

Bootstrapping for Numerical Open IE

Swarnadeep Saha

Department of CSE
I.I.T. Delhi
writetoswarna@gmail.com

Harinder Pal*

Microsoft Corporation
hapal@microsoft.com

Mausam

Department of CSE
I.I.T. Delhi
mausam@cse.iitd.ac.in

Abstract

We design and release BONIE, the first open *numerical* relation extractor, for extracting Open IE tuples where one of the arguments is a number or a quantity-unit phrase. BONIE uses bootstrapping to learn the specific dependency patterns that express numerical relations in a sentence. BONIE’s novelty lies in task-specific customizations, such as inferring *implicit* relations, which are clear due to context such as units (for e.g., ‘square kilometers’ suggests area, even if the word ‘area’ is missing in the sentence). BONIE obtains 1.5x yield and 15 point precision gain on numerical facts over a state-of-the-art Open IE system.

1 Introduction

Open Information Extraction (Open IE) systems extract relational tuples from text, without requiring a pre-specified vocabulary (Etzioni et al., 2008; Mausam, 2016), by constructing the relation phrases and arguments from within the sentences themselves. Early works on Open IE such as REVERB (Etzioni et al., 2011) extract verb-mediated relations via a handful of human-defined patterns. OLLIE improves recall by learning dependency patterns, using bootstrapping over REVERB extractions (Mausam et al., 2012). Open IE 4.2, a state-of-the-art open information extractor, is based on a combination of SRLIE, a verb-mediated extractor over SRL frames (Christensen et al., 2011), and RELNOUN 2.0, which performs special linguistic processing for extraction from complex noun phrases (Pal and Mausam, 2016).¹

*Most work was done when the author was a graduate student at IIT Delhi.

¹<https://github.com/knowitall/openie>

1. Sentence: <i>Hong Kong’s labour force is 3.5 million.</i> Open IE 4.2: (Hong Kong’s labour force; is; 3.5 million) BONIE: (Hong Kong; has labour force of; 3.5 million)
2. Sentence: <i>Microsoft has 100,000 employees.</i> Open IE 4.2: (Microsoft; has; 100,000 employees) BONIE: (Microsoft; has number of employees; 100,000)
3. Sentence: <i>James Valley is nearly 600 miles long.</i> Open IE 4.2: (James Valley; is; nearly 600 miles long) BONIE: (James Valley; has length of; nearly 600 miles)
4. Sentence: <i>Donald Trump is 70 years old.</i> Open IE 4.2: (Donald Trump; is; 70 years old) BONIE: (Donald Trump; has age of; 70 years)
5. Sentence: <i>James Valley has 5 sq kms of fruit orchards.</i> Open IE 4.2: (James Valley; has; 5 sq kms of fruit orchards) BONIE: (James Valley; has area of fruit orchards; 5 sq kms)

Table 1: Comparison of Open IE 4.2 and BONIE

In this work, we present and release² the first system for open numerical extraction, which we name BONIE for **B**ootstrapping-based **O**pen **N**umerical **I**nformation **E**xtractor. It is important to note that existing Open IE systems, like Open IE 4.2, may also extract numerical facts. However, they are oblivious to the presence of numbers in arguments. Therefore, they may miss important extractions and may not always output the best numerical facts. Table 1 compares extractions generated by Open IE 4.2 and BONIE on some of the sample sentences.

At a high level BONIE follows OLLIE’s design of identifying seed facts, constructing training data by bootstrapping sentences that may mention a seed fact, pattern learning and ranking. Madaan et al (2016) note that bootstrapping for numerical IE is challenging; it can lead to high noise and missed recall, since numbers can easily match out of context, and numbers may not match due to approximations. In response, similar to most previous works (e.g., LUCHS (Hoffmann et al., 2010)) BONIE matches a number if it is within a percentage threshold. Additionally, BONIE uses a quantity extractor (Roy et al., 2015), which provides

²Available at <https://github.com/Open-NRE>

the units mentioned in the sentence – BONIE bootstraps a sentence only when the units match.

