# KOI at SemEval-2018 Task 5: Building Knowledge Graph of Incidents

**Paramita Mirza,**[1] **Fariz Darari,**[2*] **Rahmad Mahendra**[2*]

[1] Max Planck Institute for Informatics, Germany
[2] Faculty of Computer Science, Universitas Indonesia, Indonesia
{paramita}@mpi-inf.mpg.de
{fariz,rahmad.mahendra}@cs.ui.ac.id

## Abstract

We present KOI (Knowledge of Incidents), a system that given news articles as input, builds a knowledge graph (KOI-KG) of incidental events. KOI-KG can then be used to efficiently answer questions such as *"How many killing incidents happened in 2017 that involve Sean?"* The required steps in building the KG include: *(i)* document preprocessing involving word sense disambiguation, named-entity recognition, temporal expression recognition and normalization, and semantic role labeling; *(ii)* incidental event extraction and coreference resolution via document clustering; and *(iii)* KG construction and population.

## 1 Introduction

SemEval-2018[1] Task 5: *Counting Events and Participants in the Long Tail* (Postma et al., 2018) addresses the problem of *referential quantification* that requires a system to answer numerical questions about events such as *(i) "How many killing incidents happened in June 2016 in San Antonio, Texas?"* or *(ii) "How many people were killed in June 2016 in San Antonio, Texas?"*

**Subtasks S1 and S2** For questions of type *(i)*, which are asked by the first two subtasks, participating systems must be able to identify the type (e.g., *killing*, *injuring*), time, location and participants of each event occurring in a given news article, and establish within- and cross-document event coreference links. Subtask S1 focuses on evaluating systems' performances on identifying *answer incidents*, i.e., events whose properties fit the constraints of the questions, by making sure that there is only *one* answer incident per question.

**Subtask S3** In order to answer questions of type *(ii)*, participating systems are also required to identify participant roles in each identified answer incident (e.g., *victim*, *subject-suspect*), and use such information along with victim-related numerals (*"three people were killed"*) mentioned in the corresponding *answer documents*, i.e., documents that report on the answer incident, to determine the total number of victims.

**Datasets** The organizers released two datasets: (*i*) *test data*, stemming from three domains of gun violence, fire disasters and business, and (*ii*) *trial data*, covering only the gun violence domain. Each dataset contains (*i*) an input document (in CoNLL format) that comprises news articles, and (*ii*) a set of questions (in JSON format) to evaluate the participating systems.[2]

This paper describes the KOI (Knowledge of Incidents) system submitted to SemEval-2018 Task 5, which constructs and populates a knowledge graph of incidental events mentioned in news articles, to be used to retrieve answer incidents and answer documents given numerical questions about events. We propose a fully unsupervised approach to identify events and their properties in news texts, and to resolve within- and cross-document event coreference, which will be detailed in the following section.

## 2 System Description

### 2.1 Document Preprocessing

Given an input document in CoNLL format (one token per line), for each news article, we first split the sentences following the annotation of: (*i*) whether a token is part of the article title or content; (*ii*) sentence identifier; and (*iii*) whether a to-

[*] Both share the same amount of work.
[1] http://alt.qcri.org/semeval2018/
[2] https://competitions.codalab.org/competitions/17285



*Proceedings of the 12th International Workshop on Semantic Evaluation (SemEval-2018)*, pages 81–87
New Orleans, Louisiana, June 5–6, 2018. 

ken is a newline character. We then ran several tools on the tokenized sentences to obtain the following NLP annotations.

**Word sense disambiguation (WSD)** We ran Babelfy[3] (Moro et al., 2014) to get disambiguated *concepts* (excluding stop-words), which can be multi-word expressions, e.g., *gunshot wound*. Each concept is linked to a sense in BabelNet[4] (Navigli and Ponzetto, 2012), which subsequently is also linked to a WordNet sense and a DBpedia entity (if any).

**Named-entity recognition (NER)** We relied on spaCy[5] for a statistical entity recognition, specifically for identifying persons and geopolitical entities (countries, cities, and states).

**Time expression recognition and normalization** We used HeidelTime[6] (Strötgen and Gertz, 2013) for recognizing textual spans that indicate time, e.g., *this Monday*, and normalizing the time expressions according to a given document creation time, e.g., *2018-03-05*.

