The Universal Decompositional Semantics Dataset and Decomp Toolkit

Aaron Steven White[▷], **Elias Stengel-Eskin**[◁], **Siddharth Vashishtha**[▷], **Venkata Govindarajan**[‡], **Dee Ann Reisinger**[◁], **Tim Vieira**[◁], **Keisuke Sakaguchi**[†], **Sheng Zhang**[◁], **Francis Ferraro**[○], **Rachel Rudinger**[†], **Kyle Rawlins**[◁], **Benjamin Van Durme**[◁]

[▷]University of Rochester, [◁]Johns Hopkins University, [○]University of Maryland Baltimore County

[†]Allen Institute for Artificial Intelligence, [‡]University of Texas at Austin

Abstract

We present the Universal Decompositional Semantics (UDS) dataset (v1.0), which is bundled with the Decomp toolkit (v0.1). UDS1.0 unifies five high-quality, decompositional semantics-aligned annotation sets within a single semantic graph specification—with graph structures defined by the predicative patterns produced by the PredPatt tool and real-valued node and edge attributes constructed using sophisticated normalization procedures. The Decomp toolkit provides a suite of Python 3 tools for querying UDS graphs using SPARQL. Both UDS1.0 and Decomp0.1 are publicly available at `http://decomp.io`.

Keywords: semantics, semantic roles, factuality, genericity, temporal duration, entity typing

1. Introduction

Traditional semantic annotation frameworks generally define complex, often exclusive category systems that require highly trained annotators to build (Palmer et al., 2005; Banarescu et al., 2013; Abend and Rappoport, 2013; Oepen et al., 2014; Oepen et al., 2015; Bos et al., 2017; Abzianidze et al., 2017; Abzianidze and Bos, 2017; Schneider et al., 2018). And in spite of their high quality for the cases they are designed to handle, these frameworks can be brittle to cases that (i) deviate from prototypical instances of a category; (ii) are equally good instances of multiple categories; or (iii) fall under a category that was erroneously excluded from the framework's ontology.[1]

An alternative approach to semantic annotation that addresses these issues has been growing in popularity: decompositional semantics (Reisinger et al., 2015; White et al., 2016). In this approach, which is rooted in a long tradition of theoretical approaches to lexical semantics (Pustejovsky, 1995; Levin and Rappaport Hovav, 2005, and references therein), semantic annotation takes the form of many simple questions about words or phrases (in context) that are easy for naïve native speakers to answer, thus allowing annotations to be crowd-sourced while retaining high inter-annotator agreement.

The decompositional approach can be thought of as a feature-based counterpart to traditional category-based systems, with each question determining a semantic feature. Common feature configurations often correspond to categories in a traditional framework (Reisinger et al., 2015; Govindarajan et al., 2019); but unlike such frameworks, a decompositional approach retains the ability to capture configurations that were not considered at design time. Further, unlike a categorical framework, reannotation after an overhaul of the framework's ontology is never necessary, since additional annotations simply accrue to sharpen the framework's ability to capture fine-grained semantic phenomena. A variety of semantic annotation datasets that take a decompositional approach now exist, including ones that target semantic roles (Reisinger et al., 2015; White et al., 2016), entity types (White et al., 2016), event factuality (White et al., 2016; Rudinger et al., 2018), linguistic expressions of generalizations about entities and events (Govindarajan et al., 2019), and temporal properties of and relations between events (Vashishtha et al., 2019). But despite the potential benefits of a decompositional approach—as well as the broad coverage of linguistic phenomena it has been shown to afford—a remaining obstacle to widespread adoption is the lack of a unified interface to these resources: prior work in UDS has approached annotation piecemeal—each effort focused on a restricted set of linguistic phenomena—without a broader push toward creating a unified semantic parsing resource.

To remedy this situation, we present the Universal Decompositional Semantics (UDS) dataset (v1.0) and the Decomp toolkit (v0.1), which we make publicly available at `http://decomp.io`. UDS1.0 unifies the five high-quality, decompositional semantics-aligned annotation sets listed above within a single semantic graph specification—with graph structures defined by the predicative patterns produced by the PredPatt tool and real-valued node and edge attributes constructed using sophisticated response normalization procedures. The Decomp toolkit provides a suite of Python 3 tools that make working with these data seamless, enabling a wide range of queries on Universal Decompositional Semantics graphs using the SPARQL 1.1 query language.

2. Data

UDS1.0 consists of three layers of annotations built on top of the English Web Treebank (Bies et al., 2012, EWT): (i) syntactic graphs (§2.1.) built from existing gold Universal Dependencies (UD) parses on EWT (Nivre et al., 2015); (ii) semantic graphs (§2.2.) built from the predicate-argument structures deterministically extracted from those parses using the PredPatt tool (White et al., 2016; Zhang et al., 2017); and (iii) semantic types (§2.3.) for the predicates, arguments, and their relationships, derived from five decompositional semantics-aligned datasets. Figure 1 shows an example UDS graph with all three layers of annotation.

[1]Shalev et al. (2019) discuss multiple recent, instructive examples of such brittleness.