When compared to OLLIE, BONIE contributes several numerical IE specific customizations. (1) Since no open facts are available for this task, we first manually define a set of high-precision seed patterns, which are run over a large corpus to generate seed facts. (2) Not all seeds are fit for bootstrapping – many don’t even have an entity as first argument. We develop heuristics to identify an informative subset from these. After bootstrapping and pattern learning, we find that we are missing important tuples. E.g., sentence #3 in Table 1 above has no explicit relation word – the relation ‘has length of’ is implicit via the adjective ‘long’. And, sentence #5 expresses the relation ‘area’ via the units. (3) BONIE identifies implicit relations using additional processing of units and adjectives. (4) Finally, BONIE can tag a quantity as *count* and prepends “number of” in the relation phrase (sentence #2).

2 Related Work

One of the first Open IE systems to obtain substantial recall is OLLIE (Mausam et al., 2012), which is a pattern learning approach based on a bootstrapped training data using high precision verb-based extractions. Other methods augment the linguistic knowledge in the systems – Exemplar (de Sá Mesquita et al., 2013) adds new rules over dependency parses, SRLIE develops extraction logic over SRL frames (Christensen et al., 2011). Several works identify clauses and operate over restructured sentences (Schmidek and Barbosa, 2014; Corro and Gemulla, 2013; Bast and Haussmann, 2013). Other approaches use tree kernels (Xu et al., 2013), qualia-based patterns (Xavier et al., 2015), and simple within-sentence inference (Bast and Haussmann, 2014). However, none of them handle numbers specifically, and hence do not work for our problem.

Numerical Relations: Numbers play an important role in extracting information from text. Early works have seen people working on understanding numbers that express temporal information (Ling and Weld, 2010). More recently, the focus has been on numbers that express physical quantities or measures, either mentioned in text (Chaganty and Liang, 2016) or in the context of web tables (Ibrahim et al., 2016; Neumaier et al., 2016), or on numbers that represent cardinalities of relations (Mirza et al., 2017).

One of the prior works that applies to generic numerical relations is LUCHS (Hoffmann et al., 2010), where the system uses distant supervision to create 5,000 relation extractors, which included numerical relations as well. Researchers have also specifically developed *numerical relation* extractors to extract those relations where one of the arguments is a quantity (Vlachos and Riedel, 2015; Intxaurreondo et al., 2015; Madaan et al., 2016). However, all of them extract only an ontology relation, and hence are not directly applicable to Open IE.

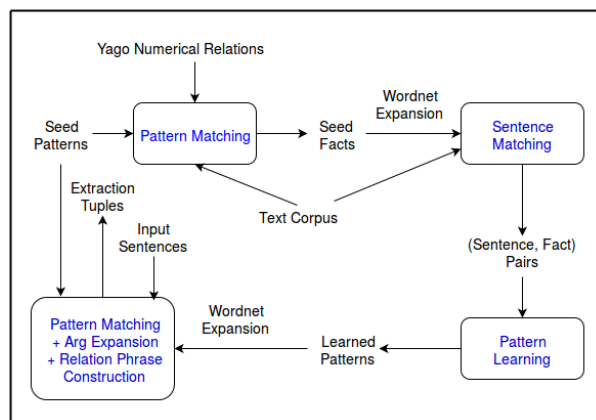


Figure 1: BONIE flow diagram

3 Open Numerical Relation Extraction

The goal of Open Numerical Relation Extraction is to process a sentence that has a quantity mention in it, and extract any tuple of the form (Arg1, relation phrase, Arg2) where Arg2 (or Arg1) is a quantity. As a first step, BONIE learns patterns where Arg2 is a quantity, as most English sentences tend to express numerical facts in active voice. Figure 1 outlines BONIE’s algorithm, which operates in two phases: training and extraction. BONIE’s training includes creation of seed facts, generation of training data via bootstrapping, and pattern learning over dependency parses. In the extraction phase, BONIE performs pattern matching and parse-based expansion to construct numerical tuples. These numerical tuples are made more coherent by a novel relation construction step.

