

Nested Propositions in Open Information Extraction

Nikita Bhutani
Department of EECS
University of Michigan
Ann Arbor
nbhutani@umich.edu

H. V. Jagadish
Department of EECS
University of Michigan
Ann Arbor
jag@umich.edu

Dragomir Radev
Department of EECS
University of Michigan
Ann Arbor
radev@umich.edu

Abstract

The challenges of Machine Reading and Knowledge Extraction at a web scale require a system capable of extracting diverse information from large, heterogeneous corpora. The Open Information Extraction (OIE) paradigm aims at extracting assertions from large corpora without requiring a vocabulary or relation-specific training data. Most systems built on this paradigm extract binary relations from arbitrary sentences, ignoring the context under which the assertions are correct and complete. They lack the expressiveness needed to properly represent and extract complex assertions commonly found in the text. To address the lack of representation power, we propose NESTIE, which uses a nested representation to extract higher-order relations, and complex, interdependent assertions. Nesting the extracted propositions allows NESTIE to more accurately reflect the meaning of the original sentence. Our experimental study on real-world datasets suggests that NESTIE obtains comparable precision with better minimality and informativeness than existing approaches. NESTIE produces 1.7-1.8 times more minimal extractions and achieves 1.1-1.2 times higher informativeness than CLAUSIE.

1 Introduction

Syntactic analyses produced by syntactic parsers are a long way from representing the full meaning of the sentences parsed. In particular, they cannot support questions like “*Who did what to whom?*”, “*Where did what happen?*”. Owing to the large, heterogeneous corpora available at web scale, traditional

approaches to information extraction (Brin, 1998; Agichtein and Gravano, 2000) fail to scale to the millions of relations found on the web. As a response, the paradigm of Open Information Extraction (OIE) (Banko et al., 2007) has seen a rise in interest as it eliminates the need for domain knowledge or relation-specific annotated data. OIE systems use a collection of patterns over the surface form or dependency tree of a sentence to extract propositions of the form $(arg1, rel, arg2)$.

However, state-of-the-art OIE systems, REVERB (Fader et al., 2011) and OLLIE (Schmitz et al., 2012) focus on extracting binary assertions and suffer from three key drawbacks. First, lack of expressivity of representation leads to significant information loss for higher-order relations and complex assertions. This results in incomplete, uninformative and incoherent prepositions. Consider Example 1 in Figure 1. Important contextual information is either ignored or is subsumed in over-specified argument and relation phrases. It is not possible to fix such nuances by post-processing the propositions. This affects downstream applications like Question Answering (Fader et al., 2014) which rely on correctness and completeness of the propositions.

Second, natural language frequently includes relations presented in a non-canonical form that cannot be captured by a small set of extraction patterns that only extract relation mediated by verbs or a subset of verbal patterns. Consider Example 2 in Figure 1 that asserts, “*Rozsa Hill is the third hill near the river*”, “*Rozsa Hill is Rose Hill*” and “*Rozsa Hill lies north of Castle Hill*”. A verb-mediated pattern would extract

1. After giving 5,000 people a second chance at life, doctors are celebrating the 25th anniversary of Britain's first heart transplant.	
R:	P1: (doctors, are celebrating the 25th anniversary of, Britain 's first heart transplant)
O:	P1: (doctors, are celebrating, the 25th anniversary of Britain's first heart transplant)
N:	P1: (doctors, are celebrating, the 25th anniversary of Britain's first heart transplant) P2: (doctors, giving, second chance at life) P3: (P1, after, P2)
2. Rozsa (Rose) Hill , the third hill near the river, lies north of Castle Hill.	
R:	P1: (the third hill, lies north of, Castle Hill)
O:	P1: (the third hill, lies north of, Castle Hill)
N:	P1: (Rozsa, lies, north of Castle Hill) P2: (Rozsa Hill, is, third hill near the river) P3: (Rozsa Hill, is, Rose)
3. "A senior official in Iraq said the body, which was found by U.S. military police, appeared to have been thrown from a vehicle."	
R:	P1: (Iraq, said, the body) P2: (the body, was found by, U.S. military police)
O:	P1: (A senior official in Iraq, said, the body which was found by U.S. military police)
N:	P1: (body, appeared to have been thrown, ∅) P2: (P1, from, vehicle) P3: (A senior official in Iraq, said, P2) P4: (U.S. military police, found, body)

Figure 1: Example propositions from OIE systems: REVERB (R), OLLIE (O) and NESTIE(N).

a triple, (the third hill, lies north of, Castle Hill) that is less informative than a triple, (Rozsa, lies, north of Castle Hill) which is not mediated by a verb in the original sentence. Furthermore, these propositions are not complete. Specifically, queries of the form ‘What is the other name of Rozsa Hill?’, ‘Where is Rozsa Hill located?’, ‘Which is the third hill near the river?’ will either return no answer or return an uninformative answer with these propositions. Since information is encoded at various granularity levels, there is a need for a representation rich enough to express such complex relations and sentence constructions.