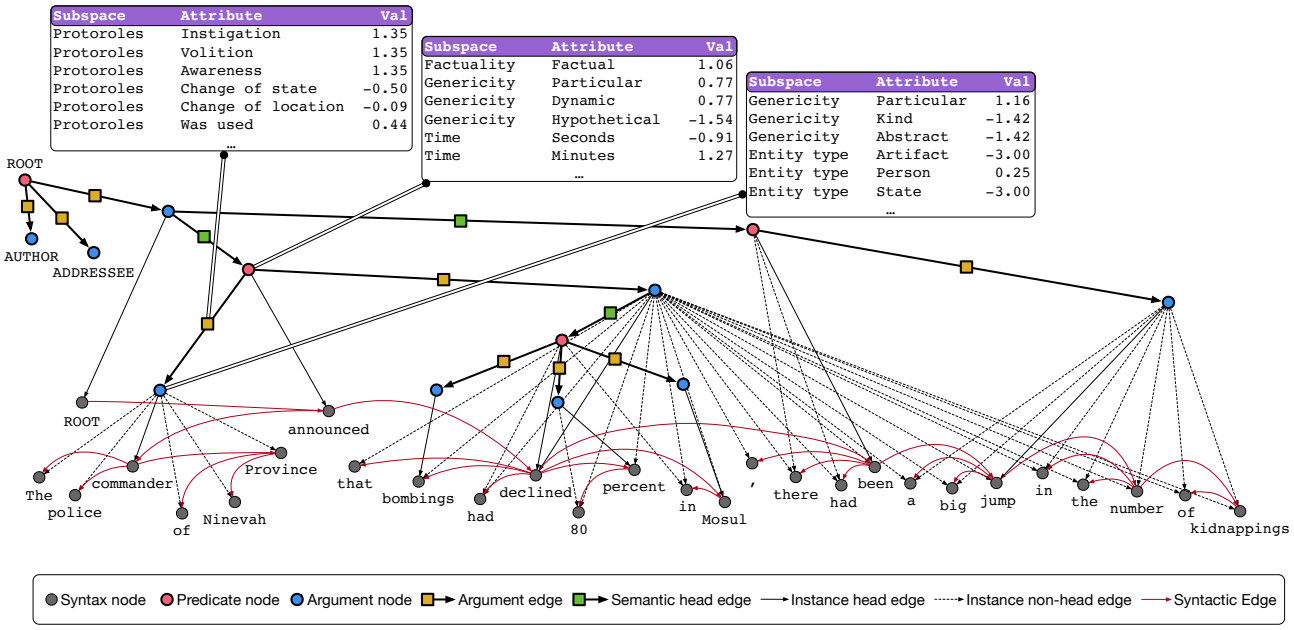

Figure 1: An example Universal Decompositional Semantics graph. Some semantic type information and most syntactic structure information (e.g. dependency relation and part-of-speech tags) are not shown but are available in the dataset.

2.1. Syntactic Graph

The syntactic graphs that form the first layer of annotation in the dataset come from gold UD dependency parses provided in the UD-EWT treebank (v1.2), which contains sentences from the Linguistic Data Consortium's constituency parsed EWT in CoNLL-U format. UDS1.0 inherits UD-EWT's training, development, and test splits.

In UDS1.0, each dependency parsed sentence in UD-EWT is represented as a rooted directed graph (digraph). Each token in a sentence is associated with a node that has, at minimum, the following attributes:

- `position` (`int`): the ordinal position of that node as an integer (1-indexed)
- `domain` (`str`): the subgraph this node is part of (always `syntax`)
- `type` (`str`): the type of the object in the particular domain (always `token`)
- `form` (`str`): the actual token
- `lemma` (`str`): the lemma corresponding to the token
- `upos` (`str`): the UD part-of-speech tag
- `xpos` (`str`): the Penn TreeBank part-of-speech tag

In addition, any attribute found in the UD features column of the CoNLL-U are inherited as node attributes by the syntactic graph.

Each graph also has a special root node that always has `domain` and `type` attributes set to `root`. Edges within the graph represent the grammatical relations annotated in UD-EWT and are directed, always pointing from the head to the dependent of the relation. At minimum, each edge has the following attributes:

- `domain` (`str`): the subgraph this node is part of (always `syntax`)
- `type` (`str`): the type of the object in the particular domain (always `dependency`)
- `deprel` (`str`): the UD dependency relation tag

For information about the values `upos`, `xpos`, `deprel`, and the attributes contained in the features column can take on, see the UD Guidelines.

2.2. Semantic Graphs

The semantic graphs that form the second layer of annotation in the dataset are produced by the PredPatt system (White et al., 2016; Zhang et al., 2017). PredPatt takes as input a UD parse and produces a set of predicates and set of arguments of each predicate. Both predicates and arguments are associated with a single head token in the sentence as well as a set of tokens that make up the predicate or argument (its span). Predicate or argument spans may be trivial in only containing the head token.

For example, given the dependency parse for the sentence (1) and its UD parse, PredPatt produces the following.

(1) Chris_1 gave_2 the_3 book_4 to_5 Pat_6

```
?a gave ?b to ?c
   ?a: Chris
   ?b: the book
   ?c: Pat
```

Assuming UD's 1-indexation, the single predicate in this sentence (*gave...to*) has a head at position 2 and a span over positions $\{2, 5\}$. This predicate has three arguments, one headed by *Chris* at position 1, with span over position $\{1\}$; one headed by *book* at position 4, with span over positions $\{3, 4\}$; and one headed by *Pat* at position 6, with span over position $\{6\}$.[2]

Each predicate and argument produced by PredPatt is associated with a node in a digraph. At minimum, each such node has the following attributes:

[2] See the PredPatt documentation tests for further examples.

- `domain` (`str`): the subgraph this node is part of (always semantics)
- `type` (`str`): the type of the object in the particular domain (either `predicate` or `argument`)

Predicate and argument nodes produced by PredPatt furthermore always have at least one outgoing instance edge that points to nodes in the syntax domain that correspond to the associated span of the predicate or argument. At minimum, each such edge has the following attributes.

- `domain` (`str`): the subgraph this node is part of (always `interface`)
- `type` (`str`): the type of the object in the particular domain (either `head` or `nonhead`)

Because PredPatt produces a unique head for each predicate and argument, there is always exactly one instance edge of type head from any particular node in the semantics domain. There may or may not be instance edges of type nonhead.
In addition to instance edges, predicate nodes always have exactly one outgoing edge connecting them to each of the nodes corresponding to their arguments. At minimum, each such edge has the following attributes.