As an example, the sentence “India has a population of 1.2 billion” matches seed pattern #2 (from Figure 2) to create a seed fact (India; population; 1.2 billion; null). This ‘null’ represents that the quantity needs no unit. While bootstrapping, this seed fact may match a sentence “India is the second most populous country in the world, with a population of 1.25 billion.” in the corpus.

Seed Dependency Patterns
1. <(#is are was were been be#verb)<(nsubj#{rel}#nnp nn)<(prep#of for#in)<(pobj#{arg}#nnp nn prp)>>(attr#{quantity}#.+)>>
2. <(#has have had having#verb)<(dobj#{rel}#nnp nn)<(prep##in)<(pobj#{quantity}#.+)>>(nsubj#{arg}#nnp nn prp)>>
3. <(#is are was were been be#verb)<(nsubj#{arg}#nnp nn prp)(acomp advmod#{rel}#jj rb)<(npadvmod#{quantity}#.+)>>
4. <(#has have had having#verb)<(nsubj#arg#nnp nn prp)(dobj#quantity#.+)<(prep#of#in)<(pobj#{rel}#nnp nn)>>>>
5. <##verb)<(attr acomp#{quantity}#.+)<(nsubj#{rel}#nnp nn)<(poss#{arg}#nnp nn)>>>>
6. <#{rel}#verb)<(auxpass#is are was were been be#verb)(nsubjpass#{arg}#nnp nn prp)(prep##in)<(pobj#{quantity}#.+)>>

Figure 2: Seed Dependency Patterns in BONIE

This training example will help learn a new pattern.³ This pattern, when applied to the sentence “Microsoft Windows is the most popular operating system, with a customer base of 300 million users”, will extract (Microsoft Windows; has customer base of; 300 million users).

While BONIE’s skeleton broadly resembles that of OLLIE’s (Mausam et al., 2012), it brings in customizations specific to the problem of numerical extraction such as a modified pattern language, heuristics for generating high quality seed set and training data, special processing for non-noun relations, and a novel relation construction step. We now describe BONIE’s algorithm in detail.

3.1 Generation of Seed Facts

Since open numerical facts are not readily available, we first write a handful of high-precision dependency patterns (see Figure 2 for a list). Each dependency pattern encodes the minimal sub-tree of the dependency parse connecting the relation, quantity and argument in that sentence. BONIE encodes a node in a pattern via ‘<depLabel>#<word>#<POSTag>’, where ‘depLabel’ is the edge connecting the node to its parent, ‘word’ is the word at the node, ‘POSTag’ is its part of speech tag; ‘#’ is a delimiter separating them. {rel}, {arg} and {quantity} in the patterns are placeholders for relation, argument and quantity headwords, respectively. BONIE generates seed facts by parsing the corpus and matching seed patterns with the parse. In case of a successful match, a seed fact of the form (arg headword; relation headword; quantity; unit) is generated. Argument and relation headwords are extracted directly from the parse. For the other two, it uses Illinois Quantifier (Roy et al., 2015), which returns both the quantity and unit separately.

Since seed facts form the basis of our training task, they must be as clean as possible – BONIE

adds several filters to reduce noise. It considers a seed fact as valid only when the quantity node in the pattern is within some quantity span given by Illinois Quantifier. It also rejects any fact whose argument is not a proper noun.

After these filters it gets high-precision extractions, but not necessarily good seeds – many seeds are generic, which may easily match unrelated sentences. E.g., (Michael; drove; 20; kms) isn’t a good seed, since ‘Michael’ isn’t specific, and could erroneously match sentences mentioning another Michael with some unrelated reference of a 20 km drive. To improve the set, BONIE checks for the presence of a seed fact in Yago KB (Hoffart et al., 2013) and keeps only those that are common. Since Yago has many numerical facts for height, area, latitude, GDP, etc., this gives BONIE a diverse set of clean facts for further training.

Finally, some numerical facts may be expressed without using a nominal relation word. BONIE uses WordNet (Miller, 1995) to generate new seeds from such seed facts using the derivationally related noun form of the relation headword. For example, (Brown ; tall ; 13 ; inches) gets transformed to (Brown; height; 13; inches), which gets added as a seed fact.

