MinIE: Minimizing Facts in Open Information Extraction

Kiril Gashteovski[1] and **Rainer Gemulla**[1] and **Luciano del Corro**[2]
[1]Universität Mannheim, Mannheim, Germany
[1]{k.gashteovski,rgemulla}@uni-mannheim.de
[2] Max-Planck-Institut für Informatik, Saarbrücken, Germany
[2]corrogg@mpi-inf.mpg.de

Abstract

The goal of Open Information Extraction (OIE) is to extract surface relations and their arguments from natural-language text in an unsupervised, domain-independent manner. In this paper, we propose MinIE, an OIE system that aims to provide useful, compact extractions with high precision and recall. MinIE approaches these goals by (1) representing information about polarity, modality, attribution, and quantities with semantic annotations instead of in the actual extraction, and (2) identifying and removing parts that are considered overly specific. We conducted an experimental study with several real-world datasets and found that MinIE achieves competitive or higher precision and recall than most prior systems, while at the same time producing shorter, semantically enriched extractions.

1 Introduction

Open Information Extraction (OIE) (Banko et al., 2007) is the task of generating a structured, machine-readable representation of information expressed in natural language text in an unsupervised, domain-independent manner. In contrast to traditional IE systems, OIE systems do not require an upfront specification of the target schema (e.g., target relations) or access to background knowledge (e.g., a knowledge base). Instead, extractions are (usually) represented in the form of surface subject-relation-object triples. OIE serves as input for deeper understanding tasks such as relation extraction (Riedel et al., 2013; Petroni et al., 2015), knowledge base construction (Dong et al., 2014), question answering (Fader et al., 2014), word analogy (Stanovsky et al., 2015), or information retrieval (Löser et al., 2012).

Consider, for example, the sentence *"Superman was born on Krypton."* An OIE system aims to extract the triple *(Superman, was born on, Krypton)*, which most of the available systems will correctly produce. As another example, consider the more involved sentence *"Pinocchio believes that the hero Superman was not actually born on beautiful Krypton"*, and the corresponding extractions of various systems in Table 1, extractions 1–6. Although most of the extractions are correct, they are often overly specific in that their constituents contain specific modifiers or even complete clauses. Such extractions severely limit the usefulness of OIE results (e.g., they are often pruned in relation extraction tasks). The main goals of OIE should be (i) to provide useful, compact extractions and (ii) to produce extractions with high precision and recall. The key challenge in OIE is how to achieve both goals simultaneously. In fact, most of the available systems (often implicitly) focus on either compactness (e.g., ReVerb (Fader et al., 2011)) or precision/recall (e.g., ClausIE (Del Corro and Gemulla, 2013)).

We propose MinIE, an OIE system that aims to address and trade-off both goals. MinIE is built on top of ClausIE, a state-of-the-art OIE system that achieves high precision and recall, but often produces overly-specific extractions. To generate more useful and semantically richer extractions, MinIE (i) provides semantic annotations for each extraction, (ii) minimizes overly-specific constituents, and (iii) produces additional extractions that capture implicit relations. Table 1 shows the output of (variants of) MinIE for the example sentence. Note that MinIE's extractions are significantly more compact but retain correctness.

MinIE's semantic annotations represent information about polarity, modality, attribution, and quantities. The idea of using annotations has al-

Pinocchio believes that the hero Superman was not actually born on beautiful Krypton.				
OLLIE	*1*	(Pinocchio,	believes that,	the hero [...] beautiful Krypton)
	2	(Superman,	was not actually born on,	beautiful Krypton)
	3	(Superman,	was not actually born on beau. Krypton in,	the hero)
ClausIE	*4*	(Pinocchio,	believes,	that the hero [...] beautiful K.)
	5	(the hero Superman,	was not born,	on beautiful Krypton)
	6	(the hero Superman,	was not born,	on beautiful Krypton actually)
Stanford OIE		*No extractions*		
MinIE-C(omplete)	*7*	(Superman,	was born actually on,	beautiful Krypton)
		A.: fact. (– [not], CT), attrib. (Pinocchio, +, PS [believes])		
	8	(Superman,	was born on,	beautiful Krypton)
		A.: fact. (– [not], CT), attrib. (Pinocchio, +, PS [believes])		
	9	(Superman,	”is”,	hero)
		A.: fact. (+, CT)		
MinIE-S(afe)	*10*	(Superman,	was born on,	beautiful Krypton)
		A.: fact. (– [not], CT), attrib. (Pinocchio, +, PS [believes]), relation (was actually born on)		
	11	(Superman,	”is”,	hero)
		A.: fact. (+, CT)		
MinIE-D(ictionary)	*12*	(Superman,	was born on,	Krypton)
		A.: fact. (– [not], CT), attrib. (Pinocchio, +, PS [bel.]), rel. (was act. born on), argument (beau. K.)		
MinIE-A(ggressive)	*13*	(Superman,	”is”,	hero)
		A.: fact. (+, CT)		

A annotation; + positive polarity, – negative polarity; *PS* possibility, *CT* certainty; *fact.* factuality; *attrib.* attribution;

Table 1: Example extractions and annotations from various OIE systems

ready been explored by OLLIE (Mausam et al., 2012) for capturing the context of an extraction. MinIE follows OLLIE, but adds semantic annotations that make the extraction *itself* more compact and useful (as opposed to capturing context). For example, MinIE detects negations in the relation, removes them from the extraction, and adds a “negative polarity” (-) annotation. In fact, MinIE treats surface relations such as *was born on* and *was not born on* as equivalent up to polarity. The absence of negative evidence is a major concern for relation extraction and knowledge base construction tasks—e.g., addressed by using a local closed world assumption (Dong et al., 2014) or negative sampling (Riedel et al., 2013; Petroni et al., 2015)—and MinIE’s annotations can help to alleviate this problem.

