

# SemEval-2019 Task 1: Cross-lingual Semantic Parsing with UCCA

Daniel Hershcovich  
Elior Sulem

Zohar Aizenbud  
Ari Rappoport

Leshem Choshen  
Omri Abend

School of Computer Science and Engineering  
Hebrew University of Jerusalem

{danielh,zohara,borgr,eliors,arir,oabend}@cs.huji.ac.il

## Abstract

We present the SemEval 2019 shared task on Universal Conceptual Cognitive Annotation (UCCA) parsing in English, German and French, and discuss the participating systems and results. UCCA is a cross-linguistically applicable framework for semantic representation, which builds on extensive typological work and supports rapid annotation. UCCA poses a challenge for existing parsing techniques, as it exhibits reentrancy (resulting in DAG structures), discontinuous structures and non-terminal nodes corresponding to complex semantic units. The shared task has yielded improvements over the state-of-the-art baseline in all languages and settings. Full results can be found in the task's website <https://competitions.codalab.org/competitions/19160>.

## 1 Overview

Semantic representation is receiving growing attention in NLP in the past few years, and many proposals for semantic schemes have recently been put forth. Examples include Abstract Meaning Representation (AMR; [Banarescu et al., 2013](#)), Broad-coverage Semantic Dependencies (SDP; [Oepen et al., 2016](#)), Universal Decompositional Semantics (UDS; [White et al., 2016](#)), Parallel Meaning Bank ([Abzianidze et al., 2017](#)), and Universal Conceptual Cognitive Annotation (UCCA; [Abend and Rappoport, 2013](#)). These advances in semantic representation, along with corresponding advances in semantic parsing, can potentially benefit essentially all text understanding tasks, and have already demonstrated applicability to a variety of tasks, including summarization ([Liu et al., 2015](#); [Dohare and Karnick, 2017](#)), paraphrase detection ([Issa et al., 2018](#)), and semantic evaluation (using UCCA; see below). In this shared task, we focus on UCCA parsing in multiple languages.

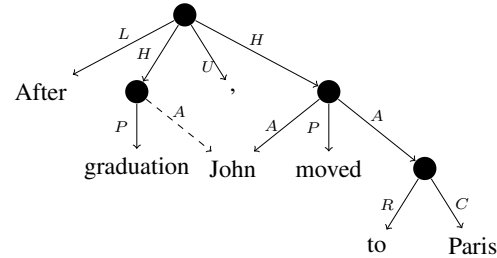


Figure 1: An example UCCA graph.

One of our goals is to benefit semantic parsing in languages with less annotated resources by making use of data from more resource-rich languages. We refer to this approach as *cross-lingual* parsing, while other works ([Zhang et al., 2017, 2018](#)) define cross-lingual parsing as the task of parsing text in one language to meaning representation in another language.

In addition to its potential applicative value, work on semantic parsing poses interesting algorithmic and modeling challenges, which are often different from those tackled in syntactic parsing, including reentrancy (e.g., for sharing arguments across predicates), and the modeling of the interface with lexical semantics.

UCCA is a cross-linguistically applicable semantic representation scheme, building on the established Basic Linguistic Theory typological framework ([Dixon, 2010b,a, 2012](#)). It has demonstrated applicability to multiple languages, including English, French and German, and pilot annotation projects were conducted on a few languages more. UCCA structures have been shown to be well-preserved in translation ([Sulem et al., 2015](#)), and to support rapid annotation by non-experts, assisted by an accessible annotation interface ([Abend et al., 2017](#)).<sup>1</sup> UCCA has already shown applicative value for text simplifica-

<sup>1</sup><https://github.com/omriabnd/UCCA-App>

Scene Elements		
P	<b>Process</b>	The main relation of a Scene that evolves in time (usually an action or movement).
S	<b>State</b>	The main relation of a Scene that does not evolve in time.
A	<b>Participant</b>	Scene participant (including locations, abstract entities and Scenes serving as arguments).
D	<b>Adverbial</b>	A secondary relation in a Scene.
Elements of Non-Scene Units		
C	<b>Center</b>	Necessary for the conceptualization of the parent unit.
E	<b>Elaborator</b>	A non-Scene relation applying to a single Center.
N	<b>Connector</b>	A non-Scene relation applying to two or more Centers, highlighting a common feature.
R	<b>Relator</b>	All other types of non-Scene relations: (1) Rs that relate a C to some super-ordinate relation, and (2) Rs that relate two Cs pertaining to different aspects of the parent unit.
Inter-Scene Relations		
H	<b>Parallel Scene</b>	A Scene linked to other Scenes by regular linkage (e.g., temporal, logical, purposive).
L	<b>Linker</b>	A relation between two or more Hs (e.g., “when”, “if”, “in order to”).
G	<b>Ground</b>	A relation between the speech event and the uttered Scene (e.g., “surprisingly”).
Other		
F	<b>Function</b>	Does not introduce a relation or participant. Required by some structural pattern.

Table 1: The complete set of categories in UCCA’s foundational layer.

tion (Sulem et al., 2018b), as well as for defining semantic evaluation measures for text-to-text generation tasks, including machine translation (Birch et al., 2016), text simplification (Sulem et al., 2018a) and grammatical error correction (Choshen and Abend, 2018).

The shared task defines a number of tracks, based on the different corpora and the availability of external resources (see §5). It received submissions from eight research groups around the world. In all settings at least one of the submitted systems improved over the state-of-the-art TUPA parser (Hershcovich et al., 2017, 2018), used as a baseline.

## 2 Task Definition

UCCA represents the semantics of linguistic utterances as directed acyclic graphs (DAGs), where terminal (childless) nodes correspond to the text tokens, and non-terminal nodes to semantic units that participate in some super-ordinate relation. Edges are labeled, indicating the role of a child in the relation the parent represents. Nodes and edges belong to one of several *layers*, each corresponding to a “module” of semantic distinctions.

