



**HAL**  
open science

# MaskParse@Deskin at SemEval-2019 Task 1: Cross-lingual UCCA Semantic Parsing using Recursive Masked Sequence Tagging

Gabriel Marzinotto, Johannes Heinecke, Geraldine Damnati

► **To cite this version:**

Gabriel Marzinotto, Johannes Heinecke, Geraldine Damnati. MaskParse@Deskin at SemEval-2019 Task 1: Cross-lingual UCCA Semantic Parsing using Recursive Masked Sequence Tagging. Proceedings of the Thirteenth International Workshop on Semantic Evaluation, Jun 2019, Minneapolis, United States. hal-02298429

**HAL Id: hal-02298429**

**<https://hal.science/hal-02298429v1>**

Submitted on 4 Oct 2019

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# MaskParse@Deskin at SemEval-2019 Task 1: Cross-lingual UCCA Semantic Parsing using Recursive Masked Sequence Tagging

Gabriel Marzinotto<sup>1,2</sup> Johannes Heinecke<sup>1</sup> Géraldine Damnati<sup>1</sup>

(1) Orange Labs / Lannion France

(2) Aix Marseille Univ, CNRS, LIS / Marseille France

{gabriel.marzinotto, johannes.heinecke, geraldine.damnati}@orange.com

## Abstract

This paper describes our recursive system for SemEval-2019 *Task 1: Cross-lingual Semantic Parsing with UCCA*. Each recursive step consists of two parts. We first perform semantic parsing using a sequence tagger to estimate the probabilities of the UCCA categories in the sentence. Then, we apply a decoding policy which interprets these probabilities and builds the graph nodes. Parsing is done recursively, we perform a first inference on the sentence to extract the main scenes and links and then we recursively apply our model on the sentence using a masking feature that reflects the decisions made in previous steps. Process continues until the terminal nodes are reached. We choose a standard neural tagger and we focused on our recursive parsing strategy and on the cross lingual transfer problem to develop a robust model for the French language, using only few training samples.

## 1 Introduction

Semantic representation is an essential part of NLP. For this reason, several semantic representation paradigms have been proposed. Among them we find PropBank (Palmer et al., 2005) and FrameNet Semantics (Baker et al., 1998), Abstract Meaning Representation (AMR) (Banarescu et al., 2013), Universal Compositional Semantics (White et al., 2016) and Universal Conceptual Cognitive Annotation (UCCA) (Abend and Rapoport, 2013). These constantly improving representations, along with the advances in semantic parsing, have proven to be beneficial in many NLU tasks such as Question Answering (Shen and Lapata, 2007), text summarization (Genest and Lapalme, 2011), dialog systems (Tur et al., 2005), information extraction (Bastianelli et al., 2013) and machine translation (Liu and Gildea, 2010).

UCCA is a cross-lingual semantic representation scheme, has demonstrated applicability in En-

glish, French and German (with pilot annotation projects on Czech, Russian and Hebrew). Despite the newness of UCCA, it has proven useful for defining semantic evaluation measures in text-to-text generation and machine translation (Birch et al., 2016). UCCA represents the semantics of a sentence using directed acyclic graphs (DAGs), where terminal nodes correspond to text tokens, and non-terminal nodes to higher level semantic units. Edges are labelled, indicating the role of a child in the relation to its parent. UCCA parsing is a recent task and since UCCA has several unique properties, adapting syntactic parsers or parsers from other semantic representations is not straight-forward. Current state of the art parser TUPA (Hershcovich et al., 2017) uses a transition based parsing to build UCCA representations.

Building over previous work on FrameNet Semantic Parsing (Marzinotto et al., 2018a,b) we chose to perform UCCA parsing using sequence tagging methods along with a graph decoding policy. To do this we propose a recursive strategy in which we perform a first inference on the sentence to extract the main scenes and links and then we recursively apply our model on the sentence with a masking mechanism at the input in order to feed information about the previous parsing decisions.

## 2 Model

Our system consists of a sequence tagger that is first applied on the sentence to extract the main scenes and links and then it is recursively applied on the extracted element to build the semantic graph. At each step of the recursion we use a masking mechanism to feed information about the previous stages into the model. In order to convert the sequence labels into nodes of the UCCA graph we also apply a decoding policy at each stage.