3.2 Bootstrapping

Similar to OLLIE, BONIE finds sentences that contain all words in a seed fact and generates (*sentence, fact*) pairs. But unlike OLLIE, BONIE has quantities and units, and matching them as words isn’t appropriate. Illinois Quantifier performs an internal normalization for both, e.g, changes ‘dollars’ and ‘\$’ to ‘US\$', and ‘%’ to ‘percent’. Since seed facts also have normalized units, we run Illinois quantifier on candidate sentences and match normalized units directly. Moreover, BONIE maintains a percentage threshold δ to control the amount of allowed difference between quantities in the sentence and seed fact. Once all constituents of a fact match with a sentence, BONIE generates the (*sentence, fact*) pair.

³ <(#is#verb)<(nsubj#{arg}#nnp|nn)(prep#with#in)<(pobj#{rel}#nnp|nn)<(prep#of#in)<(pobj#{quantity}#.+)>>>>>>

3.3 Open Pattern Learning

For each (*sentence, fact*) pair, BONIE parses the sentence, and replaces the argument and relation words of the fact with ‘{arg}’ and ‘{rel}’ placeholders. For quantity and unit words, BONIE replaces the one at a higher level in the parse with ‘{quantity}’. The minimal path containing ‘{arg}’, ‘{rel}’ and ‘{quantity}’ is learned as a pattern. Since quantity and unit are typically expected to remain close to each other in a sentence, BONIE rejects all such patterns where the distance between them exceeds a certain threshold value.

Some patterns are learned with specific words such as ‘contains’ in example (partial) $\langle(\#contains\#verb)\langle(dobj\#\{quantity\}\#.\+)\rangle\dots$. We believe that this pattern should work with all inflections and synonyms of ‘contain’, BONIE uses WordNet to expand the pattern by including all inflections and synset synonyms. Each pattern is scored based on the number of times it is learned from the data.

3.4 Constructing Extractions

After matching a pattern to a new sentence, similar to OLLIE, arg/rel phrases are completed by expanding the extracted headnouns on poss, det, num, neg, amod, quantmod, nn, and pobj edges. If one of the children of the argument headword is a prep, rcmmod or partmod edge, the whole subtree under that is extracted. Quantity phrase is extracted by Illinois quantifier, but if any sibling node of the quantity node is connected by a prep edge, with the word ‘of’, BONIE expands the entire subtree below it. This allows “10 percent of 100 dollars” to be included in the quantity phrase.

Relation Phrase Construction: Whenever the relation headword is an adjective or an adverb, BONIE uses WordNet to get its derivationally related noun form and that becomes the new relation. This transforms the tuple (Donald Trump ; old ; about 70 years) to (Donald Trump ; has age of ; about 70 years).

Sometimes, sentences don’t use a numerical relation word – it is obvious from the units. E.g., sentence #4 on page 1 expresses the ‘area’ relation implicitly. BONIE infers these implicit relations using the unit analysis in UnitTagger (Sarawagi and Chakrabarti, 2014). Whenever BONIE sees a unit (sq kms) getting mistreated as a relation it uses UnitTagger to infer relations from units and postprocesses the extraction accordingly. The ex-

traction, as a result changes from (James Valley; has sq kms of ; 5 of fruit orchards) to (James Valley ; has area of fruit orchards ; 5 sq kms).

Finally, in cases when a plural noun relation word also appears as a unit in the quantifier, BONIE hypothesizes that it is a count extraction, and prepends ‘number of’ to the relation headword and removes the unit from the quantity phrase. E.g., (Microsoft; has employees; 100,000 employees) from sentence #2 becomes (Microsoft; has number of employees; 100,000).