UCCA’s *foundational layer* covers the predicate-argument structure evoked by predicates of all grammatical categories (verbal, nominal, adjectival and others), the inter-relations between them, and other major linguistic phenomena such as semantic heads and multi-word expressions. It is the only layer for which annotated corpora exist at the moment, and is thus the target of this shared task. The layer’s basic notion is the *Scene*, describing a state, action,

movement or some other relation that evolves in time. Each Scene contains one main relation (marked as either a Process or a State), as well as one or more Participants. For example, the sentence “After graduation, John moved to Paris” (Figure 1) contains two Scenes, whose main relations are “graduation” and “moved”. “John” is a Participant in both Scenes, while “Paris” only in the latter. Further categories account for inter-Scene relations and the internal structure of complex arguments and relations (e.g., coordination and multi-word expressions). Table 1 provides a concise description of the categories used by the UCCA foundational layer.

UCCA distinguishes *primary* edges, corresponding to explicit relations, from *remote* edges (appear dashed in Figure 1) that allow for a unit to participate in several super-ordinate relations. Primary edges form a tree in each layer, whereas remote edges enable reentrancy, forming a DAG.

UCCA graphs may contain *implicit* units with no correspondent in the text. Figure 2 shows the annotation for the sentence “A similar technique is almost impossible to apply to other crops, such as cotton, soybeans and rice.”<sup>2</sup> It includes a single Scene, whose main relation is “apply”, a secondary relation “almost impossible”, as well as two complex arguments: “a similar technique” and the coordinated argument “such as cotton, soybeans, and rice.” In addition, the Scene includes an implicit argument, which represents the agent of the “apply” relation.

While parsing technology is well-established

<sup>2</sup>The same example was used by Oepen et al. (2015) to compare different semantic dependency schemes.

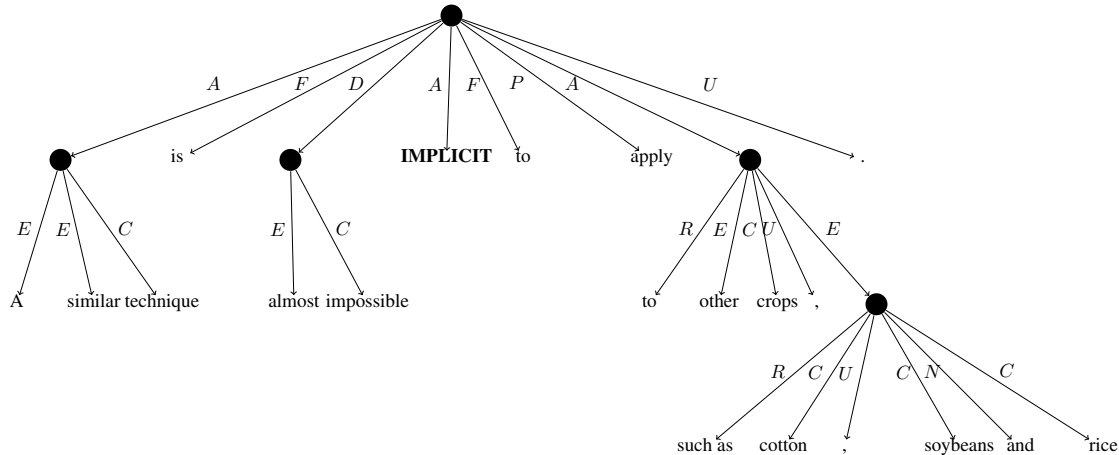


Figure 2: UCCA example with an implicit unit.

for syntactic parsing, UCCA has several formal properties that distinguish it from syntactic representations, mostly UCCA’s tendency to abstract away from syntactic detail that do not affect argument structure. For instance, consider the following examples where the concept of a Scene has a different rationale from the syntactic concept of a clause. First, non-verbal predicates in UCCA are represented like verbal ones, such as when they appear in copula clauses or noun phrases. Indeed, in Figure 1, “graduation” and “moved” are considered separate Scenes, despite appearing in the same clause. Second, in the same example, “John” is marked as a (remote) Participant in the graduation Scene, despite not being explicitly mentioned. Third, consider the possessive construction in “John’s trip home”. While in UCCA “trip” evokes a Scene in which “John” is a Participant, a syntactic scheme would analyze this phrase similarly to “John’s shoes”.

The differences in the challenges posed by syntactic parsing and UCCA parsing, and more generally by semantic parsing, motivate the development of targeted parsing technology to tackle it.

### 3 Data & Resources

All UCCA corpora are freely available.<sup>3</sup> For English, we use v1.2.3 of the Wikipedia UCCA corpus (Wiki), v1.2.2 of the UCCA *Twenty Thousand Leagues Under the Sea* English-French parallel corpus (20K), which includes UCCA manual annotation for the first five chapters in French and English, and v1.0.1 of the UCCA German *Twenty*

*Thousand Leagues Under the Sea* corpus, which includes the entire book in German. For consistent annotation, we replace any Time and Quantifier labels with Adverbial and Elaborator in these data sets. The resulting training, development<sup>4</sup> and test sets<sup>5</sup> are publicly available, and the splits are given in Table 2. Statistics on various structural properties are given in Table 3.

The corpora were manually annotated according to v1.2 of the UCCA guidelines,<sup>6</sup> and reviewed by a second annotator. All data was passed through automatic validation and normalization scripts.<sup>7</sup> The goal of validation is to rule out cases that are inconsistent with the UCCA annotation guidelines. For example, a Scene, defined by the presence of a Process or a State, should include at least one Participant.

Due to the small amount of annotated data available for French, we only provided a minimal training set of 15 sentences, in addition to the development and test set. Systems for French were expected to pursue semi-supervised approaches, such as cross-lingual learning or structure projection, leveraging the parallel nature of the corpus, or to rely on datasets for related formalisms, such as Universal Dependencies (Nivre et al., 2016). The full unannotated 20K Leagues corpus in English and French was released as well, in order to facilitate pursuing cross-lingual approaches.

