

Acquiring Predicate Paraphrases from News Tweets

Vered Shwartz

Gabriel Stanovsky

Ido Dagan

Computer Science Department, Bar-Ilan University, Ramat-Gan, Israel

{vered1986, gabriel.satanovsky}@gmail.com, dagan@cs.biu.ac.il

Abstract

We present a simple method for ever-growing extraction of predicate paraphrases from news headlines in Twitter. Analysis of the output of ten weeks of collection shows that the accuracy of paraphrases with different support levels is estimated between 60-86%. We also demonstrate that our resource is to a large extent complementary to existing resources, providing many novel paraphrases. Our resource is publicly available, continuously expanding based on daily news.

1 Introduction

Recognizing that various textual descriptions across multiple texts refer to the same event or action can benefit NLP applications such as recognizing textual entailment (Dagan et al., 2013) and question answering. For example, to answer “when did the US Supreme Court approve same-sex marriage?” given the text “In June 2015, the Supreme Court ruled for same-sex marriage”, *approve* and *ruled for* should be identified as describing the same action.

To that end, much effort has been devoted to identifying predicate paraphrases, some of which resulted in releasing resources of predicate entailment or paraphrases. Two main approaches were proposed for that matter; the first leverages the similarity in argument distribution across a large corpus between two predicates (e.g. [a]₀ buy [a]₁ / [a]₀ acquire [a]₁) (Lin and Pantel, 2001; Berant et al., 2010). The second approach exploits bilingual parallel corpora, extracting as paraphrases pairs of texts that were translated identically to foreign languages (Ganitkevitch et al., 2013).

While these methods have produced exhaustive resources which are broadly used by applications,

[a] ₀ introduce [a] ₁	[a] ₀ welcome [a] ₁
[a] ₀ appoint [a] ₁	[a] ₀ to become [a] ₁
[a] ₀ die at [a] ₁	[a] ₀ pass away at [a] ₁
[a] ₀ hit [a] ₁	[a] ₀ sink to [a] ₁
[a] ₀ be investigate [a] ₁	[a] ₀ be probe [a] ₁
[a] ₀ eliminate [a] ₁	[a] ₀ slash [a] ₁
[a] ₀ announce [a] ₁	[a] ₀ unveil [a] ₁
[a] ₀ quit after [a] ₁	[a] ₀ resign after [a] ₁
[a] ₀ announce as [a] ₁	[a] ₀ to become [a] ₁
[a] ₀ threaten [a] ₁	[a] ₀ warn [a] ₁
[a] ₀ die at [a] ₁	[a] ₀ live until [a] ₁
[a] ₀ double down on [a] ₁	[a] ₀ stand by [a] ₁
[a] ₀ kill [a] ₁	[a] ₀ shoot [a] ₁
[a] ₀ approve [a] ₁	[a] ₀ pass [a] ₁
[a] ₀ would be cut under [a] ₁	[a] ₁ slash [a] ₀
seize [a] ₀ at [a] ₁	to grab [a] ₀ at [a] ₁

Table 1: A sample from the top-ranked predicate paraphrases.

their accuracy is limited. Specifically, the first approach may extract antonyms, that also have similar argument distribution (e.g. [a]₀ raise to [a]₁ / [a]₀ fall to [a]₁) while the second may conflate multiple senses of the foreign phrase.

A third approach was proposed to harvest paraphrases from multiple mentions of the same event in news articles.¹ This approach assumes that various redundant reports make different lexical choices to describe the same event. Although there has been some work following this approach (e.g. Shinyama et al., 2002; Shinyama and Sekine, 2006; Roth and Frank, 2012; Zhang and Weld, 2013), it was less exhaustively investigated and did not result in creating paraphrase resources.

In this paper we present a novel unsupervised method for ever-growing extraction of lexically-divergent predicate paraphrase pairs from news tweets. We apply our methodology to create a resource of predicate paraphrases, exemplified in Table 1.

Analysis of the resource obtained after ten

¹This corresponds to instances of event coreference (Bagga and Baldwin, 1999).

weeks of acquisition shows that the set of paraphrases reaches accuracy of 60-86% at different levels of support. Comparison to existing resources shows that, even as our resource is still smaller in orders of magnitude from existing resources, it complements them with non-consecutive predicates (e.g. take [a]₀ from [a]₁) and paraphrases which are highly context specific.