- `domain` (`str`): the subgraph this node is part of (always `semantics`)
- `type` (`str`): the type of the object in the particular domain (always `dependency`)

There is one special case where an argument node has an outgoing edge that points to a predicate node: clausal subordination. For example, given the dependency parse for the sentence (2), PredPatt produces the following:

(2) Gene_1 thought_2 that_3 Chris_4 gave_5 the_6 book_7 to_8 Pat_9

```
?a thinks ?b
   ?a: Gene
   ?b: SOMETHING := that Chris gave
                    the book to Pat

?a gave ?b to ?c
   ?a: Chris
   ?b: the book
   ?c: Pat
```

In this case, the second argument of the predicate headed by *thinks* is the argument that *Chris gave the book to Pat*, which is headed by *gave*. This argument is associated with a node of type argument with span over positions {3, 4, 5, 6, 7, 8, 9}. In addition, there is a predicate headed by *gave*. This predicate is associated with a node with span over positions {5, 8}. This predicate node in turn has an outgoing edge pointing to the argument node. At minimum, each such edge has the following attributes:

- `domain` (`str`): the subgraph this node is part of (always `semantics`)
- `type` (`str`): the type of the object in the particular domain (always `head`)

The type attribute in this case has the same value as instance edges (`head`), but crucially the domain attribute is distinct. In the case of instance edges, it is `interface` and in the case of clausal subordination, it is `semantics`. This matters when making queries against the graph and serializing to an RDF-based format.
Every semantic graph contains at least four additional *performative* nodes that are not produced by PredPatt.[3]

- an argument node representing the entire sentence in the same way complement clauses are represented
- a predicate node representing the authors production of the entire sentence directed at the addressee
- an argument node representing the author
- an argument node representing the addressee

All of these nodes have a `domain` attribute with value `semantics`. Unlike nodes associated with PredPatt predicates and arguments, the predicate node representing the authors production, the argument node representing the author, and the argument node representing the addressee. The argument node representing the entire sentence does, however, have an instance head edge to the syntactic root node. This node also has semantics head edges to each of the predicate nodes in the graph that are not dominated by any other semantics node. The predicate node representing the author's production in turn has an argument edge to each of the three argument nodes listed above.
These performative nodes are included for purposes of forward compatibility. None of them currently have attributes, but future releases of decomp will include annotations on either them or their edges.

2.3. Semantic Types

PredPatt makes very coarse-grained typing distinctions—between predicate and argument nodes, on the one hand, and between dependency and head edges, on the other. UDS provides ultra fine-grained typing distinctions, represented as collections of real-valued attributes. The collection of all node and edge attributes defined in UDS determines the *UDS type space*; any cohesive subset determines a *UDS type subspace*.[4]

2.3.1. Deriving attribute values and confidence scores

UDS attributes are derived from crowd-sourced annotations of the heads or spans corresponding to predicates and/or arguments (described in §2.3.2. and §2.3.3.) and are represented in the dataset as node or edge attributes. All of

[3]The term *performative* because these nodes are intended to represent something akin to analogous syntactically represented nodes argued for by Ross (1972).

[4]It is important to note that, though all nodes and edges in the semantics domain have a type attribute, UDS does not afford any special status to these types. That is, the only thing that UDS "sees" are the nodes and edges in the semantics domain. This implies that the set of nodes and edges visible to UDS is, in principle, a superset of those associated with PredPatt predicates and their arguments. For UDS1.0, however, we only include attributes of nodes and edges built from predicate-argument patterns produced by PredPatt. In future releases, additional edge types will be added. For instance, predicate-predicate temporal relations are currently annotated in the temporal relations dataset from which we extract temporal durations (Vashishtha et al., 2019).

Annotated Nodes

Train

	Factuality	Genericity	Time	Entity Type
Factuality	21,092	20,929	20,733	0
Genericity		56,594	26,314	16,873
Time			26,324	0
Entity Type				17,192

Dev

	Factuality	Genericity	Time	Entity Type
Factuality	2,476	2,456	2,320	0
Genericity		6,858	3,051	1,894
Time			3,051	0
Entity Type				1,943

Test

	Factuality	Genericity	Time	Entity Type
Factuality	2,413	2,394	2,275	0
Genericity		6,602	2,927	1,847
Time			2,927	0
Entity Type				1,876

Table 1: The number of predicate and argument nodes that are annotated for both the node type subspace along the columns and the one along the rows. The diagonal elements show the total number of nodes annotated for a particular node type subspace. The Entity Type subspace is not annotated on any of the same nodes as Factuality and Time because Entity Type is only annotated on arguments and Factuality and Time are only annotated on predicates.

Annotated Edges + Nodes

Train

	Factuality	Genericity	Time	Entity Type
Factuality	0	3,935	0	2,670
Genericity		4,200	4,163	2,903
Time			0	2,883
Entity Type				0

Dev

	Factuality	Genericity	Time	Entity Type
Factuality	0	536	0	340
Genericity		570	542	365
Time			0	344
Entity Type				0

Test

	Factuality	Genericity	Time	Entity Type
Factuality	0	481	0	298
Genericity		507	471	320
Time			0	296
Entity Type				0

Table 2: The number of edges that are annotated for semantic protoroles and which touch predicate and argument nodes, where one is annotated for at least the node type subspace along the columns and the other is annotated for the one along the rows. Zeros arise for node type subspaces that are only annotated for either predicate or argument nodes—e.g. because the Time subspace is only annotated on predicates, Time cannot be annotated for both nodes. (This will change in future versions of UDS.)

these attributes come from existing datasets that are publicly available at `http://decomp.io`.[5] Table 1 provides a breakdown of the number of PredPatt argument and predicate nodes annotated for different node type subspaces. In total, there are 57,080 annotated nodes in the training set, 6,927 in the development set, and 6,650 in the test set. Table 2 provides a similar breakdown, showing the number of PredPatt edges that are annotated for an edge type subspace and touch nodes that are annotated for different node type subspaces. In total, there are 5,669 annotated edges in the training set, 751 in the development set, and 670 in the test set.

There are currently four node type subspaces in UDS1.0: (i) factuality; (ii) genericity; (iii) time; and (iv) entity type. There is currently one edge type subspace: semantic protoroles. For each attribute annotated on a particular node or edge, UDS1.0 provides two values: (i) the *attribute value* itself (a real value) and (ii) a *researcher confidence score* (a value on $[0, 1]$). The attribute value combines information about both (a) whether the attribute holds—in its sign—and (ii) the *annotator confidence responses* made available with each dataset (in a form that depends on the particular dataset). The researcher confidence score quantifies our certainty that the attribute value accurately reflects all annotators' responses. This has the consequence that the more variable annotators responses are, the lower the researcher confidence score will be.

Both the attribute value and the researcher confidence are derived from mixed effects models (MEMs). The goal in using these models is derive a single attribute value for each attribute on each node and edge that adjusts for idiosyncracies in how annotators use the particular instruments through which their annotations were collected, while simultaneously capturing variability across annotators' responses to the same item. Such adjustment is not possible with simpler methods—e.g. just taking the mean response across annotators.