In addition to the semantic annotations, MinIE minimizes its extractions by identifying and removing parts that are considered overly specific. In general, such minimization is inherently limited in scope due to the absence of domain knowledge. Thus MinIE does not and cannot correctly minimize all its extractions in all cases. Instead, MinIE supports multiple minimization modes, which differ in their aggressiveness and effectively control the usefulness-precision trade-off. In particular, MinIE’s complete mode (C) does not perform any minimizations. MinIE’s safe mode (S) only performs minimizations that are considered universally safe. MinIE’s dictionary mode (D) makes use of corpus-level statistics to inform the minimization process. Finally, MinIE’s aggressive mode (A) only keeps parts that are considered universally necessary. The use of corpus-level statistics by MinIE-D is inspired by the pruning techniques of ReVerb, although we use these statistics for minimization instead of pruning (see Sec. 2). Tab. 1 shows the output of MinIE’s various modes.

We conducted an experimental study with several real-world datasets and found that the various modes of MinIE produced much shorter extractions than most prior systems, while simultaneously achieving competitive or higher precision (depending on the mode being used). MinIE sometimes fell behind prior systems in terms of the total number of extractions. We found that in almost all of these cases, MinIE became competitive once redundant extractions were removed.

2 Related work

OIE was introduced by Banko et al. (2007). Since then, many different OIE systems have been proposed. Earlier systems—e.g., Fader et al.

(2011)—relied mostly on shallower NLP techniques such as POS tagging and chunking, while later systems often use dependency parsing in addition (Gamallo et al., 2012; Wu and Weld, 2010). Most OIE systems represent extractions in the form of triples, although some also produce n-ary extractions (Akbik and Löser, 2012; Del Corro and Gemulla, 2013) or nested representations (Bast and Haussmann, 2013; Bhutani et al., 2016). Some systems focus on non-verb-mediated relations (Yahya et al., 2014). MinIE is based on the state-of-the-art OIE system ClausIE (Del Corro and Gemulla, 2013).

A general challenge in OIE is to avoid both uninformative and overly-specific extractions. ReVerb (Fader et al., 2011) proposed to avoid overly-specific relations by making use of *lexical constraints*: relations that occur infrequently in a large corpus were considered overly-specific and pruned. MinIE's dictionary mode also makes use of the corpus frequency of constituents. In contrast to ReVerb, MinIE uses frequency to inform minimization (instead of to prune) and applies it to subjects and arguments as well. Perhaps the closest system in spirit to MinIE is Stanford OIE (Angeli et al., 2015), which uses aggressive minimization. Stanford OIE deletes all subconstituents connected by certain typed dependencies (e.g., *amod*). For some dependencies (e.g., *prep* or *dobj*), it uses a frequency constraint along the lines of ReVerb. MinIE differs from Stanford OIE in that it (i) separates out polarity, modality, attribution, and quantities; (ii) uses a different, more principled (and more precise) approach to minimization.

Annotated OIE extractions were introduced by OLLIE (Mausam et al., 2012), which uses two types of annotations: *attribution* (the supplier of information) and *clause modifier* (a clause modifying the triple). MinIE extends OLLIE's attribution by additional semantic annotations for polarity, modality, and quantities. Such annotations are not provided by prior OIE systems. CSD-IE (Bast and Haussmann, 2013) introduced the notion of nested facts (termed "minimal" in their paper) and produce extractions with "pointers" to other extractions. NestIE (Bhutani et al., 2016) takes up this idea. OLLIE's clause modifier has a similar purpose. MinIE currently does not handle nested extractions.

Another line of research explores the integration of background knowledge into OIE (Nakashole et al., 2012; Moro and Navigli, 2012, 2013). In general, OIE systems should use background knowledge when available, but remain open when not. MinIE currently does not use background knowledge, although it allows providing domain-dependent dictionaries.

3 Overview

The goal of MinIE is to provide minimized, semantically annotated OIE extractions. While the techniques employed here can potentially be integrated into any OIE system, we built MinIE on top of ClausIE. We chose ClausIE because (i) it separates the identification of the extractions from the generation of propositions, (ii) it detects clause types, which are also useful for MinIE, and (iii) it is a state-of-the-art OIE system with high precision and recall.

As ClausIE, MinIE focuses on extractions obtained from individual clauses (with the exception of attribution; Sec. 5.3). Each clause consists of one subject (S), one verb (V) and alternatively an indirect object (O_i), a direct object (O), a complement (C) and one or more adverbials (A). ClausIE identifies the clause type, which indicates which constituents are obligatory or optional from a syntactic point of view. Quirk et al. (1985) identified seven clause types for English: SV, SVA, SVC, SVO, SVOO, SVOA, and SVOC, where letters refer to obligatory constituents and each clause can be accompanied by additional optional adverbials.

