on that of the dependency parser, we also tagged each fact with the information on whether the underlying phrase was correctly parsed. As a result, we could calculate precision and recall values for a filtered set of facts produced solely based on correctly parsed sentences. In that respect GerIE achieved 0.91 precision and 0.88 recall for the GerNews data set.

Our system had a poorer performance on the GerBH data set, with only 217 correctly extracted facts, yielding a precision of 0.54 and a recall of 0.48. 9 of the extracted facts were correct but not complete, and 24 were correct but not minimal. The total number of extracted facts was 407, leading to 4.07 facts per sentence, and also precision was low (0.54). Nearly all of the missed or incorrect facts were attributable to incorrectly parsed sentences. Considering facts based only on correctly parsed sentences improved the results (i.e. a precision of 0.88 and a recall of 0.90).

In addition, we conducted a manual evaluation of *PropsDE* [Falke, 2016] on our data sets to put the results for GerIE into perspective. Due to differences in the representations of extracted tuples, we restricted our evaluation to a proportion of correctly extracted facts. The parser component of PropsDE failed to produce correct results for some of the sentences, which were subsequently excluded from the evaluation, resulting in 94 and 41 for the GerNews and GerBH corpora, respectively. We established that the precision was 0.85 for GerNews and 0.68 for GerBH.

4.5 Discussion

The precision and recall of both datasets used in our study are quite dissimilar, when all sentences are included. GerNews achieves better results, which can partly be due to the parser being trained on the TIGER corpus that also contains news articles. This difference is even more pronounced for *PropsDE*.

Further analyses of incorrect or missing facts revealed that erroneous dependency parses were mainly responsible for incorrectly extracted facts, which prompted us to limit the performance figures to correctly parsed sentences. Another reason for errors include too general extraction patterns, missed essential phrases and dependent subordinate clauses. The missing facts were also caused by rejected subordinate clauses and accidental filtering, for instance, due to punctuation marks. For correctly parsed sentences, the extraction performance of both data sets is comparable (identical F_1 of 0.89), suggesting that our system can be applied to other domains as well.

Compared to the two state-of-the-art English OIE systems ClausIE and CSD-IE, the number of extracted facts and the precision values are similar, when considering the results of GerIE's GerNews evaluation and those published by [Bast and Haussmann, 2013]. All systems extract on average more than 3 facts per sentence.

5 Conclusions

In this paper, we presented GerIE, an OIE system for the German language. As a starting point for our work, we surveyed the existing systems generally tailored to the English language. Following some of these methods, we established the core of our system as a set of manually crafted rules, which were applied on the dependency tree produced by parser components.

To evaluate the German language OIE systems, we generated two dedicated data sets, including manually annotated gold facts. They consisted of articles from two domains, news articles and encyclopaedic articles, that differ with regard to writing style and information density. Our system's evaluation indicates that GerIE's performance is comparable to its English counterparts, in terms of key characteristics. The quality of the preceding dependency parsing step determines the final quality of the extracted facts.

For future research, we plan to investigate methods based on machine learning rather than on hand crafted rules. Furthermore, we plan to apply GerIE to evaluate its suitability under a fact-checking scenario.

Acknowledgments

The Know-Center is funded within the Austrian COMET Program - Competence Centers for Excellent Technologies - under the auspices of the Austrian Federal Ministry of Transport, Innovation and Technology, the Austrian Federal Ministry of Economy, Family and Youth and by the State of Styria. COMET is managed by the Austrian Research Promotion Agency FFG.

References

[Akbik and Broß, 2009] Akbik, A. and Broß, J. (2009). Wanderlust: Extracting semantic relations from natural language text using dependency grammar patterns. In *Proceedings of the 2009 Semantic Search Workshop at the 18th International World Wide Web Conference*.

[Akbik and Löser, 2012] Akbik, A. and Löser, A. (2012). Kraken: N-ary facts in open information extraction. In *Proceedings of the Joint Workshop on Automatic Knowledge Base Construction and Web-scale Knowledge Extraction*, AKBC-WEKEX '12. Association for Computational Linguistics.

[Banko et al., 2007] Banko, M., Cafarella, M. J., Soderland, S., Broadhead, M., and Etzioni, O. (2007). Open information extraction for the web. In *Proceedings of the International Joint Conference on Artificial Intelligence*, IJCAI '07.

[Bast and Haussmann, 2013] Bast, H. and Haussmann, E. (2013). Open information extraction via contextual sentence decomposition. In *Proceedings of the 2013 IEEE 7th International Conference on Semantic Computing*, ICSC '13. IEEE.

[Castella Xavier et al., 2013] Castella Xavier, C., de Lima, S., Lucia, V., and Souza, M. (2013). Open information extraction based on lexical-syntactic patterns. In *Proceedings of the Brazilian Conference on Intelligent Systems*, BRACIS '13. IEEE.

