Broad-Coverage Transition-Based UCCA Parsing

Daniel Hershcovich[1,2] & Omri Abend[2] & Ari Rappoport[2]

[1]The Edmond and Lily Safra Center for Brain Sciences
[2]School of Computer Science and Engineering
The Hebrew University of Jerusalem {danielh,oabend,arir}@cs.huji.ac.il

UCCA supports **reentrancy**, **discontinuity** and **non-terminal nodes**, which are essential for representing natural language semantics.
We present the **first parser for UCCA** and the first to support these properties.

Introduction

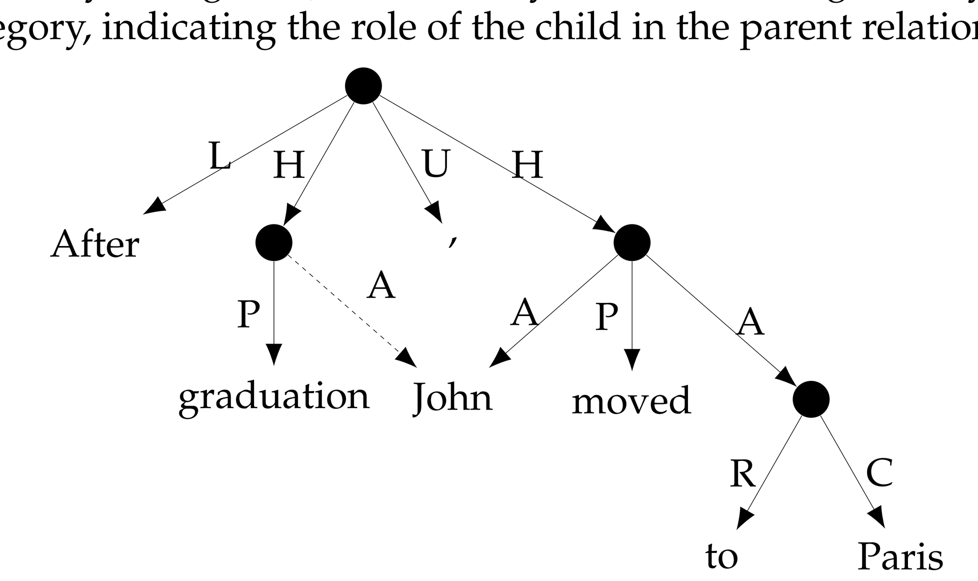
Universal Conceptual Cognitive Annotation [1] is a cross-linguistically applicable semantic representation scheme.

- Builds on typological and Cognitive Linguistics literature.
- Demonstrated applicability to English, French, German & Czech.
- Support for rapid annotation.
- Semantic stability in translation [8].
- Proven useful for machine translation evaluation [3].
- Applicability has been so far limited by the absence of a parser.

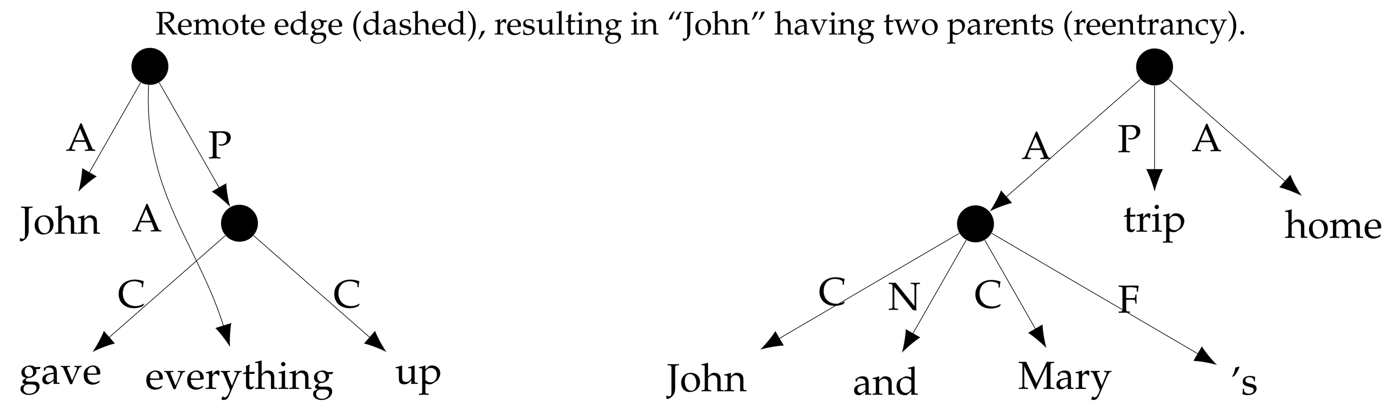
UCCA graphs are labeled, directed acyclic graphs (DAGs). Leaves correspond to the tokens of the text. A node corresponds to one or more terminals (not necessarily contiguous) semantically viewed as a single entity. Edges bear a category, indicating the role of the child in the parent relation.