MinIE consists of three phases. (1) Each input sentence is run through ClausIE and a separate extractor for implicit facts (Sec. 4.2). We rewrite ClausIE's extractions to make relations more informative (Sec. 4.1). We refer to the resulting extractions as *input extractions*. (2) MinIE then detects information about polarity (Sec. 5.1), modality (Sec. 5.2), attribution (Sec. 5.3), and quantities (Sec. 5.4) and represents it with semantic annotations. (3) To further minimize the resulting *annotated extractions*, MinIE provides various minimization modes (Sec. 6) with increasing levels of aggressiveness: MinIE-C(omplete), MinIE-S(afe), MinIE-D(ictionary), and MinIE-A(ggressive). The modes differ in the amount of minimizations being applied. The result of this phase is a *minimized extraction*.

Finally, MinIE outputs each minimized extraction along with its annotations. Semantic annotations (such as polarity) are crucial to correctly rep-

resent the extraction, whereas other annotations (such as original relation) provide additional information about the minimization process.

4 Input Extractions

We first describe how MinIE obtains meaningful input extractions.

4.1 Enriching Relations

As mentioned before, MinIE uses ClausIE as its underlying OIE system. The relations extracted by ClausIE consist of only verbs and negation particles (cf. Tab. 1). Fader et al. (2011) argue that such an approach can lead to uninformative relations. For example, from the sentence *"Faust made a deal with the Devil"*, ClausIE extracts triple *(Faust, made, a deal with the Devil)*, whereas the extraction *(Faust, made a deal with, the Devil)* has a more informative relation and a shorter argument. Indeed, the relation *made* is highly polysemous (49 synsets in WordNet), whereas *made a deal with* is not. MinIE aims to produce informative relations by deciding which constituents of the input sentence should be pushed into the relation. Our goal is to retain only one of the constituents of the input clause in the argument of the extraction whenever possible, while simultaneously retaining coherence. In particular, our approach uses the clause types detected by ClausIE to ensure that MinIE never removes obligatory constituents from a clause (which would lead to incoherent extractions); it instead may opt to move such constituents to the relation. Our approach is inspired by the syntactic patterns of ReVerb—which is similar to our handling of the SVA and SVO clause types—but, in contrast, applies to all clause types. Note that the relations produced in this step may sometimes be considered overly specific; they will be minimized further in subsequent steps.

SVA. If the adverbial is a prepositional complement, we push the preposition into the relation. For example, we rewrite *(Superman, lives, in Metropolis)* to *(Superman, lives in, Metropolis)*. This allows us to distinguish *live in* from relations such as *live during*, *live until*, *live through*, and so on.

SVO$_i$O, SVOC. We generally push the indirect object (SVO$_i$O) or direct object (SVOC) into the relation. In both cases, the verb requires two additional constituents: we use the first one to enrich the relation and the second one as an argument. For example, we rewrite *(Superman, declared, the city safe)* to *(Superman, declared the city, safe)*. As this example indicates, this rewrite is somewhat unsatisfying; further exploration is an interesting direction for future work.

SVOA. If the adverbial consists of a single adverb, we push it to the relation and use the object as an argument. This approach retains coherence because such adverbials are "fluent", i.e., they do not have a fixed position. Otherwise, we proceed as in SVOC, but additionally push the starting preposition (if present) of the adverbial to the relation. For example, *(Ana, turned, the light off)* becomes *(Ana, turned off, the light)*, and *(The doorman, leads, visitors to their destination)* becomes *(The doorman, leads visitors to, their destination)*.

Optional adverbials. If the clause contains optional adverbials, ClausIE creates one extraction without any optional adverbial and one additional extraction per optional adverbial. The former extractions are processed as above. The latter extractions are treated as if the adverbial were obligatory. For example, the extraction *(Faust, made, a deal with the Devil)* becomes *(Faust, made a deal with, the Devil)*. Here the actual clause type is SVO, but we process it as if it were SVOA.

Infinitive forms. If the argument starts with a to-infinitive verb, we move it to the relation. For example, *(Superman, needs, to defeat Lex)* becomes *(Superman, needs to defeat, Lex)*.

4.2 Implicit Extractions

ClausIE produces non-verb-mediated extractions from appositions and possessives. We refer to these extractions as *implicit extractions*. MinIE makes use of additional implicit extractors. In particular, we use the patterns of FINET (Del Corro et al., 2015) to detect explicit type mentions. For example, if the sentence contains "*president Barack Obama*", we obtain *(Barack Obama, is, president)*. We also include certain patterns involving named entities: pattern *ORG IN LOC* for extraction *(ORG, is IN, LOC)*; pattern *"Mr." PER* for *(PER, is, male)* (similarly, Ms. or Mrs.); and pattern *ORG POS? NP PER* for *(PER, is NP of, ORG)* from RelNoun (Pal and Mausam, 2016). Apart from providing additional high-quality extractions, we use implicit extractions as a signal for minimization (Sec. 6.2). The extractors above have thus been included both to increase recall and to be able to provide more effective minimizations.

