ERS for Everyone
Meaning Representation Parsing Shared Tasks

http://sdp.delph-in.net

http://mrp.nlpl.eu

Stephan Oepen

University of Oslo, Department of Informatics

oe@ifi.uio.no

DELPH-IN May at Times
Feel Somewhat Peripheral to
Mainstream Natural Language Processing

DELPH-IN May at Times
Feel Somewhat Peripheral to
Mainstream Natural Language Processing

Everyone who is Anyone

DELPH-IN May at Times
Feel Somewhat Peripheral to
Mainstream Natural Language Processing

Everyone who is Anyone
Uses ERG Meaning Representations

English Resource Semantics Increasingly Popular

DELPH-IN May at Times
Feel Somewhat Peripheral to
Mainstream Natural Language Processing

Everyone who is Anyone
Uses ERG Meaning Representations
(Simplified as Directed Graphs,

DELPH-IN May at Times
Feel Somewhat Peripheral to
Mainstream Natural Language Processing

Everyone who is Anyone
Uses ERG Meaning Representations
(Simplified as Directed Graphs,
Preferably Bi-Lexical)

Modeling Graph Languages with Grammars Extracted via Tree Decompositions

Bevan Keeley Jones[*,†] B.K.Jones@sms.ed.ac.uk
Sharon Goldwater[*] sgwater@inf.ed.ac.uk
Mark Johnson[†] mark.johnson@mq.edu.au

[*] School of Informatics, University of Edinburgh, Edinburgh, UK
[†] Department of Computing, Macquarie University, Sydney, Australia

Abstract

Work on probabilistic models of natural language tends to focus on strings and trees, but there is increasing interest in more general graph-shaped structures since they seem to be better suited for representing natural language semantics, ontologies, or other varieties of knowledge structures. However, while there are relatively simple approaches to defining generative models over strings and trees, it has proven more challenging for more general graphs. This paper describes a natural generalization of the n-gram to graphs, making use of Hyperedge Re-

on graphs has been hampered, due, in part, to the absence of a general agreed upon formalism for processing and modeling such data structures. Where string and tree modeling benefits from the wildly popular Probabilistic Context Free Grammar (PCFG) and related formalisms such as Tree Substitution Grammar, Regular Tree Grammar, Hidden Markov Models, and n-grams, there is nothing of similar popularity for graphs. We need a slightly different formalism, and Hyperedge Replacement Grammar (HRG) (Drewes et al., 1997), a variety of context-free grammar for graphs, suggests itself as a reasonable choice given its close analogy with CFG. Of course, in order to make use of the formalism we need actual grammars, and this paper fills that gap by in-

Transition-Based Parsing for Deep Dependency Structures

Xun Zhang*
Peking University

Yantao Du*
Peking University

Weiwei Sun*
Peking University

Xiaojun Wan*
Peking University

Derivations under different grammar formalisms allow extraction of various dependency structures. Particularly, bilexical deep dependency structures beyond surface tree representation

Parsing to Noncrossing Dependency Graphs

Marco Kuhlmann and **Peter Jonsson**
Department of Computer and Information Science
Linköping University, Sweden
marco.kuhlmann@liu.se and peter.jonsson@liu.se

Abstract

We study the generalization of maximum spanning tree dependency parsing to maximum acyclic subgraphs. Because the underlying optimization problem is intractable even under an arc-factored model, we consider the restriction to *noncrossing* dependency graphs. Our main contribution is a cubic-time exact inference algorithm for this class. We extend this algorithm into a practical parser and evaluate its performance on four linguistic data sets used in semantic dependency parsing. We also explore

While a maximum spanning tree of a weighted digraph can be found in polynomial time (Tarjan, 1977), computing a maximum acyclic subgraph is intractable, and even good approximate solutions are hard to find (Guruswami et al., 2011). In this paper we therefore address maximum acyclic subgraph parsing under the restriction that the subgraph should be *noncrossing*, which informally means that its arcs can be drawn on the half-plane above the sentence in such a way that no two arcs cross (and without changing the order of the words). The main contribution

Robust Incremental Neural Semantic Graph Parsing

Jan Buys[1] **and Phil Blunsom**[1,2]
[1]Department of Computer Science, University of Oxford [2]DeepMind
{jan.buys,phil.blunsom}@cs.ox.ac.uk

Abstract

Parsing sentences to linguistically-expressive semantic representations is a key goal of Natural Language Processing. Yet statistical parsing has focussed almost exclusively on bilexical dependencies or domain-specific logical forms. We propose a neural encoder-decoder transition-based parser which is the first full-coverage semantic graph parser for Minimal Recursion Semantics (MRS).

However the linguistic structure used in applications has predominantly been shallow, restricted to bilexical dependencies or trees.

In this paper we focus on robust parsing into linguistically deep representations. The main representation that we use is Minimal Recursion Semantics (MRS) (Copestake et al., 1995, 2005), which serves as the semantic representation of the English Resource Grammar (ERG) (Flickinger, 2000). Existing parsers for full MRS (as opposed to bilexical semantic graphs derived from, but simplifying MRS) are grammar-based, per-

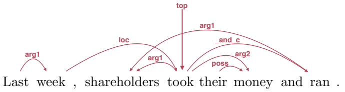
Big Conferences (ACL 2017)

Deep Multitask Learning for Semantic Dependency Parsing

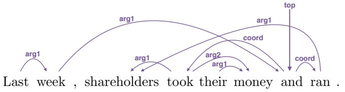
Hao Peng[*] **Sam Thomson**[†] **Noah A. Smith**[*]
[*]Paul G. Allen School of Computer Science & Engineering, University of Washington, Seattle, WA, USA
[†]School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, USA
{hapeng,nasmith}@cs.washington.edu, sthomson@cs.cmu.edu

Abstract

We present a deep neural architecture that parses sentences into three semantic dependency graph formalisms. By using efficient, nearly arc-factored inference and a bidirectional-LSTM composed with a multi-layer perceptron, our base system is able to significantly improve the state of the art for semantic dependency parsing, without using hand-engineered features or syntax. We then explore two multitask



(a) DM



(b) PAS

Semantic Dependency Parsing via Book Embedding

Weiwei Sun, Junjie Cao and **Xiaojun Wan**
Institute of Computer Science and Technology, Peking University
The MOE Key Laboratory of Computational Linguistics, Peking University
{ws,junjie.cao,wanxiaojun}@pku.edu.cn

Abstract

We model a dependency graph as a book, a particular kind of topological space, for semantic dependency parsing. The spine of the book is made up of a sequence of words, and each page contains a subset of noncrossing arcs. To build a semantic graph for a given sentence, we design new Maximum Subgraph algorithms to generate noncrossing graphs on each page, and a Lagrangian Relaxation-based algorithm to combine pages into a book. Experi-

complexity to low degrees. For transition-based parsing, no principled decoding algorithms, e.g. dynamic programming (DP), has been developed for existing transition systems.

In this paper, we borrow the idea of book embedding from graph theory, and propose a novel framework to build parsers for flexible dependency representations. In graph theory, a *book* is a kind of topological space that consists of a spine and a collection of one or more half-planes. In our "book model" of semantic dependency graph, the spine is made up of a sequence of words, and

Simpler but More Accurate Semantic Dependency Parsing

Timothy Dozat
Stanford University
tdozat@stanford.edu

Christopher D. Manning
Stanford University
manning@stanford.edu

Abstract

While syntactic dependency annotations concentrate on the surface or functional structure of a sentence, semantic dependency annotations aim to capture between-word relationships that are more closely related to the meaning of a sentence, using graph-structured representations. We extend the LSTM-based syntactic parser of Dozat and Manning (2017) to train on and generate these graph structures. The resulting system on its own achieves state-

strict tree structure in favor of a richer graph-structured representation, allowing them to capture more linguistic information about a sentence. This opens up the possibility of providing more useful information to downstream tasks (Reddy et al., 2017; Schuster et al., 2017), but increases the difficulty of automatically extracting that information, since most previous work on parsing has focused on generating trees.