Datasets were released in an XML format, including tokenized text automatically pre-

<sup>3</sup><https://github.com/UniversalConceptualCognitiveAnnotation>

<sup>4</sup><http://bit.ly/semEval2019task1traindev>

<sup>5</sup><http://bit.ly/semEval2019task1test>

<sup>6</sup><http://bit.ly/semEval2019task1guidelines>

<sup>7</sup><https://github.com/huji-nlp/ucca/tree/master/scripts>

corpus	train/trial		dev		test		total		
	sentences	tokens	sentences	tokens	sentences	tokens	passages	sentences	tokens
English-Wiki	4,113	124,935	514	17,784	515	15,854	367	5,142	158,573
English-20K	0	0	0	0	492	12,574	154	492	12,574
French-20K	15	618	238	6,374	239	5,962	154	492	12,954
German-20K	5,211	119,872	651	12,334	652	12,325	367	6,514	144,531

Table 2: Data splits of the corpora used for the shared task.

	En-Wiki	En-20K	Fr-20K	De-20K
# passages	367	154	154	367
# sentences	5,141	492	492	6,514
# tokens	158,739	12,638	13,021	144,529
# non-terminals	62,002	4,699	5,110	51,934
% discontinuous	1.71	3.19	4.64	8.87
% reentrant	1.84	0.89	0.65	0.31
# edges	208,937	16,803	17,520	187,533
% primary	97.40	96.79	97.02	97.32
% remote	2.60	3.21	2.98	2.68
by category				
% Participant	17.17	18.1	17.08	19.86
% Center	18.74	16.31	18.03	14.32
% Adverbial	3.65	5.25	4.18	5.67
% Elaborator	18.98	18.06	18.65	14.88
% Function	3.38	3.58	2.58	2.98
% Ground	0.03	0.56	0.37	0.57
% Parallel Scene	6.02	6.3	6.15	7.54
% Linker	2.19	2.66	2.57	2.49
% Connector	1.26	0.93	0.84	0.65
% Process	7.1	7.51	6.91	7.03
% Relator	8.58	8.09	9.6	7.54
% State	1.62	2.1	1.88	3.34
% Punctuation	11.28	10.55	11.16	13.15

Table 3: Statistics of the corpora used for the shared task.

processed using spaCy (see §5), and gold-standard UCCA annotation for the train and development sets.<sup>8</sup> To facilitate the use of existing NLP tools, we also released the data in SDP, AMR, CoNLL-U and plain text formats.

#### 4 TUPA: The Baseline Parser

We use the TUPA parser, the only parser for UCCA at the time the task was announced, as a baseline (Herscovich et al., 2017, 2018). TUPA is a transition-based DAG parser based on a BiLSTM-based classifier.<sup>9</sup> TUPA in itself has been found superior to a number of conversion-based parsers that use existing parsers for other formalisms to parse UCCA by constructing a two-way conversion protocol between the formalisms. It can thus be regarded as a strong baseline for sys-

<sup>8</sup><https://github.com/UniversalConceptualCognitiveAnnotation/docs/blob/master/FORMAT.md>

<sup>9</sup><https://github.com/huji-nlp/tupa>

tem submissions to the shared task.

## 5 Evaluation

**Tracks.** Participants in the task were evaluated in four settings:

1. English in-domain setting, using the Wiki corpus.
2. English out-of-domain setting, using the Wiki corpus as training and development data, and 20K Leagues as test data.
3. German in-domain setting, using the 20K Leagues corpus.
4. French setting with no training data, using the 20K Leagues as development and test data.

In order to allow both even ground comparison between systems and using hitherto untried resources, we held both an **open** and a **closed** track for submissions in the English and German settings. Closed track submissions were only allowed to use the gold-standard UCCA annotation distributed for the task in the target language, and were limited in their use of additional resources. Concretely, the only additional data they were allowed to use is that used by TUPA, which consists of automatic annotations provided by spaCy.<sup>10</sup> POS tags, syntactic dependency relations, and named entity types and spans. In addition, the closed track only allowed the use of word embeddings provided by fastText (Bojanowski et al., 2017)<sup>11</sup> for all languages.

Systems in the open track, on the other hand, were allowed to use any additional resource, such as UCCA annotation in other languages, dictionaries or datasets for other tasks, provided that they make sure not to use any additional gold standard annotation over the same text used in the UCCA

<sup>10</sup><http://spacy.io>

<sup>11</sup><http://fasttext.cc>

corpora.<sup>12</sup> In both tracks, we required that submitted systems are not trained on the development data. We only held an open track for French, due to the paucity of training data. The four settings and two tracks result in a total of 7 competitions.

**Scoring.** The following scores an output graph  $G_1 = (V_1, E_1)$  against a gold one,  $G_2 = (V_2, E_2)$ , over the same sequence of terminals (tokens)  $W$ . For a node  $v$  in  $V_1$  or  $V_2$ , define  $yield(v) \subseteq W$  as is its set of terminal descendants. A pair of edges  $(v_1, u_1) \in E_1$  and  $(v_2, u_2) \in E_2$  with labels (categories)  $\ell_1, \ell_2$  is *matching* if  $yield(u_1) = yield(u_2)$  and  $\ell_1 = \ell_2$ . Labeled precision and recall are defined by dividing the number of matching edges in  $G_1$  and  $G_2$  by  $|E_1|$  and  $|E_2|$ , respectively.  $F_1$  is their harmonic mean:

$$2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

Unlabeled precision, recall and  $F_1$  are the same, but without requiring that  $\ell_1 = \ell_2$  for the edges to match. We evaluate these measures for primary and remote edges separately. For a more fine-grained evaluation, we additionally report precision, recall and  $F_1$  on edges of each category.<sup>13</sup>

## 6 Participating Systems

We received a total of eight submissions to the different tracks: *MaskParse@Deskiñ* (Marzinotto et al., 2019) from Orange Labs and Aix-Marseille University, *HLT@SUDA* (Jiang et al., 2019) from Soochow University, *TüPa* (Pütz and Glocker, 2019) from the University of Tübingen, *UC Davis* (Yu and Sagae, 2019) from the University of California, Davis, *GCN-Sem* (Taslimipour et al., 2019) from the University of Wolverhampton, *CUNY-PekingU* (Lyu et al., 2019) from the City University of New York and Peking University, *DANGNT@UIT.VNU-HCM* (Nguyen and Tran, 2019) from the University of Information Technology VNU-HCM, and *XLangMo* from Zhejiang University. Some of the teams participated in more than one track and two systems (*HLT@SUDA* and *CUNY-PekingU*) participated in all the tracks.