Sentence	Factuality
S. does live in Metropolis.	(+, CT)
S. does not live in M.	(– [not], CT)
S. does probably live in M.	(+, PS [probably])
S. probably does not live in M.	(– [not], PS [probably])

Table 2: Factuality examples. MinIE extracts triple *(Superman; does live in; Metropolis)* from each sentence but the factuality annotations differ.

5 Semantic Annotations

Once input extractions have been created, MinIE detects information about polarity (Sec. 5.1), modality (Sec. 5.2), attribution (Sec. 5.3), and quantities (Sec. 5.4) and represents it using semantic annotations. Our focus is on simple, rule-based methods that are both domain-independent and (considered) safe to use in that they do not harm the accuracy of the extraction.

5.1 Polarity

MinIE annotates each extraction with information about its *factuality*. Following Saurí and Pustejovsky (2012), we represent the factuality of an extraction with two pieces of information: polarity (+ or -) and modality (CT or PS; for certainty or possibility, resp.). Tab. 2 lists some examples.

The *polarity* indicates whether or not a triple occurred in negated form. In order to assign a polarity value to a triple, we aim to detect whether the relation indicates a negative polarity. If so, we assign negative polarity to the whole triple. We detect negations using a small lexicon of negation words (e.g., *no, not, never, none*). If a word from the lexicon is detected, it is dropped from the relation and the triple is annotated with negative polarity (-) and the negation word. In Tab. 2, the extractions from sentences 2 and 4 are annotated as negative.

We found that this simple approach successfully spots many negations present in the input relations. Note that whenever a negation is present but not detected, MinIE still produces correct results because such negations are retained in the triple. For example, if a negations occurs in the subject or argument of the extraction, MinIE does not detect it. E.g., from sentence *"No people were hurt in the fire"*, MinIE extracts *(Q_1 people, were hurt in; fire)* with quantity Q_1=*no* (see Sec. 5.4). This extraction is correct, but can be further minimized to *(people; were hurt in; fire)* with a negative polarity annotation. We consider such advanced minimizations too dangerous to use.

Generally, negation detection is a hard problem and involves questions such as negation scope resolution, focus detection, and double negation (Blanco and Moldovan, 2011). MinIE does not address these problems, but restricts attention to the simple, safe cases.

5.2 Modality

The *modality* indicates whether the triple is a *certainty* (CT) or a *possibility* (PS) according to the clause in which it occurs. We proceed similarly as for the detection of negations and consider a triple certain unless we find evidence of possibility.

To find such evidence, MinIE searches the relation for (1) modal verbs such as *may* or *can*, (2) possibility-indicating words, and (3) certain infinitive verb phrases. For (2) and (3), we make use of a small domain-independent lexicon. Our lexicon is based on the lexicon of Saurí and Pustejovsky (2012) and the words in the corresponding WordNet synsets. It mainly contains adverbs such as *probably, possibly, maybe, likely* and infinitive verb phrases such as *is going to*, *is planning to*, or *intends to*. Whenever words indicating possibility are detected, we remove these words from the triple and annotate the triple as possible (PS) along with the words just removed. For example, sentences 3 and 4 in Tab. 2 are annotated PS with the possibility-indicating word *probably*.

5.3 Attribution

The *attribution* of a triple is the supplier of information given in the input sentence, if any. We adapt our attribution annotation from the notion of *source* of Saurí and Pustejovsky (2012), i.e., the attribution consists of a supplier of information (as in OLLIE) and an additional factuality (polarity and modality). The factuality is independent from the factuality of the extracted triple; it indicates whether the supplier expresses a negation or a possibility. Tab. 1 shows some examples.

We extract attributions from subordinate clauses and from "according to" patterns.

Subordinate clauses. MinIE searches for extractions that contain entire clauses as arguments. We then compare the relation against a domain-independent dictionary of relations indicating attributions (e.g., *say* or *believe*).[1] If we find a

[1] As with modality, the dictionary is based on Saurí and

match, we create an attribution annotation and use the subject of the extraction as the supplier of information. Each entry in the attribution dictionary is annotated with a modality. For example, relations such as *know*, *say*, or *write* express certainty, whereas relations such as *believe* or *guess* express possibility. If the relation is modified by a negation word, we mark the attribution with negative polarity (e.g., *never said that*). After the attribution has been established, we run ClausIE on the main clause and add the attribution to each extracted triple.

***"according to"* adverbial patterns.** We search for adverbials that start with *according to* and take whatever follows as the supplier with factuality (+, CT). The remaining part of the clause is processed as before.

5.4 Quantities

A *quantity* is a phrase that expresses an amount (or the absence) of something. It either modifies a noun phrase (e.g., *9 cats*) or is an independent complement (e.g., *I have 3*). Quantities include cardinals (*9*), determiners (*all*) or phrases (*almost 10*). If we detect a quantity, we replace it by a placeholder *Q* and add an annotation with the original quantity. The goal of this step is to unify extractions that only differ in quantities. For example, the phrases *9 cats*, *all cats* and *almost about 100 cats* are all rewritten to *Q cats*, only the quantity annotation differs.