The resource and the source code are available at <http://github.com/vered1986/Chirps>.² As of the end of May 2017, it contains 456,221 predicate pairs in 1,239,463 different contexts. Our resource is ever-growing and is expected to contain around 2 million predicate paraphrases within a year. Until it reaches a large enough size, we will release a daily update, and at a later stage, we plan to release a periodic update.

2 Background

2.1 Existing Paraphrase Resources

A prominent approach to acquire predicate paraphrases is to compare the distribution of their arguments across a corpus, as an extension to the distributional hypothesis (Harris, 1954). DIRT (Lin and Pantel, 2001) is a resource of 10 million paraphrases, in which the similarity between predicate pairs is estimated by the geometric mean of the similarities of their argument slots. Berant (2012) constructed an entailment graph of distributionally similar predicates by enforcing transitivity constraints and applying global optimization, releasing 52 million directional entailment rules (e.g. [a]₀ shoot [a]₁ → [a]₀ kill [a]₁).

A second notable source for extracting paraphrases is multiple translations of the same text (Barzilay and McKeown, 2001). The Paraphrase Database (PPDB) (Ganitkevitch et al., 2013; Pavlick et al., 2015) is a huge collection of paraphrases extracted from bilingual parallel corpora. Paraphrases are scored heuristically, and the database is available for download in six increasingly large sizes according to scores (the smallest size being the most accurate). In addition to lexical paraphrases, PPDB also consists of 140 million syntactic paraphrases, some of which include predicates with non-terminals as arguments.

2.2 Using Multiple Event Descriptions

Another line of work extracts paraphrases from redundant comparable news articles (e.g. Shinyama

et al., 2002; Barzilay and Lee, 2003). The assumption is that multiple news articles describing the same event use various lexical choices, providing a good source for paraphrases. Heuristics are applied to recognize that two news articles discuss the same event, such as lexical overlap and same publish date (Shinyama and Sekine, 2006). Given such a pair of articles, it is likely that predicates connecting the same arguments will be paraphrases, as in the following example:

- | |
|--|
| 1. GOP lawmakers <i>introduce</i> new health care plan |
| 2. GOP lawmakers <i>unveil</i> new health care plan |

Zhang and Weld (2013) and Zhang et al. (2015) introduced methods that leverage parallel news streams to cluster predicates by meaning, using temporal constraints. Since this approach acquires paraphrases from descriptions of *the same event*, it is potentially more accurate than methods that acquire paraphrases from the entire corpus or translation phrase table. However, there is currently no paraphrase resource acquired in this approach.³

Finally, Xu et al. (2014) developed a supervised model to collect *sentential* paraphrases from Twitter. They used Twitter’s trending topic service, and considered two tweets from the same topic as paraphrases if they shared a single anchor word.

3 Resource Construction

We present a methodology to automatically collect binary verbal predicate paraphrases from Twitter. We first obtain news related tweets (§3.1) from which we extract propositions (§3.2). For a candidate pair of propositions, we assume that if both arguments can be matched then the predicates are likely paraphrases (§3.3). Finally, we rank the predicate pairs according to the number of instances in which they were aligned (§3.4).

3.1 Obtaining News Headlines

We use Twitter as a source of readily available news headlines. The 140 characters limit makes tweets concise, informative and independent of each other, obviating the need to resolve document-level entity coreference. We query the Twitter Search API⁴ via Twitter Search.⁵ We use

³Zhang and Weld (2013) released a small collection of 10k predicate paraphrase clusters (with average cluster size of 2.4) produced by the system.

⁴<https://apps.twitter.com/>

⁵<https://github.com/ckoepp/TwitterSearch>

²Chirp is a paraphrase of *tweet*.