**Semantic role labeling (SRL)** Senna[7] (Collobert et al., 2011) was used to run semantic parsing on the input text, for identifying sentence-level events (i.e., predicates) and their participants.

### 2.2 Event Extraction and Coreference Resolution

**Identifying document-level events** Sentence-level events, i.e., *predicates* recognized by the SRL tool, were considered as the candidates for the document-level events. Note that predicates containing other predicates as the patient argument, e.g., *'says'* with arguments *'police'* as its agent and *'one man was shot to death'* as its patient, were not considered as candidate events.

Given a predicate, we simultaneously determined whether it is part of document-level events and also identified its type, based on the occurrence of *BabelNet concepts* that are related to four event types of interest stated in the task guidelines: *killing*, *injuring*, *fire burning* and *job firing*. A predicate is automatically labeled as a sentence-level event with one of the four types if such related concepts occur either in the predicate itself or in one of its arguments. For example, a predicate *'shot'*, with arguments *'one man'* as its patient and *'to death'* as its manner, will be considered as a *killing* event because of the occurrence of *'death'* concept.[8]

Concept relatedness was computed via path-based WordNet similarity (Hirst et al., 1998) of a given BabelNet concept, which is linked to a WordNet sense, with a predefined set of related WordNet senses for each event type (e.g., *wn30:killing.n.02* and *wn30:kill.v.01* for the killing event), setting 5.0 as the threshold. Related concepts were also annotated with the corresponding event types, to be used for the mention-level event coreference evaluation.

We then assumed all identified sentence-level events in a news article belonging to the same event type to be automatically regarded as *one* document-level event, meaning that each article may contain at most four document-level events (i.e., at most one event per event type).

**Identifying document-level event participants** Given a predicate as an identified event, its participants were simply extracted from the occurrence of named entities of type *person*, according to both Senna and spaCy, in the agent and patient arguments of the predicate. Furthermore, we determined the role of each participant as *victim*, *perpetrator* or *other*, based on its mention in the predicate. For example, if *'Randall'* is mentioned as the agent argument of the predicate *'shot'*, then he is a perpetrator. Note that a participant can have multiple roles, as is the case for a person who kills himself.

Taking into account all participants of a set of identified events (per event type) in a news article, we extracted document-level event participants by resolving name coreference. For instance, *'Randall'*, *'Randall R. Coffland'*, and *'Randall Coffland'* all refer to the same person.

**Identifying document-level number of victims** For each identified predicate in a given document, we extracted the first existing *numeral* in the patient argument of the predicate, e.g., *one* in *'one man'*. The normalized value of the numeral was then taken as the number of victims, as long as the predicate is not suspect-related predicates such

[3] http://babelfy.org/
[4] http://babelnet.org/
[5] https://spacy.io/
[6] https://github.com/HeidelTime/heideltime
[7] https://ronan.collobert.com/senna/

[8] We assume that a predicate that is labeled as a *killing* event cannot be labeled as an *injuring* event even though an injuring-related concept such as *'shot'* occurs.

as *suspected* or *charged*. The number of victims of document-level events is simply the maximum value of identified number of victims per predicate.

**Identifying document-level event locations** To retrieve candidate event locations given a document, we relied on disambiguated DBpedia entities as a result of Babelfy annotation. We utilized SPARQL queries over the *DBpedia SPARQL endpoint*[9] to identify whether a DBpedia entity is a *city* or a *state*, and whether it is *part of* or *located in* a city or a state. Specifically, an entity is considered to be a city whenever it is of type `dbo:City` or its equivalent types (e.g., `schema:City`). Similarly, it is considered to be a state whenever it is either of type `yago:WikicatStatesOfTheUnitedStates`, has a senator (via the property `dbp:senators`), or has `dbc:States_of_the_United_States` as a subject.

Assuming that document-level events identified in a given news article happen at *one* certain location, we simply ranked the candidate event locations, i.e., pairs of city and state, based on their frequencies, and took the one with the highest frequency.

**Identifying document-level event times** Given a document $D$, suppose we have $dct$ as the document creation time and $T$ as a list of normalized time expressions returned by HeidelTime, whose types are either `date` or `time`. We considered a time expression $t_i \in T$ as one of candidate event times $T' \subseteq T$, if $dct - t_i$ is a non-negative integer less than $n$ days.[10] We hypothesize that the event reported in a news article may have happened several days before the news is published.