Dozat and Manning (2017) developed a successful syntactic dependency parsing system with few task-specific sources of complexity. In this

Multitask Parsing Across Semantic Representations

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
Hebrew University of Jerusalem
{danielh,oabend,arir}@cs.huji.ac.il

Abstract

The ability to consolidate information of different types is at the core of intelligence, and has tremendous practical value in allowing learning for one task to benefit from generalizations learned for others. In this paper we tackle the challenging task of improving semantic parsing performance, taking UCCA parsing as a test case, and AMR, SDP and Universal Dependencies (UD) parsing as auxiliary tasks. We experiment on three languages, using a uni-

fectively extend the training data, and has greatly advanced with neural networks and representation learning (see §2). We build on these ideas and propose a general transition-based DAG parser, able to parse UCCA, AMR, SDP and UD (Nivre et al., 2016). We train the parser using MTL to obtain significant improvements on UCCA parsing over single-task training in (1) in-domain and (2) out-of-domain settings in English; (3) an in-domain setting in German; and (4) an in-domain setting in French, where training data is scarce.

The novelty of this work is in proposing a gen-

Semantics as a Foreign Language

Gabriel Stanovsky*[2,3] and Ido Dagan[1]

[1]Bar-Ilan University Computer Science Department, Ramat Gan, Israel
[2]Paul G. Allen School of Computer Science & Engineering, University of Washington, Seattle, WA
[3]Allen Institute for Artificial Intelligence, Seattle, WA
gabis@cs.washington.edu
dagan@cs.biu.ac.il

Abstract

We propose a novel approach to semantic dependency parsing (SDP) by casting the task as an instance of multi-lingual machine translation, where each semantic representation is a different foreign dialect. To that end, we first generalize syntactic linearization techniques to account for the richer semantic dependency graph structure. Following, we design a neural sequence-to-sequence framework which can effectively recover our graph linearizations, performing almost on-par with previous SDP state-of-the-art while requiring less parallel

Lexical Dependencies (DM) (Flickinger, 2000),[1] (2) Enju Predicate-Argument Structures (PAS) (Miyao et al., 2014), and (3) Prague Semantic Dependencies (PSD) (Hajic et al., 2012). These annotations have garnered recent attention (e.g., (Buys and Blunsom, 2017; Peng et al., 2017a)), and were consistently annotated in parallel on over more than 30K sentences of the Wall Street Journal corpus (Charniak et al., 2000).

In this work we take a novel approach to graph parsing, casting sentence-level semantic parsing as a multilingual machine-translation task (MT). We

Multi-Task Semantic Dependency Parsing with Policy Gradient for Learning Easy-First Strategies

Shuhei Kurita
Center for Advanced Intelligence Project
RIKEN
Tokyo, Japan
shuhei.kurita@riken.jp

Anders Søgaard
Department of Computer Science
University of Copenhagen
Copenhagen, Denmark
soegaard@di.ku.dk

Abstract

In Semantic Dependency Parsing (SDP), semantic relations form directed acyclic graphs, rather than trees. We propose a new iterative predicate selection (IPS) algorithm for SDP. Our IPS algorithm combines the graph-based and transition-based parsing approaches in order to handle *multiple* semantic head words. We train the IPS model using a combination of multi-task learning and task-specific policy gradient training. Trained this way, IPS achieves a new state of the art on the SemEval 2015 Task 18 datasets. Furthermore, we ob-

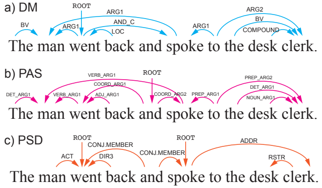


Figure 1: Semantic dependency parsing arcs of DM, PAS and PSD formalisms.

Compositional Semantic Parsing Across Graphbanks

Matthias Lindemann* and **Jonas Groschwitz*** and **Alexander Koller**
Department of Language Science and Technology
Saarland University
{mlinde|jonasg|koller}@coli.uni-saarland.de

Abstract

Most semantic parsers that map sentences to graph-based meaning representations are hand-designed for specific graphbanks. We present a compositional neural semantic parser which achieves, for the first time, competitive accuracies across a diverse range of graphbanks. Incorporating BERT embeddings and multi-task learning improves the accuracy further, setting new states of the art on DM, PAS, PSD, AMR 2015 and EDS.

2018) assumes dependency graphs and cannot be directly applied to EDS or AMR. Conversely, top AMR parsers (Lyu and Titov, 2018) invest heavily into identifying AMR-specific alignments and concepts, which may not be useful in other graphbanks. Hershcovich et al. (2018) parse across different semantic graphbanks (UCCA, DM, AMR), but focus on UCCA and do poorly on DM. The system of Buys and Blunsom (2017) set a state of the art on EDS at the time, but does poorly on AMR.

In this paper, we present a single semantic parser that does very well across all of DM, PAS, PSD,

Even Bigger Conferences (EMNLP 2019)

Broad-Coverage Semantic Parsing as Transduction

Sheng Zhang **Xutai Ma** **Kevin Duh** **Benjamin Van Durme**
Johns Hopkins University
{zsheng2, xutai_ma}@jhu.edu
{kevinduh, vandurme}@cs.jhu.edu

Abstract

We unify different broad-coverage semantic parsing tasks under a transduction paradigm, and propose an attention-based neural framework that *incrementally* builds a meaning representation via a sequence of semantic relations. By leveraging multiple attention mechanisms, the transducer can be effectively trained without relying on a pre-trained aligner. Experiments conducted on three separate broad-coverage semantic parsing tasks – AMR, SDP and UCCA – demonstrate that our attention-based neural transducer improves the state of

Manning, 2018; Peng et al., 2017a) are not directly transferable to AMR and UCCA because of the lack of explicit alignments between tokens in the sentence and nodes in the semantic graph.

While transition-based approaches are adaptable to different broad-coverage semantic parsing tasks (Wang et al., 2018; Hershcovich et al., 2018; Damonte et al., 2017), when it comes to representations such as AMR whose nodes are *unanchored* to tokens in the sentence, a pre-trained aligner has to be used to produce the reference transition sequences (Wang et al., 2015; Damonte et al.,

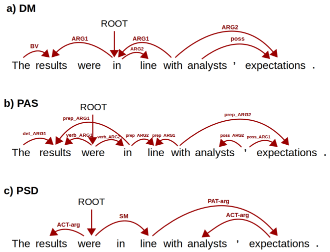
Last Week (ACL 2020)

Transition-based Semantic Dependency Parsing with Pointer Networks

Daniel Fernández-González and **Carlos Gómez-Rodríguez**
Universidade da Coruña, CITIC
FASTPARSE Lab, LyS Group
Depto. de Ciencias de la Computación y Tecnologías de la Información
Campus de Elviña, s/n, 15071 A Coruña, Spain
d.fgonzalez@udc.es, carlos.gomez@udc.es

Abstract

Transition-based parsers implemented with Pointer Networks have become the new state of the art in dependency parsing, excelling in producing labelled syntactic trees and outperforming graph-based models in this task. In order to further test the capabilities of these powerful neural networks on a harder NLP problem, we propose a transition system that, thanks to Pointer Networks, can straightforwardly produce labelled directed acyclic graphs and perform semantic dependency parsing. In addition, we enhance our approach



Semi-Supervised Semantic Dependency Parsing Using CRF Autoencoders

Zixia Jia[◇], **Youmi Ma**[†], **Jiong Cai**[◇], **Kewei Tu**[◇*]
[◇]School of Information Science and Technology, ShanghaiTech University
Shanghai Institute of Microsystem and Information Technology, Chinese Academy of Sciences
University of Chinese Academy of Sciences
Shanghai Engineering Research Center of Intelligent Vision and Imaging
[†]Tokyo Institute of Technology
{jiazx, caijiong, tukw}@shanghaitech.edu.cn
youmi.ma@nlp.c.titech.ac.jp

Abstract

Semantic dependency parsing, which aims to find rich bi-lexical relationships, allows words to have multiple dependency heads, resulting in graph-structured representations. We propose an approach to semi-supervised learning of semantic dependency parsers based on the CRF autoencoder framework. Our encoder is a discriminative neural semantic dependency parser that predicts the latent parse graph of the input sentence. Our decoder is a generative neural model that reconstructs the input sentence conditioned on the latent parse graph.