<sup>12</sup>We are not aware of any such annotation, but include this restriction for completeness.

<sup>13</sup>The official evaluation script providing both coarse-grained and fine-grained scores can be found in [https://github.com/huji-nlp/ucca/blob/master/scripts/evaluate\\_standard.py](https://github.com/huji-nlp/ucca/blob/master/scripts/evaluate_standard.py).

In terms of parsing approaches, the task was quite varied. *HLT@SUDA* converted UCCA graphs to constituency trees and trained a constituency parser and a recovery mechanism of remote edges in a multi-task framework. *MaskParse@Deskiñ* used a bidirectional GRU tagger with a masking mechanism. *TüPa* and *XLangMo* used a transition-based approach. *UC Davis* used an encoder-decoder architecture. *GCN-SEM* uses a BiLSTM model to predict Semantic Dependency Parsing tags, when the syntactic dependency tree is given in the input. *CUNY-PKU* is based on an ensemble that includes different variations of the TUPA parser. *DANGNT@UIT.VNU-HCM* converted syntactic dependency trees to UCCA graphs.

Different systems handled remote edges differently. *DANGNT@UIT.VNU-HCM* and *GCN-SEM* ignored remote edges. *UC Davis* used a different BiLSTM for remote edges. *HLT@SUDA* marked remote edges when converting the graph to a constituency tree and trained a classification model for their recovery. *MaskParse@Deskiñ* handles remote edges by detecting arguments that are outside of the parent’s node span using a detection threshold on the output probabilities.

In terms of using the data, all teams but one used the UCCA XML format, two used the CoNLL-U format, which is derived by a lossy conversion process, and only one team found the other data formats helpful. One of the teams (*MaskParse@Deskiñ*) built a new training data adapted to their model by repeating each sentence  $N$  times,  $N$  being the number of non-terminal nodes in the UCCA graphs. Three of the teams adapted the baseline TUPA parser, or parts of it to form their parser, namely *TüPa*, *CUNY-PekingU* and *XLangMo*; *HLT@SUDA* used a constituency parser (Stern et al., 2017) as a component in their model; *DANGNT@UIT.VNU-HCM* is a rule-based system over the Stanford Parser, and the rest are newly constructed parsers.

All teams found it useful to use external resources beyond those provided by the Shared Task. Four submissions used external embeddings, MUSE (Conneau et al., 2017) in the case of *MaskParse@Deskiñ* and *XLangMo*, ELMo (Peters et al., 2018) in the case of *TüPa*,<sup>14</sup> and BERT (Devlin et al., 2018) in the case of *HLT@SUDA*.

<sup>14</sup>GCN-Sem used ELMo in the closed tracks, training on the available data.



#	Team	Labeled			Unlabeled		
		All	Prim.	Rem.	All	Prim.	Rem.
English-Wiki (closed)							
1	HLT@SUDA	77.4	77.9	52.2	87.2	87.9	52.5
2	baseline	72.8	73.3	47.2	85.0	85.8	48.4
3	Davis	72.2	73.0	0	85.5	86.4	0
4	CUNY-PekingU	71.8	72.3	49.5	84.5	85.2	50.1
5	DANGNT@UIT. VNU-HCM	70.0	70.7	0	81.7	82.6	0
6	GCN-Sem	65.7	66.4	0	80.9	81.8	0
English-Wiki (open)							
1	HLT@SUDA	80.5	81.0	58.8	89.7	90.3	60.7
2	CUNY-PekingU	80.0	80.2	66.6	89.4	89.9	67.4
3	baseline	73.5	73.9	53.5	85.1	85.7	54.3
3	TüPa	73.5	74.1	42.5	85.3	86.2	43.1
4	XLangMo	73.1	73.5	53.2	85.1	85.7	53.5
5	DANGNT@UIT. VNU-HCM	70.3	71.1	0	81.7	82.6	0
English-20K (closed)							
1	HLT@SUDA	72.7	73.6	31.2	85.2	86.4	32.1
2	baseline	67.2	68.2	23.7	82.2	83.5	24.3
3	CUNY-PekingU	66.9	67.9	27.9	82.3	83.6	29.0
4	GCN-Sem	62.6	63.7	0	80.0	81.4	0
English-20K (open)							
1	HLT@SUDA	76.7	77.7	39.2	88.0	89.2	41.4
2	CUNY-PekingU	73.9	74.6	45.7	86.4	87.4	48.1
3	TüPa	70.9	71.9	29.6	84.4	85.7	30.7
4	XLangMo	69.5	70.4	36.6	83.5	84.6	38.5
5	baseline	68.4	69.4	25.9	82.5	83.9	26.2
German-20K (closed)							
1	HLT@SUDA	83.2	83.8	59.2	92.0	92.6	60.9
2	CUNY-PekingU	79.7	80.2	59.3	90.2	90.9	59.9
3	baseline	73.1	73.6	47.8	85.9	86.7	48.2
4	GCN-Sem	71.0	72.0	0	85.1	86.2	0
German-20K (open)							
1	HLT@SUDA	84.9	85.4	64.1	92.8	93.4	64.7
2	CUNY-PekingU	84.1	84.5	66.0	92.3	93.0	66.6
3	baseline	79.1	79.6	59.9	90.3	91.0	60.5
4	TüPa	78.1	78.8	40.8	89.4	90.3	41.2
5	XLangMo	78.0	78.4	61.1	89.4	90.1	61.4
French-20K (open)							
1	CUNY-PekingU	79.6	80.0	64.5	89.1	89.6	71.1
2	HLT@SUDA	75.2	76.0	43.3	86.0	87.0	45.1
3	XLangMo	65.6	66.6	13.3	81.5	82.8	14.1
4	MaskParse@Deskiñ	65.4	66.6	24.3	80.9	82.5	25.8
5	baseline	48.7	49.6	2.4	74.0	75.3	3.2
6	TüPa	45.6	46.4	0	73.4	74.6	0