Our tagger is implemented using deep bi-

directional GRU (*biGRU*). This simple architecture is frequently used in semantic parsers across different representation paradigms. Besides its flexibility, it is a powerful model, with close to state of the art performance on both PropBank (He et al., 2017) and FrameNet semantic parsing (Yang and Mitchell, 2017; Marzinotto et al., 2018b).

More precisely, the model consists of a 4 layer bi-directional Gated Recurrent Unit (GRU) with highway connections (Srivastava et al., 2015). Our model uses has a rich set of features including syntactic, morphological, lexical and surface features, which have shown to be useful in language abstracted representations. The list is given below:

- Word embeddings of 300 dimensions<sup>1</sup>.
- Syntactic dependencies of each token<sup>2</sup>.
- Part-of-speech and morphological features such as gender, number, voice and degree<sup>2</sup>.
- Capitalization and word length encoding.
- Prefixes and Suffixes of 2 and 3 characters.
- A language indicator feature.
- Boolean indicator of idioms and multi word expression. Detailed in section 3.2.
- Masking mechanism, which indicates, for a given node in the graph, the tokens within the span as well as the arc label between the node and its parent. See details in section 2.1.

Except for words where we use pre-trained embeddings, we use randomly initialized embedding layers for categorical features.

## 2.1 Masking Mechanism

We introduce an original masking mechanism in order to feed information about the previous parsing stages into the model. During parsing, we first do an initial inference step to extract the main scenes and links. Then, for each resulting node, we build a new input which is essentially the same, but with a categorical sequence masking feature. For the input tokens in the node span, this feature is equal to the label of the arc between the node and its parent. Outside of the node span, this mask is equal to  $\emptyset$ . A diagram of this masking process is shown in figure 1. The process continues and the model recursively extracts the inner semantic structures (the node’s children) in the graph, until the terminal nodes are reached.

<sup>1</sup>Obtained from <https://github.com/facebookresearch/MUSE>

<sup>2</sup> Using Universal Dependencies categories.

To train such a model, we build a new training corpus in which the sentences are repeated several times. More precisely, a sentence appears  $N$  times ( $N$  being the number of non terminal nodes in the UCCA graph) each one with different mask.

## 2.2 Multi-Task UCCA Objective

Along with the UCCA-XML graph representations, a simplified tree representation in CoNLL format was also provided. Our model combines both representations using a multitask objective with two tasks. *TASK1* consists in, for a given node and its corresponding mask, predicting the children and their arc labels. *TASK1* encodes the children spans using a BIO scheme. The *TASK2* consists in predicting the CoNLL simplified UCCA structure of the sentence. More precisely, *TASK2* is a sequence tagger that predicts the UCCA-CoNLL function of each token. *TASK2* is not used for inference purposes. It is only a support that help the model to extract relevant features, allowing it to model the whole sentence even when parsing small pre-terminal nodes.

## 2.3 Label Encoding

We have previously stated that *TASK1* uses BIO encoded labels to model the structure of the children of each node in the semantic graph. In some rare cases, the BIO encoding scheme is not sufficient to model the interaction between parallel scenes. For example, when we have two parallel scenes and one of them appears as a clause inside the other. In such cases, BIO encoding does not allow to determine whether the last part of the sentence belongs to the first scene or to the clause. Despite this issue, prior experiments testing more complete label encoding schemes (BIEO, BIEOW) showed that BIO outperforms the other schemes on the validation sets.

## 2.4 Graph Decoding

During the decoding phase, we convert the BIO labels into graph nodes. To do so, we add a few constraints to ensure the outputs are feasible UCCA graphs that respect the sentence’s structure:

- We merge parallel scenes (H) that do not have either a verb or an action noun to the nearest previous scene having one.
- Within each parallel scene, we force the existence of one and only one State (S) or Process (P) by selecting the token with the highest probability of State or Process.