2018) or graph-based (Martins and Almeida, 2014; Peng et al., 2017; Dozat and Manning, 2018; Wang et al., 2019).

One limitation of supervised SDP is that labeled SDP data resources are limited in scale and diversity. Due to the rich relationships in SDP, the annotation of semantic dependency graphs is expensive and difficult, calling for professional linguists to design rules and highly skilled annotators to annotate sentences. This limitation becomes more severe with the rise of deep learning, because neural approaches are more data-hungry and susceptible to

Last Week (ACL 2020)

Parsing into Variable-in-situ Logico-Semantic Graphs

Yufei Chen[1] and **Weiwei Sun**[12]
[1]Wangxuan Institute of Computer Technology
[1]The MOE Key Laboratory of Computational Linguistics
[2]Center for Chinese Linguistics
Peking University
{yufei.chen,ws}@pku.edu.cn

Abstract

We propose variable-in-situ logico-semantic graphs to bridge the gap between semantic graph and logical form parsing. The new type of graph-based meaning representation allows us to include analysis for scope-related phenomena, such as quantification, negation and modality, in a way that is consistent with the state-of-the-art underspecification approach. Moreover, the well-formedness of such a graph is clear, since model-theoretic interpretation is available. We demonstrate the effectiveness of this new perspective by de-

Partly due to the lack of model-theoretic semantics, it is rather difficult to add scope information related to quantification, negation and modality to a graph. Partly due to the lack of logical deduction engines, it is rather difficult to directly perform automated reasoning over graphs.

This paper proposes to express logical forms with variable-in-situ graphs for the ongoing advances in graph-centric formalisms, algorithms and neural architectures. This leads us to a novel neural graph rewriting system that combines the strengths of Hyperedge Replacement Grammar

Exact yet Efficient Graph Parsing, Bi-directional Locality and the Constructivist Hypothesis

Yajie Ye[1] and **Weiwei Sun**[12]
[1]Wangxuan Institute of Computer Technology
[1]The MOE Key Laboratory of Computational Linguistics
[2]Center for Chinese Linguistics
Peking University
{yeyajie,ws}@pku.edu.cn

Abstract

A key problem in processing graph-based meaning representations is *graph parsing*, i.e. computing all possible derivations of a given graph according to a (competence) grammar. We demonstrate, for the first time, that exact graph parsing can be efficient for large graphs and with large Hyperedge Replacement Grammars (HRGs). The advance is achieved by exploiting locality as terminal edge-adjacency in HRG rules. In particular, we highlight the importance of 1) a terminal edge-first parsing strategy, 2) a categorization of a subclass production, a reversed direction to language understanding. We discuss locality in a sense of terminal edge-adjacency and develop a locality-centric complexity analysis of the *de facto* algorithm introduced by Chiang et al. (2013). Our analysis motivates (1) a terminal edge-first parsing strategy, (2) a categorization of a subclass of HRG, i.e. what we call Weakly Regular Graph Grammar, and (3) a computational support in the constructivist hypothesis in theoretical linguistics. Altogether, our analysis leads to a substantial improvement in practical graph parsing. An MR with the number of concep-

Semantic Parsing for English as a Second Language

Yuanyuan Zhao[1,2], Weiwei Sun[1,3], Junjie Cao[1*] and Xiaojun Wan[1]
[1]Wangxuan Institute of Computer Technology, Peking University
[1]The MOE Key Laboratory of Computational Linguistics, Peking University
[2]Academy for Advanced Interdisciplinary Studies, Peking University
[3]Center for Chinese Linguistics, Peking University
{zhao_yy,ws,wanxiaojun}@pku.edu.cn
junjie.junjiecao@alibaba-inc.com

Abstract

This paper is concerned with semantic parsing for English as a second language (ESL). Motivated by the theoretical emphasis on the learning challenges that occur at the syntax-semantics interface during second language acquisition, we formulate the task based on the divergence between literal and intended meanings. We combine the complementary strengths of English Resource Grammar, a linguistically-precise hand-crafted deep grammar, and TLE, an existing manually annotated ESL UD-TreeBank with a novel reranking

ena (Gass, 2013). This direction has been recently explored by the NLP community (Nagata and Sakaguchi, 2016; Berzak et al., 2016a; Lin et al., 2018).

Different from *standard* English, ESL may preserve many features of learners' first languages[1]. The difference between learner texts and benchmark training data, e.g. Penn TreeBank (PTB; Marcus et al., 1993), is more related to linguistic competence, rather than performance (Chomsky, 2014). This makes processing ESL different from almost all the existing discussions on domain

Kind of a Reference Task (NAACL 2019)

Linguistic Knowledge and Transferability of Contextual Representations

Nelson F. Liu[♠♡*] **Matt Gardner**[♣] **Yonatan Belinkov**[♢]
Matthew E. Peters[♣] **Noah A. Smith**[♠♣]

[♠]Paul G. Allen School of Computer Science & Engineering,
University of Washington, Seattle, WA, USA
[♡]Department of Linguistics, University of Washington, Seattle, WA, USA
[♣]Allen Institute for Artificial Intelligence, Seattle, WA, USA
[♢]Harvard John A. Paulson School of Engineering and Applied Sciences and
MIT Computer Science and Artificial Intelligence Laboratory, Cambridge, MA, USA

{nfliu,nasmith}@cs.washington.edu
{mattg,matthewp}@allenai.org, belinkov@seas.harvard.edu

Abstract

Contextual word representations derived from large-scale neural language models are successful across a diverse set of NLP tasks, suggesting that they encode useful and transferable features of language. To shed light on the linguistic knowledge they capture, we study the representations produced by several recent pretrained contextualizers (variants of ELMo, the OpenAI transformer language model, and BERT) with a suite of sixteen diverse probing tasks. We find that linear models trained on top of frozen contextual repre-

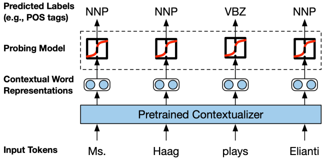


Figure 1: An illustration of the probing model setup used to study the linguistic knowledge within contextual word representations.

The Thirty-Third International
FLAIRS Conference (FLAIRS-33)

Establishing Strong Baselines for the New Decade: Sequence Tagging, Syntactic and Semantic Parsing with BERT

Han He
Computer Science
Emory University
Atlanta GA 30322, USA
han.he@emory.edu

Jinho D. Choi
Computer Science
Emory University
Atlanta GA 30322, USA
jinho.choi@emory.edu

Abstract

This paper presents new state-of-the-art models for three tasks, part-of-speech tagging, syntactic parsing, and semantic parsing, using the cutting-edge contextualized embedding framework known as BERT. For each task, we first replicate and simplify the current state-of-the-art approach to enhance its model efficiency. We then evaluate our simplified approaches on those three tasks using token embeddings generated by BERT. 12 datasets in both English and Chinese are used for our experiments. The BERT models outperform the previously best-performing models by 2.5% on average (7.5% for the most significant case). All models and source codes are available in public so that researchers can improve upon and utilize them to establish strong baselines for the next decade. We also provide a dedicated error analysis and extensive dissections in

on 3.3B word corpus. After scaling the model size to hundreds of millions parameters, BERT brings markedly huge improvement to a wide range of tasks without substantial task-specific modifications.