Table 4: Official F1-scores for each system in each track. Prim.: primary edges, Rem.: remote edges.

Other resources included additional unlabeled data (*TüPa* and *CUNY-PekingU*), a list of multi-word expressions (*MaskParse@Deskiñ*), and the Stanford parser in the case of *DANGNT@UIT.VNU-HCM*. Only *CUNY-PKU* used the 20K unlabeled parallel data in English and French.

A common trend for many of the systems was the use of cross-lingual projection or transfer (*MaskParse@Deskiñ*, *HLT@SUDA*, *TüPa*, *GCN-Sem*, *CUNY-PKU* and *XLangMo*). This was necessary for French, and was found helpful for German as well (*CUNY-PKU*).

## 7 Results

Table 4 shows the labeled and unlabeled F1 for primary and remote edges, for each system in each track. Overall F1 (All) is the F1 calculated over both primary and remote edges. Full results are available online.<sup>15</sup>

Figure 3 shows the fine-grained evaluation by

<sup>15</sup><http://bit.ly/semEval2019task1results>

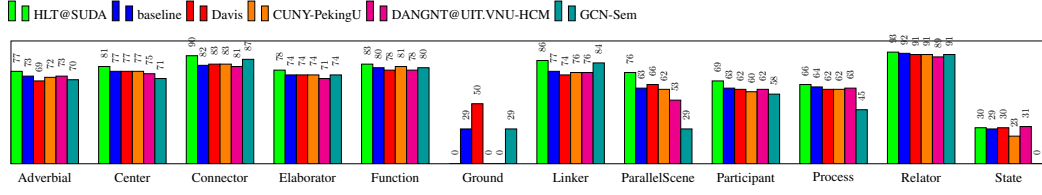
labeled F1 per UCCA category, for each system in each track. While Ground edges were uniformly difficult to parse due to their sparsity in the training data, Relators were the easiest for all systems, as they are both common and predictable. The Process/State distinction proved challenging, and most main relations were identified as the more common Process category. The winning system in most tracks (*HLT@SUDA*) performed better on almost all categories. Its largest advantage was on Parallel Scenes and Linkers, showing was especially successful at identifying Scene boundaries relative to the other systems, which requires a good understanding of syntax.

## 8 Discussion

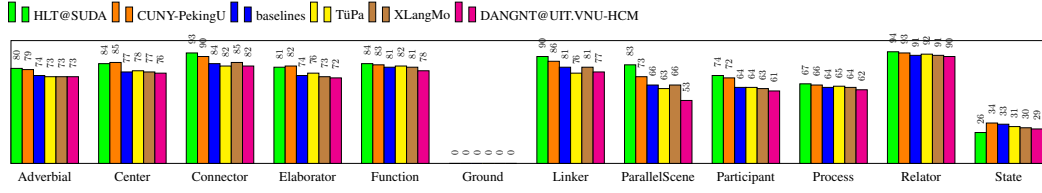
The *HLT@SUDA* system participated in all the tracks, obtaining the first place in the six English and German tracks and the second place in the French open track. The system is based on the conversion of UCCA graphs into constituency trees, marking remote and discontinuous edges for recovery. The classification-based recovery of the remote edges is performed simultaneously with the constituency parsing in a multi-task learning framework. This work, which further connects between semantic and syntactic parsing, proposes a recovery mechanism that can be applied to other grammatical formalisms, enabling the conversion of a given formalism to another one for parsing. The idea of this system is inspired by the pseudo non-projective dependency parsing approach proposed by *Nivre and Nilsson (2005)*.

The *MaskParse@Deskiñ* system only participated to the French open track, focusing on cross-lingual parsing. The system uses a semantic tagger, implemented with a bidirectional GRU and a masking mechanism to recursively extract the inner semantic structures in the graph. Multilingual word embeddings are also used. Using the English and German training data as well as the small French trial data for training, the parser ranked fourth in the French open track with a labeled F1 score of 65.4%, suggesting that this new model could be useful for low-resource languages.

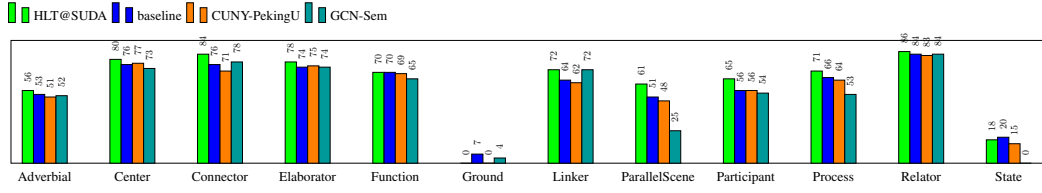
The *TüPa* system takes a transition-based approach, building on the TUPA transition system and oracle, but modifies its feature representations. Specifically, instead of representing the parser configuration using LSTMs over the partially parsed graph, stack and buffer, they use feed-