Third, OIE systems tend to extract propositions with long argument phrases that are not minimal and are difficult to disambiguate or aggregate for downstream applications. For instance, the argu-

ment phrase, *body which was found by U.S. military police*, is less likely to be useful than the argument phrase, *body* in Example 3 in Figure 1.

In this paper we present NESTIE, which overcomes these limitations by 1) expanding the proposition representation to nested expressions so additional contextual information can be captured, 2) expanding the syntactic scope of relation phrases to allow relations mediated by other syntactic entities like nouns, adjectives and nominal modifiers. NESTIE bootstraps a small set of extraction patterns that cover simple sentences and learns broad-coverage relation-independent patterns. We believe that it is possible to adapt OIE systems that extract verb-based relations to process assertions denoting events with many arguments, and learn other non-clausal relations found in the text. With weakly-supervised learning techniques, patterns encoding these relations can be learned from a limited amount of data containing sentence equivalence pairs.

This article is organized as follows. We provide background on OIE in Sec. 2 followed by an overview of our proposed solution in Sec. 3. We then discuss how the extraction patterns for nested representations are learned in Sec. 4. In Sec. 5, we compare NESTIE against alternative methods on two datasets: Wikipedia and News. In Sec. 6, we discuss related work on pattern-based information extraction.

2 Background

The key goal of OIE is to obtain a shallow semantic representation of the text in the form of tuples consisting of argument phrases and a phrase that expresses the relation between the arguments. The phrases are identified automatically using domain-independent syntactic and lexical constraints. Some OIE systems are:

TextRunner (Yates et al., 2007) **WOE** (Wu and Weld, 2010): They use a sequence-labeling graphical model on extractions labeled automatically using heuristics or distant supervision. Consequently, long-range dependencies, holistic and lexical aspects of relations tend to get ignored.

ReVerb (Fader et al., 2011): Trained with shallow syntactic features, REVERB uses a logistic regression classifier to extract relations that begin with a

verb and occur between argument phrases.

Ollie (Schmitz et al., 2012): Bootstrapping from REVERB extractions, OLLIE learns syntactic and lexical dependency parse-tree patterns for extraction. Some patterns reduce higher order relations to ReVerb-style relation phrases. Also, representation is extended optionally to capture contextual information about conditional truth and attribution for extractions.

ClausIE (Del Corro and Gemulla, 2013): Using linguistic knowledge and a small set of domain-independent lexica, CLAUSIE identifies and classifies clauses into clause types, and then generates extractions based on the clause type. It relies on a predefined set of rules on how to extract assertions instead of learning extraction patterns. Also, it doesn't capture the relations between the clauses.

There has been some work in open-domain information extraction to extract higher-order relations. KRAKEN (Akbik and Löser, 2012) uses a predefined set of rules based on dependency parse to identify fact phrases and argument heads within fact phrases. But unlike alternative approaches, it doesn't canonicalize the fact phrases. There is another body of work in natural language understanding that shares tasks with OIE. AMR parsing (Banarescu et al.,), semantic role labeling (SRL) (Toutanova et al., 2008; Punyakanok et al., 2008) and frame-semantic parsing (Das et al., 2014). In these tasks, verbs or nouns are analyzed to identify their arguments. The verb or noun is then mapped to a semantic frame and roles of each argument in the frame are identified. These techniques have gained interest with the advent of hand-constructed semantic resources like PropBank and FrameNet (Kingsbury and Palmer, 2002; Baker et al., 1998). Generally, the verb/noun and the semantically labeled arguments correspond to OIE propositions and, therefore, the two tasks are considered similar. Systems like SRL-IE (Christensen et al., 2010) explore if these techniques can be used for OIE. However, while OIE aims to identify the relation/predicate between a pair of arguments, frame-based techniques aim to identify arguments and their roles with respect to a predicate. Hence, the frames won't correspond to propositions when both the arguments cannot be identified for a binary relation or when the correct argument is buried in long argument phrases.