In this paper, we verify the effectiveness and conciseness of BERT by first generating token-level embeddings from it, then integrating them to task-oriented yet efficient model structures (Section 3). With careful investigation and engineering, our simplified models significantly outperform many of the previous state-of-the-art models, achieving the highest scores for 11 out of 12 datasets (Section 4).

To the best of our knowledge, it is the first work that tightly integrates BERT embeddings to these three downstream tasks and present such high performance. All our resources including the models and the source codes are publicly available [1]

Compatibility, Comparison, Competition

Joint Work with Linköping, Prague, and Yusuke Miyao

- Three parallel semantic annotations over the venerable WSJ text;
- sentence- and token-aligned (PTB tokenization, Unicode punctuation);
- bi-lexical DM: DELPH-IN MRS–Derived Bi-Lexical Dependencies;
- SemEval 2014 & 2015 parsing shared tasks; reference release via LDC;

Compatibility, Comparison, Competition

Joint Work with Linköping, Prague, and Yusuke Miyao

- Three parallel semantic annotations over the venerable WSJ text;
- sentence- and token-aligned (PTB tokenization, Unicode punctuation);
- bi-lexical DM: DELPH-IN MRS–Derived Bi-Lexical Dependencies;
- SemEval 2014 & 2015 parsing shared tasks; reference release via LDC;

→ now known as SDP: (Broad-Coverage) Semantic Dependency Parsing:

`http://sdp.delph-in.net`

Compatibility, Comparison, Competition

Joint Work with Linköping, Prague, and Yusuke Miyao

- Three parallel semantic annotations over the venerable WSJ text;
- sentence- and token-aligned (PTB tokenization, Unicode punctuation);
- bi-lexical DM: DELPH-IN MRS–Derived Bi-Lexical Dependencies;
- SemEval 2014 & 2015 parsing shared tasks; reference release via LDC;

→ now known as SDP: (Broad-Coverage) Semantic Dependency Parsing:

`http://sdp.delph-in.net`

Joint Work with Brandeis & Colorado, Jerusalem, and Groningen

- Beyond bi-lexical simplifications: general (directed) semantic graphs;
- relate to perceived ‘mainstream’: Abstract Meaning Representation;
- a ‘new kid’ on the block: Universal Conceptual Cognitive Annotation;

Compatibility, Comparison, Competition

Joint Work with Linköping, Prague, and Yusuke Miyao

- Three parallel semantic annotations over the venerable WSJ text;
- sentence- and token-aligned (PTB tokenization, Unicode punctuation);
- bi-lexical DM: DELPH-IN MRS–Derived Bi-Lexical Dependencies;
- SemEval 2014 & 2015 parsing shared tasks; reference release via LDC;

→ now known as SDP: (Broad-Coverage) Semantic Dependency Parsing:

`http://sdp.delph-in.net`

Joint Work with Brandeis & Colorado, Jerusalem, and Groningen

- Beyond bi-lexical simplifications: general (directed) semantic graphs;
- relate to perceived 'mainstream': Abstract Meaning Representation;
- a 'new kid' on the block: Universal Conceptual Cognitive Annotation;

→ Meaning Representation Parsing (MRP) tasks at CoNLL 2019 and 2020:

`http://mrp.nlpl.eu`

Brief Genealogy of ERS Simplifications

Native Interface: Minimal Recursion Semantics (Copestake et al., 2005)

- Logic-inspired tradition in (computational) natural language semantics;
- designer 'logic' for scope underspecification: labeled tree fragments;
- grammar provides (highly) partial information about possible scopings;
- downstream usages (so far) predominantly on unscoped representations.

Brief Genealogy of ERS Simplifications

Native Interface: Minimal Recursion Semantics (Copestake et al., 2005)

- Logic-inspired tradition in (computational) natural language semantics;
- designer 'logic' for scope underspecification: labeled tree fragments;
- grammar provides (highly) partial information about possible scopings;
- downstream usages (so far) predominantly on unscoped representations.

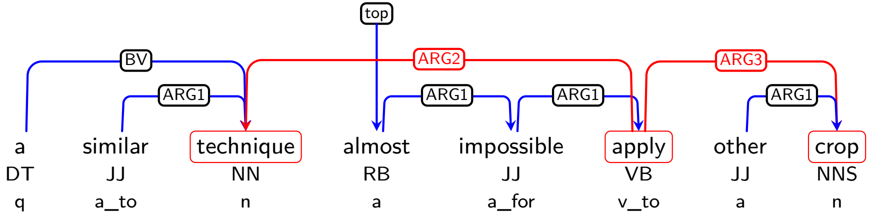
Dependency-Inspired, (Mostly) Graph-Based Alternate Renderings

2000	K2Y	Callmeier & Flickinger (Driving up to Napa Valley)
2002	EDS	Variable-free dependency graph (Oepen & Lønning, 2006)
2009	DMRS	Extend EDS with underspedified scope (Copestake, 2009)
2012	DM	Reduce EDS to bi-lexical form (Ivanova et al., 2012)

(0) Two Bi-Lexical Frameworks: DM & PSD

DM: DELPH-IN MRS Bi-Lexical Dependencies (Ivanova et al., 2012)

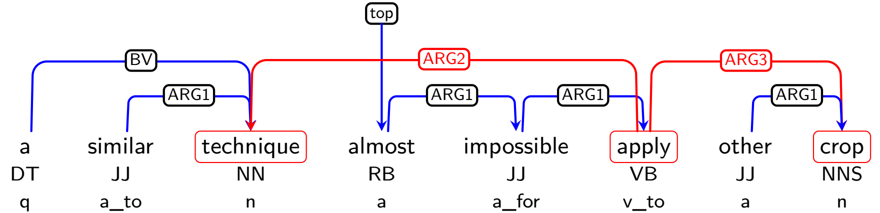
- Simplification from underspecified logical forms (ERS; coming later);



(0) Two Bi-Lexical Frameworks: DM & PSD

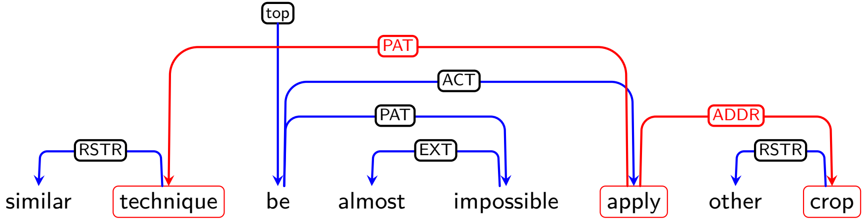
DM: DELPH-IN MRS Bi-Lexical Dependencies (Ivanova et al., 2012)

- Simplification from underspecified logical forms (ERS; coming later);



PSD: Prague Semantic Dependencies (Hajič et al., 2012)

- Simplification from FGD tectogrammatical trees (Sgall et al., 1986).