[Christensen et al., 2010] Christensen, J., Soderland, S., Etzioni, O., et al. (2010). Semantic role labeling for open information extraction. In *Proceedings of the NAACL HLT 2010 1st International Workshop on Formalisms and Methodology for Learning by Reading*, pages 52–60. Association for Computational Linguistics.

[Cimiano and Wenderoth, 2005] Cimiano, P. and Wenderoth, J. (2005). Automatically learning qualia structures from the web. In *Proceedings of the ACL-SIGLEX Workshop on Deep Lexical Acquisition.* Association for Computational Linguistics.

[Del Corro and Gemulla, 2013] Del Corro, L. and Gemulla, R. (2013). ClausIE: Clause-based open information extraction. In *Proceedings of the 22nd International World Wide Web Conference.* International World Wide Web Conferences Steering Committee.

[Fader et al., 2011] Fader, A., Soderland, S., and Etzioni, O. (2011). Identifying relations for open information extraction. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing.* Association for Computational Linguistics.

[Falke, 2016] Falke, T. (2016). Porting an open information extraction system from english to german. *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP-16)*, pages 892–898.

[Fellbaum, 1998] Fellbaum, C. (1998). *WordNet: An electronic lexical database.* The MIT press.

[Gamallo and Garcia, 2015] Gamallo, P. and Garcia, M. (2015). *Progress in Artificial Intelligence: 17th Portuguese Conference on Artificial Intelligence*, chapter Multilingual Open Information Extraction. Springer International Publishing, Cham, Switzerland.

[Gamallo et al., 2012] Gamallo, P., Garcia, M., and Fernández-Lanza, S. (2012). Dependency-based open information extraction. In *Proceedings of the Joint Workshop on Unsupervised and Semi-Supervised Learning in NLP*. Association for Computational Linguistics.

[Groeben and Vorderer, 1982] Groeben, N. and Vorderer, P. (1982). *Leserpsychologie: Textverständnis-Textverständlichkeit*. Aschendorff Münster, Stroudsburg, PA, USA.

[Hentschel and Weydt, 2003] Hentschel, E. and Weydt, H. (2003). *Handbuch der deutschen Grammatik*. Walter de Gruyter, Berlin, Germany, 3 edition.

[Kok and Domingos, 2005] Kok, S. and Domingos, P. (2005). Learning the structure of markov logic networks. In *Proceedings of the 22nd International Conference on Machine Learning*, ICML '05. ACM.

[Moschitti, 2006] Moschitti, A. (2006). *17th European Conference on Machine Learning*, chapter Efficient Convolution Kernels for Dependency and Constituent Syntactic Trees. Springer Berlin Heidelberg.

[Nakashole et al., 2012] Nakashole, N., Weikum, G., and Suchanek, F. (2012). Patty: A taxonomy of relational patterns with semantic types. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*. Association for Computational Linguistics.

[Piskorski and Yangarber, 2013] Piskorski, J. and Yangarber, R. (2013). *Multi-source, Multilingual Information Extraction and Summarization*, chapter Information Extraction: Past, Present and Future. Springer Berlin Heidelberg.

[Schmitz et al., 2012] Schmitz, M., Bart, R., Soderland, S., Etzioni, O., et al. (2012). Open language learning for information extraction. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

[Sleator and Temperley, 1995] Sleator, D. D. and Temperley, D. (1995). Parsing English with a link grammar. *CoRR*, abs/cmp-lg/9508004.

[Wang et al., 2014] Wang, M., Li, L., and Huang, F. (2014). Semi-supervised Chinese open entity relation extraction. In *Proceedings of the 3rd IEEE International Conference on Cloud Computing and Intelligence Systems*. IEEE.

[Wu and Weld, 2010] Wu, F. and Weld, D. S. (2010). Open information extraction using wikipedia. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, ACL '10. Association for Computational Linguistics.

[Xavier and de Lima, 2014] Xavier, C. C. and de Lima, V. L. S. (2014). Boosting open information extraction with noun-based relations. In *Proceedings of the 9th International Conference on Language Resources and Evaluation*, LREC '14.

[Xu et al., 2013] Xu, Y., Kim, M.-Y., Quinn, K., Goebel, R., and Barbosa, D. (2013). Open information extraction with tree kernels. In *North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, HLT-NAACL '13.

[Yahya et al., 2014] Yahya, M., Whang, S., Gupta, R., and Halevy, A. Y. (2014). Renoun: Fact extraction for nominal attributes. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '14.

[Zhila and Gelbukh, 2013] Zhila, A. and Gelbukh, A. (2013). Comparison of open information extraction for English and Spanish. In *Proceedings of the 19th Annual International Conference Dialog 2013*.

[Zhu et al., 2009] Zhu, J., Nie, Z., Liu, X., Zhang, B., and Wen, J.-R. (2009). Statsnowball: a statistical approach to extracting entity relationships. In *Proceedings of the 18th International World Wide Web Conference*. ACM.