Because each dataset uses a distinct annotation protocol with distinct annotation instruments—e.g. some datasets, such as the factuality dataset (White et al., 2016; Rudinger et al., 2018), are collected using a binary instrument with ordinal confidence responses, while others, such as the semantic protoroles dataset (Reisinger et al., 2015; White et al., 2016) use an ordinal instrument with binary confidence responses—the particular mixed model used for each differs (see §2.3.2. and §2.3.3. for details). But each model conforms to the same overarching principle: for each attribute and each node or edge that that attribute applies to, we assume (i) that there is some true, fixed real value for the node or edge on that attribute; (ii) that each annotator (drawn randomly from the annotator population) maps that attribute value to the response scale in a way specific to

[5] All of these datasets are introduced in peer-reviewed publications which report extensive interannotator agreement statistics, and so we do not report such statistics here.

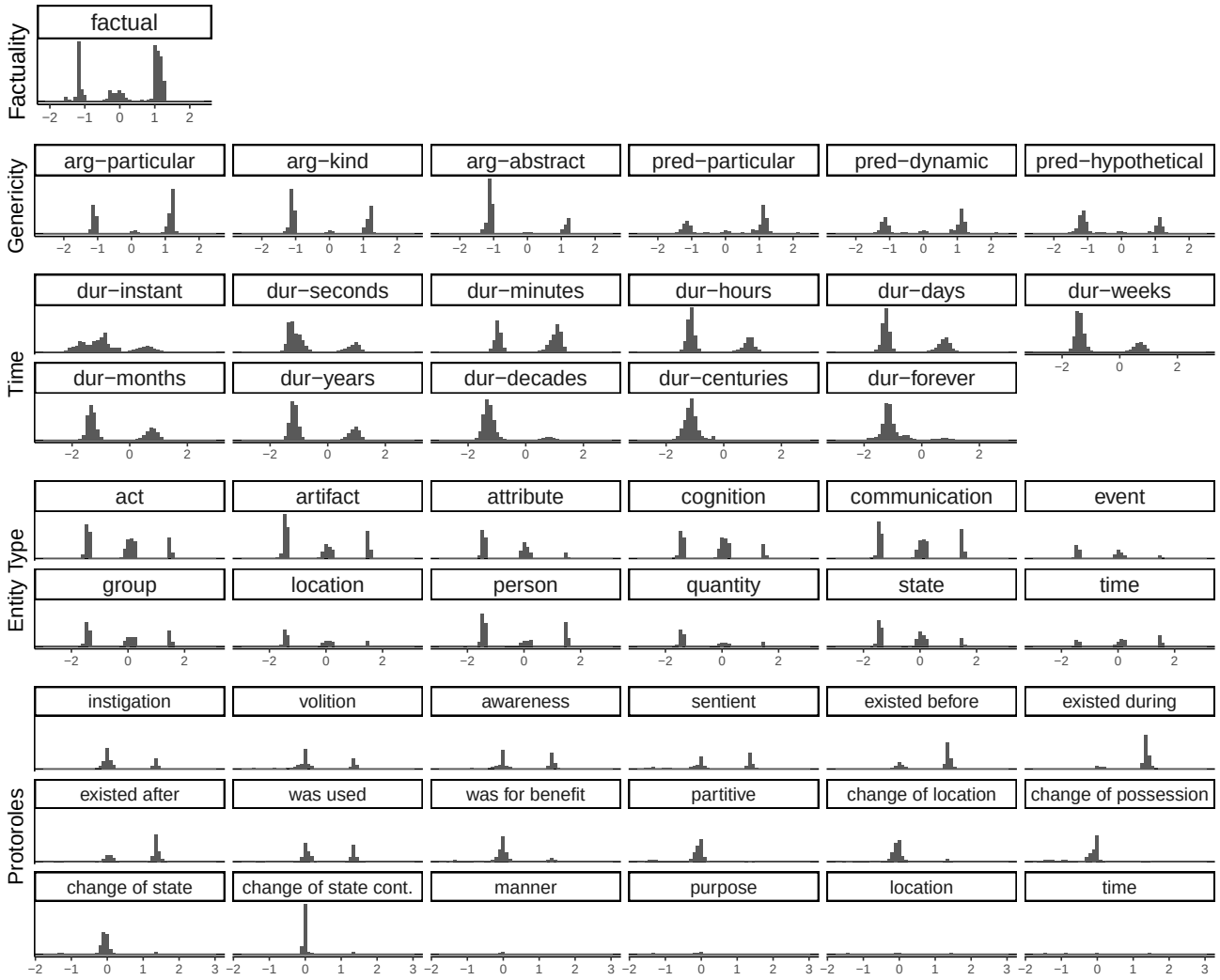

Figure 2: Distribution of attribute values in training and development sets. Only a subset of entity types are shown. Cases where the entity type attribute values default to the minimum are also excluded.

that annotator; and (iii) that each response by an annotator should be weighted by their reported confidence for that response (normalized to account for the fact that some annotators take different approaches to reporting confidence). Thus, we treat the attribute value as a fixed effect in a MEM (subsequently z-scored), the annotator response mappings as random effects, and the normalized annotator confidence score (always on $[0, 1]$) as a weight on the MEM's loss function. The distribution of these attribute value scores is shown in Figure 2.

We derive our researcher confidence scores from these MEMs. Each MEM we use is probabilistic in the sense that the fixed attribute value and the random annotator mappings are optimized to maximize the likelihood of the observed responses by mapping these values through some *link function*—e.g. for the datasets that use binary responses, we use a standard logistic as the link function. We compute the researcher confidence score for a particular attribute value of a particular node or edge as the mean of the likelihoods assigned to annotator responses for that attribute on that node or edge, weighted by the normalized annotator confidence response for that attribute on that node or edge. This has the effect that, if all annotators agree on a particular annotation with high (normalized) confidence, then the attribute value will be extreme and researcher confidence will be high; if half the annotators respond one way with high confidence and the other half responds another with high confidence, then researcher confidence will be low, since the value will be middling and will not assign particularly high likelihood to an particular response; and if half the annotators respond one way with high confidence and the other half responds another with low confidence, then researcher confidence will be middling.