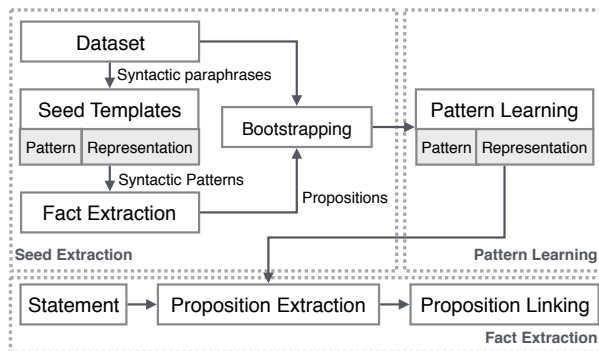


Figure 2: System Architecture of NESTIE.

3 Task Definition and NestIE Overview

Task: We focus on the task of OIE, where the system takes a natural language statement and extracts the supported assertions. This is achieved by using an extractor that uses nested representations to extract propositions and a linker that connects extracted propositions to capture context.

Proposition-based Extractor: We propose a framework to extend open-domain binary-relation extractors to extract n-ary and complex relations. As not all assertions can be expressed as $(arg1, rel, arg2)$, we learn syntactic patterns for relations that are expressed as nested templates like, $(arg1, rel, (arg2, rel2, arg3))$, $((arg1, rel, arg2), rel2, arg3)$.

Proposition Linking: In practice, it is infeasible to enumerate simple syntactic pattern templates that capture the entire meaning of a sentence. Also, increasing the complexity of templates would lead to sparsity issues while bootstrapping. We assume that there is a finite set of inter-proposition relations that can be captured using a small set of rules which take into account the structural properties of the propositions and syntactic dependencies between the relation phrases of the propositions.

System Evaluation: To compare NESTIE to other alternative methods, we conduct an experimental study on two real-world datasets: Wikipedia and News. Propositions from each system are evaluated for correctness, minimality, and informativeness.

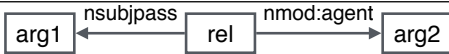
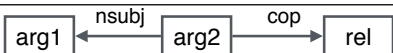
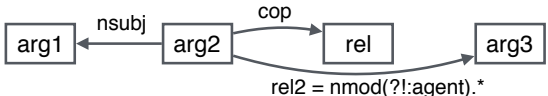
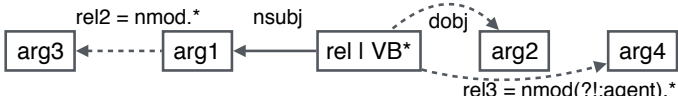
Template	Example
Pattern:  Representation: T: (arg1, [rel, by], arg2)	A body has been found by police. (body, [found, by], police)
Pattern:  Representation: T: (arg1, be, arg2)	Fallujah is an Iraqi city. (Fallujah, is, city)
Pattern:  Representation: T: (arg1, be, [arg2, rel2, arg3])	Ghazi al-Yawar is new president of Iraq. (Yawar, is, [president, of, Iraq])
Pattern:  Representation: T1: ([arg1, rel2, arg3], rel, arg2) T2: (T1, rel3, arg4)	10,000 people in Africa died of Ebola. T1: ([people, in, Africa], died, ∅) T2: (T1, of, Ebola)

Figure 3: Seed templates and corresponding representation.

4 Proposition Extraction

Figure 2 illustrates the system architecture of NESTIE. First, a set of high-precision seed templates is used to extract propositions. A template maps a dependency parse-tree pattern to a triple representation such as $(arg1, rel, arg2)$ for binary relations, or a nested triple representation such as $((arg1, rel, arg2), rel2, arg3)$ for n-ary relations. Furthermore, an argument is allowed to be a sequence of words, “arg2 rel2 arg3” to capture its nominal modifiers. Then, using a RTE dataset that contains syntactic paraphrases, NESTIE learns equivalent parse-tree patterns for each template in the seed set. These patterns are used to extract propositions which are then linked.