P process
S state
A participant
L linker
H linked scene
C center
E elaborator
D adverbial
R relator
N connector
U punctuation
F function unit
G ground

Edge labels.



Remote edge (dashed), resulting in “John” having two parents (reentrancy).



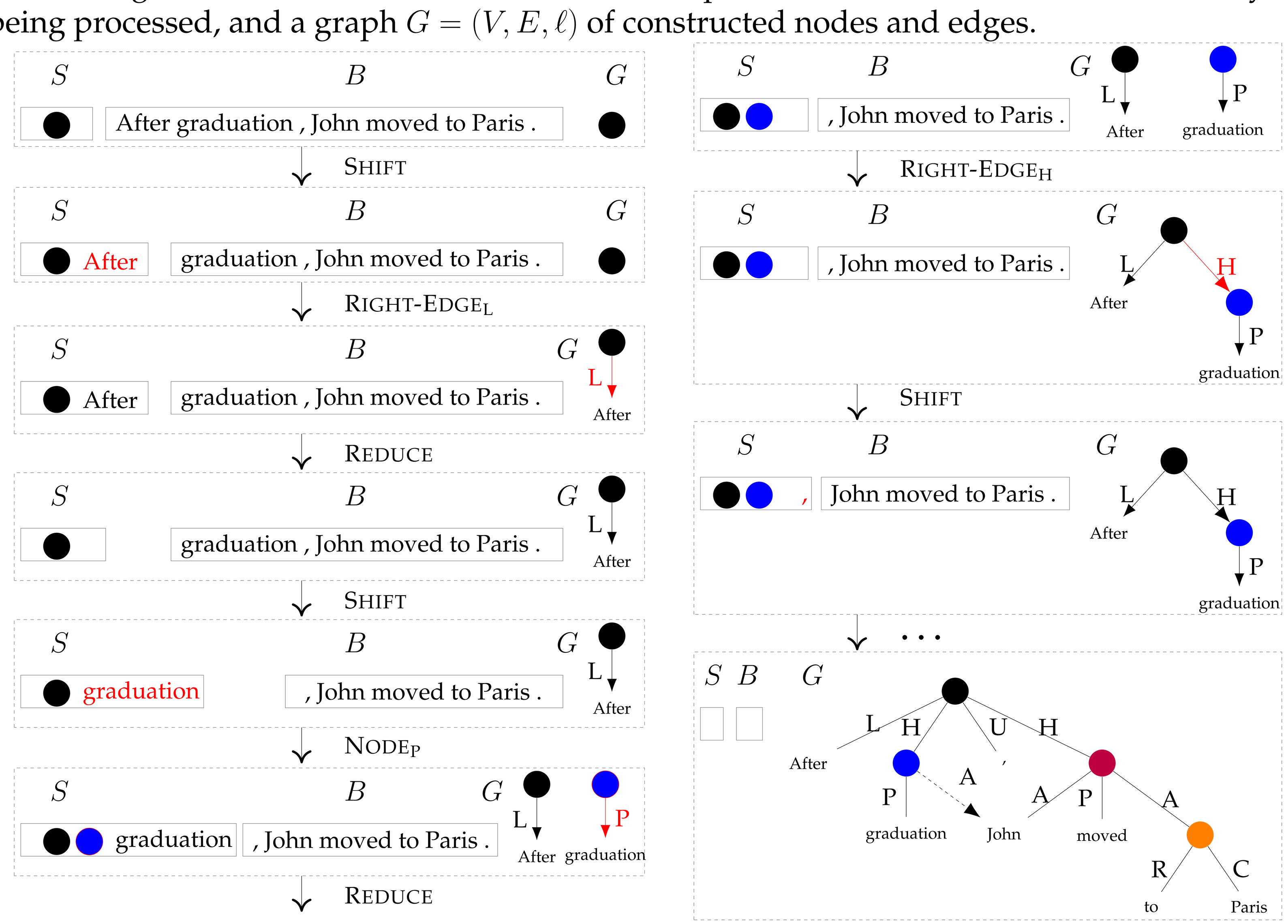
Discontinuous unit (“gave ... up”). Coordination construction (“John and Mary”).