2.3.2. Node attributes

The four node type subspaces in UDS1.0—factuality, genericity, time, and entity type—are all derived from datasets collected by presenting annotators with a sentence containing a highlighted (possibly discontiguous) span corresponding to either a predicate (factuality, genericity, time) or an argument (genericity, entity type).

Factuality The UDS-Factuality dataset (v2.0) annotates predicates for whether or not the event or state that they refer to happened according to the author of the containing sentence (White et al., 2016; Rudinger et al., 2018). Annotations are collected as binary responses along with an

ordinal (1-5) confidence rating.

To normalize the ordinal confidence ratings across participants, we ridit score each participants' confidence ratings (Agresti, 2014). In ridit scoring, ordinal labels are normalized to (0, 1) using the empirical cumulative distribution function of the ratings given by each annotator. Specifically, for the responses $\mathbf{y}_a$ given by annotator a:[6]

$$\text{ridit}_{\mathbf{y}_a}(y_{ai}) = \text{ECDF}_{\mathbf{y}_a}(y_{ai} - 1) + 0.5 \times \text{ECDF}_{\mathbf{y}_a}(y_{ai})$$

These normalized confidence ratings are entered as weights on the likelihood in a logistic mixed effects model. This model has real-valued fixed effects β_i for each annotated predicate token i and real-valued random intercepts u_a for each annotator a. As is standard for mixed effects models, these parameters are optimized against a cross-entropy loss—in this case, a Bernoulli likelihood—with an additional term to enforce that random intercepts are normally distributed with mean 0 and unknown variance σ^2.[7]

$$\mathcal{L} = \sum_i \sum_{a \in \alpha(i)} r_{ai} \log \text{Bern}\left(y_{a\rho_a(i)}; p_{ai}\right) + \sum_a \log \mathcal{N}\left(u_a; 0, \text{Var}(\mathbf{u})\right)$$

where $\alpha(i)$ is the set of annotators that annotated predicate token i for factuality, $\rho_a(i)$ is the index of the response to predicate i within y_{ai}, $r_{ai} = \text{ridit}_{\mathbf{y}_a}\left(y_{a\rho_a(i)}\right)$, and $p_{ai} = \text{logit}^{-1}(\beta_i + u_a)$. We then take β_i as the attribute value for predicate token i.

To derive the researcher confidence score c_i for a predicate token i, we compute the mean of the probabilities of the annotations weighted by r_{ai}:

$$c_i = \frac{\sum_{a \in \alpha(i)} r_{ai} \text{Bern}\left(y_{a\rho_a(i)}; p_{ai}\right)}{\sum_{a \in \alpha(i)} r_{ai}}$$

Genericity The UDS-Genericity dataset (v1.0) annotates both predicates and arguments for a variety of attributes relevant to linguistic expression of generalization, including (i) whether or not a predicate refers to (a) a particular event or state (or some collection thereof); (b) a dynamic event; (c) a hypothetical situation; and (ii) whether or not an argument refers to (a) a particular thing or collection thereof; (b) a kind of thing; or (c) an abstract object (Govindarajan et al., 2019). Like UDS-Factuality, annotations are collected as binary responses along with an ordinal (1-5) confidence rating, and so we use the same approach for constructing attribute values and research confidence scores used there.

Time The UDS-Time dataset (v1.0) annotates predicates for the likely duration of the event or state they refer to—whether it was *instantaneous* or lasted *seconds*, *minutes*, *hours*, *days*, *weeks*, *months*, *years*, *decades*, *centuries*, or *forever*—along with an ordinal (1-5) confidence response. It also annotates pairs of predicates for the continuous temporal relation between the events or states the predicates in that pair refer to, along with an ordinal confidence response. We include only the duration annotations in UDS1.0 for reasons mentioned above.

We normalize the confidence ratings using ridit scoring, as for UDS-Factuality and UDS-Genericity. For normalizing the duration responses themselves, there are two reasonable options that are both generalizations of the approach taken for binary responses. The first would be to treat the duration responses as ordinal variables and induce a single duration value using an ordinal link logit model (Agresti, 2014). The second is to treat them as nominal variables and induce a real value for each duration using a multinomial logistic mixed model.

We disprefer the first approach for two reasons. First, in mapping this response to a single real value, we lose information about the real world duration. The duration could be recovered by providing the ordinal model's binning of the real scale into duration values, but we take this indirection to be suboptimal. Second, we lose information about possible ambiguity in the duration leading to multimodal responses—e.g. being sick could be something that lasts for days or weeks, but it could also refer to a lifelong affliction, lasting decades. Ambiguity-driven multimodality is problematic for all our annotations—this is one reason why we include a researcher confidence score—but it is particularly problematic here in light of the first problem.

As such, we implement the second approach, which yields a real-valued attribute corresponding to each duration. We derive this attribute from a multinomial logistic mixed effects model analogously to how we derive values for binary responses. In this case, the fixed effects $\boldsymbol{\beta}_i$ for each predicate token i and the random effects $\mathbf{u}_a$ for annotator a are vectors of length equal to the number of duration responses.

$$\mathcal{L} = \sum_i \sum_{a \in \alpha(i)} r_{ai} \log \text{Cat}\left(y_{a\rho_a(i)}; \mathbf{p}_{ai}\right) + \sum_{a,k} \log \mathcal{N}\left(u_{ak}; 0, \text{Var}(\mathbf{u}_{\cdot k})\right)$$

where $\mathbf{p}_{ai} = \text{softmax}(\boldsymbol{\beta}_i + \mathbf{u}_a)$ and $\text{Var}(\mathbf{U})$ is a covariance matrix estimated from $\mathbf{U}$. We then take β_{ik} as the attribute value for duration k for predicate token i. We derive the corresponding research confidence score c_{ik} analogously to what was done for binary responses.

$$c_{ik} = \frac{\sum_{a \in \alpha(i)} r_{ai} \text{Cat}\left(y_{a\rho_a(i)}; \mathbf{p}_{ai}\right)}{\sum_{a \in \alpha(i)} r_{ai}}$$

Entity type The UDS-WordSense dataset (v1.0) annotates (the nominal heads of) arguments for the WordNet 3.0 (Miller, 1995; Fellbaum, 1998) senses that those (nominal heads of) arguments can have. For any particular argument, annotators were presented with all of the definitions of senses listed in WordNet for the head of that argument and asked to select all that were applicable using check boxes. After MEM-based normalization of the sense responses (described below), we extract entity types for these annotations by mapping the selected senses to their supersenses/lexicographer classes (Ciaramita and Johnson, 2003) and deriving a real-valued attribute value for each

[6] see Govindarajan et al. 2019 for a recent use of such scoring in an NLP context, along with an intuitive explanation of its use.