We detect quantities by looking for numbers (NER types such as NUMBER or PERCENT) or words expressing quantities (such as *all, some, many*). We then extend these words via relevant typed dependencies, such as quantity modifiers (*quantmod*) and adverbial modifiers (*advmod*).

6 Minimization

After adding semantic annotations, MinIE minimizes extractions by dropping additional words. Since such minimization is risky, MinIE employs various minimization modes with different levels of aggressiveness, which effectively control the minimality-precision trade-off.

MinIE represents each constituent of an annotated extraction by its words, its dependency structure, its POS tags, and its named entities (detected by a named-entity recognizer). In general, each mode defines a set of *stable subconstituents*, which will always be fully retained, and subsequently searches for candidate words to drop outside of the stable subconstituents. Whenever a word is dropped from a constituent, we add an annotation with the original, unmodified constituent.

In all of MinIE's modes, noun sequences (which include the head) and named entities (from NER) are considered stable subconstituents. MinIE's minimization can be augmented with domain knowledge by providing information about additional stable subconstituents (e.g., collocations).

6.1 Complete Mode (MinIE-C)

MinIE's *complete mode* (MinIE-C) prunes all the extractions that contain subordinate clauses but does not otherwise modify the annotated extractions. The rationale is that extractions containing subordinate clauses are almost always overly specific. MinIE-C serves as a baseline.

6.2 Safe Mode (MinIE-S)

MinIE's *safe mode* only drops words which we consider universally safe to drop. We first drop all constituents that are covered by the implicit extractions discussed in Sec. 4.2 (e.g., *"Mr."* before persons). We then drop all determiners, possessive pronouns, adverbs modifying the verb in the relation, as well as adjectives and adverbs modifying words tagged as PERSON by the NER. An exception to these rules is given by named entities, which we consider as stable subconstituents (e.g., we do not drop *"Mr."* in (*Joe*, *cleans with*, *Mr. Muscle*)).

Note that this procedure cannot be considered safe when used on input extractions. We consider it safe, however, when applied to annotated extractions. In particular, all determiners, pronouns, and adverbs indicating negation, modality, or quantities are already processed and captured in annotations. The safe mode thus only performs simple rewrites such as *the great city* to *great city*, *his car* to *car*, *had also* to *had*, and *the eloquent president Mr. Barack Obama* to *Barack Obama*.

6.3 Dictionary Mode (MinIE-D)

Our dictionary mode uses a *dictionary $\mathcal{D}$ of stable constituents*. We first discuss how the dictionary is being used and subsequently how we construct it. An example is given in Fig. 1.

MinIE-D first performs all the minimizations of the safe mode and then searches for maximal noun phrases of the form $P \equiv [adverbial|adjective]^+$

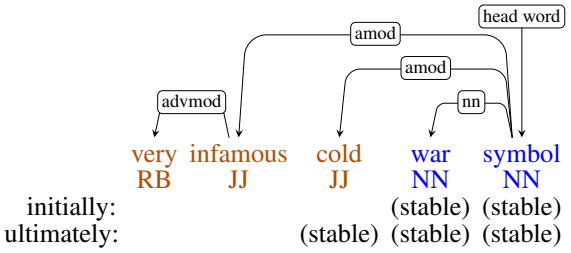
Pustejovsky (2012) plus WordNet synonyms.

[noun$^+$|ner]. For each instance of P, we drop a certain subset of its words. For example, a suitable minimization for *very infamous cold war symbol* (i.e., the Berlin wall) is *cold war symbol*, i.e., we consider *cold* as essential to the meaning of the constituent and *very infamous* as overly specific. The decision of what is considered essential and what overly specific is informed by dictionary $\mathcal{D}$. Note that in order to minimize mistakes, we consider for dropping only words in instances of pattern P. In particular, we do not touch subconstituents that contain prepositions because these are notoriously difficult to handle (e.g., we do not want to minimize *Bill of Rights* to *Bill*).

Our goal is to retain phrases occurring in $\mathcal{D}$, even if they occur in different order or with additional modifiers. We proceed as follows for each instance I of P. We first mark all nouns modifying the root (or the named entity) as *stable*. Afterwards, we create a set of *potentially stable subconstituents* (PSS). Each PSS is queried against dictionary $\mathcal{D}$. If it occurs in $\mathcal{D}$, all of its words are marked as stable. Once all PSS have been processed, we drop all words from I that are not marked stable. In our example, if $\{cold\ war\} \in \mathcal{D}$, we obtain *cold war symbol*.

To generate the set of PSS, we enumerate all syntactically valid subconstituents of I. For example, *infamous symbol* or *cold infamous war* are syntactically valid, whereas *very symbol* or *very cold war* are not. Conceptually,[2] we enumerate all subsequences of I and check whether (1) at least one noun (or named entity) is retained, and (2) whenever an adverb or adjective is not retained, neither are its modifiers. For each such subsequence, we generate all permutations of adverbial and adjective modifiers originating from the same dependency node, and each result as a PSS. This step ensures that the order of modifiers in I does influence whether or not a word is marked stable. The set of PSS for *very infamous cold war symbol* contains 22 entries.