SDP 2014 (Task 8 at SemEval): Nine Teams

SemEval 2014 Task 8: Broad-Coverage Semantic Dependency Parsing

Stephan Oepen[♣♠], Marco Kuhlmann[♡], Yusuke Miyao[◇], Daniel Zeman[◦], Dan Flickinger[•], Jan Hajič[◦], Angelina Ivanova[♣], and Yi Zhang[⋆]

[♣] University of Oslo, Department of Informatics

[♠] Potsdam University, Department of Linguistics

[♡] Linköping University, Department of Computer and Information Science

[◇] National Institute of Informatics, Tokyo

[◦] Charles University in Prague, Faculty of Mathematics and Physics, Institute of Formal and Applied Linguistics

[•] Stanford University, Center for the Study of Language and Information

[⋆] Nuance Communications Aachen GmbH

sdp-organizers@emmtee.net

Abstract

Task 8 at SemEval 2014 defines *Broad-Coverage Semantic Dependency Parsing* (SDP) as the problem of recovering sentence-internal predicate–argument relationships for *all content words*, i.e. the semantic structure constituting the relational core of sentence meaning. In this task description, we position the problem in

Unfortunately, tree-oriented parsers are ill-suited for producing meaning representations, i.e. moving from the analysis of grammatical structure to sentence semantics. Even if syntactic parsing arguably can be limited to tree structures, this is not the case in semantic analysis, where a node will often be the argument of multiple predicates (i.e. have more than one incoming arc), and it will often be desirable to leave nodes corresponding to se-

SemEval 2015 Task 18: Broad-Coverage Semantic Dependency Parsing

Stephan Oepen[♣♠], Marco Kuhlmann[♡], Yusuke Miyao[♢], Daniel Zeman[∘], Silvie Cinková[∘], Dan Flickinger[•], Jan Hajič[∘], and Zdeňka Urešová[∘]

[♣] University of Oslo, Department of Informatics
[♠] Potsdam University, Department of Linguistics
[♡] Linköping University, Department of Computer and Information Science
[♢] National Institute of Informatics, Tokyo
[∘] Charles University in Prague, Faculty of Mathematics and Physics, Institute of Formal and Applied Linguistics
[•] Stanford University, Center for the Study of Language and Information

sdp-organizers@emmtee.net

Abstract

Task 18 at SemEval 2015 defines *Broad-Coverage Semantic Dependency Parsing* (SDP) as the problem of recovering sentence-internal predicate–argument relationships for *all content words*, i.e. the semantic structure constituting the relational core of sentence meaning. In this task description, we position the problem in comparison to other language analysis sub-tasks, introduce and compare the semantic

more general graph processing, to thus enable a more direct analysis of *Who did What to Whom?*

Extending the very similar predecessor task SDP 2014 (Oepen et al., 2014), we make use of three distinct, parallel semantic annotations over the same common texts, viz. the venerable Wall Street Journal (WSJ) and Brown segments of the Penn Treebank (PTB; Marcus et al., 1993) for English, as well as comparable resources for Chinese and Czech. Figure 1 below shows example target representations,

High-Level Goals of the MRP Shared Tasks

Cross-Framework Comparability and Interoperability

- Vast, complex landscape of representing natural language meaning;
- diverse linguistic traditions, modeling assumptions, levels of ambition;
- some differences are superficial (e.g. terminology), others run deeper;

$\rightarrow$ clarify concepts and terminology; unify representations and evaluation.

High-Level Goals of the MRP Shared Tasks

Cross-Framework Comparability and Interoperability

- Vast, complex landscape of representing natural language meaning;
- diverse linguistic traditions, modeling assumptions, levels of ambition;
- some differences are superficial (e.g. terminology), others run deeper;

$\rightarrow$ clarify concepts and terminology; unify representations and evaluation.

Parsing into Graph-Structured Representations

- Cottage industry of parsers with output structures beyond rooted trees;
- different families: factorization, transitions, composition, 'translation';
- much framework-internal evolution: design reflects specific assumptions;

$\rightarrow$ evaluate across frameworks; learning from complementary knowledge.

High-Level Goals of the MRP Shared Tasks

Cross-Framework Comparability and Interoperability

- Vast, complex landscape of representing natural language meaning;
- diverse linguistic traditions, modeling assumptions, levels of ambition;
- some differences are superficial (e.g. terminology), others run deeper;

$\rightarrow$ clarify concepts and terminology; unify representations and evaluation.

Parsing into Graph-Structured Representations

- Cottage industry of parsers with output structures beyond rooted trees;
- different families: factorization, transitions, composition, 'translation';
- much framework-internal evolution: design reflects specific assumptions;

$\rightarrow$ evaluate across frameworks; learning from complementary knowledge.

Learning from Complementary Knowledge

- Cross-Framework Perspective: Seek commonality and complementarity.

$$\mathbb{G} = \langle N, E, T \rangle$$

- $\mathbb{G}$ is a directed graph: N is set of nodes; $E \subseteq N \times N$ is set of edges;
- $T \subseteq N$ is possibly empty set of top node(s): the 'main' predicate(s);

$$\mathbb{G} = \langle N, E, T \rangle$$

- $\mathbb{G}$ is a directed graph: N is set of nodes; $E \subseteq N \times N$ is set of edges;
- $T \subseteq N$ is possibly empty set of top node(s): the 'main' predicate(s);
- ***in***- and ***out***-degree of $n \in N$ count edges to and from n; ***in*** $= 0$: root;
- top in *Abrams arrived quickly.* is *arrive*, but can be argument of *quick*;
- semantic graphs often multi-rooted: rootness just a structural property;

Graph Theory 101

$$\mathbb{G} = \langle N, E, T \rangle$$

- $\mathbb{G}$ is a directed graph: N is set of nodes; $E \subseteq N \times N$ is set of edges;
- $T \subseteq N$ is possibly empty set of top node(s): the 'main' predicate(s);
- ***in***- and ***out***-degree of $n \in N$ count edges to and from n; ***in*** $= 0$: root;
- top in *Abrams arrived quickly.* is *arrive*, but can be argument of *quick*;
- semantic graphs often multi-rooted: rootness just a structural property;
- a node n is reentrant if ***in***$(n) > 1$ (shared argument across predicates);

$$\mathbb{G} = \langle N, E, T \rangle$$

- $\mathbb{G}$ is a directed graph: N is set of nodes; $E \subseteq N \times N$ is set of edges;
- $T \subseteq N$ is possibly empty set of top node(s): the 'main' predicate(s);
- ***in***- and ***out***-degree of $n \in N$ count edges to and from n; ***in*** $= 0$: root;
- top in *Abrams arrived quickly.* is *arrive*, but can be argument of *quick*;
- semantic graphs often multi-rooted: rootness just a structural property;
- a node n is reentrant if ***in***$(n) > 1$ (shared argument across predicates);
- cycles can occur: directed path from m to n and ('back') from n to m;

Graph Theory 101

$$\mathbb{G} = \langle N, E, T \rangle$$

- $\mathbb{G}$ is a directed graph: N is set of nodes; $E \subseteq N \times N$ is set of edges;
- $T \subseteq N$ is possibly empty set of top node(s): the ‘main’ predicate(s);
- ***in***- and ***out***-degree of $n \in N$ count edges to and from n; ***in*** $= 0$: root;
- top in *Abrams arrived quickly.* is *arrive*, but can be argument of *quick*;
- semantic graphs often multi-rooted: rootness just a structural property;
- a node n is reentrant if ***in***$(n) > 1$ (shared argument across predicates);
- cycles can occur: directed path from m to n and (‘back’) from n to m;
- $\mathbb{G}$ is connected if there is an undirected path between all pairs of nodes;

$$\mathbb{G} = \langle N, E, T \rangle$$

- $\mathbb{G}$ is a directed graph: N is set of nodes; $E \subseteq N \times N$ is set of edges;
- $T \subseteq N$ is possibly empty set of top node(s): the 'main' predicate(s);
- ***in***- and ***out***-degree of $n \in N$ count edges to and from n; ***in*** $= 0$: root;
- top in *Abrams arrived quickly.* is *arrive*, but can be argument of *quick*;
- semantic graphs often multi-rooted: rootness just a structural property;
- a node n is reentrant if ***in***$(n) > 1$ (shared argument across predicates);
- cycles can occur: directed path from m to n and ('back') from n to m;
- $\mathbb{G}$ is connected if there is an undirected path between all pairs of nodes;
- $\mathbb{G}$ is a tree if $|T| = 1$ and there is a unique path to all other nodes.