4 Experiments

We build BONIE over data from ClueWeb12,⁴ filtered so as to keep only the sentences that contain numbers. We further remove those where quantity represents a date, time, or duration, and where the quantity is accompanied by document words like ‘Section’, ‘Table’, or ‘Figure’. We use the dependency parser from ClearNLP⁵. We generate about 21,000 seed facts from roughly 20 million numerical sentences. These are matched against 7 million numerical sentences obtaining about 18,500 (sentence, fact) pairs.

We tried different values of δ (the matching threshold) and found results to not be sensitive as long as δ varies in the range of 2% to 5%. So we set $\delta=2\%$ during the final evaluation. The distance threshold between the quantity and unit mentioned in Section 2.3 is set to 3 and is based on our general understanding of parse trees.

BONIE learns around 7,000 new patterns. Since pattern frequency is a good indicator of pattern quality (Wu and Weld, 2010), we rank the patterns on the basis of frequency and take the top 1,000 patterns for further analysis. We find that almost all patterns beyond the top 1,000 are learned only once or twice on our training set. Our decision to ignore all patterns beyond the top 1,000 is so that we have a support of at least three for each pattern.

We sample a random testset of 2,000 numerical sentences from ClueWeb12 (not used in training). Two annotators with NLP experience annotate each extraction for correctness. We obtain an inter-annotator agreement of 97%, and report the results on the subset where both annotators agree.

Since there are no open numerical extractors available, we compare BONIE against an Open IE system and another closed numerical IE system.

⁴<http://www.lemurproject.org/clueweb12.php/>

⁵<https://github.com/clir/clearnlp>

Setting	Precision	Yield
NumberRule	50.00	6
Open IE 4.2	62.50	296
BONIE (seed patterns only)	85.71	72
+ learned patterns	13.88	362
+ fact filters	55.27	351
+ Yago + WordNet expansion on facts	72.69	418
+ Relation phrase construction	77.91	448
+ WordNet expansion on patterns	77.23	458

Table 2: Precision and Yield (#correct numerical extractions) on a dataset of 2000 ClueWeb12 numerical sentences.

Table 2 reports the precisions and yields of all systems. We first compare against numerical tuples from Open IE 4.2,⁶ a publicly available state-of-the-art Open IE system that combines SRLIE and RELNOUN. BONIE outputs a much higher precision and yield on numerical facts, as compared to Open IE 4.2. We also compare against NumberRule, a state-of-the-art closed numerical IE system (Madaan et al., 2016). NumberRule⁷ can be quickly re-targeted to any new semantic relation by inputting keywords. We feed all of Yago numerical relation words as keywords to NumberRule, but still find that it was able to generate only 12 extractions on our testset.

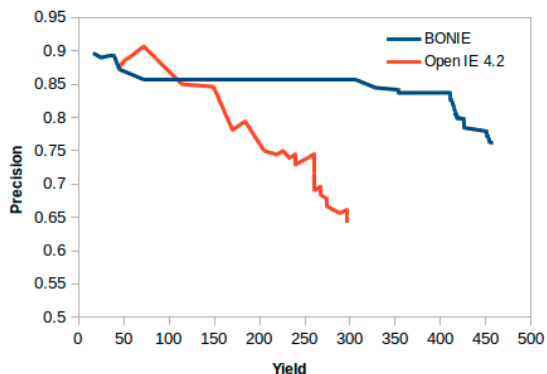


Figure 3: Comparison of Precision-Yield curves for BONIE and Open IE 4.2. BONIE achieves substantially larger area under the curve than Open IE 4.2.

We also perform additional ablation study to evaluate the value of each component. Just the seed patterns themselves have a significantly higher precision but much smaller yield. This is expected, since the seeds must be highly precise for bootstrapping. If Yago matching and other seed filtering heuristics are turned off, the precision of the system goes down drastically due to a very noisy bootstrapped set. If the post-processing of relation phrase construction is turned off, there is a 5 point precision loss and about 7% yield reduction due to some incorrect extracted tuples, which are corrected by post-processing. Finally,

⁶<https://github.com/knowitall/openie>

⁷<https://github.com/NEO-IE>

Wordnet-based expansion has marginal increase in yield and slight precision loss.