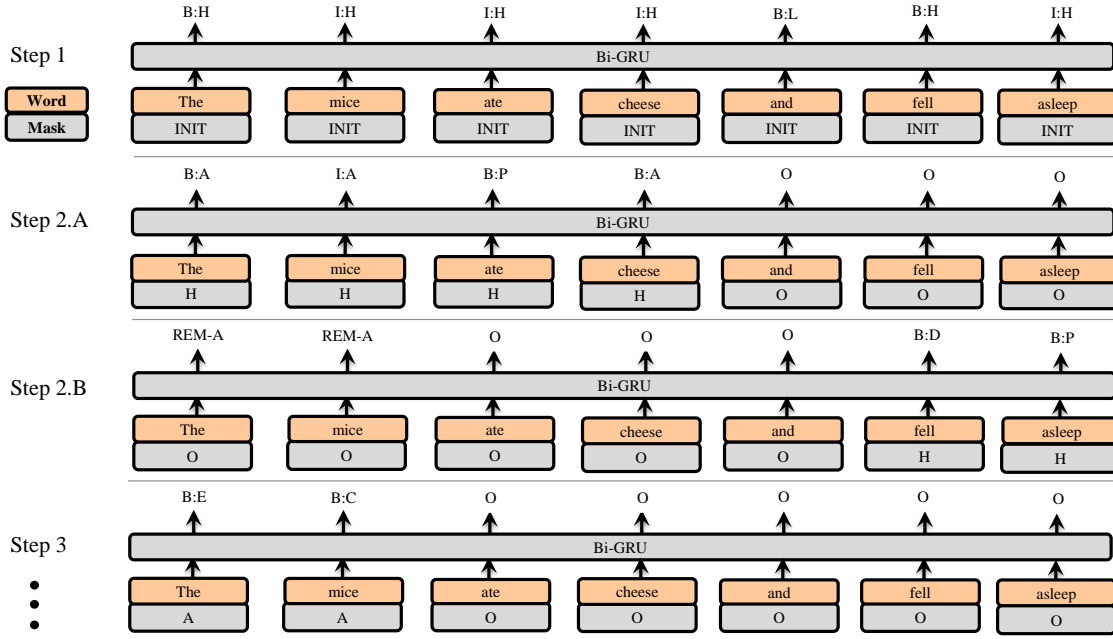


Figure 1: Masking mechanism through recursive calls. Step 1 parses the sentence to extract parallel scenes (H) and links (L). Then Steps 2.A 2.B use a different mask to parse these scenes and extract arguments (A) and processes (P) which will be recursively parsed until terminal nodes are reached.

- For scenes (H) and arguments (A) we do not allow to split multi word expressions (MWE) and chunks into different graph nodes. If the boundary between two segments lies inside a chunk or MWE segments are merged.

## 2.5 Remote Edges

Our approach easily handles remote edges. We consider remote arguments as those detected outside the parent’s node span (see REM in Fig.1). Our earlier models showed low recall on remotes. To fix this, we introduced a detection threshold on the output probabilities. This increased the recall at the cost of some precision. The optimal detection threshold was optimized on the validation set.

## 3 Data

### 3.1 UCCA Task Data

In table 1 we show the number of annotations for each language and domain. Our objective is to build a model that generalizes to the French language despite of having only 15 training samples.

When we analyse data in details we observe that there are several tokenization errors. Specially in the French corpus. These errors propagate to the POS tagging and dependency parsing as well. For this reason, we retokenized and parsed all the corpus using a enriched version of UDpipe that we trained ourselves (Straka and Straková, 2017) us-

Corpus	Train	Dev	Test
English Wiki	4113	514	515
English 20K	-	-	492
German 20K	5211	651	652
French 20K	15	238	239

Table 1: number of UCCA annotated sentences in the partitions for each language and domain

ing the Treebanks from Universal Dependencies<sup>3</sup>. For French we enriched the Treebank with XPOS from our lexicon. Finally, since tokenization is pre-established in the UCCA corpus we projected the improved POS and dependency parsing into the original tokenization of the task.

### 3.2 Supplementary lexicon

We observed that a major difficulty in UCCA parsing is analyzing idioms and phrases. The unawareness about these expressions, which are mostly used as links between scenes, mislead the model during the early stages of the inference and errors get propagated through the graph. To boost the performance of our model when detecting links and parallel scenes we developed an internal list with about 500 expression for each language. These lists include prepositional, adverbial and conjunctive expressions and are used to compute Boolean features indicating the words in the sentence which are part of an expression.