Anchoring in the Surface String

Relating Pieces of Meaning to the Linguistic Signal

- Intuitively, sub-structures of meaning relate to sub-parts of the input;
- semantic frameworks vary in how much weight to put on this relation;

Anchoring in the Surface String

Relating Pieces of Meaning to the Linguistic Signal

- Intuitively, sub-structures of meaning relate to sub-parts of the input;
- semantic frameworks vary in how much weight to put on this relation;
- anchoring of graph elements in sub-strings of the underlying utterance;
- can be part of semantic annotations or not; can take different forms;

Anchoring in the Surface String

Relating Pieces of Meaning to the Linguistic Signal

- Intuitively, sub-structures of meaning relate to sub-parts of the input;
- semantic frameworks vary in how much weight to put on this relation;
- anchoring of graph elements in sub-strings of the underlying utterance;
- can be part of semantic annotations or not; can take different forms;
- hierarchy of anchoring types: Flavor (0)–(2); bilexical graphs strictest;

Flavor	Name	Example	Type of Anchoring
(0)	bilexical	DM, PSD	nodes are sub-set of surface tokens
(1)	anchored	EDS, UCCA	free node–sub-string correspondences
(2)	unanchored	AMR	no explicit sub-string correspondences

Anchoring in the Surface String

Relating Pieces of Meaning to the Linguistic Signal

- Intuitively, sub-structures of meaning relate to sub-parts of the input;
- semantic frameworks vary in how much weight to put on this relation;
- anchoring of graph elements in sub-strings of the underlying utterance;
- can be part of semantic annotations or not; can take different forms;
- hierarchy of anchoring types: Flavor (0)–(2); bilexical graphs strictest;
- anchoring central in parsing, explicit or latent; aka ‘alignment’ for AMR;
- relevant to at least some downstream tasks; should impact evaluation.

Flavor	**Name**	**Example**	**Type of Anchoring**
(0)	bilexical	DM, PSD	nodes are sub-set of surface tokens
(1)	anchored	EDS, UCCA	free node–sub-string correspondences
(2)	unanchored	AMR	no explicit sub-string correspondences

(1) Elementary Dependency Structures (EDS)

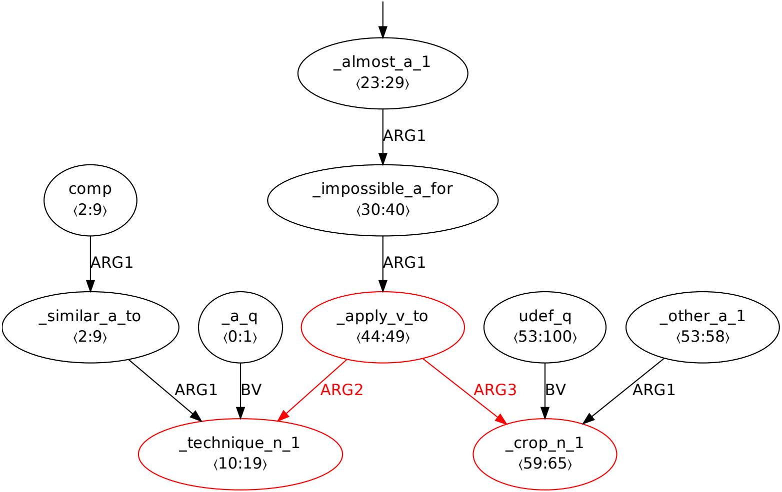
Break Free of Bi-Lexical Limitations (Oepen & Lønning, 2006)

- Decomposition or construction meaning; anchors: arbitrary sub-strings.

(1) Elementary Dependency Structures (EDS)

Break Free of Bi-Lexical Limitations (Oepen & Lønning, 2006)

- Decomposition or construction meaning; anchors: arbitrary sub-strings.

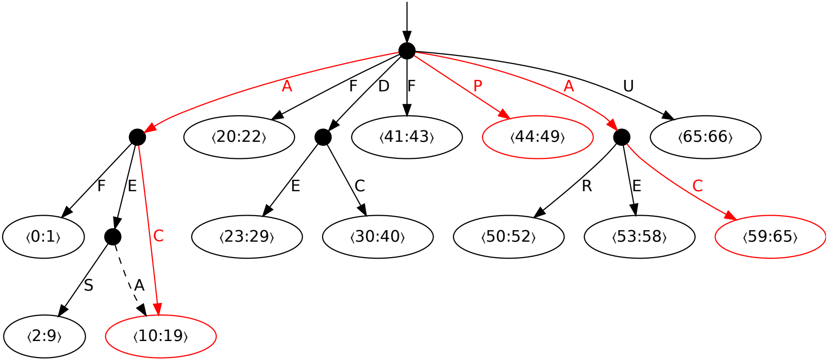


A similar technique is almost impossible to apply to other crops.

(1) Universal Conceptual Cognitive Annotation (UCCA)

Multi-Layered Design (Abend & Rappoport, 2013); Foundational Layer

- Tree backbone: semantic ‘constituents’ are scenes (‘clauses’) and units;

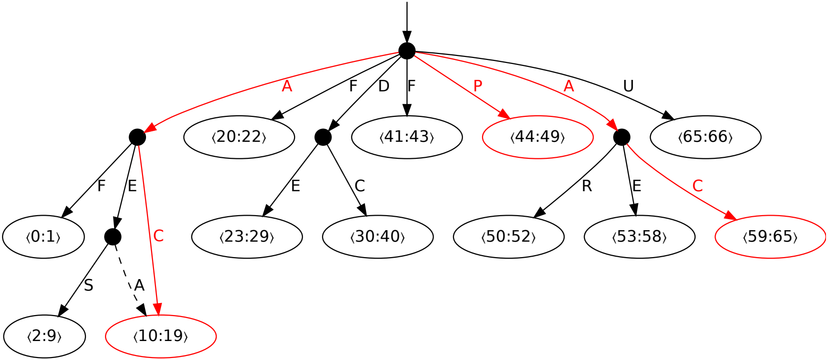


A similar technique is almost impossible to apply to other crops.

(1) Universal Conceptual Cognitive Annotation (UCCA)

Multi-Layered Design (Abend & Rappoport, 2013); Foundational Layer

- Tree backbone: semantic ‘constituents’ are scenes (‘clauses’) and units;
- scenes (Process or State): pArticipants and aDverbials (plus F and U);

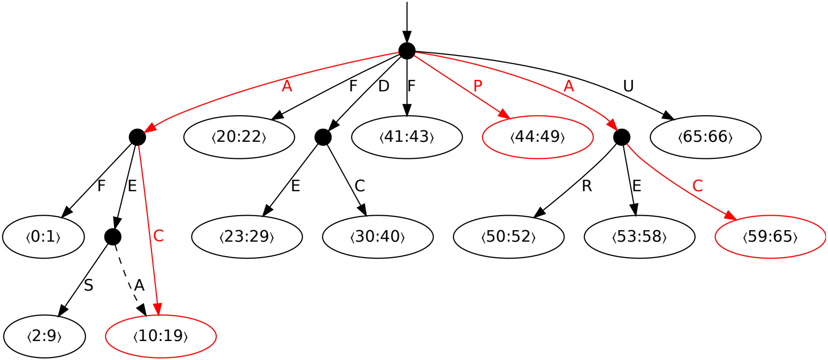


A similar technique is almost impossible to apply to other crops.