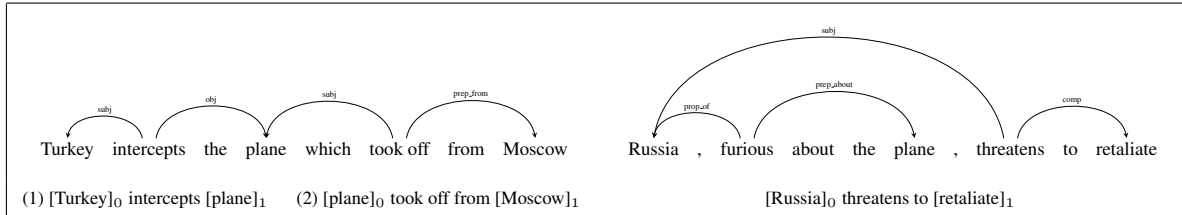


Figure 1: PropS structures and the corresponding propositions extracted by our process. Left: multi-word predicates and multiple extractions per tweet. Right: argument reduction.

Manafort hid payments from Ukraine party with Moscow ties	[a] ₀ hide [a] ₁	Paul Manafort	payments
Manafort laundered the payments through Belize	[a] ₀ launder [a] ₁	Manafort	payments
Send immigration judges to cities to speed up deportations	to send [a] ₀ to [a] ₁	immigration judges	cities
Immigration judges headed to 12 cities to speed up deportations	[a] ₀ headed to [a] ₁	immigration judges	12 cities

Table 2: Examples of predicate paraphrase instances in our resource: each instance contains two tweets, predicate types extracted from them, and the instantiations of arguments.

Twitter’s news filter that retrieves tweets containing links to news websites, and limit the search to English tweets.

3.2 Proposition Extraction

We extract propositions from news tweets using PropS (Stanovsky et al., 2016), which simplifies dependency trees by conveniently marking a wide range of predicates (e.g, verbal, adjectival, non-lexical) and positioning them as direct heads of their corresponding arguments. Specifically, we run PropS over dependency trees predicted by spaCy⁶ and extract predicate types (as in Table 1) composed of *verbal predicates*, datives, prepositions, and auxiliaries.

Finally, we employ a pre-trained argument reduction model to remove non-restrictive argument modifications (Stanovsky and Dagan, 2016). This is essential for our subsequent alignment step, as it is likely that short and concise phrases will tend to match more frequently in comparison to longer, more specific arguments. Figure 1 exemplifies some of the phenomena handled by this process, along with the automatically predicted output.

3.3 Generating Paraphrase Instances

Following the assumption that different descriptions of the same event are bound to be redundant (as discussed in Section 2.2), we consider two predicates as paraphrases if: (1) They appear on the same day, and (2) Each of their arguments aligns with a unique argument in the other predicate, either by *strict matching* (short edit distance, abbreviations, etc.) or by a looser matching (par-

tial token matching or WordNet synonyms).⁷ Table 2 shows examples of predicate paraphrase instances in the resource.

3.4 Resource Release

The resource release consists of two files:

1. **Instances:** the specific contexts in which the predicates are paraphrases (as in Table 2). In practice, to comply with Twitter policy, we release predicate paraphrase pair types along with their arguments and tweet IDs, and provide a script for downloading the full texts.
2. **Types:** predicate paraphrase pair types (as in Table 1). The types are ranked in a descending order according to a heuristic accuracy score:

$$s = count \cdot \left(1 + \frac{d}{N}\right)$$

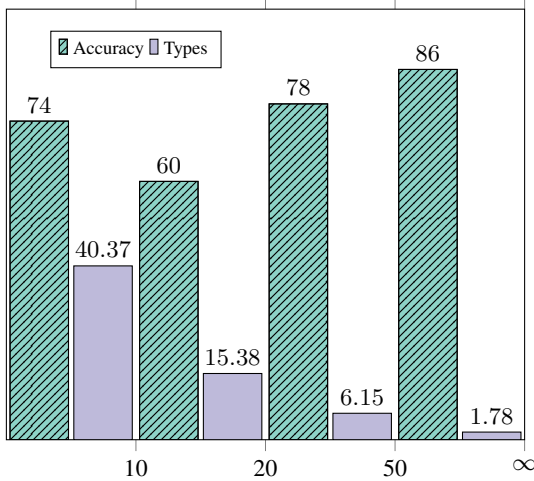
where *count* is the number of instances in which the predicate types were aligned (Section 3.3), *d* is the number of different days in which they were aligned, and *N* is the number of days since the resource collection began.