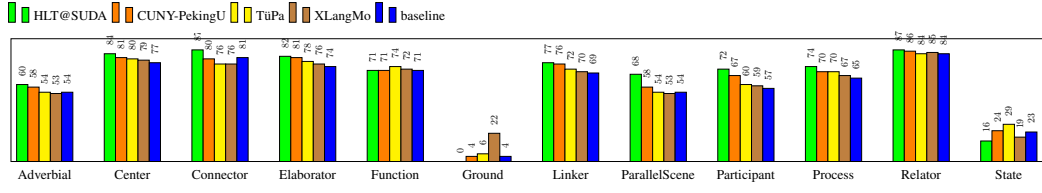
(a) English Wiki (closed)



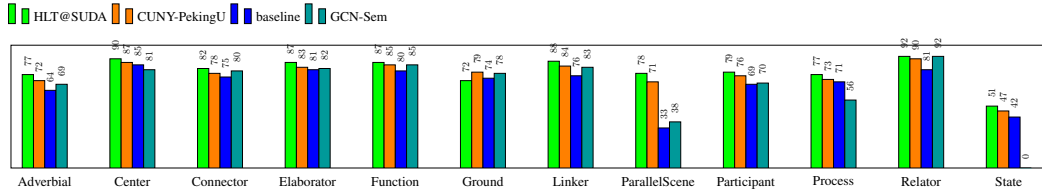
(b) English Wiki (open)



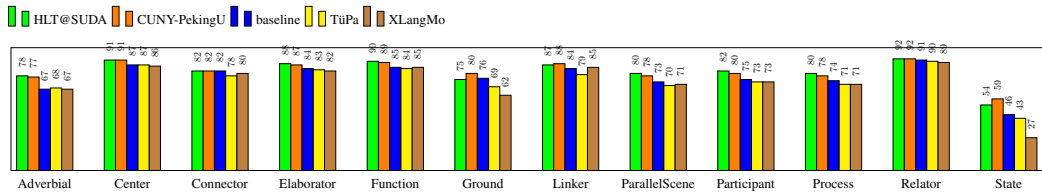
(c) English 20K (closed)



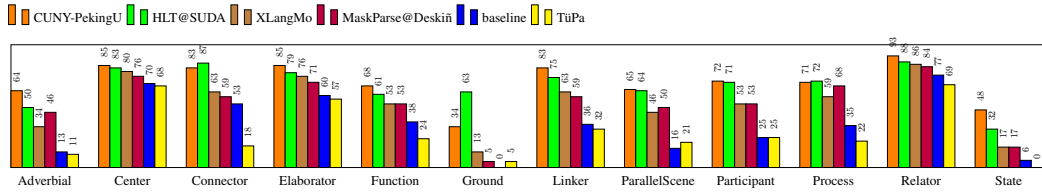
(d) English 20K (open)



(e) German 20K (closed)



(f) German 20K (open)



(g) French 20K (open)

Figure 3: Each system’s labeled F1 per UCCA category in each track.

forward networks with ELMo contextualized embeddings. The stack and buffer are represented by the top three items on them. For the partially parsed graph, they extract the rightmost and leftmost parents and children of the respective items, and represent them by the ELMo embedding of their form, the embedding of their dependency heads (for terminals, for non-terminals this is replaced with a learned embedding) and the embeddings of all terminal children. Results are generally on-par with the TUPA baseline, and surpass it from the out-of-domain English setting. This suggests that the TUPA architecture may be simplified, without compromising performance.

The *UC Davis* system participated only in the English closed track, where they achieved the second highest score, on par with TUPA. The proposed parser has an encoder-decoder architecture, where the encoder is a simple BiLSTM encoder for each span of words. The decoder iteratively and greedily traverses the sentence, and attempts to predict span boundaries. The basic algorithm yields an unlabeled contiguous phrase-based tree, but additional modules predict the labels of the spans, discontinuous units (by joining together spans from the contiguous tree under a new node), and remote edges. The work is inspired by [Kitaev and Klein \(2018\)](#), who used similar methods for constituency parsing.

The *GCN-SEM* system uses a BiLSTM encoder, and predicts bi-lexical semantic dependencies (in the SDP format) using word, token and syntactic dependency parses. The latter is incorporated into the network with a graph convolutional network (GCN). The team participated in the English and German closed tracks, and were not among the highest-ranking teams. However, scores on the UCCA test sets converted to the bi-lexical CoNLL-U format were rather high, implying that the lossy conversion could be much of the reason.

The *CUNY-PKU* system was based on an ensemble. The ensemble included variations of TUPA parser, namely the MLP and BiLSTM models ([Hershcovich et al., 2017](#)) and the BiLSTM model with an additional MLP. The system also proposes a way to aggregate the ensemble going through CKY parsing and accounting for remotes and discontinuous spans. The team participated in all tracks, including additional information in the open domain, notably synthetic data based on automatically translating annotated texts. Their sys-

tem ranks first in the French open track.

The *DANGNT@UIT.VNU-HCM* system participated only in the English Wiki open and closed tracks. The system is based on graph transformations from dependency trees into UCCA, using heuristics to create non-terminal nodes and map the dependency relations to UCCA categories. The manual rules were developed based on the training and development data. As the system converts trees to trees and does not add reentrancies, it does not produce remote edges. While the results are not among the highest-ranking in the task, the primary labeled F1 score of 71.1% in the English open track shows that a rule-based system on top of a leading dependency parser (the Stanford parser) can obtain reasonable results for this task.

## 9 Conclusion

The task has yielded substantial improvements to UCCA parsing in all settings. Given that the best reported results were achieved with different parsing and learning approaches than the baseline model TUPA (which has been the only available parser for UCCA), the task opens a variety of paths for future improvement. Cross-lingual transfer, which capitalizes on UCCA’s tendency to be preserved in translation, was employed by a number of systems and has proven remarkably effective. Indeed, the high scores obtained for French parsing in a low-resource setting suggest that high quality UCCA parsing can be straightforwardly extended to additional languages, with only a minimal amount of manual labor.