(1) Universal Conceptual Cognitive Annotation (UCCA)

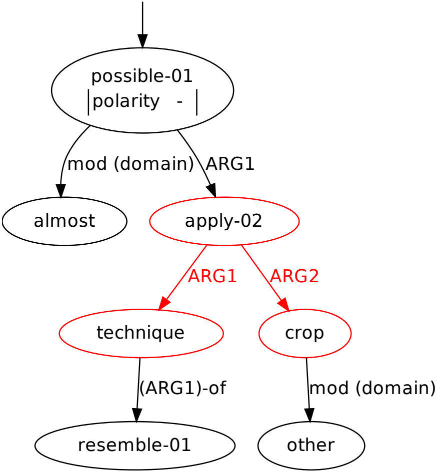
Multi-Layered Design (Abend & Rappoport, 2013); Foundational Layer

- Tree backbone: semantic 'constituents' are scenes ('clauses') and units;
- scenes (Process or State): pArticipants and aDverbials (plus F and U);
- complex units distinguish Center and Elaborator(s); allow remote edges.



A similar technique is almost impossible to apply to other crops.

(2) Abstract Meaning Representation (AMR)



Banarescu et al. (2013)

- Abstractly (if not linguistically) similar to EDS, but unanchored;
- verbal senses from PropBank^{++};
- negation as node-local property;
- tree-like annotation: inversed edges normalized for evaluation;
- originally designed for (S)MT; various NLU applications to date.

A similar technique is almost impossible to apply to other crops.

MRP 2019 (CoNLL Shared Task): Eighteen Teams

MRP 2019: Cross-Framework Meaning Representation Parsing

Stephan Oepen♣**, Omri Abend**♠**, Jan Hajič**♡**, Daniel Hershcovich**♢**, Marco Kuhlmann**°**,**
Tim O'Gorman⋆**, Nianwen Xue**•**, Jayeol Chun**•**, Milan Straka**♡**, and Zdeňka Urešová**♡

♣ University of Oslo, Department of Informatics
♠ The Hebrew University of Jerusalem, School of Computer Science and Engineering
♡ Charles University in Prague, Faculty of Mathematics and Physics, Institute of Formal and Applied Linguistics
♢ University of Copenhagen, Department of Computer Science
° Linköping University, Department of Computer and Information Science
⋆ University of Colorado at Boulder, Department of Linguistics
• Brandeis University, Department of Computer Science

mrp-organizers@nlpl.eu,
jchun@brandeis.edu, {straka|uresova}@ufal.mff.cuni.cz

Abstract

The 2019 Shared Task at the Conference for Computational Language Learning (CoNLL) was devoted to Meaning Representation Parsing (MRP) across frameworks. Five distinct approaches to the representation of sentence meaning in the form of directed graphs were represented in the training and evaluation data for the task, packaged in a uniform graph abstraction and serialization. The task received

Representation Parsing (MRP 2019). The goal of the task is to advance data-driven parsing into *graph-structured* representations of *sentence meaning*. For the first time, this task combines *formally* and *linguistically* different approaches to meaning representation in graph form in a uniform training and evaluation setup. Participants were invited to develop parsing systems that support five distinct semantic graph frameworks (see §3 below)—

Training and Evaluation Data in the Shared Task

		DM	PSD	EDS	UCCA	AMR
	Flavor	0	0	1	1	2
train	Text Type	newspaper	newspaper	newspaper	mixed	mixed
	Sentences	35,656	35,656	35,656	6,572	56,240
	Tokens	802,717	802,717	802,717	138,268	1,000,217
test	Text Type	mixed	mixed	mixed	mixed	mixed
	Sentences	3,359	3,359	3,359	1,131	1,998
	Tokens	64,853	64,853	64,853	21,647	39,520

- DM, PSD, and EDS annotate the same text (Sections 00–20 of WSJ);
- UCCA: samples of EWT & Wikipedia; AMR: twelve different sources;

Training and Evaluation Data in the Shared Task

		DM	PSD	EDS	UCCA	AMR
	Flavor	0	0	1	1	2
train	Text Type	newspaper	newspaper	newspaper	mixed	mixed
	Sentences	35,656	35,656	35,656	6,572	56,240
	Tokens	802,717	802,717	802,717	138,268	1,000,217
test	Text Type	mixed	mixed	mixed	mixed	mixed
	Sentences	3,359	3,359	3,359	1,131	1,998
	Tokens	64,853	64,853	64,853	21,647	39,520

- DM, PSD, and EDS annotate the same text (Sections 00–20 of WSJ);
- UCCA: samples of EWT & Wikipedia; AMR: twelve different sources;
- linguistics: 100-item WSJ sample in all frameworks publicly available;

Training and Evaluation Data in the Shared Task

		DM	PSD	EDS	UCCA	AMR
	Flavor	0	0	1	1	2
train	Text Type	newspaper	newspaper	newspaper	mixed	mixed
	Sentences	35,656	35,656	35,656	6,572	56,240
	Tokens	802,717	802,717	802,717	138,268	1,000,217
test	Text Type	mixed	mixed	mixed	mixed	mixed
	Sentences	3,359	3,359	3,359	1,131	1,998
	Tokens	64,853	64,853	64,853	21,647	39,520

- DM, PSD, and EDS annotate the same text (Sections 00–20 of WSJ);
- UCCA: samples of EWT & Wikipedia; AMR: twelve different sources;
- linguistics: 100-item WSJ sample in all frameworks publicly available;
- evaluation: subset of 100 sentences from *The Little Prince* is public.

Graphbank Statistics (Kuhlmann & Oepen, 2016)

			DM	PSD	EDS	UCCA	AMR^{-1}
counts	(01)	number of graphs	35,656	35,656	35,656	6,572	56,240
	(01)	number of tokens	802,717	802,717	802,717	138,268	1,000,217
	(02)	average number of tokens	22.51	22.51	22.51	21.03	17,78
	(03)	average nodes per token	0.77	0.64	1.29	1.37	0.65
	(04)	number of edge labels	59	90	10	15	101
treeness	(05)	$\%_g$ trees	2.31	42.26	0.09	34.83	22.24
	(06)	$\%_g$ treewidth one	69.82	43.08	68.99	41.57	50.00
	(07)	average treewidth	1.30	1.61	1.31	1.61	1.56
	(08)	maximal treewidth	3	7	3	4	5
	(09)	average edge density	1.019	1.073	1.015	1.053	1.092
	(10)	$\%_n$ reentrant	27.43	11.41	32.78	4.98	19.89
	(11)	$\%_g$ cyclic	0.00	0.00	0.12	0.00	0.38
	(12)	$\%_g$ not connected	6.57	0.70	1.74	0.00	0.00
	(13)	$\%_g$ multi-rooted	97.47	40.60	99.93	0.00	71.37
	(14)	percentage non-top roots	44.94	4.34	54.85	0.00	20.09
order	(15)	average edge length	2.684	3.320	–	–	–
	(16)	$\%_g$ noncrossing	69.21	64.61	–	–	–
	(17)	$\%_g$ pagenumber two	99.59	98.08	–	–	–

Graphbank Statistics (Kuhlmann & Oepen, 2016)

			DM	PSD	EDS	UCCA	AMR^{-1}
counts	(01)	number of graphs	35,656	35,656	35,656	6,572	56,240
	(01)	number of tokens	802,717	802,717	802,717	138,268	1,000,217
	(02)	average number of tokens	22.51	22.51	22.51	21.03	17,78
	(03)	average nodes per token	0.77	0.64	1.29	1.37	0.65
	(04)	number of edge labels	59	90	10	15	101
treeness	(05)	$\%_g$ trees	2.31	42.26	0.09	34.83	22.24
	(06)	$\%_g$ treewidth one	69.82	43.08	68.99	41.57	50.00
	(07)	average treewidth	1.30	1.61	1.31	1.61	1.56
	(08)	maximal treewidth	3	7	3	4	5
	(09)	average edge density	1.019	1.073	1.015	1.053	1.092
	(10)	$\%_n$ reentrant	27.43	11.41	32.78	4.98	19.89
	(11)	$\%_g$ cyclic	0.00	0.00	0.12	0.00	0.38
	(12)	$\%_g$ not connected	6.57	0.70	1.74	0.00	0.00
	(13)	$\%_g$ multi-rooted	97.47	40.60	99.93	0.00	71.37
	(14)	percentage non-top roots	44.94	4.34	54.85	0.00	20.09
order	(15)	average edge length	2.684	3.320	–	–	–
	(16)	$\%_g$ noncrossing	69.21	64.61	–	–	–
	(17)	$\%_g$ pagenumber two	99.59	98.08	–	–	–