[7] The variance σ^2 is not optimized because that would result in driving it toward ∞; rather, it is estimated from $\mathbf{u}$. This implies that the second term remains constant, and it is correct behavior, since this term is merely included to encode the assumption of random sampling over annotators by controlling the shape of the distribution of $\mathbf{u}$.

supersense from the normalized values associated with the senses that fall under it.

To normalize the sense responses, we use a logistic mixed effects model with real-valued fixed effects β_{ik} for each annotated argument (head) token i and potential sense k and real-valued random intercepts u_a for each annotator a:[8]

$$\mathcal{L} = \sum_i \sum_{k \in \pi(i)} \sum_{a \in \alpha(i)} \log \text{Bern}\left(y_{a\rho_a(i)k}; p_{aik}\right) + \sum_a \log \mathcal{N}\left(u_a; 0, \text{Var}(\mathbf{u})\right)$$

where $\pi(i)$ is the set of potential senses for argument (head) token i, and $p_{aik} = \text{logit}^{-1}(\beta_{ik} + u_a)$. We then take β_{ik} as the attribute value for predicate token i and sense k.

To derive the researcher confidence score c_{ik} for a predicate token i and potential sense k, we compute the mean of the probabilities of the annotations:

$$c_{ik} = \frac{\sum_{a \in \alpha(i)} \text{Bern}\left(y_{a\rho_a(i)k}; p_{aik}\right)}{|\alpha(i)|}$$

We compute the attribute value γ_{il} and research confidence d_{ik} for each argument head token and supersense l from β_{ik} and c_{ik} for all senses k that fall under supersense l:

$$\gamma_{il} = \begin{cases} \max_{k \in \pi(i) \cap \psi(l)} \beta_{ik} & \pi(i) \cap \psi(l) \neq \emptyset \\ \min_{i,k} \beta_{ik} & \text{otherwise} \end{cases}$$

where $\psi(l)$ is the set of senses that fall under supersense l. The research confidence is computed analogously:

$$d_{il} = \begin{cases} \max_{k \in \pi(i) \cap \psi(l)} c_{ik} & \pi(i) \cap \psi(l) \neq \emptyset \\ 1 & \text{otherwise} \end{cases}$$

We default to a confidence of 1 here because if no sense of an argument (head) can fall under a particular supersense, then we have high confidence that the value should be low.

2.3.3. Edge attributes

The single node type subspaces in UDS1.0—the UDS-Protoroles (v2.x) dataset (White et al., 2016)—is derived from a dataset collected by presenting annotators with a sentence containing two highlighted (possibly discontiguous) spans corresponding to a predicate and an argument. Annotators responded to 18 questions on an ordinal (1-5) scale, all starting with *how likely or unlikely is it that...*[9]

1. `instigation`: ...ARG caused the PRED to happen?
2. `volition`: ...ARG chose to be involved in the PRED?
3. `awareness`: ...ARG was/were aware of being involved in the PRED?
4. `sentient`: ...ARG was/were sentient?
5. `change of location`: ...ARG changed location during the PRED?
6. `existed before`: ...ARG existed before the PRED began?
7. `existed during`: ...ARG existed during the PRED?
8. `existed after`: ...ARG existed after the PRED stopped?
9. `change of possession`: ...ARG changed possession during the PRED?
10. `change of state`: ...ARG was/were altered or somehow changed during or by the end of the PRED?
11. `change of state continuous`: ...the change in ARG happened throughout the PRED? (*only shown if the answer to* `change of state` *was 4 or 5*)
12. `was used`: ...ARG was/were used in carrying out the PRED?
13. `was for benefit`: ...PRED happened for the benefit of ARG?
14. `partitive`: ...*only* a part or portion of ARG was involved in the PRED?
15. `manner`: ...ARG was the manner of the PRED?
16. `purpose`: ...ARG was the purpose of the PRED?
17. `location`: ...ARG was the location of the PRED?
18. `time`: ...ARG was when the PRED happened?

UDS-Protoroles does not provides confidence annotations beyond whatever notion of confidence is part of giving the ordinal response to these questions; however, if the annotator responded with a 3 or less, an additional question was revealed asking whether the question was applicable. We normalize the ordinal response and applicability response separately and then combine the normalized ratings.

For the ordinal response, we use an ordinal link logit mixed effects model (Agresti, 2014) with real-valued fixed effects β_{ik} for each annotated predicate-argument pair i and each property k and real-valued random intercepts $\mathbf{u}_a$ for each annotator a. This sort of model, which has been used in prior work on the semantic protoroles (v1.0) dataset (White et al., 2017), is a straightforward generalization of logistic regression to more than two labels. The random intercepts $\mathbf{u}_a$ are a vector with four monotonically increasing elements that separates the real values into five bins corresponding to the five ordinal responses. These intercepts (or *cutpoints*) are used to define a cumulative categorical probability distribution, from which the probability mass function for said distribution can be reconstructed.

$$\mathbb{P}(y_{a\rho_a(i)k} \leq l) = \begin{cases} \text{logit}^{-1}(\beta_{ik} - u_{al}) & \text{if } l \in \{1, ..., 4\} \\ 1 & \text{if } l = 5 \\ 0 & \text{otherwise} \end{cases}$$

The probability p_{aikl} of an ordinal response l for a predicate-argument pair i on property k by annotator a is thus defined as:

$$p_{aikl} = \mathbb{P}(y_{a\rho_a(i)k} \leq l) - \mathbb{P}(y_{a\rho_a(i)k} \leq l - 1)$$

The likelihood is then a simple categorical likelihood. The loss term for the random effects places a distribution with

[8] Unlike the annotations from which we derive the other three node type subspaces, UDS-WordSense does not contain annotator confidence responses. We thus do not weight the likelihood of the MEM by normalized annotator confidence responses.