Assuming that document-level events identified in a given news article happen at *one* certain time, we determine which one is the document-level event time from the set of candidates $T'$ by applying two heuristics: A time expression $t_j \in T'$ is considered as the event time, if *(i)* $t_j$ is mentioned in sentences containing event-related concepts, and *(ii)* $t_j$ is the earliest time expression in the candidate set.

**Cross-document event coreference resolution** We approached cross-document event coreference by clustering similar document-level events that are of the same type, via their *provenance*, i.e., news articles where they were mentioned. From each news article we derived TF-IDF-based vectors of *(i)* BabelNet senses and *(ii)* spaCy's persons and geopolitical entities, which are then used to compute cosine similarities among the articles.

Two news articles will be clustered together if *(i)* the computed similarity is above a certain threshold, which was optimized using the trial data, and *(ii)* the event time distance of document-level events found in the articles does not exceed a certain threshold, i.e., 3 days. All document-level events belonging to the same document cluster are assumed to be coreferring events and to have properties resulting from the aggregation of locations, times and participants of contributing events, with the exception of number of victims where the maximum value was taken instead.

| Resource Type | Properties |
|---|---|
| *IncidentEvent* | *eventType*, *eventDate*, *location*, *participant*, *numOfVictims* |
| *Document* | *docDate*, *docID*, *event* |
| *Participant* | *fullname*, *firstname*, *lastname*, *role* |
| *Location* | *city*, *state* |
| *Date* | *value*, *day*, *month*, *year* |

Table 1: KOI-KG ontology

## 2.3 Constructing, Populating and Querying the Knowledge Graph

We first built an *OWL ontology*[11] to capture the knowledge model of incidental events and documents. We rely on reification (Noy and Rector, 2006) for modeling entities, that is, incident events, documents, locations, participants and dates are all resources of their own. Each resource is described through its corresponding properties, as shown in Table 1.

An incident event can be of type *injuring*, *killing*, *fire_burning*, and *job_firing*. Documents are linked to incident events through the property *event*, and different documents may refer to the same corresponding incident event. We borrow URIs from DBpedia for values of the properties *city* and *state*. Participant roles can be either *victim*, *perpetrator* or *other*. A date has a unified literal value of the format "yyyy-mm-dd", as well as separated values for the day, month, and year.

To build the KOI knowledge graph (KOI-KG)

[9] `https://dbpedia.org/sparql`

[10] Based on our empirical observations on the trial data we found $n = 7$ to be the best parameter.

[11] Available at `https://koi.cs.ui.ac.id/ns`

```
SELECT ?event ?document
WHERE {
  ?event koi:eventType koi:killing .
  ?event koi:eventDate [
     koi:year "2017" ] .
  ?event koi:participant [
     koi:firstname "Sean" ] .
  ?document koi:event ?event .
}
```

Figure 1: A SPARQL query over KOI-KG for "Which killing events happened in 2017 that involve persons with Sean as first name?"

we relied on Apache Jena,[12] a Java-based Semantic Web framework. The output of the previously explained event extraction and coreference resolution steps was imported into the Jena TDB triple store as RDF triples. This facilitates SPARQL querying, which can be done using the Jena ARQ module. The whole dump of KOI-KG is available for download at https://koi.cs.ui.ac.id/incidents.

Given a question in JSON format, we applied mapping rules to transform it into a SPARQL query, which was then used to retrieve corresponding *answer incidents* and *answer documents*. Constraints of questions such as event type, participant, date, and location were mapped into SPARQL join conditions (that is, triple patterns). Figure 1 shows a SPARQL representation for the question "Which killing events happened in 2017 that involve persons with Sean as first name?". The prefix *koi* is for the KOI ontology namespace (https://koi.cs.ui.ac.id/ns#). In the SPARQL query, the join conditions are over the event type *killing*, the date *'2017'* (as year) and the participant *'Sean'* (as firstname).

For Subtask S2, we extended the SPARQL query with counting feature to retrieve the total number of unique events. Analogously, for Subtask S3, we retrieve number of victims by counting event participants having *victim* as their roles, and by getting the value of the *numOfVictims* property (if any). The value of the *numOfVictims* property was preferred as the final value for an incident if it exists, otherwise, KOI relied on counting event participants.