UCCA structures demonstrating **reentrancy**, **discontinuity** and **non-terminal nodes**.

Transition-based UCCA Parsing

TUPA, our transition-based parser, supports the structural properties of UCCA.

Transition-based parsers work by applying a *transition* at each step to the parser state, defined using a buffer B of tokens and nodes to be processed, a stack S of nodes currently being processed, and a graph $G = (V, E, \ell)$ of constructed nodes and edges.



Example for intermediate states during transition-based UCCA parsing.

A classifier selects the next transition based on the current state’s features. It is trained by an oracle based on gold-standard annotations. We experiment with three classifiers:

TUPA$_{\text{Sparse}}$	Perceptron with features: words, POS, dependency & edge label combinations.
TUPA$_{\text{MLP}}$	2-layer NN, learned embedding features + external word embeddings.
TUPA$_{\text{BiLSTM}}$	2-layer bidirectional RNN to encode features, 2-layer NN for classification.

For all classifiers, inference is performed greedily, i.e., without beam search.
Parser code available at https://github.com/danielhers/tupa
All corpora available at http://www.cs.huji.ac.il/~oabend/ucca.html
Parser written in Python using DyNet: https://github.com/clab/dynet

Experimental Setup

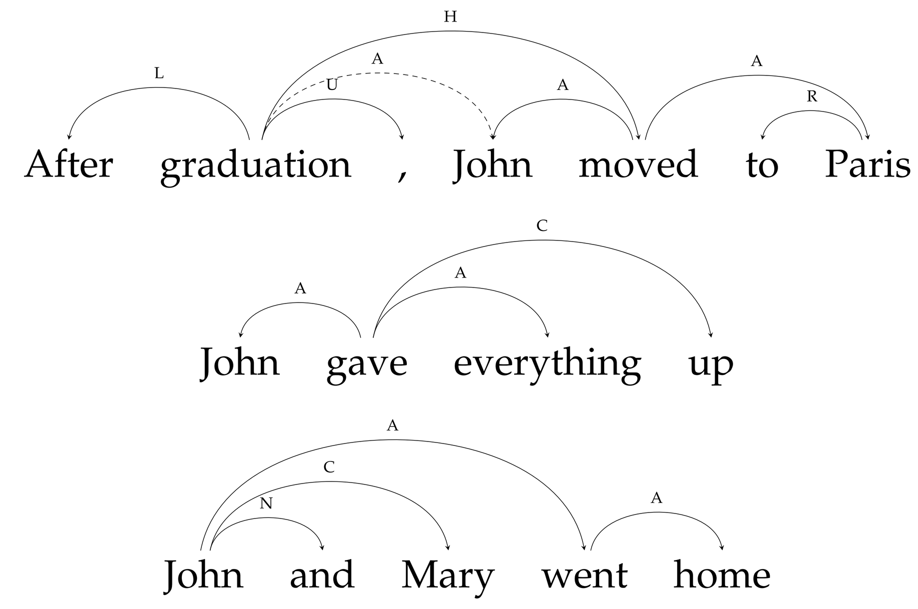
Corpora. English Wikipedia (in-domain), English part of *Twenty Thousand Leagues Under the Sea* English-French parallel corpus (out-of-domain).

Evaluation. Labeled precision, recall and F-score on graph edges, represented by their terminal yields. Primary and remote evaluated separately.

Baselines. Parsers trained on bilexical graphs and trees converted from UCCA training set, and evaluated by converting test set output to UCCA.