4.1 Constructing Seed Set

We use a set of 13 hand-written templates. Each template maps an extraction pattern for a simple sentence to corresponding representation. A subset of these templates is shown in Figure 3. To create a seed set of propositions, we use the RTE dataset which is comprised of statements and their entailed hypotheses. We observed that most of the hypotheses were syntactic variants of the facts in their corresponding statements. These hypotheses were also short with a single, independent clause. These shared sentence constructions could be cap-

tured with a small set of templates. We iteratively create templates until at least one proposition could be extracted for each hypothesis. The propositions from the hypotheses form the set for bootstrapping.

For each seed proposition extracted from a hypothesis, the statement entailing the hypothesis contains all the content words of the proposition and expresses the same information as the proposition. However, there is a closed class of words, such as prepositions, a subset of adverbs, determiners, verbs etc. that does not modify the underlying meaning of the hypothesis or the statement and can be considered auxiliary. These were ignored while constructing the seed set.

Example 1 Consider a statement-hypothesis pair, Statement: *Paul Bremer, the top U.S. civilian administrator in Iraq, and Iraq’s new president, Ghazi al-Yawar, visited the northern Iraqi city of Kirkuk.*

Hypothesis: *Ghazi al-Yawar is the president of Iraq.* The hypothesis is entailed in the statement. The seed templates extract propositions from the hypothesis: $(al-Yawar, is, president, (al-Yawar, is, president\ of\ Iraq))$, and $(al-Yawar, is\ president\ of, Iraq)$.

Bootstrapping is a popular technique to generate positive training data for information extraction (Collins and Singer, 1999; Hoffmann et al., 2011). We extend the bootstrapping techniques employed

in OLLIE and RENOUN, for n-ary and complex relations. First, instead of learning dependency parse-tree patterns connecting the heads of the argument phrases and the relation phrase connecting them, we learn the dependency parse-tree patterns connecting the heads of all argument and relation phrases in the template. This allows greater coverage of context for the propositions and prevents the arguments/relations from being over-specified and/or uninformative. Second, some of the relations in the representation are derived from the type of dependency, e.g. type of nominal modifier. As these relations are implicit, and might not be present in the paraphrase, they are ignored for learning. Intuitively, with such constraints, paraphrases “*Mary gave John a car*” and “*Mary gave a car to John*” can map to the same representation.

4.2 Extraction Pattern Learning

The biggest challenge in information extraction is the multitude of ways in which information can be expressed. Since it is not possible to enumerate all the different syntactic variations of an assertion, there is a need to learn general patterns that encode the various ways of expressing the assertion. In particular, we learn the various syntactic patterns that can encode the same information as the seed patterns and hence can be mapped to same representation.

NESTIE tries to learn the different ways in which the content words of a seed proposition from a hypothesis can be expressed in the statement that entails this hypothesis. We use the Stanford dependency parser (De Marneffe et al., 2006) to parse the statement and identify the path connecting the content words in the parse tree. If such a path exists, we retain the syntactic constraints on the nodes and edges in the path and ignore the surface forms of the nodes in the path. This helps generalize the learned patterns to unseen relations and arguments. NESTIE could learn 183 templates from the 13 seed templates. Figure 4 shows a subset of these patterns.

Example 2 Consider dependency parse-subtree of the statement and hypothesis from Example 1,
Statement: *Iraq* $\xrightarrow{\text{poss}}$ *president* $\xrightarrow{\text{appos}}$ *al - Yawar*
Hypothesis: *al - Yawar* $\xleftarrow{\text{nsubj}}$ *president* $\xrightarrow{\text{of}}$ *Iraq*
A seed extraction pattern maps the parse-tree of the hypothesis to the representation,

(arg1, be, arg2), returning proposition, (al-Yawar, is, president of Iraq).

With bootstrapping, the syntactic pattern from the statement is mapped to the same representation.

4.3 Pattern Matching

Once the extraction patterns are learned, we use these patterns to extract propositions from new unseen sentences. We first parse a new sentence and match the patterns against the parse tree. As the patterns only capture the heads of the arguments and relations, we expand the extracted propositions to increase the coverage of context of the arguments as in the original sentence.

Example 3 In the statement from Example 1, the extraction patterns capture the dependency path connecting the head words: Iraq, administrator and Paul Bremer. However, to capture the contextual information, further qualification of the argument node, administrator, is required.

Following this observation, we expand the arguments on nmod, amod, compound, nummod, det, neg edges. We expand the relations on advmod, neg, aux, auxpass, cop, nmod edges. Only the dependency edges not captured in the pattern are considered for expansion. Also, the order of words from the original sentence is retained in the argument phrases.