[9] Questions 1-14 are modified versions of the questions used for the UDS-Protoroles (v1.0) dataset (Reisinger et al., 2015) and were asked about arguments headed by the subject or direct object of the predicate's head. Questions 15-18 were asked about a distinct (and much smaller) set of arguments/adjuncts that were dependents of the predicate's head, but not subjects or direct objects. This is why the histograms for these attributes in Figure 2 are so low compared to the other questions.

strictly positive support—here, an exponential—on the distance between the random intercepts.

$$\mathcal{L} = \sum_i \sum_{a \in \alpha(i)} \log \text{Cat}\left(y_{a\rho_a(i)k}; \mathbf{p}_{aik}\right) + \sum_{a,l} \log \text{Exp}\left(u_{al} - u_{a(l-1)}; \frac{1}{\text{Var}(\mathbf{u}_{\cdot l} - \mathbf{u}_{\cdot(l-1)})}\right)$$

We derive the corresponding researcher confidence score c_{ik} analogously to the node type subspaces.

$$c_{ik} = \frac{\sum_{a \in \alpha(i)} \text{Cat}\left(y_{a\rho_a(i)k}; \mathbf{p}_{aik}\right)}{|\alpha(i)|}$$

The applicability ratings are normalized using a logisitic mixed effects model to yield fixed effects δ_{ik} for each predicate argument pair i and property k. The final normalized score for a predicate argument pair i and property k is then computed as $\text{logit}^{-1}(\delta_{ik})\beta_{ik}$. This pulls properties that are not applicable for a particular pair toward zero.

3. Toolkit

The Decomp toolkit (v0.1) is a Python 3 package that provides utilities for:

1. reading the the UDS dataset from the underlying treebank and annotations or directly from its native JSON format, including facilities for quickly adding user-defined annotations to the graphs
2. serializing UDS graphs to many common formats, such as Notation3, N-Triples, turtle, and JSON-LD, as well as any other format supported by NetworkX
3. querying both the syntactic and semantic subgraphs of UDS (as well as pointers between them) using SPARQL 1.1 queries

This last feature is particularly useful for quickly and easily searching for sentences based on complex syntactic and semantic constraints. These queries can be relatively simple. For example, if one were interested in extracting only predicates referring to events that likely happened and likely lasted for minutes:

```
SELECT ?pred
WHERE { ?pred <domain> <semantics> ;
              <type> <predicate> ;
              <factual> ?factual ;
              <dur-minutes> ?duration
              FILTER ( ?factual > 0 &&
                       ?duration > 0
                     )
      }
```

But they can also be arbitrarily sophisticated. For instance, if one were interested in extracting all predicate-argument edges that (i) touch a predicate referring to an event that is likely spatiotemporally delimited; and (ii) touch at least one argument that refers to a spatiotemporally delimited participant that was volitional in the event:

```
SELECT ?edge
WHERE { ?pred ?edge ?arg ;
              <domain> <semantics> ;
              <type>   <predicate> ;
              <pred-particular> ?ppart
              FILTER ( ?ppart > 0 ) .
        ?arg  <domain> <semantics> ;
              <type>   <argument>  ;
              <arg-particular> ?apart
              FILTER ( ?apart > 0 ) .
        { ?edge <volition> ?volition
          FILTER ( ?volition > 0 )
        } UNION
        { ?edge <sentient> ?sentient
          FILTER ( ?sentient > 0 )
        }
      }
```

Further, syntactic and semantic constraints can be mixed. For instance, if one were interested in all copular predicates with at least one argument that refers to a spatiotemporally delimited participant that was sentient in the event referred to by the predicate:

```
SELECT ?pred
WHERE { ?pred ?semedge ?arg ;
              <domain> <semantics> ;
              <type>   <predicate> .
        ?arg  <domain> <semantics> ;
              <type>   <argument>  ;
              <arg-particular> ?apart
              FILTER ( ?apart > 0 ) .
        ?semedge <sentient> ?sentient
           FILTER ( ?sentient > 0 ) .
        ?pred ?instedge ?head .
        ?instedge <domain> <interface> ;
                  <type>   <head> .
        ?head ?synedge ?syndep .
        ?syndep <deprel> ?relation
           FILTER ( ?relation = "cop" ) .
      }
```

4. Conclusion

We presented the Universal Decompositional Semantics dataset (v1.0), which is bundled with the Decomp toolkit (v0.1) and discussed how we construct the Universal Decompositional Semantics (UDS) dataset (v1.0) by unifying five high-quality, decompositional semantics-aligned annotation sets within a single semantic graph specification based on the predicative patterns produced by the PredPatt tool. We also presented the Decomp toolkit (v0.1), which provides a suite of Python 3 tools for querying Universal Decompositional Semantics graphs using SPARQL 1.1. We believe these resources will be helpful to (i) those wishing to pursue corpus linguistic studies on the existing annotations; and (ii) those pursuing broad-coverage semantic parsing algorithms, where the structural distinctions in UDS, as compared to previously existing semantic annotated corpora, may offer unique challenges.

Acknowledgements

This research was supported by the University of Rochester, JHU HLTCOE, the National Science Foundation (BCS-1748969/BCS-1749025), DARPA AIDA, DARPA KAIROS, and IARPA BETTER. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes. The views and conclusions contained in this publication are those of the authors and should not be interpreted as representing official policies or endorsements of DARPA or the U.S. Government.

References

Abend, O. and Rappoport, A. (2013). Universal Conceptual Cognitive Annotation (UCCA). In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 228–238, Sofia, Bulgaria, August. Association for Computational Linguistics.

Abzianidze, L. and Bos, J. (2017). Towards Universal Semantic Tagging. In *IWCS 2017 12th International Conference on Computational Semantics Short papers*.

Abzianidze, L., Bjerva, J., Evang, K., Haagsma, H., van Noord, R., Ludmann, P., Nguyen, D.-D., and Bos, J. (2017). The Parallel Meaning Bank: Towards a Multilingual Corpus of Translations Annotated with Compositional Meaning Representations. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 242–247, Valencia, Spain, April. Association for Computational Linguistics.

Agresti, A. (2014). *Categorical Data Analysis*. John Wiley & Sons.