We also provide a SPARQL query interface for KOI-KG at https://koi.cs.ui.ac.id/dataset.html?tab=query&ds=/incidents.

[12] http://jena.apache.org/

## 3 Results and Discussion

**Evaluation results** Participating systems were evaluated according to three evaluation schemes: *(i) mention-level evaluation*, for resolving cross-document coreference of event mentions, *(ii) document-level evaluation* (`doc-f1`), for identifying events and their properties given a document, and *(iii) incident-level evaluation*, for combining event extraction and within-/cross-document event coreference resolution to answer numerical questions in terms of exact matching (`inc-acc`) and Root Mean Square Error (`inc-rmse`). Furthermore, the percentage of questions in each subtask that can be answered by the systems (`%ans`) also contributes to the final ranking.

Regarding the *mention-level evaluation*, KOI achieves an average F1-score of 42.8% (36.3 percentage point increase over the baseline) from several established metrics for evaluating coreference resolution systems. For *document-level* and *incident-level evaluation* schemes, we report in Table 2 the performance of three different system runs of KOI:

- `v1` Submitted version of KOI during the *evaluation period*.
- `v2` Similar as `v1`, however, instead of giving no answers when we found no matching *answer incidents*, KOI simply returns *zero* as the numerical answer with an empty list of *answer documents*.
- `v3` Submitted version of KOI during the *post-evaluation period*, which incorporates improvement on document-level event time identification leading to enhanced cross-document event coreference.[13]

Compared to the baseline provided by the task organizers, the performance of KOI is considerably better, specifically of KOI `v3` for subtask S2 with `doc-f1` and `inc-acc` around twice as much as of the baseline. Hereafter, our quantitative and qualitative analyses are based on KOI `v3`, and mentions of the KOI system refer to this system run.

**Subtask S1** We detail in Table 3, the performance of KOI on retrieving relevant *answer documents* given questions with event constraints,

[13] Submission `v1` and `v2` did not consider heuristic *(i)* that we have discussed in Section 2.2.

| system run | subtask S1 | | subtask S2 | | | | subtask S3 | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | %ans | doc-f1 | %ans | doc-f1 | inc-acc | inc-rmse | %ans | doc-f1 | inc-acc | inc-rmse |
| *baseline* | *16.5* | *67.3* | *100.0* | *26.4* | *18.3* | *8.5* | - | - | - | - |
| KOI v1* | 44.2 | 83.0 | 67.5 | 55.2 | 20.4 | 6.2 | 66.6 | 69.6 | 19.3 | 7.9 |
| KOI v2 | 44.2 | 83.0 | 100.0 | 51.2 | 25.6 | 5.2 | 100.0 | 49.1 | 24.8 | 7.1 |
| KOI v3 | 55.1 | 85.7 | 100.0 | 54.8 | 27.4 | 5.3 | 100.0 | 50.9 | 23.0 | 7.7 |

Table 2: KOI performance results at SemEval-2018 Task 5 (in percentages) for three subtasks, *baseline* was provided by the task organizers, *) denotes the system run that we submitted during the evaluation period.

| | micro-averaged | | | macro-averaged | | |
|---|---|---|---|---|---|---|
| | p | r | f1 | p | r | f1 |
| **Overall** | | | | | | |
| answered questions | 86.6 | 74.0 | 79.8 | 94.2 | 83.6 | 85.7 |
| all questions | 86.6 | 41.6 | 56.2 | 51.7 | 45.9 | 47.1 |
| **Event type** | | | | | | |
| killing | 88.5 | 43.2 | 58.1 | 56.8 | 48.6 | 50.3 |
| injuring | 82.8 | 37.4 | 51.5 | 46.4 | 40.1 | 41.4 |
| job_firing | 100.0 | 8.7 | 16.0 | 15.4 | 15.4 | 15.4 |
| fire_burning | 96.9 | 66.2 | 78.7 | 65.5 | 66.2 | 65.7 |
| **Event constraint** | | | | | | |
| participant | 84.8 | 43.0 | 57.0 | 61.1 | 51.1 | 53.2 |
| location | 89.1 | 39.4 | 54.6 | 46.7 | 42.8 | 43.6 |
| time | 86.0 | 42.4 | 56.8 | 51.7 | 46.3 | 47.4 |

Table 3: KOI performance results for subtask S1, on *answer document* retrieval (p for precision, r for recall and f1 for F1-score).

in terms of *micro-averaged* and *macro-averaged* scores. Note that the official doc-f1 scores reported in Table 2 correspond to macro-averaged F1-scores.