Moreover, given the conceptual similarity between the different semantic representations ([Abend and Rappoport, 2017](#)), it is likely the parsers developed for the shared task will directly contribute to the development of other semantic parsing technology. Such a contribution is facilitated by the available conversion scripts available between UCCA and other formats.

## Acknowledgments

We are deeply grateful to Dotan Dvir and the UCCA annotation team for their diligent work on the corpora used in this shared task.

This work was supported by the Israel Science Foundation (grant No. 929/17), and by the HUJI Cyber Security Research Center in conjunction with the Israel National Cyber Bureau in the Prime Minister’s Office.



## References

- Omri Abend and Ari Rappoport. 2013. [Universal Conceptual Cognitive Annotation \(UCCA\)](#). In *Proc. of ACL*, pages 228–238.
- Omri Abend and Ari Rappoport. 2017. [The state of the art in semantic representation](#). In *Proc. of ACL*, pages 77–89.
- Omri Abend, Shai Yershalimi, and Ari Rappoport. 2017. [UCCAApp: Web-application for syntactic and semantic phrase-based annotation](#). *Proc. of ACL System Demonstrations*, pages 109–114.
- Lasha Abzianidze, Johannes Bjerva, Kilian Evang, Hessel Haagsma, Rik van Noord, Pierre Ludmann, Duc-Duy Nguyen, and Johan Bos. 2017. [The parallel meaning bank: Towards a multilingual corpus of translations annotated with compositional meaning representations](#). *CoRR*, abs/1702.03964.
- Laura Banarescu, Claire Bonial, Shu Cai, Madalina Georgescu, Kira Griffitt, Ulf Hermjakob, Kevin Knight, Martha Palmer, and Nathan Schneider. 2013. [Abstract Meaning Representation for sem-banking](#). In *Proc. of the Linguistic Annotation Workshop*.
- Alexandra Birch, Omri Abend, Ondřej Bojar, and Barry Haddow. 2016. [HUME: Human UCCA-based evaluation of machine translation](#). In *Proc. of EMNLP*, pages 1264–1274.
- Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. [Enriching word vectors with subword information](#). *TACL*, 5:135–146.
- Leshem Choshen and Omri Abend. 2018. [Reference-less measure of faithfulness for grammatical error correction](#). In *Proc. of NAACL (Short papers)*, pages 124–129.
- Alexis Conneau, Guillaume Lample, Marc’Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. 2017. Word translation without parallel data. *arXiv preprint arXiv:1710.04087*.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. [BERT: Pre-training of deep bidirectional transformers for language understanding](#). In *Proc. of NAACL (to appear)*.
- Robert M. W. Dixon. 2010a. *Basic Linguistic Theory: Grammatical Topics*, volume 2. Oxford University Press.
- Robert M. W. Dixon. 2010b. *Basic Linguistic Theory: Methodology*, volume 1. Oxford University Press.
- Robert M. W. Dixon. 2012. *Basic Linguistic Theory: Further Grammatical Topics*, volume 3. Oxford University Press.
- Shibhansh Dohare and Harish Karnick. 2017. [Text summarization using abstract meaning representation](#). *CoRR*, abs/1706.01678.
- Daniel Hershcovich, Omri Abend, and Ari Rappoport. 2017. [A transition-based directed acyclic graph parser for UCCA](#). In *Proc. of ACL*, pages 1127–1138.
- Daniel Hershcovich, Omri Abend, and Ari Rappoport. 2018. [Multitask parsing across semantic representations](#). In *Proc. of ACL*, pages 373–385.
- Fuad Issa, Marco Damonte, Shay B. Cohen, Xiaohui Yan, and Yi Chang. 2018. [Abstract meaning representation for paraphrase detection](#). In *Proc. of NAACL*, pages 442–452.
- Wei Jiang, Yu Zhang, Zhenghua Li, and Min Zhang. 2019. HLT@SUDA at semEval-2019 Task 1: UCCA graph parsing as constituent tree parsing. In *Proc. of SemEval-2019*.
- Nikita Kitaev and Dan Klein. 2018. [Consistency parser with self-attentive decoder](#). In *ACL*, pages 2676–2686.
- Fei Liu, Jeffrey Flanigan, Sam Thomson, Norman Sadeh, and Noah A. Smith. 2015. [Toward abstractive summarization using semantic representations](#). In *Proc. of NAACL-HLT*, pages 1077–1086, Denver, Colorado.
- Weimin Lyu, Sheng Huang, Abdul Rafae Khan, Shengqiang Zhang, Weiwei Sun, and Jia Xu. 2019. CUNY-PKU parser at SemEval-2019 Task 1: Cross-lingual semantic parsing with UCCA. In *Proc. of SemEval-2019*.
- Gabriel Marzinotto, Johannes Heinecke, and Géraldine Damnati. 2019. MaskParse@Deskiñ at semEval-2019 Task 1: Cross-lingual UCCA semantic parsing with recursive masked sequence tagging. In *Proc. of SemEval-2019*.
- Dang Tuan Nguyen and Trung Tran. 2019. DAN-GNT@UIT.VNU-HCM at SemEval-2019 Task 1: Graph transformation system from Stanford basic dependencies to Universal Conceptual Cognitive Annotation (UCCA). In *Proc. of SemEval-2019*.
- Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Yoav Goldberg, Jan Hajic, Christopher D. Manning, Ryan McDonald, Slav Petrov, Sampo Pyysalo, Natalia Silveira, Reut Tsarfaty, and Daniel Zeman. 2016. [Universal dependencies v1: A multilingual treebank collection](#). In *Proc. of LREC*.
- Joakim Nivre and Jens Nilsson. 2005. [Pseudo-projective dependency parsing](#). In *Proc. of ACL*, pages 99–106.
- Stephan Oepen, Marco Kuhlmann, Yusuke Miyao, Daniel Zeman, Silvie Cinkova, Dan Flickinger, Jan Hajic, Angelina Ivanova, and Zdenka Uresova. 2016. [Towards comparability of linguistic graph banks for semantic parsing](#). In *Proc. of LREC*.