Banarescu, L., Bonial, C., Cai, S., Georgescu, M., Griffitt, K., Hermjakob, U., Knight, K., Koehn, P., Palmer, M., and Schneider, N. (2013). Abstract meaning representation for sembanking. In *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*, pages 178–186.

Bies, A., Mott, J., Warner, C., and Kulick, S. (2012). English web treebank. *Linguistic Data Consortium, Philadelphia, PA*.

Bos, J., Basile, V., Evang, K., Venhuizen, N., and Bjerva, J. (2017). The Groningen Meaning Bank. In Nancy Ide et al., editors, *Handbook of Linguistic Annotation*. Springer, Berlin.

Ciaramita, M. and Johnson, M. (2003). Supersense Tagging of Unknown Nouns in WordNet. In *Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing*, pages 168–175.

Fellbaum, C. (1998). *WordNet*. Wiley Online Library.

Govindarajan, V., Van Durme, B., and White, A. S. (2019). Decomposing Generalization: Models of Generic, Habitual, and Episodic Statements. *Transactions of the Association for Computational Linguistics*, 7:501–517.

Levin, B. and Rappaport Hovav, M. (2005). *Argument Realization*. Cambridge University Press, Cambridge.

Miller, G. A. (1995). WordNet: a lexical database for English. *Communications of the ACM*, 38(11):39–41.

Nivre, J., Agic, Z., Aranzabe, M. J., Asahara, M., Atutxa, A., Ballesteros, M., Bauer, J., Bengoetxea, K., Bhat, R. A., Bosco, C., Bowman, S., Celano, G. G. A., Connor, M., de Marneffe, M.-C., Diaz de Ilarraza, A., Dobrovoljc, K., Dozat, T., Erjavec, T., Farkas, R., Foster, J., Galbraith, D., Ginter, F., Goenaga, I., Gojenola, K., Goldberg, Y., Gonzales, B., Guillaume, B., Haji, J., Haug, D., Ion, R., Irimia, E., Johannsen, A., Kanayama, H., Kanerva, J., Krek, S., Laippala, V., Lenci, A., Ljubei, N., Lynn, T., Manning, C., Mrnduc, C., Mareek, D., Martnez Alonso, H., Maek, J., Matsumoto, Y., McDonald, R., Missil, A., Mititelu, V., Miyao, Y., Montemagni, S., Mori, S., Nurmi, H., Osenova, P., vrelid, L., Pascual, E., Passarotti, M., Perez, C.-A., Petrov, S., Piitulainen, J., Plank, B., Popel, M., Prokopidis, P., Pyysalo, S., Ramasamy, L., Rosa, R., Saleh, S., Schuster, S., Seeker, W., Seraji, M., Silveira, N., Simi, M., Simionescu, R., Simk, K., Simov, K., Smith, A., tpnek, J., Suhr, A., Sznt, Z., Tanaka, T., Tsarfaty, R., Uematsu, S., Uria, L., Varga, V., Vincze, V., abokrtsk, Z., Zeman, D., and Zhu, H. (2015). Universal Dependencies 1.2. *http://universaldependencies.github.io/docs/*.

Oepen, S., Kuhlmann, M., Miyao, Y., Zeman, D., Flickinger, D., Haji, J., Ivanova, A., and Zhang, Y. (2014). SemEval 2014 Task 8: Broad-Coverage Semantic Dependency Parsing. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 63–72, Dublin, Ireland, August. Association for Computational Linguistics.

Oepen, S., Kuhlmann, M., Miyao, Y., Zeman, D., Cinkov, S., Flickinger, D., Haji, J., and Ureov, Z. (2015). SemEval 2015 Task 18: Broad-Coverage Semantic Dependency Parsing. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 915–926, Denver, Colorado, June. Association for Computational Linguistics.

Palmer, M., Gildea, D., and Kingsbury, P. (2005). The proposition bank: An annotated corpus of semantic roles. *Computational Linguistics*, 31(1):71–106.

Pustejovsky, J. (1995). *The Generative Lexicon*. MIT Press, Cambridge, MA.

Reisinger, D., Rudinger, R., Ferraro, F., Harman, C., Rawlins, K., and Van Durme, B. (2015). Semantic Proto-Roles. *Transactions of the Association for Computational Linguistics*, 3:475–488.

Ross, J. R. (1972). Act. In Donald Davidson et al., editors, *Semantics of Natural Language*, pages 70–126. Springer.

Rudinger, R., White, A. S., and Van Durme, B. (2018). Neural Models of Factuality. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 731–744, New Orleans, Louisiana. Association for Computational Linguistics.

Schneider, N., Hwang, J. D., Srikumar, V., Prange, J., Blodgett, A., Moeller, S. R., Stern, A., Bitan, A., and Abend, O. (2018). Comprehensive Supersense Disambiguation of English Prepositions and Possessives. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 185–196, Melbourne, Australia, July. Association for Computational Linguistics.

Shalev, A., Hwang, J. D., Schneider, N., Srikumar, V., Abend, O., and Rappoport, A. (2019). Preparing SNACS for Subjects and Objects. In *Proceedings of the First International Workshop on Designing Meaning Representations*, pages 141–147, Florence, Italy, August. Association for Computational Linguistics.

Vashishtha, S., Van Durme, B., and White, A. S. (2019). Fine-Grained Temporal Relation Extraction. In *Proceedings of the 57th Annual Meeting of the Association for*

Computational Linguistics, pages 2906–2919, Florence, Italy, July. Association for Computational Linguistics.

White, A. S., Reisinger, D., Sakaguchi, K., Vieira, T., Zhang, S., Rudinger, R., Rawlins, K., and Van Durme, B. (2016). Universal decompositional semantics on universal dependencies. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1713–1723, Austin, TX. Association for Computational Linguistics.

White, A. S., Rawlins, K., and Van Durme, B. (2017). The Semantic Proto-Role Linking Model. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, volume 2, pages 92–98, Valencia, Spain. Association for Computational Linguistics.

Zhang, S., Rudinger, R., and Van Durme, B. (2017). An Evaluation of PredPatt and Open IE via Stage 1 Semantic Role Labeling. In *IWCS 2017 12th International Conference on Computational Semantics Short papers*.