We first analyzed the system performance only on *answered questions*, i.e., for which KOI returns the relevant answer documents (55.1% of all questions), yielding 79.8% and 85.7% micro-averaged and macro-averaged F1-scores, respectively.

In order to have a fair comparison with systems that are able to answer *all questions*, we also report the performance of KOI that returns empty sets of answer documents for unanswered questions. In this evaluation scheme, the macro-averaged precision is significantly lower than the micro-averaged one (51.7% vs 86.6%), because systems are heavily penalized for not retrieving relevant answer documents per question, i.e., given zero precision score, which brings the average over all questions down. Meanwhile, the micro-averaged precision measures the systems' ability in returning relevant documents for all questions regardless of whether the questions were answered or not. KOI focuses on yielding high quality answer documents, which is reflected by high micro-averaged precision of above 80% in general. The following result analyses are based on the *all questions* scheme.

| | subtask S2 | | subtask S3 | |
|---|---|---|---|---|
| | inc-acc | inc-rmse | inc-acc | inc-rmse |
| overall | 27.4 | 5.3 | 23.0 | 7.7 |
| zero | 96.3 | 0.2 | 55.2 | 6.8 |
| non-zero | 18.9 | 5.6 | 11.9 | 8.0 |

Table 4: KOI performance results for subtasks S2 and S3, on answering numerical questions, i.e., number of incidents and number of victims.

By analyzing the document retrieval per event type, we found that KOI can identify *fire_burning* events in documents quite well, yielding the highest recall among all event types, but the contrary for *job_firing* events. With respect to event constraints, answering questions with *location* constraint results in the worst performance, meaning that our method is still lacking in identifying and/or disambiguating event locations from news documents. Specifically, questions with *city* constraint are more difficult to answer compared to the ones with *state* constraint (49.6% vs 61.5% micro-averaged F1-scores, respectively).

**Subtask S2** The key differences between Subtask S1 and S2 are: *(i)* questions with *zero* as an answer are included, and *(ii)* there can be more than one *answer incidents* per question, hence, systems must be able to cluster *answer documents* into the correct number of clusters, i.e., incidents.

As shown in Table 4, KOI is able to answer questions with zero as the true answer with 96.3% accuracy. Meanwhile, for questions with non-zero number of incidents as the answers, KOI gives numerical answers with 18.9% accuracy, resulting in overall accuracy (inc-acc) of 27.4% and RMSE inc-rmse) of 5.3.

We also analyzed questions (with non-zero answer incidents) for which KOI yields perfect sets of answer documents with 100% F1-score, i.e., 7.7% of all questions. For 61.8% of such answered questions, KOI returns the perfect number of inci-

| Event ID: 22409 |
|---|
| *2016-06-19*<br>**Man playing with gun while riding in a car fatally shoots, kills driver**<br>A man was fatally shot early Sunday morning after the passenger in the car he was driving accidentally discharged the gun, according to the San Antonio Police Department. The shooting occurred about 3 a.m. when group of four men were driving out of the Iron Horse Apartments at 8800 Village Square on the Northeast Side. The passenger in the front seat was playing with a gun and allegedly shot himself in the hand, according to officers at the scene. The bullet went through his hand and struck the driver in the abdomen. The men then drove to Northeast Baptist Hospital, which was nearby, but the driver was pronounced dead at the hospital, according to investigators. Police believe the driver and passenger to be related and are still investigating the incident. The other two men in the vehicle were detained. No charges have been filed. |
| *2016-06-19*<br>**41-year - old man killed in overnight shooting**<br>SAN ANTONIO - A 41-year-old man is dead after a shooting police say may have been accidental. The victim died after another man drove him to Northeast Baptist Hospital for treatment of that gunshot wound. Police say they got a call at around 2:45 a.m. for the shooting in the 8800 block of Village Drive. The man told them he and the victim were in a pickup when he fired the shot, but police say it's not known why the men were in the truck. Investigators say the man told them he fired the shot accidentally and struck the victim. Police say the shooter took the victim to the emergency room at Northeast Baptist, where hospital personnel pronounced him dead. Police are questioning the man who did the shooting. |

Table 5: An identified *'killing'* event by KOI for *"Which killing incidents happened in June 2016 in San Antonio, Texas?"* with two supporting documents.

dents. For the rest, KOI tends to overestimate the number of incidents, i.e., for 30.9% of the cases, KOI fails to establish cross-document event coreference links with the current document clustering method.