	Wiki			20K
	Train	Dev	Test	Leagues
# passages	300	34	33	154
# sentences	4268	454	503	506
# nodes	298,993	33,704	35,718	29,315
% terminal	42.96	43.54	42.87	42.09
% non-term.	58.33	57.60	58.35	60.01
% discont.	0.54	0.53	0.44	0.81
% reentrant	2.38	1.88	2.15	2.03
# edges	287,914	32,460	34,336	27,749
% primary	98.25	98.75	98.74	97.73
% remote	1.75	1.25	1.26	2.27
Average per non-terminal node				
# children	1.67	1.68	1.66	1.61

Statistics of *Wiki* and *20K Leagues* corpora.



Bilexical graph approximation.

As no direct comparison with existing parsers is possible, we compare TUPA to bilexical dependency graph parsers, which support reentrancy and discontinuity but not non-terminal nodes. We also convert UCCA to (bilexical) trees and evaluate constituency and dependency tree parsers on them, by simply removing remote edges from the graph.

Results

TUPA$_{\text{BiLSTM}}$ obtains the highest F-scores in all metrics:

	Wiki (in-domain)						20K Leagues (out-of-domain)					
	Primary			Remote			Primary			Remote		
	LP	**LR**	**LF**	**LP**	**LR**	**LF**	**LP**	**LR**	**LF**	**LP**	**LR**	**LF**
TUPA$_{\text{Sparse}}$	64.5	63.7	64.1	19.8	13.4	16	59.6	59.9	59.8	22.2	7.7	11.5
TUPA$_{\text{MLP}}$	65.2	64.6	64.9	23.7	13.2	16.9	62.3	62.6	62.5	20.9	6.3	9.7
TUPA$_{\text{BiLSTM}}$	74.4	72.7	**73.5**	47.4	51.6	**49.4**	68.7	68.5	**68.6**	38.6	18.8	**25.3**
Bilexical Approximation (Dependency DAG Parsers)												
Upper Bound			91			58.3			91.3			43.4
DAGParser [7]	61.8	55.8	58.6	9.5	0.5	1	56.4	50.6	53.4	–	0	0
TurboParser [2]	57.7	46	51.2	77.8	1.8	3.7	50.3	37.7	43.1	100	0.4	0.8
Tree Approximation (Constituency Tree Parser)												
Upper Bound			100			–			100			–
UPARSE [5]	60.9	61.2	61.1	–	–	–	52.7	52.8	52.8	–	–	–
Bilexical Tree Approximation (Dependency Tree Parsers)												
Upper Bound			91			–			91.3			–
MaltParser [6]	62.8	57.7	60.2	–	–	–	57.8	53	55.3	–	–	–
LSTM Parser [4]	73.2	66.9	69.9	–	–	–	66.1	61.1	63.5	–	–	–

The performance is encouraging in light of UCCA’s inter-annotator agreement of 80–85% F-score on primary edges [1].