Taking into account the number of different days in which predicates were aligned reduces the noise caused by two entities that undergo two different actions on the same day. For example, the following tweets from the day of Chuck Berry’s death:

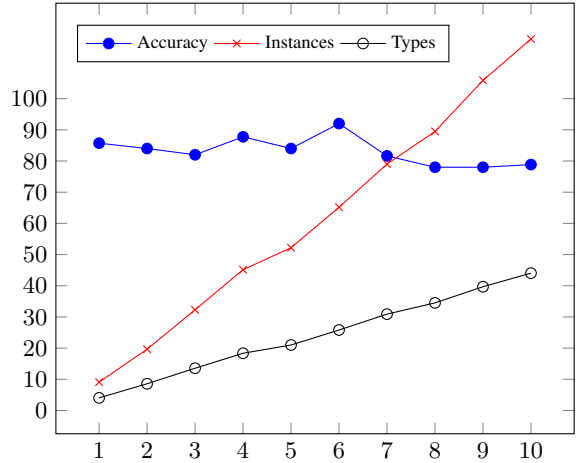
1. Last year when Chuck Berry turned 90
2. Chuck Berry dies at 90

⁶<https://spacy.io>

⁷In practice, our publicly available code requires that *at least one* pair of arguments will strictly match.



(a) Estimated accuracy (%) and number of types ($\times 1K$) of predicate pairs with at least 5 instances in different score bins.



(b) Estimated accuracy (%), number of instances ($\times 10K$) and types ($\times 10K$) in the first 10 weeks.

Figure 2: Resource statistics after ten weeks of collection.

yield the incorrect type $[a]_0$ turn $[a]_1 / [a]_0$ die at $[a]_1$. While there may be several occurrences of that type on the same day, it is not expected to re-occur in other news events (in different days), yielding a low accuracy score.

4 Analysis of Resource Quality

We estimate the quality of the resource obtained after ten weeks of collection by annotating a sample of the extracted paraphrases.

The annotation task was carried out in Amazon Mechanical Turk.⁸ To ensure the quality of workers, we applied a qualification test and required a 99% approval rate for at least 1,000 prior tasks. We assigned each annotation to 3 workers and used the majority vote to determine the correctness of paraphrases.

We followed a similar approach to instance-based evaluation (Szpektor et al., 2007), and let workers judge the correctness of a predicate pair (e.g. $[a]_0$ purchase $[a]_1/[a]_0$ acquire $[a]_1$) through 5 different instances (e.g. Intel purchased Mobileye/Intel acquired Mobileye). We considered the type as correct if at least one of its instance-pairs were judged as correct. The idea that lies behind this type of evaluation is that predicate pairs are difficult to judge out-of-context.

Differently from Szpektor et al. (2007), we used the instances in which the paraphrases appeared originally, as those are available in the resource.

⁸<https://www.mturk.com/mturk>

4.1 Quality of Extractions and Ranking

To evaluate the resource accuracy, and following the instance-based evaluation scheme, we only considered paraphrases that occurred in at least 5 instances (which currently constitute 10% of the paraphrase types). We partition the types into four increasingly large bins according to their scores (the smallest bin being the most accurate), similarly to PPDB (Ganitkevitch et al., 2013), and annotate a sample of 50 types from each bin. Figure 2(a) shows that the frequent types achieve up to 86% accuracy.

The accuracy expectedly increases with the score, except for the lowest-score bin $((0, 10])$ which is more accurate than the next one $((10, 20])$. At the current stage of the resource there is a long tail of paraphrases that appeared few times. While many of them are incorrect, there are also true paraphrases that are infrequent and therefore have a low accuracy score. We expect that some of these paraphrases will occur again in the future and their accuracy score will be strengthened.

4.2 Size and Accuracy Over Time

To estimate future usefulness, Figure 2(b) plots the resource size (in terms of types and instances) and estimated accuracy through each week in the first 10 weeks of collection.