**Subtask S3** We also show in Table 4, the KOI performance on answering numerical questions about number of victims. KOI is able to answer correctly 55.2% of questions with zero answers, and 11.9% of the ones with non-zero answers.

Analyzing the questions with zero as the true answer, for which KOI is able to answer correctly, in 41.1% of the cases KOI is able to identify the non-existence of victims when the set of answer documents is not empty. In 40.0% of the cases, the correctly predicted zero answers are actually by chance, i.e., because KOI fails to identify relevant answer documents.

Meanwhile, for questions with gold numerical answers greater than zero, KOI returns wrong answers in 88.1% of the cases. Among these answers, 66.9% of the answers are lower than the true number of victims, and 33.1% are higher. This means that KOI tends to underestimate the number of victims with 6.6 RMSE.

For 22.5% of all questions, KOI is able to identify the perfect sets of answer documents with 100% F1-score. Among these questions, 34.3% were answered correctly with the exact number of victims, for which: 52.7% of correct answers result from solely *counting participants (as victims)*, 35.3% were inferred only from *numeral mentions*, and the rest of 12.0% were answered by combining both victim counting and numeral mentions.

**Qualitative Analysis** Recalling the example questions mentioned in the beginning of Section 1, for the first question, KOI is able to perfectly identify 2 killing incidents with 5 supporting documents pertaining to the event-time and -location constraints. One of the identified answer incidents with two supporting documents is shown in Table 5, which shows how well the system is able to establish cross-document event coreference, given overlapping concepts and entities. However, in answering the second question, KOI returns one less number of victims since it cannot identify the killed victim in the answer incident shown in Table 5, due to the lack of numeral mentions and named event participants as victims.

## 4 Conclusion

We have introduced a system called KOI (Knowledge of Incidents), that is able to build a knowledge graph (KG) of incidental events by extracting relevant event information from news articles. The resulting KG can then be used to efficiently answer numerical questions about events such as *"How many people were killed in June 2016 in San Antonio, Texas?"* We have submitted KOI as a participating system at SemEval-2018 Task 5, which achieved competitive results. A live demo of our system is available at `https://koi.cs.ui.ac.id/`. Future directions of this work include the incorporation of supervised (or semi-supervised) approaches for specific steps of KOI such as the extraction of numeral information (Mirza et al., 2017), as well as the investigation of applying our approach to other domains such as disease outbreaks and natural disasters.

## References


Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. *J. Mach. Learn. Res.*, 12:2493–2537.

Graeme Hirst, David St-Onge, et al. 1998. Lexical chains as representations of context for the detection and correction of malapropisms. *WordNet: An electronic lexical database*, 305:305–332.

Paramita Mirza, Simon Razniewski, Fariz Darari, and Gerhard Weikum. 2017. Cardinal virtues: Extracting relation cardinalities from text. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 2: Short Papers*, pages 347–351.

Andrea Moro, Alessandro Raganato, and Roberto Navigli. 2014. Entity Linking meets Word Sense Disambiguation: a Unified Approach. *Transactions of the Association for Computational Linguistics (TACL)*, 2:231–244.

Roberto Navigli and Simone Paolo Ponzetto. 2012. BabelNet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artificial Intelligence*, 193:217–250.

Natasha Noy and Alan Rector, editors. 2006. *Defining N-ary Relations on the Semantic Web*. W3C Working Group Note. Retrieved Jan 10, 2017 from https://www.w3.org/TR/2006/NOTE-swbp-n-aryRelations-20060412/.

Marten Postma, Filip Ilievski, and Piek Vossen. 2018. Semeval-2018 task 5: Counting events and participants in the long tail. In *Proceedings of the 12th International Workshop on Semantic Evaluation (SemEval-2018)*. Association for Computational Linguistics.

Jannik Strötgen and Michael Gertz. 2013. Multilingual and cross-domain temporal tagging. *Language Resources and Evaluation*, 47(2):269–298.