The construction of dictionary $\mathcal{D}$ is inspired by the lexical constraint of Fader et al. (2011): Our assumption is that everything sufficiently frequent in a large corpus is not overly specific. To obtain $\mathcal{D}$, we process the entire corpus using the safe mode and include all frequent (e.g., frequency $\geq$ 10) subjects, relations, and arguments into $\mathcal{D}$. Applications can extend the dictionary using suitable collocations, either from domain-dependent dictionaries or by using methods to automatically extract collocations from a corpus (Gries, 2013).



PSS include: cold war symbol, cold symbol, **cold war**, infamous war symbol, infamous symbol, ...

Figure 1: Illustration of PSS generation in MinIE-D. Initially stable words are marked blue. Entries in dictionary $\mathcal{D}$ are printed in bold face.

6.4 Aggressive Mode (MinIE-A)

All previous modes aimed to be conservative. MinIE-A proceeds the other way around: all words for which we are not sure if they need to be retained are dropped. For every word in a constituent of an annotated extraction, we drop all adverbial, adjective, possessive, and temporal modifiers (along with their modifiers). We also drop prepositional attachments (e.g., *man with apples* becomes *man*), quantities modifying nouns, auxiliary modifiers to the main verb (e.g., *have escalated* becomes *escalated*), and all compound nouns that have a different named-entity type than their head word (e.g., *European Union official* becomes *official*). In most cases, after applying these steps, only a single word, named entity, or a sequence of nouns remains for subject and argument constituents.

7 Experimental Study

The goal of our experimental study was to investigate the differences in the various modes of MinIE w.r.t. precision, recall, and extraction length as well as to compare it with popular prior methods.

7.1 Experimental Setup

Source code, dictionaries, datasets, extractions, labels, and labeling guidelines are made available.[3]

Datasets. We used (1) 10,000 random sentences from the New York Times Corpus (NYT-

[2] We generate both instances of P as well as the set of PSS directly from the dependency structure of the constituent.

[3] http://dws.informatik.uni-mannheim.de/en/resources/software/minie/

10k) (Sandhaus, 2008), (2) a random sample of 200 sentences from the same corpus (NYT), and (3) a random sample of 200 sentences from Wikipedia (Wiki). NYT and Wiki were used in the evaluation of ClausIE and NestIE.[4]

Methods. We used ClausIE, OLLIE, and Stanford OIE as baseline systems. We adapted the publicly available version of ClausIE to Stanford CoreNLP 3.8.0 and implemented MinIE on top. For MinIE-D, we built dictionary $\mathcal{D}$ from the entire NYT and Wikipedia corpus, respectively.

Labeling. Labelers provided two labels per extraction of NYT and Wiki: one for the triple (without attribution) and one for the attribution. A triple is labeled as correct if it is entailed by its corresponding clause; here factuality annotations are taken into account but attribution errors are ignored. For example, all triples except #3 of Tab. 1 are considered correct. An attribution is incorrect if there is an attribution in the sentence which is neither present in the triple nor in the attribution annotation. In Tab. 1, the attribution is incorrect for extractions #2, #3, #5, and #6. Attribution is labeled only when the fact triple is labeled correct. See the labeling guidelines for further details.

Overall, there were more than 9,400 distinct extractions on NYT and Wiki. Each extraction was labeled by two independent labelers. We treat an extraction as *correct* if both labelers labeled it as *correct*. The inter-annotator agreement was moderate (NYT: Cohen's $\kappa = 0.53$, 78% of labels agree; Wiki: $\kappa = 0.5$, 79% of labels agree).

Measures. For each system, we measured the total number of extractions, the total number of correct triples (*recall*), the fraction of correct triples out of all extractions (*factual precision*), and the fraction of correct triples that have correct attributions (*attribution precision*). We also determined the mean word count per triple (μ) and its standard deviation (σ) as a proxy for minimality. Finally, as some systems produced a large number of redundant extractions, we also report the number of non-redundant extractions. For simplicity, we consider a triple t_1 redundant if it appears as subsequence in some other triple t_2 produced by the same extractor from the same sentence (e.g., extraction #5 in Tab. 1 is redundant given extraction #6).

[4]We did not use the OIE benchmark of Stanovsky and Dagan (2016) because it treats an extraction as correct if the heads of each constituent match the ones of a gold extraction. This is not suitable for us because it does not account for minimization (which does not change grammatical heads).

7.2 Extraction Statistics

In our first experiment, we used the larger but unlabeled NYT-10k dataset. The goal of this experiment was to investigate the total number of redundant and non-redundant extractions produced by each system and how frequently semantic annotations were produced (Tab. 3). For MinIE, we show the fraction of negative polarity and possibility annotations for triples only (i.e., we exclude the attribution polarity annotations).

In terms of number of extractions, MinIE (all modes) and Stanford OIE were roughly on par; OLLIE fell behind and ClausIE went ahead. The reason why ClausIE has more extractions than MinIE is that different (partly redundant) extractions from ClausIE may lead to the same minimized extraction. This is also also the reason why extraction numbers drop in the more aggressive modes of MinIE. We also determined the number of non-redundant extractions produced by each system and found that most systems produced only a moderate number of redundant extractions. A notable exception is Stanford OIE, which produced many extraction variants by dropping different subsets of words.