Open IE 4.2 associates a confidence value with each extraction - ranking against which generates a precision-yield curve. For BONIE, we rank the patterns in such a way that the seed patterns are at the top, followed by the learned patterns. The learned patterns are ordered based on their frequencies. Figure 3 reports the curves for both the systems and we find that BONIE has a larger area under the curve as compared to Open IE 4.2.

Estimating recall in Open IE is difficult since it requires annotators to exhaustively tag all open extractions in a sentence. To get an estimate, an author manually tagged 100 sentences with all numerical extractions. We find that BONIE’s recall is about 48%. Two-thirds of missed recall is because of missing conjuncts. E.g., it misses the tuple relating retirement age with 68 years in “*The retirement age for men is 65 years and 68 years for women.*” Other missed recall is due to complexity of sentences or inaccuracy of parsers.

5 Conclusions

We release BONIE⁸, the first open numerical relation extractor and other resources for further research. BONIE is based on bootstrapping and pattern learning and follows previous similar works such as OLLIE. However, for effective bootstrapping and training, it implements various customizations specific to numerical relations in curation of seed fact set, matching of sentences, and construction of relation phrase at the time of extraction. BONIE significantly outperforms both open non-numerical IE, and closed numerical IE systems with 1.5x yield and 15 point precision gain over a state-of-the-art Open IE system. We find that better conjunction processing is an important future step for improving BONIE’s recall even further.

Acknowledgements

This work is supported by Google language understanding and knowledge discovery focused research grants, a Bloomberg award, a Visvesvaraya faculty award by Govt. of India and an IRD seed grant at I.I.T, Delhi. We also thank Microsoft for the Microsoft Azure sponsorship and a Microsoft Travel grant in support of the work.