4.4 Proposition Linking

NESTIE uses a nested representation to capture the context of extracted propositions. The context could include condition, attribution, belief, order, reason and more. Since it is not possible to generate or learn patterns that can express these complex assertions as a whole, NESTIE links the various propositions from the previous step to generate nested propositions that are complete and closer in meaning to the original statement.

The proposition linking module is based on the assumption that the inter-proposition relation can be inferred from the dependency parse of the sentence from which propositions were extracted. Some of the rules employed to link the propositions are:

- The relation of proposition P1 has a relationship to the relation of proposition P2.

Template	Seed Pattern	Learned Pattern
Pattern:		
Representation:	T: (arg1, [rel, by], arg2)	
Pattern:		
Representation:	T: (arg1, be, arg2)	
Pattern:		
Representation:	T: (arg1, be, [arg2, rel2, arg3])	
Pattern:		
Representation:	T1:([arg1, rel2, arg3], rel, arg2), T2: (T1, rel3, arg4)	

Figure 4: Syntactic Patterns learned using bootstrapping.

Consider the statement, “The accident happened after the chief guest had left the event.” and propositions, P1: (accident, happen, ϕ) and P2: (chief guest, had left, event). Using dependency edge, `nmod:after`, the linking returns (P1, after, P2).

- Proposition P1 is argument in proposition P2.

Consider the statement, “A senior official said the body appeared to have been thrown from a vehicle.” and propositions, P1: (body, appeared to have been thrown from, vehicle) and P2: (senior official, said, ϕ). The linking updates P2 to (senior official, said, P1).

- An inner nested proposition is replaced with a more descriptive alternative proposition.

We use dependency parse patterns to link propositions. We find correspondences between: a `ccomp` edge and a clausal complement, an `advcl` edge and a conditional, a `nmod` edge and a relation modifier. For clausal complements, a null argument in the source proposition is updated with the target proposition. For conditionals and nominal modifiers, a new proposition is created with the source and target propositions as arguments. The relation of the new proposition is derived from the target of the `mark` edge from the relation head of target proposition.

4.5 Comparison with Ollie

NESTIE uses an approach similar to OLLIE and WOE to learn dependency parse based syntactic patterns. However, there are significant differences. First, OLLIE and WOE rely on extractions from REVERB and Wikipedia info-boxes respectively for bootstrapping. Most of these relations are binary. On the contrary, our algorithm is based on high-confidence seed templates that are more expressive and hence learn patterns expressing different ways in which the proposition as a whole can be expressed. Though the arguments in OLLIE can be expanded to include the n-ary arguments, NESTIE encodes them in the seed templates and learns different ways of expressing these arguments. Also, similar to OLLIE, NESTIE can extract propositions that are not just mediated by verbs.

5 Experiments

We conducted an experimental study to compare NESTIE to other state-of-the-art extractors. We found that it achieves higher informativeness and produces more correct and minimal propositions than other extractors.

5.1 Experimental Setup

We used two datasets released by (Del Corro and Gemulla, 2013) in our experiments: 200 random sentences from Wikipedia, and 200 random sentences from New York Times (NYT). We compared

Dataset		Reverb	Ollie	ClausIE	NestIE
NYT dataset	Avg. Informativeness	1.437/5	2.09/5	2.32/5	2.762/5
	Correct	187/275 (0.680)	359/529 (0.678)	527/882 (0.597)	469/914 (0.513)
	Minimal (among correct)	161/187 (0.861)	238/359 (0.663)	199/527 (0.377)	355/469 (0.757)
Wikipedia dataset	Avg. Informativeness	1.63/5	2.267/5	2.432/5	2.602/5
	Correct	194/258 (0.752)	336/582 (0.577)	453/769 (0.589)	415/827 (0.501)
	Minimal (among correct)	171/194 (0.881)	256/336 (0.761)	214/453 (0.472)	362/415 (0.872)

Figure 5: Informativeness and number of correct and minimal extractions as fraction of total extractions.

NESTIE against three OIE systems: REVERB, OLLIE and CLAUSIE. Since the source code for each of the extractors was available, we independently ran the extractors on the two datasets. Next, to make the extractions comparable, we configured the extractors to generate triple propositions. REVERB and CLAUSIE extractions were available as triples by default. OLLIE extends its triple proposition representation. So, we generated an additional extraction for each of the possible extensions of a proposition. NESTIE uses a nested representation. So, we simply extracted the innermost proposition in a nested representation as a triple and allowed the subject and the object in the outer proposition to contain a *reference* to the inner triple. By preserving references the context of a proposition is retained while allowing for queries at various granularity levels.