The accuracy at a specific time was estimated by annotating a sample of 50 predicate pair types with accuracy score ≥ 20 in the resource obtained

drag [a] ₀ from [a] ₁	[a] ₀ remove from [a] ₁
leak [a] ₀ to [a] ₁	to share [a] ₀ with [a] ₁
oust [a] ₀ from [a] ₁	[a] ₀ be force out at [a] ₁
reveal [a] ₀ to [a] ₁	share [a] ₀ with [a] ₁
[a] ₀ add [a] ₁	[a] ₀ beef up [a] ₁
[a] ₀ admit to [a] ₁	[a] ₀ will attend [a] ₁
[a] ₀ announce as [a] ₁	[a] ₀ to become [a] ₁
[a] ₀ arrest in [a] ₁	[a] ₀ charge in [a] ₁
[a] ₀ attack [a] ₁	[a] ₀ clash with [a] ₁
[a] ₀ be force out at [a] ₁	[a] ₀ have be fire from [a] ₁
[a] ₀ eliminate [a] ₁	[a] ₀ slash [a] ₁
[a] ₀ face [a] ₁	[a] ₀ hit with [a] ₁
[a] ₀ mock [a] ₁	[a] ₀ troll [a] ₁
[a] ₀ open up about [a] ₁	[a] ₀ reveal [a] ₁
[a] ₀ get [a] ₁	[a] ₀ sentence to [a] ₁

Table 3: A sample of types from our resource that are not found in *Berant* or in *PPDB*.

at that time, which roughly correspond to the top ranked 1.5% types.

Figure 2(b) demonstrates that these types maintain a level of around 80% in accuracy. The resource growth rate (i.e. the number of new types) is expected to change with time. We predict that the resource will contain around 2 million types in one year.⁹

5 Comparison to Existing Resources

The resources which are most similar to ours are *Berant* (Berant, 2012), a resource of predicate entailments, and *PPDB* (Pavlick et al., 2015), a resource of paraphrases, both described in Section 2.

We expect our resource to be more accurate than resources which are based on the distributional approach (Berant, 2012; Lin and Pantel, 2001). In addition, in comparison to *PPDB*, we specialize on binary verbal predicates, and apply an additional phase of proposition extraction, handling various phenomena such as non-consecutive particles and minimality of arguments.

Berant (2012) evaluated their resource against a dataset of predicate entailments (Zeichner et al., 2012), using a recall-precision curve to show the performance obtained with a range of thresholds on the resource score. This kind of evaluation is less suitable for our resource; first, predicate entailment is directional, causing paraphrases with the wrong entailment direction to be labeled negative in the dataset. Second, since our resource is still relatively small, it is unlikely to have sufficient coverage of the dataset at that point. We therefore

⁹For up-to-date resource statistics, see: <https://github.com/vered1986/Chirps/tree/master/resource>.

leave this evaluation to future work.

To demonstrate the added value of our resource, we show that even in its current size, it already contains accurate predicate pairs which are absent from the existing resources. Rather than comparing against labeled data, we use types with score ≥ 50 from our resource (1,778 pairs), which were assessed as accurate (Section 4.2).

We checked whether these predicate pairs are covered by *Berant* and *PPDB*. To eliminate directionality, we looked for types in both directions, i.e. for a predicate pair $(p1, p2)$ we searched for both $(p1, p2)$ and $(p2, p1)$. Overall, we found that 67% of these types do not exist in *Berant*, 62% in *PPDB*, and 49% in neither.

Table 3 exemplifies some of the predicate pairs that do not exist in both resources. Specifically, our resource contains many non-consecutive predicates (e.g. reveal [a]₀ to [a]₁ / share [a]₀ with [a]₁) that by definition do not exist in *Berant*.

Some pairs, such as [a]₀ get [a]₁ / [a]₀ sentence to [a]₁, are context-specific, occurring when [a]₀ is a person and [a]₁ is the time they are about to serve in prison. Given that *get* has a broad distribution of argument instantiations, this paraphrase and similar paraphrases are less likely to exist in resources that rely on the distribution of arguments in the entire corpus.

6 Conclusion

We presented a new unsupervised method to acquire fairly accurate predicate paraphrases from news tweets discussing the same event. We release a growing resource of predicate paraphrases. Qualitative analysis shows that our resource adds value over existing resources. In the future, when the resource is comparable in size to the existing resources, we plan to evaluate its intrinsic accuracy on annotated test sets, as well as its extrinsic benefits in downstream NLP applications.

Acknowledgments

This work was partially supported by an Intel ICRI-CI grant, the Israel Science Foundation grant 880/12, and the German Research Foundation through the German-Israeli Project Cooperation (DIP, grant DA 1600/1-1).