We observed that all modes of MinIE achieved significantly smaller extractions than ClausIE (its underlying OIE system), and that the average extraction length indeed dropped as we used more aggressive modes. Only MinIE-A produced shorter extractions than Stanford OIE. The main reason for the short extraction length of Stanford OIE is its aggressive creation of short redundant extractions (at the cost of precision; see below). We also found that to further minimize the extractions of MinIE-D, it is often necessary to minimize subjects and objects with prepositional modifiers (which MinIE currently avoids).

Only OLLIE and MinIE make use of annotations. The fraction of extracted attribution annotations was significantly smaller for OLLIE than for MinIE, mainly because OLLIE's attribution detection is limited to the *ccomp* dependency relation. Our results also indicate that MinIE frequently provides semantic annotations (with the notable exception of negative polarity).

7.3 Precision

In our second experiment, we compared the precision and recall of the various systems on the

	OLLIE	ClausIE	Stanford	MinIE-C	MinIE-S	MinIE-D	MinIE-A
# non-redundant extr.	20,557	36,173	16,350	**37,465**	37,093	36,921	36,474
# with redundant extr.	24,316	58,420	43,360	47,637	45,492	45,318	42,842
$\mu \pm \sigma$	9.9 ± 5.8	10.9 ± 7.0	6.6 ± 3.0	8.3 ± 4.9	7.2 ± 4.2	7.0 ± 4.1	$\mathbf{4.7 \pm 1.9}$
with attributions	6.8%	-	-	10.8%	10.8%	10.7%	10.8%
with negative polarity	-	-	-	3.8%	3.7%	3.7%	3.8%
with possibility	-	-	-	10.1%	9.9%	10.0%	9.7%
with quantity	-	-	-	17.6%	17.8%	17.8%	1.9%

Table 3: Results on the unlabeled NYT-10k dataset (μ=avg. extraction length, σ=standard deviation)

	OLLIE	ClausIE	Stanford	MinIE-C	MinIE-S	MinIE-D	MinIE-A
			NYT				
# non-redundant (correct/total)	246/414	505/821	178/342	**581/785**	574/781	569/777	439/753
# w/ redundant (correct/total)	302/497	**792/1300**	530/1052	727/970	690/924	681/916	505/860
factual prec.	(0.61)	(0.61)	(0.5)	**(0.75)**	**(0.75)**	(0.74)	(0.59)
attr. prec.	(0.9)	-	-	**(0.94)**	(0.93)	(0.93)	(0.93)
			Wiki				
# non-redundant (correct/total)	229/479	424/704	217/398	**500/666**	489/661	486/669	401/658
# w/ redundant (correct/total)	284/565	628/1002	651/1519	**635/851**	602/816	593/816	474/783
factual prec.	(0.50)	(0.63)	(0.43)	**(0.75)**	(0.74)	(0.73)	(0.61)
attr. prec.	**(0.97)**	-	-	**(0.97)**	(0.96)	(0.96)	**(0.97)**

Table 4: Results on the labeled NYT and Wiki datasets

smaller NYT and Wiki datasets. Our results are summarized in Tab. 4.

We found that Stanford OIE had the lowest factual precision and recall for non-redundant extractions throughout; it produced many incorrect and many redundant extractions (e.g., Stanford OIE produced 400 extractions from five sentences on NYT). For MinIE, the factual precision dropped as expected when we use more aggressive modes. Interestingly, the drop in precision between MinIE-C and MinIE-D was quite low, even though extractions get shorter. The aggressive minimization of MinIE-A led to a more severe drop in precision. Surprisingly to us, even MinIE's aggressive mode achieved precision comparable to ClausIE and higher than Stanford OIE. Note that MinIE-C, MinIE-S, and MinIE-D had higher precision than ClausIE. Reasons include that MinIE produces additional high-precision implicit extractions and breaks up very long and thus error-prone extractions.We also tried enriching the dictionary of MinIE-D with WordNet and Wiktionary collocations; the precision was almost the same.

As for attribution precision, most of the sentences in our samples did not contain attributions; these numbers thus have low accuracy. OLLIE and MinIE achieved similar results, even though MinIE additionally annotated attributions with factuality information.

Errors. For all modes, errors in dependency parsing transfer over to errors in MinIE, which we believe was the main source of error in MinIE-C and MinIE-S. For MinIE-D, we sometimes drop adjectives which in fact form collocations (e.g., "*assistant* director") with the noun they are modifying. This happens when the collocation is not present in the dictionary; better collocation dictionaries may address this problem. Another source of error stems from the NER (e.g., the first word of the entity *Personal Ensign* was not recognized).

8 Conclusions

We believe that the use of minimized extractions with semantic annotations are a promising direction for OIE. The techniques presented here can be seen as a step towards this goal, but there are still many open questions. Important directions include additional annotation types (e.g., temporal/spatial), use of background knowledge, better handling of collocations, the use of nested representations, and multilingual OIE.

Acknowledgments

We would like to thank Simone Paolo Ponzetto, Goran Glavaš, Stefano Faralli, Daniel Ruffinelli, and the anonymous reviewers for their invaluable feedback and support.

References

Alan Akbik and Alexander Löser. 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*, pages 52–56. Association for Computational Linguistics.