We manually labeled the extractions obtained from all extractors to 1) maintain consistency, 2) additionally, assess if extracted triples were informative and minimal. Some extractors use heuristics to identify arguments and/or relation phrase boundaries, which leads to over-specific arguments that render the extractions unusable for other downstream applications. To assess the usability of extractions, we evaluated them for minimality (Bast and Haussmann, 2013). Furthermore, the goal of our system is to extract as many propositions as possible and lose as little information as possible. We measure this as *informativeness* of the set of the extractions for a sentence. Since computing informativeness as a percentage of text contained in at least one extraction could be biased towards long extractions, we used an explicit rating scale to measure informativeness.

Two CS graduate student labeled each extraction for correctness (0 or 1) and minimality (0 or 1). For

each sentence, they label the set of extractions for informativeness (0-5). An extraction is marked correct if it is asserted in the text and correctly captures the contextual information. An extraction is considered minimal if the arguments are not over-specified i.e. they don't subsume another extraction or have conjunctions or are excessively long. Lastly, they rank the set of extractions on a scale of 0-5 (0 for bad, 5 for good) based on the coverage of information in the original sentence. The agreement between labelers was measured in terms of Cohens Kappa.

5.2 Comparative Results

The results of our experimental study are summarized in Figure 5 which shows the number of correct and minimal extractions, as well as the total number of extractions for each extractor and dataset. For each dataset, we also report the macro-average of informativeness reported by the labelers. We found moderate inter-annotator agreement: 0.59 on correctness and 0.53 on minimality for both the datasets. Each extractor also includes a confidence score for the propositions. But since each extractor has its unique method to find confidence, we compare the precision over all the extractions instead of a subset of high-confidence extractions.

NESTIE produced many more extractions, and more informative extractions than other systems. There appears to be a trade-off between informativeness and correctness (which are akin to recall and precision, respectively). CLAUSIE is the system with results closer to NESTIE than other systems. However, the nested representation and proposition linking used by NESTIE produce substantially more (1.7-1.8 times more) minimal extractions than CLAUSIE, which generates propositions from the constituents of the clause. Learning non-verb medi-

ated extraction patterns and proposition linking also increase the syntactic scope of relation expressions and context. This is also reflected in the average informativeness score of the extractions. NESTIE achieves 1.1-1.9 times higher informativeness score than the other systems.

We believe that nested representation directly improves minimality, independent of other aspects of extractor design. To explore this idea, we conducted experiments on OLLIE, which does not expand the context of the arguments heuristically unlike other extractors. Of the extractions labeled correct but not minimal by the annotators on the Wikipedia dataset, we identified extractions that satisfy one of: 1) has an argument for which there is an equivalent extraction (nested extractions), 2) shares the same subject with another extraction whose relation phrase contains the relation and object of this extraction (n-ary extractions), 3) has an object with conjunction. Any such extractions can be made minimal and informative with a nested representation. 73.75% of the non-minimal correct extractions met at least one of these conditions, so by a post-processing step, we could raise the minimality score of OLLIE by 17.65%, from 76.1% to 93.75%.

5.3 Error Analysis of NestIE

We did a preliminary analysis of the errors made by NESTIE. We found that in most of the cases (about 33%-35%), extraction errors were due to incorrect dependency parsing. This is not surprising as NESTIE relies heavily on the parser for learning extraction patterns and linking propositions. An incorrect parse affects NESTIE more than other systems which are not focused on extracting finer grained information and can trade-off minimality for correctness. An incorrect parse not only affects the pattern matching but also proposition linking which either fails to link two propositions or produces an incorrect proposition.

Example 4 Consider the statement, “A day after strong winds stirred up the Hauraki Gulf and broke the mast of Team New Zealand, a lack of wind caused Race 5 of the America’s Cup to be abandoned today.”. The statement entails following assertions:

A1: “strong winds stirred up the Hauraki Gulf”

A2: “strong winds broke the mast of Team New Zealand”

A3: “a lack of wind caused Race 5 of the America’s Cup to be abandoned”

A1 and A2 are parsed correctly. A3 is parsed incorrectly with *Race 5* as object of the verb *caused*. Some extractors either don’t capture A3 or return an over-specified extraction, (*a lack of wind, caused, Race 5 of the America’s Cup to be abandoned today*). Such an extraction is correct but not minimal.