References

- Amit Bagga and Breck Baldwin. 1999. Cross-document event coreference: Annotations, experiments, and observations. In *Workshop on Coreference and its Applications*.
- Regina Barzilay and Lillian Lee. 2003. Learning to paraphrase: An unsupervised approach using multiple-sequence alignment. In *Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*. <http://aclweb.org/anthology/N03-1003>.
- Regina Barzilay and R. Kathleen McKeown. 2001. Extracting paraphrases from a parallel corpus. In *Proceedings of the 39th Annual Meeting of the Association for Computational Linguistics*. <http://aclweb.org/anthology/P01-1008>.
- Jonathan Berant. 2012. *Global Learning of Textual Entailment Graphs*. Ph.D. thesis, Tel Aviv University.
- Jonathan Berant, Ido Dagan, and Jacob Goldberger. 2010. Global learning of focused entailment graphs. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, pages 1220–1229. <http://aclweb.org/anthology/P10-1124>.
- Ido Dagan, Dan Roth, and Mark Sammons. 2013. Recognizing textual entailment .
- Juri Ganitkevitch, Benjamin Van Durme, and Chris Callison-Burch. 2013. PPDB: The paraphrase database. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, pages 758–764. <http://aclweb.org/anthology/N13-1092>.
- Zellig S Harris. 1954. Distributional structure. *Word* 10(2-3):146–162.
- Dekang Lin and Patrick Pantel. 2001. Dirt – Discovery of inference rules from text. In *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, pages 323–328.
- Ellie Pavlick, Pushpendre Rastogi, Juri Ganitkevitch, Benjamin Van Durme, and Chris Callison-Burch. 2015. PPDB 2.0: Better paraphrase ranking, fine-grained entailment relations, word embeddings, and style classification. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*. Association for Computational Linguistics, pages 425–430. <https://doi.org/10.3115/v1/P15-2070>.
- Michael Roth and Anette Frank. 2012. Aligning predicate argument structures in monolingual comparable texts: A new corpus for a new task. In **SEM 2012: The First Joint Conference on Lexical and Computational Semantics – Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation (SemEval 2012)*. Association for Computational Linguistics, pages 218–227. <http://aclweb.org/anthology/S12-1030>.
- Yusuke Shinyama and Satoshi Sekine. 2006. Preemptive information extraction using unrestricted relation discovery. In *Proceedings of the Human Language Technology Conference of the NAACL, Main Conference*. <http://aclweb.org/anthology/N06-1039>.
- Yusuke Shinyama, Satoshi Sekine, and Kiyoshi Sudo. 2002. Automatic paraphrase acquisition from news articles. In *Proceedings of the second international conference on Human Language Technology Research*. Morgan Kaufmann Publishers Inc., pages 313–318.
- Gabriel Stanovsky and Ido Dagan. 2016. Annotating and predicting non-restrictive noun phrase modifications. In *Proceedings of the 54rd Annual Meeting of the Association for Computational Linguistics (ACL 2016)*.
- Gabriel Stanovsky, Jessica Fidler, Ido Dagan, and Yoav Goldberg. 2016. Getting more out of syntax with props. *CoRR* abs/1603.01648. <http://arxiv.org/abs/1603.01648>.
- Idan Szpektor, Eyal Shnarch, and Ido Dagan. 2007. Instance-based evaluation of entailment rule acquisition. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*. Association for Computational Linguistics, pages 456–463. <http://aclweb.org/anthology/P07-1058>.
- Wei Xu, Alan Ritter, Chris Callison-Burch, William B Dolan, and Yangfeng Ji. 2014. Extracting lexically divergent paraphrases from twitter. *Transactions of the Association for Computational Linguistics* 2:435–448.
- Naomi Zeichner, Jonathan Berant, and Ido Dagan. 2012. Crowdsourcing inference-rule evaluation. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. Association for Computational Linguistics, pages 156–160. <http://aclweb.org/anthology/P12-2031>.
- Congle Zhang, Stephen Soderland, and Daniel S Weld. 2015. Exploiting parallel news streams for unsupervised event extraction. *Transactions of the Association for Computational Linguistics* 3:117–129.
- Congle Zhang and Daniel S Weld. 2013. Harvesting parallel news streams to generate paraphrases of event relations. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*. Seattle, Washington, USA, pages 1776–1786.