<sup>3</sup><https://universaldependencies.org/>

Open Tracks	Ours Labeled			Ours Unlabeled			TUPA Labeled			TUPA Unlabeled		
	Avg F1	Prim F1	Rem F1	Avg F1	Prim F1	Rem F1	Avg F1	Prim F1	Rem F1	Avg F1	Prim F1	Rem F1
Dev English Wiki	<b>70.8</b>	71.3	58.7	82.5	83.8	37.5	<b>74.8</b>	75.3	51.4	86.3	87.0	51.4
Dev German 20K	<b>74.7</b>	75.4	40.5	87.4	88.6	40.9	<b>79.2</b>	79.7	58.7	90.7	91.5	59.0
Dev French 20K	<b>63.6</b>	64.4	19.0	78.9	79.6	20.5	<b>51.4</b>	52.3	1.6	74.9	76.2	1.6
Test English Wiki	<b>68.9</b>	69.4	42.5	82.3	83.1	42.8	<b>73.5</b>	73.9	53.5	85.1	85.7	54.3
Test English 20K	<b>66.6</b>	67.7	24.6	82.0	83.4	24.9	<b>68.4</b>	69.4	25.9	82.5	83.9	26.2
Test German 20K	<b>74.2</b>	74.8	47.3	87.1	88.0	47.6	<b>79.1</b>	79.6	59.9	90.3	91.0	60.5
Test French 20K	<b>65.4</b>	66.6	24.3	80.9	82.5	25.8	<b>48.7</b>	49.6	2.4	74.0	75.3	3.2

Table 2: Our model vs TUPA baseline performance for each open track

Tracks	D	C	N	E	F	G	L	H	A	P	U	R	S
EN Wiki	64.3	71.4	68.5	69.6	76.7	0.0	71.4	61.3	60.0	64.0	99.7	89.2	25.1
EN 20K	47.2	75.2	62.5	72.3	71.5	0.2	57.9	49.5	55.7	69.8	99.7	83.2	19.5
DE 20K	69.4	83.8	57.7	80.5	83.8	59.2	68.4	62.2	67.5	68.9	97.1	86.9	25.9
FR 20K	46.1	76.0	58.9	71.2	53.3	4.8	59.4	50.4	52.8	67.6	99.6	83.5	16.9

Table 3: Our model’s Fine-grained F1 by label on Test Open Tracks

### 3.3 Multilingual Training

This model uses multilingual word embeddings trained using fastText (Bojanowski et al., 2017) and aligned using MUSE (Conneau et al., 2017). This is done in order to ease cross-lingual training. In prior experiments we introduced an adversarial objective similar to (Kim et al., 2017; Marzinotto et al., 2019) to build a language independent representation. However, the language imbalance on the training data did not allow us to take advantage from this technique. Hence, we simply merged training data from different languages.

## 4 Experiments

We focus on obtaining the model that best generalizes on the French language. We trained our model for 50 epochs and we selected the best one on the validation set. In our experiments we did not use any product of experts or bagging technique and we did not run any hyper parameter optimization.

We trained several models building different training corpora composed of different language combinations. We obtained our best model using the training data for all the languages. This model FR+DE+EN achieved 63.6% avg. F1 on the French validation set. Compared to 63.1% for FR+DE, 62.9% for FR+EN and 50.8% for only FR.

### 4.1 Main Results

In Table 2 we provide the performance of our model for all the open tracks and we provide the results for TUPA baseline in order to establish a

comparison. Our model finishes 4th in the French Open Track with an average F1 score of 65.4%, very close to the 3rd place which had a 65.6% F1. For languages with larger training corpus, our model did not outperform the monolingual TUPA.

### 4.2 Error Analysis

In Table 3 we give the performance by arc type. We observe that the main performance bottleneck is in the parallel scene segmentation (H). Due to our recursive parsing approach, this kind of error is particularly harmful to the model performance, because scene segmentation errors at the early steps of the parsing may induce errors in the rest of the graph. To assert this, we used the validation set to compare the performance of the mono scene sentences (with no potential scene segmentation problems) with the multi scene sentences. For the French track we obtained 67.2% avg. F1 on the 114 mono scene sentences compared to 61.9% avg. F1 on the 124 multi scene sentences.

## 5 Conclusions

We described an original approach to recursively build the UCCA semantic graph using a sequence tagger along with a masking mechanism and a decoding policy. Even though this approach did not yield the best results in the UCCA task, we believe that our original recursive, mask-based parsing can be helpful in low resource languages. Moreover, we believe that this model could be further improved by introducing a global criterion and by performing further hyper parameter tuning.