To maintain minimality, NESTIE aims to extract propositions, P1: (*Race 5 of the America’s Cup, be abandoned, ϕ*) and P2: (*a lack of wind, caused, P1*). However, it fails because of parser errors. It extracts incorrect proposition, P3: (*a lack of wind, caused, Race 5*) corresponding to A3 and links it to propositions for A1 and A2. Linking an incorrect proposition generates more incorrect propositions which hurt the system performance.

However, we hope this problem can be alleviated to some extent as parsers become more robust. Another approach could be to use clause segmentation to first identify clause boundaries and then use NESTIE on reduced clauses. As the problem becomes more severe for longer sentences, we wish to explore clause processing for complex sentences in future.

Another source of errors was under-specified propositions. Since our nested representation allows null arguments for intransitive verb phrases and for linking propositions, failure to find an argument/proposition results in an under-specified extraction. We found that 27% of the errors were because of null arguments. However, by ignoring extractions with null arguments we found that precision increases by only 4%-6% (on Wikipedia). This explains that many of the extractions with empty arguments were correct, and need special handling. Other sources of errors were: aggressive generalization of an extraction pattern to unseen relations (24%), unidentified dependency types while parsing long, complex sentences (21%), and errors in expanding the scope of arguments and linking extractions (20%).

6 Related Work

As OIE has gained popularity to extract propositions from large corpora of unstructured text, the problem of the extractions being uninformative and incomplete has surfaced. A recent paper (Bast and Haussmann, 2014) pointed out that a significant fraction of the extracted propositions is not informative. A simple inference algorithm was proposed that uses generic rules for each semantic class of predicate to derive new triples from extracted triples. Though it improved the informativeness of extracted triples, it did not alleviate the problem of lost context in complex sentences. We, therefore, create our own extractions.

Some recent works (Bast and Haussmann, 2013; Angeli et al., 2015) have tried to address the problem of long and uninformative extractions in open-domain information extraction by finding short entailment or clusters of semantically related constituents from a longer utterance. These clusters are reduced to triples using schema mapping to known relation types or using a set of hand-crafted rules. NESTIE shares similar objectives but uses bootstrapping to learn extraction patterns.

Bootstrapping and pattern learning has a long history in traditional information extraction. Systems like DIPRE (Brin, 1998), SNOWBALL (Agichtein and Gravano, 2000), NELL (Mitchell, 2010), and OLLIE bootstrap based on seed instances of a relation and then learn patterns for extraction. We follow a similar bootstrapping algorithm to learn extraction patterns for n-ary and nested propositions.

Using a nested representation to express complex and n-ary assertions has been studied in closed-domain or ontology-aided information extraction. Yago (Suchanek et al., 2008) and (Nakashole and Mitchell, 2015) extend binary relations to capture temporal, geospatial and prepositional context information. We study such a representation for open-domain information extraction.

7 Conclusions

We presented NESTIE, a novel open information extractor that uses nested representation for expressing complex propositions and inter-propositional relations. It extends the bootstrapping techniques of previous approaches to learn syntactic extraction pat-

terns for the nested representation. This allows it to obtain higher informativeness and minimality scores for extractions at comparable precision. It produces 1.7-1.8 times more minimal extractions and achieves 1.1-1.2 times higher informativeness than CLAUSEIE. Thus far, we have tested our bootstrap learning and proposition linking approaches only on a small dataset. We believe that its performance will improve with larger datasets. NESTIE can be seen as a step towards a system that has a greater awareness of the context of each extraction and provides informative extractions to downstream applications.

Acknowledgments

This research was supported in part by NSF grants IIS 1250880 and IIS 1017296.