⁸Available at <https://github.com/Open-NRE>

References

- Hannah Bast and Elmar Haussmann. 2013. Open information extraction via contextual sentence decomposition. In *2013 IEEE Seventh International Conference on Semantic Computing, Irvine, CA, USA, September 16-18, 2013*. pages 154–159.
- Hannah Bast and Elmar Haussmann. 2014. More informative open information extraction via simple inference. In *Advances in Information Retrieval - 36th European Conference on IR Research, ECIR 2014, Amsterdam, The Netherlands, April 13-16, 2014. Proceedings*. pages 585–590.
- Arun Tejasvi Chaganty and Percy Liang. 2016. How much is 131 million dollars? putting numbers in perspective with compositional descriptions. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*.
- Janara Christensen, Mausam, Stephen Soderland, and Oren Etzioni. 2011. An analysis of open information extraction based on semantic role labeling. In *Proceedings of the 6th International Conference on Knowledge Capture (K-CAP 2011), June 26-29, 2011, Banff, Alberta, Canada*. pages 113–120.
- Luciano Del Corro and Rainer Gemulla. 2013. Clause: clause-based open information extraction. In *22nd International World Wide Web Conference, WWW '13, Rio de Janeiro, Brazil, May 13-17, 2013*. pages 355–366.
- Filipe de Sá Mesquita, Jordan Schmeidek, and Denilson Barbosa. 2013. Effectiveness and efficiency of open relation extraction. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, EMNLP 2013, 18-21 October 2013, Grand Hyatt Seattle, Seattle, Washington, USA, A meeting of SIGDAT, a Special Interest Group of the ACL*. pages 447–457.
- Oren Etzioni, Michele Banko, Stephen Soderland, and Daniel S. Weld. 2008. Open information extraction from the web. *Commun. ACM* 51(12):68–74.
- Oren Etzioni, Anthony Fader, Janara Christensen, Stephen Soderland, and Mausam. 2011. Open information extraction: The second generation. In *IJCAI 2011, Proceedings of the 22nd International Joint Conference on Artificial Intelligence, Barcelona, Catalonia, Spain, July 16-22, 2011*. pages 3–10.
- Johannes Hoffart, Fabian M. Suchanek, Klaus Berberich, and Gerhard Weikum. 2013. YAGO2: A spatially and temporally enhanced knowledge base from wikipedia. *Artif. Intell.* 194:28–61.
- Raphael Hoffmann, Congle Zhang, and Daniel S. Weld. 2010. Learning 5000 relational extractors. In *ACL 2010, Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, July 11-16, 2010, Uppsala, Sweden*. pages 286–295.
- Yusra Ibrahim, Mirek Riedewald, and Gerhard Weikum. 2016. Making sense of entities and quantities in web tables. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management, CIKM 2016, Indianapolis, IN, USA, October 24-28, 2016*. pages 1703–1712.
- Ander Intxaurreondo, Eneko Agirre, Oier Lopez de Lacalle, and Mihai Surdeanu. 2015. Diamonds in the rough: Event extraction from imperfect microblog data. In *NAACL HLT 2015, The 2015 Conference of the North American Chapter of the Association for Computational Linguistics, Denver, Colorado, USA, May 31 - June 5, 2015*. pages 641–650.
- Xiao Ling and Daniel S. Weld. 2010. Temporal information extraction. In *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2010, Atlanta, Georgia, USA, July 11-15, 2010*.
- Aman Madaan, Ashish Mittal, Mausam, Ganesh Ramakrishnan, and Sunita Sarawagi. 2016. Numerical relation extraction with minimal supervision. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, February 12-17, 2016, Phoenix, Arizona, USA*. pages 2764–2771.
- Mausam. 2016. Open information extraction systems and downstream applications. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI 2016, New York, NY, USA, 9-15 July 2016*. pages 4074–4077.
- Mausam, Michael Schmitz, Stephen Soderland, Robert Bart, 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, EMNLP-CoNLL 2012, July 12-14, 2012, Jeju Island, Korea*. pages 523–534.
- George A. Miller. 1995. Wordnet: A lexical database for english. *Commun. ACM* 38(11):39–41.
- Paramita Mirza, Simon Razniewski, Fariz Darari, and Gerhard Weikum. 2017. Cardinal virtues: Extracting relation cardinalities from text.
- Sebastian Neumaier, Jürgen Umbrich, Josiane Xavier Parreira, and Axel Polleres. 2016. Multi-level semantic labelling of numerical values. In *The Semantic Web - ISWC 2016 - 15th International Semantic Web Conference, Kobe, Japan, October 17-21, 2016, Proceedings, Part I*. pages 428–445.
- Harinder Pal and Mausam. 2016. Donyms and compound relational nouns in nominal open IE. In *Proceedings of the 5th Workshop on Automated Knowledge Base Construction, AKBC@NAACL-HLT 2016, San Diego, CA, USA, June 17, 2016*. pages 35–39.

- Subhro Roy, Tim Vieira, and Dan Roth. 2015. Reasoning about quantities in natural language. *TACL* 3:1–13.
- Sunita Sarawagi and Soumen Chakrabarti. 2014. Open-domain quantity queries on web tables: annotation, response, and consensus models. In *The 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '14, New York, NY, USA - August 24 - 27, 2014*. pages 711–720.
- Jordan Schmidek and Denilson Barbosa. 2014. Improving open relation extraction via sentence restructuring. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC-2014), Reykjavik, Iceland, May 26-31, 2014*. pages 3720–3723.
- Andreas Vlachos and Sebastian Riedel. 2015. Identification and verification of simple claims about statistical properties. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*. pages 2596–2601.
- Fei Wu and Daniel S. Weld. 2010. Open information extraction using wikipedia. In *ACL 2010, Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, July 11-16, 2010, Uppsala, Sweden*. pages 118–127.
- Clarissa Castellã Xavier, Vera Lúcia Strube de Lima, and Marlo Souza. 2015. Open information extraction based on lexical semantics. *J. Braz. Comp. Soc.* 21(1):4:1–4:14.
- Ying Xu, Mi-Young Kim, Kevin Quinn, Randy Goebel, and Denilson Barbosa. 2013. Open information extraction with tree kernels. In *Human Language Technologies: Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings, June 9-14, 2013, Westin Peachtree Plaza Hotel, Atlanta, Georgia, USA*. pages 868–877.