## References

- Omri Abend and Ari Rappoport. 2013. Universal conceptual cognitive annotation (UCCA). In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 228–238.
- Collin F. Baker, Charles J. Fillmore, and John B. Lowe. 1998. The Berkeley FrameNet project. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics-Volume 1*, pages 86–90. Association for Computational Linguistics.
- Laura Banarescu, Claire Bonial, Shu Cai, Madalina Georgescu, Kira Griffitt, Ulf Hermjakob, Kevin Knight, Philipp Koehn, Martha Palmer, and Nathan Schneider. 2013. Abstract meaning representation for sembanking. In *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*, pages 178–186.
- Emanuele Bastianelli, Giuseppe Castellucci, Danilo Croce, and Roberto Basili. 2013. [Textual inference and meaning representation in human robot interaction](#). In *Proceedings of the Joint Symposium on Semantic Processing. Textual Inference and Structures in Corpora*, pages 65–69.
- Alexandra Birch, Omri Abend, Ondřej Bojar, and Barry Haddow. 2016. HUME: Human UCCA-based evaluation of machine translation. *arXiv preprint arXiv:1607.00030*.
- Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.
- Alexis Conneau, Guillaume Lample, Marc’Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. 2017. Word translation without parallel data. *arXiv preprint arXiv:1710.04087*.
- Pierre-Etienne Genest and Guy Lapalme. 2011. Framework for abstractive summarization using text-to-text generation. In *Proceedings of the Workshop on Monolingual Text-To-Text Generation*, Portland, Oregon, USA. Association for Computational Linguistics, Association for Computational Linguistics.
- Luheng He, Kenton Lee, Mike Lewis, and Luke Zettlemoyer. 2017. Deep semantic role labeling: What works and what’s next. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*.
- Daniel Hershcovich, Omri Abend, and Ari Rappoport. 2017. A transition-based directed acyclic graph parser for UCCA. *arXiv preprint arXiv:1704.00552*.
- Joo-Kyung Kim, Young-Bum Kim, Ruhi Sarikaya, and Eric Fosler-Lussier. 2017. [Cross-lingual transfer learning for pos tagging without cross-lingual resources](#). In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2832–2838. Association for Computational Linguistics.
- Ding Liu and Daniel Gildea. 2010. [Semantic role features for machine translation](#). In *Proceedings of the 23rd International Conference on Computational Linguistics, COLING ’10*, pages 716–724, Stroudsburg, PA, USA. Association for Computational Linguistics.
- Gabriel Marzinotto, Jeremy Auguste, Frédéric Béchet, Géraldine Damnati, and Alexis Nasr. 2018a. [Semantic Frame Parsing for Information Extraction : the CALOR corpus](#). In *LREC 2018*, Miyazaki, Japan.
- Gabriel Marzinotto, Frédéric Béchet, Géraldine Damnati, and Alexis Nasr. 2018b. [Sources of Complexity in Semantic Frame Parsing for Information Extraction](#). In *International FrameNet Workshop 2018*, Miyazaki, Japan.
- Gabriel Marzinotto, Géraldine Damnati, Frédéric Béchet, and Benoît Favre. 2019. Robust semantic parsing with adversarial learning for domain generalization. In *Proc. of NAACL*.
- Martha Palmer, Daniel Gildea, and Paul Kingsbury. 2005. The proposition bank: An annotated corpus of semantic roles. *Computational linguistics*, 31(1):71–106.
- Dan Shen and Mirella Lapata. 2007. Using semantic roles to improve question answering. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 12–21, Prague. Association for Computational Linguistics, Association for Computational Linguistics.
- Rupesh Kumar Srivastava, Klaus Greff, and Jürgen Schmidhuber. 2015. [Training very deep networks](#). In *NIPS 2015*, pages 2377–2385, Montreal, Quebec, Canada.
- Milan Straka and Jana Straková. 2017. [Tokenizing, pos tagging, lemmatizing and parsing ud 2.0 with udpipes](#). In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 88–99, Vancouver, Canada. Association for Computational Linguistics.
- Gokhan Tur, Dilek Hakkani-Tür, and Ananlada Chotimongkol. 2005. [Semi-supervised learning for spoken language understanding semantic role labeling](#). In *IEEE Workshop on Automatic Speech Recognition and Understanding*, pages 232 – 237.
- Aaron Steven White, Drew Reisinger, Keisuke Sakaguchi, Tim Vieira, Sheng Zhang, Rachel Rudinger, Kyle Rawlins, and Benjamin Van Durme. 2016. Universal decompositional semantics on universal

dependencies. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1713–1723.

Bishan Yang and Tom Mitchell. 2017. [A joint sequential and relational model for frame-semantic parsing](#). In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1247–1256. Association for Computational Linguistics.