References

- Eugene Agichtein and Luis Gravano. 2000. Snowball: Extracting relations from large plain-text collections. In *Proceedings of the fifth ACM conference on Digital libraries*, pages 85–94.
- Alan Akbik and Alexander Löser. 2012. Kraken: N-ary facts in open information extraction. In *Proceedings of the AKBC-WEKEX*, pages 52–56.
- Gabor Angeli, Melvin Johnson Premkumar, and Christopher D Manning. 2015. Leveraging linguistic structure for open domain information extraction. In *Proceedings of ACL*, pages 26–31.
- Collin F Baker, Charles J Fillmore, and John B Lowe. 1998. The berkeley framenet project. In *Proceedings of the 17th international conference on Computational linguistics-Volume 1*, pages 86–90.
- Laura Banarescu, Claire Bonial, Shu Cai, Madalina Georgescu, Kira Griffitt, Ulf Hermjakob, Kevin Knight, Philipp Koehn, Martha Palmer, and Nathan Schneider. Abstract meaning representation (amr) 1.0 specification.
- Michele Banko, Michael J Cafarella, Stephen Soderland, Matthew Broadhead, and Oren Etzioni. 2007. Open information extraction for the web. In *IJCAI*, volume 7, pages 2670–2676.
- Hannah Bast and Elmar Haussmann. 2013. Open information extraction via contextual sentence decomposition. In *IEEE-ICSC 2013*, pages 154–159.
- Hannah Bast and Elmar Haussmann. 2014. More informative open information extraction via simple inference. In *Advances in information retrieval*, pages 585–590. Springer.

- Sergey Brin. 1998. Extracting patterns and relations from the world wide web. In *The World Wide Web and Databases*, pages 172–183. Springer.
- Janara Christensen, Stephen Soderland, Oren Etzioni, et al. 2010. Semantic role labeling for open information extraction. In *Proceedings of the NAACL-HLT 2010*, pages 52–60.
- Michael Collins and Yoram Singer. 1999. Unsupervised models for named entity classification. In *Proceedings of the joint SIGDAT conference on empirical methods in natural language processing and very large corpora*, pages 100–110. Citeseer.
- Dipanjan Das, Desai Chen, André FT Martins, Nathan Schneider, and Noah A Smith. 2014. Frame-semantic parsing. *Computational linguistics*, 40(1):9–56.
- Marie-Catherine De Marneffe, Bill MacCartney, Christopher D Manning, et al. 2006. Generating typed dependency parses from phrase structure parses. In *Proceedings of LREC*, volume 6, pages 449–454.
- Luciano Del Corro and Rainer Gemulla. 2013. Clausie: clause-based open information extraction. In *Proceedings of the IW3C2*, pages 355–366.
- Anthony Fader, Stephen Soderland, and Oren Etzioni. 2011. Identifying relations for open information extraction. In *Proceedings of EMNLP*, pages 1535–1545.
- Anthony Fader, Luke Zettlemoyer, and Oren Etzioni. 2014. Open question answering over curated and extracted knowledge bases. In *Proceedings of ACM-SIGKDD*, pages 1156–1165. ACM.
- Raphael Hoffmann, Congle Zhang, Xiao Ling, Luke Zettlemoyer, and Daniel S Weld. 2011. Knowledge-based weak supervision for information extraction of overlapping relations. In *Proceedings of ACL-HLT*, pages 541–550.
- Paul Kingsbury and Martha Palmer. 2002. From treebank to propbank. In *LREC*.
- Tom Mitchell. 2010. Never-ending learning. Technical report, DTIC Document, Carnegie Mellon University.
- Ndapandula Nakashole and Tom M Mitchell. 2015. A knowledge-intensive model for prepositional phrase attachment. In *Proceedings of ACL*, pages 365–375.
- V. Punyakanok, D. Roth, and W. Yih. 2008. The importance of syntactic parsing and inference in semantic role labeling. *Computational Linguistics*, 34(2).
- Michael Schmitz, Robert Bart, Stephen Soderland, Oren Etzioni, et al. 2012. Open language learning for information extraction. In *Proceedings of EMNLP-CoNLL 2012*, pages 523–534.
- Fabian M Suchanek, Gjergji Kasneci, and Gerhard Weikum. 2008. Yago: A large ontology from wikipedia and wordnet. *Web Semantics: Science, Services and Agents on the World Wide Web*, 6(3):203–217.
- Kristina Toutanova, Aria Haghighi, and Christopher D Manning. 2008. A global joint model for semantic role labeling. *Computational Linguistics*, 34(2):161–191.
- Fei Wu and Daniel S Weld. 2010. Open information extraction using wikipedia. In *Proceedings of the ACL*, pages 118–127.
- Alexander Yates, Michael Cafarella, Michele Banko, Oren Etzioni, Matthew Broadhead, and Stephen Soderland. 2007. Texrunner: open information extraction on the web. In *Proceedings of NAACL-HLT: Demonstrations*, pages 25–26.