- Stephan Oepen, Marco Kuhlmann, Yusuke Miyao, Daniel Zeman, Silvie Cinková, Dan Flickinger, Jan Hajič, and Zdeňka Urešová. 2015. [SemEval 2015 task 18: Broad-coverage semantic dependency parsing](#). In *Proc. of SemEval*, pages 915–926.
- Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. [Deep contextualized word representations](#). 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 2227–2237. Association for Computational Linguistics.
- Tobias Pütz and Kevin Glocker. 2019. Tüpa at SemEval-2019 Task 1: (Almost) feature-free semantic parsing. In *Proc. of SemEval-2019*.
- Mitchell Stern, Jacob Andreas, and Dan Klein. 2017. [A minimal span-based neural constituency parser](#). In *ACL*, pages 818–827.
- Elior Sulem, Omri Abend, and Ari Rappoport. 2015. [Conceptual annotations preserve structure across translations: A French-English case study](#). In *Proc. of S2MT*, pages 11–22.
- Elior Sulem, Omri Abend, and Ari Rappoport. 2018a. [Semantic structural annotation for text simplification](#). In *NAACL 2018*, pages 685–696.
- Elior Sulem, Omri Abend, and Ari Rappoport. 2018b. [Simple and effective text simplification using semantic and neural methods](#). In *Proc. of ACL*, pages 162–173.
- Shiva Taslimipour, Omid Rohanian, and Sara Može. 2019. GCN-Sem at SemEval-2019 Task 1: Semantic parsing using graph convolutional and recurrent neural networks. In *Proc. of SemEval-2019*.
- Aaron Steven White, Drew Reisinger, Keisuke Sakaguchi, Tim Vieira, Sheng Zhang, Rachel Rudinger, Kyle Rawlins, and Benjamin Van Durme. 2016. [Universal compositional semantics on universal dependencies](#). In *Proc. of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1713–1723.
- Dian Yu and Kenji Sagae. 2019. UC Davis at SemEval-2019 Task 1: DAG semantic parsing with attention-based decoder. In *Proc. of SemEval-2019*.
- Sheng Zhang, Kevin Duh, and Benjamin Van Durme. 2017. [Selective decoding for cross-lingual open information extraction](#). In *Proc. of IJCNLP*, pages 832–842.
- Sheng Zhang, Xutai Ma, Rachel Rudinger, Kevin Duh, and Benjamin Van Durme. 2018. [Cross-lingual compositional semantic parsing](#). In *Proc. of EMNLP*, pages 1664–1675.

# HLT@SUDA at SemEval-2019 Task 1: UCCA Graph Parsing as Constituent Tree Parsing

Wei Jiang, Zhenghua Li\*, Yu Zhang, Min Zhang

School of Computer Science and Technology, Soochow University, China

{wjiang0501, yzhang25}@stu.suda.edu.cn, {zhli13, minzhang}@suda.edu.cn

## Abstract

This paper describes a simple UCCA semantic graph parsing approach. The key idea is to convert a UCCA semantic graph into a constituent tree, in which extra labels are deliberately designed to mark remote edges and discontinuous nodes for future recovery. In this way, we can make use of existing syntactic parsing techniques. Based on the data statistics, we recover discontinuous nodes directly according to the output labels of the constituent parser and use a biaffine classification model to recover the more complex remote edges. The classification model and the constituent parser are simultaneously trained under the multi-task learning framework. We use the multilingual BERT as extra features in the open tracks. Our system ranks the first place in the six English/German closed/open tracks among seven participating systems. For the seventh cross-lingual track, where there is little training data for French, we propose a language embedding approach to utilize English and German training data, and our result ranks the second place.

## 1 Introduction

Universal Conceptual Cognitive Annotation (UCCA) is a multi-layer linguistic framework for semantic annotation proposed by [Abend and Rappoport \(2013\)](#). Figure 1 shows an example sentence and its UCCA graph. Words are represented as terminal nodes. Circles denote non-terminal nodes, and the semantic relation

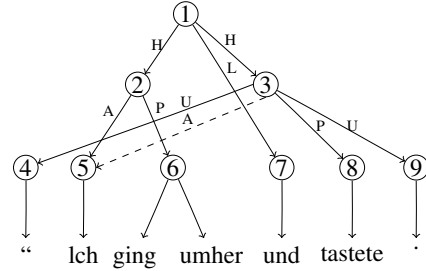


Figure 1: A UCCA graph example from the German data. The English translation is “I went around and groped . We assign a number to each non-terminal node to facilitate illustration.

between two non-terminal nodes is represented by the label on the edge. One node may have multiple parents, among which one is annotated as the primary parent, marked by solid line edges, and others as remote parents, marked by dashed line edges. The primary edges form a tree structure, whereas the remote edges enable reentrancy, forming directed acyclic graphs (DAGs).<sup>1</sup> The second feature of UCCA is the existence of nodes with discontinuous leaves, known as discontinuity. For example, node 3 in Figure 1 is discontinuous because some terminal nodes it spans are not its descendants.

[Hershcovich et al. \(2017\)](#) first propose a transition-based UCCA Parser, which is used as the baseline in the closed tracks of this shared task. Based on the recent progress on transition-based parsing techniques, they propose a novel set of transition actions to handle both discontinuous and remote nodes and design useful features based on bidirectional LSTMs. [Hershcovich et al. \(2018\)](#) then extend their previous approach and propose to utilize the annotated data with other

<sup>1</sup>The full UCCA scheme also has implicit and linkage relations, which are overlooked in the community so far.

\*Corresponding author, [hlt.suda.edu.cn/zhenghua](mailto:hlt.suda.edu.cn/zhenghua)