Gabor Angeli, Melvin Johnson Premkumar, and Christopher D Manning. 2015. Leveraging linguistic structure for open domain information extraction. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics.

Michele Banko, Michael J Cafarella, Stephen Soderland, Matthew Broadhead, and Oren Etzioni. 2007. Open information extraction from the web. In *International Joint Conference on Artificial Intelligence*, volume 7, pages 2670–2676.

Hannah Bast and Elmar Haussmann. 2013. Open information extraction via contextual sentence decomposition. In *International Conference on Semantic Computing*, pages 154–159. IEEE.

Nikita Bhutani, H V Jagadish, and Dragomir Radev. 2016. Nested propositions in open information extraction. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 55–64.

Eduardo Blanco and Dan I Moldovan. 2011. Some issues on detecting negation from text. In *Proceedings of the Twenty-Fourth International Florida Artificial Intelligence Research Society Conference*, pages 228–233.

Luciano Del Corro, Abdalghani Abujabal, Rainer Gemulla, and Gerhard Weikum. 2015. Finet: Context-aware fine-grained named entity typing. In *Conference on Empirical Methods in Natural Language Processing*, pages 868–878. Association for Computational Linguistics.

Luciano Del Corro and Rainer Gemulla. 2013. Clausie: clause-based open information extraction. In *Proceedings of the 22nd international conference on World Wide Web*, pages 355–366. Association for Computing Machinery.

Xin Dong, Evgeniy Gabrilovich, Geremy Heitz, Wilko Horn, Ni Lao, Kevin Murphy, Thomas Strohmann, Shaohua Sun, and Wei Zhang. 2014. Knowledge vault: A web-scale approach to probabilistic knowledge fusion. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 601–610. Association for Computing Machinery.

Anthony Fader, Stephen Soderland, and Oren Etzioni. 2011. Identifying relations for open information extraction. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 1535–1545. Association for Computational Linguistics.

Anthony Fader, Luke Zettlemoyer, and Oren Etzioni. 2014. Open question answering over curated and extracted knowledge bases. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1156–1165. Association for Computing Machinery.

Pablo Gamallo, Marcos Garcia, and Santiago Fernández-Lanza. 2012. Dependency-based open information extraction. In *Proceedings of the Joint Workshop on Unsupervised and Semi-Supervised Learning in NLP*, pages 10–18. Association for Computational Linguistics.

Stefan Th Gries. 2013. 50-something years of work on collocations: what is or should be next. *International Journal of Corpus Linguistics*, 18(1):137–166.

Alexander Löser, Sebastian Arnold, and Tillmann Fiehn. 2012. The goolap fact retrieval framework. In *Business Intelligence*, pages 84–97. Springer.

Mausam, Michael Schmitz, Robert Bart, Stephen Soderland, and Oren Etzioni. 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*, pages 523–534. Association for Computational Linguistics.

Andrea Moro and Roberto Navigli. 2012. Wisenet: Building a wikipedia-based semantic network with ontologized relations. In *Proceedings of the 21st ACM international conference on Information and knowledge management*, pages 1672–1676. ACM.

Andrea Moro and Roberto Navigli. 2013. Integrating syntactic and semantic analysis into the open information extraction paradigm. In *IJCAI*.

Ndapandula Nakashole, Gerhard Weikum, and Fabian Suchanek. 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*, pages 1135–1145. Association for Computational Linguistics.

Harinder Pal and Mausam. 2016. Demonyms and compound relational nouns in nominal open ie. In *Proceedings of Workshop on Automated Knowledge Base Construction*, pages 35–39.

Fabio Petroni, Luciano del Corro, and Rainer Gemulla. 2015. Core: Context-aware open relation extraction with factorization machines. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1763–1773. Association for Computational Linguistics.

Randolph Quirk, Sidney Greenbaum, Geoffrey Leech, Jan Svartvik, and David Crystal. 1985. *A comprehensive grammar of the English language*, volume 397. Cambridge Univ Press.

Sebastian Riedel, Limin Yao, Andrew McCallum, and Benjamin M Marlin. 2013. Relation extraction with matrix factorization and universal schemas. In *Joint Human Language Technology Conference/Annual Meeting of the North American Chapter of the Association for Computational Linguistics*.

Evan Sandhaus. 2008. The new york times annotated corpus. *Linguistic Data Consortium, Philadelphia*, 6(12):e26752.

Roser Saurí and James Pustejovsky. 2012. Are you sure that this happened? assessing the factuality degree of events in text. *Computational Linguistics*, 38(2):261–299.

Gabriel Stanovsky and Ido Dagan. 2016. Creating a large benchmark for open information extraction. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Austin, Texas. Association for Computational Linguistics.

Gabriel Stanovsky, Mausam, and Ido Dagan. 2015. Open ie as an intermediate structure for semantic tasks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics*, pages 303–308. Association for Computational Linguistics.

Fei Wu and Daniel S Weld. 2010. Open information extraction using wikipedia. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 118–127. Association for Computational Linguistics.

Mohamed Yahya, Steven Whang, Rahul Gupta, and Alon Y Halevy. 2014. Renoun: Fact extraction for nominal attributes. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 325–335. Association for Computational Linguistics.