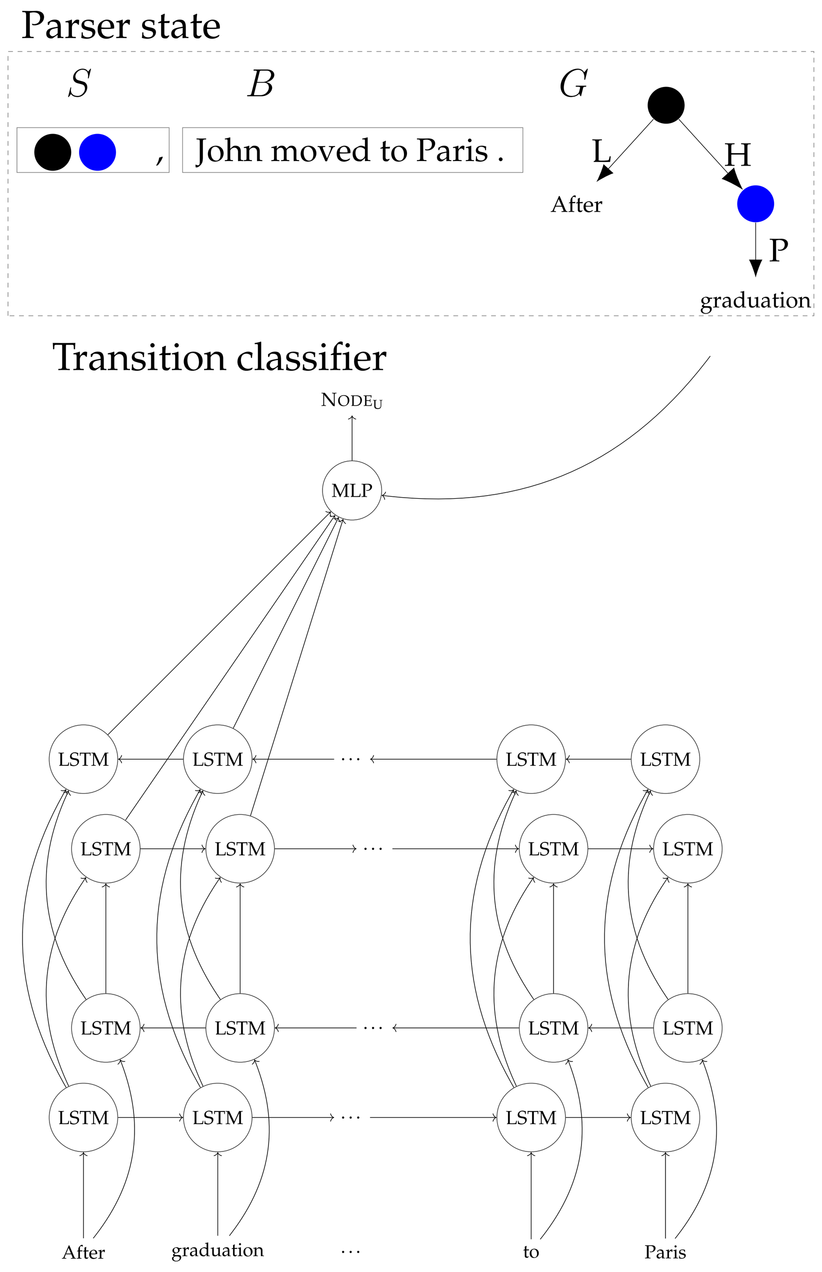


Illustration of the TUPA model.

Conclusions

We present TUPA, the first parser for UCCA, and show that with a NN classifier and BiLSTM feature extractor, it accurately predicts UCCA graphs from text, outperforming various strong baselines. Future work will explore different target representations, and apply the parser to more languages, demonstrating the importance of broad-coverage parsing. A parser for UCCA will enable using the framework for new tasks.

References

[1] Omri Abend and Ari Rappoport. Universal Conceptual Cognitive Annotation (UCCA). In *Proc. of ACL*, pages 228–238, August 2013.

[2] Mariana S. C. Almeida and André F. T. Martins. Lisbon: Evaluating TurboSemanticParser on multiple languages and out-of-domain data. In *Proc. of SemEval*, pages 970–973, June 2015.

[3] Alexandra Birch, Omri Abend, Ondřej Bojar, and Barry Haddow. HUME: Human UCCA-based evaluation of machine translation. In *Proc. of EMNLP*, pages 1264–1274, November 2016.

[4] Chris Dyer, Miguel Ballesteros, Wang Ling, Austin Matthews, and Noah A. Smith. Transition-based dependeny parsing with stack long short-term memory. In *Proc. of ACL*, pages 334–343, 2015.

[5] Wolfgang Maier and Timm Lichte. Discontinuous parsing with continuous trees. In *Proc. of Workshop on Discontinuous Structures in NLP*, pages 47–57, June 2016.

[6] Joakim Nivre, Johan Hall, Jens Nilsson, Atanas Chanev, Gülsen Eryigit, Sandra Kübler, Svetoslav Marinov, and Erwin Marsi. MaltParser: A language-independent system for data-driven dependency parsing. *Natural Language Engineering*, 13(02):95–135, 2007.

[7] Corentin Ribeyre, Eric Villemonte de la Clergerie, and Djamé Seddah. Alpage: Transition-based semantic graph parsing with syntactic features. In *Proc. of SemEval*, pages 97–103, August 2014.

[8] Elior Sulem, Omri Abend, and Ari Rappoport. Conceptual annotations preserve structure across translations: A French-English case study. In *Proc. of S2MT*, pages 11–22, 2015.