Graphbank Statistics (Kuhlmann & Oepen, 2016)

			DM	PSD	EDS	UCCA	AMR^{-1}
counts	(01)	number of graphs	35,656	35,656	35,656	6,572	56,240
	(01)	number of tokens	802,717	802,717	802,717	138,268	1,000,217
	(02)	average number of tokens	22.51	22.51	22.51	21.03	17,78
	(03)	average nodes per token	0.77	0.64	1.29	1.37	0.65
	(04)	number of edge labels	59	90	10	15	101
treeness	(05)	$\%_g$ trees	2.31	42.26	0.09	34.83	22.24
	(06)	$\%_g$ treewidth one	69.82	43.08	68.99	41.57	50.00
	(07)	average treewidth	1.30	1.61	1.31	1.61	1.56
	(08)	maximal treewidth	3	7	3	4	5
	(09)	average edge density	1.019	1.073	1.015	1.053	1.092
	(10)	$\%_n$ reentrant	27.43	11.41	32.78	4.98	19.89
	(11)	$\%_g$ cyclic	0.00	0.00	0.12	0.00	0.38
	(12)	$\%_g$ not connected	6.57	0.70	1.74	0.00	0.00
	(13)	$\%_g$ multi-rooted	97.47	40.60	99.93	0.00	71.37
	(14)	percentage non-top roots	44.94	4.34	54.85	0.00	20.09
order	(15)	average edge length	2.684	3.320	–	–	–
	(16)	$\%_g$ noncrossing	69.21	64.61	–	–	–
	(17)	$\%_g$ pagenumber two	99.59	98.08	–	–	–

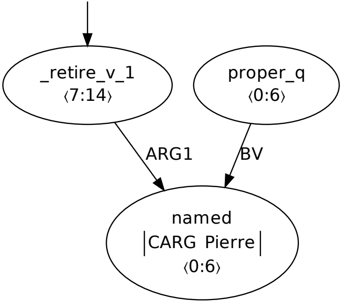
Graphbank Statistics (Kuhlmann & Oepen, 2016)

			DM	PSD	EDS	UCCA	AMR^{-1}
counts	(01)	number of graphs	35,656	35,656	35,656	6,572	56,240
	(01)	number of tokens	802,717	802,717	802,717	138,268	1,000,217
	(02)	average number of tokens	22.51	22.51	22.51	21.03	17,78
	(03)	average nodes per token	0.77	0.64	1.29	1.37	0.65
	(04)	number of edge labels	59	90	10	15	101
treeness	(05)	$\%_g$ trees	2.31	42.26	0.09	34.83	22.24
	(06)	$\%_g$ treewidth one	69.82	43.08	68.99	41.57	50.00
	(07)	average treewidth	1.30	1.61	1.31	1.61	1.56
	(08)	maximal treewidth	3	7	3	4	5
	(09)	average edge density	1.019	1.073	1.015	1.053	1.092
	(10)	$\%_n$ reentrant	27.43	11.41	32.78	4.98	19.89
	(11)	$\%_g$ cyclic	0.00	0.00	0.12	0.00	0.38
	(12)	$\%_g$ not connected	6.57	0.70	1.74	0.00	0.00
	(13)	$\%_g$ multi-rooted	97.47	40.60	99.93	0.00	71.37
	(14)	percentage non-top roots	44.94	4.34	54.85	0.00	20.09
order	(15)	average edge length	2.684	3.320	–	–	–
	(16)	$\%_g$ noncrossing	69.21	64.61	–	–	–
	(17)	$\%_g$ pagenumber two	99.59	98.08	–	–	–

Graphbank Statistics (Kuhlmann & Oepen, 2016)

			DM	PSD	EDS	UCCA	AMR^{-1}
counts	(01)	number of graphs	35,656	35,656	35,656	6,572	56,240
	(01)	number of tokens	802,717	802,717	802,717	138,268	1,000,217
	(02)	average number of tokens	22.51	22.51	22.51	21.03	17,78
	(03)	average nodes per token	0.77	0.64	1.29	1.37	0.65
	(04)	number of edge labels	59	90	10	15	101
treeness	(05)	$\%_g$ trees	2.31	42.26	0.09	34.83	22.24
	(06)	$\%_g$ treewidth one	69.82	43.08	68.99	41.57	50.00
	(07)	average treewidth	1.30	1.61	1.31	1.61	1.56
	(08)	maximal treewidth	3	7	3	4	5
	(09)	average edge density	1.019	1.073	1.015	1.053	1.092
	(10)	$\%_n$ reentrant	27.43	11.41	32.78	4.98	19.89
	(11)	$\%_g$ cyclic	0.00	0.00	0.12	0.00	0.38
	(12)	$\%_g$ not connected	6.57	0.70	1.74	0.00	0.00
	(13)	$\%_g$ multi-rooted	97.47	40.60	99.93	0.00	71.37
	(14)	percentage non-top roots	44.94	4.34	54.85	0.00	20.09
order	(15)	average edge length	2.684	3.320	–	–	–
	(16)	$\%_g$ noncrossing	69.21	64.61	–	–	–
	(17)	$\%_g$ pagenumber two	99.59	98.08	–	–	–

- Break down graphs into types of information: per-type and overall F_1;



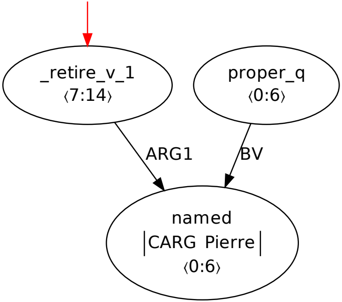
Pierre retired.

Different Types of Semantic Graph 'Atoms'

	DM	PSD	EDS	UCCA	AMR
Top Nodes	✓	✓	✓	✓	✓
Labeled Edges	✓	✓	✓	✓	✓
Node Labels	✓	✓	✓	✗	✓
Node Properties	✓	✓	✓	✗	✓
Node Anchoring	✓	✓	✓	✓	✗
Edge Attributes	✗	✗	✗	✓	✗

Cross-Framework Evaluation: MRP Graph Similarity

- Break down graphs into types of information: per-type and overall F_1;
- tops



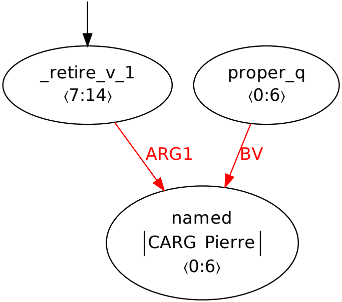
Pierre retired.

Different Types of Semantic Graph 'Atoms'

	DM	PSD	EDS	UCCA	AMR
Top Nodes	✓	✓	✓	✓	✓
Labeled Edges	✓	✓	✓	✓	✓
Node Labels	✓	✓	✓	✗	✓
Node Properties	✓	✓	✓	✗	✓
Node Anchoring	✓	✓	✓	✓	✗
Edge Attributes	✗	✗	✗	✓	✗

Cross-Framework Evaluation: MRP Graph Similarity

- Break down graphs into types of information: per-type and overall F_1;
- tops and (labeled) edges;

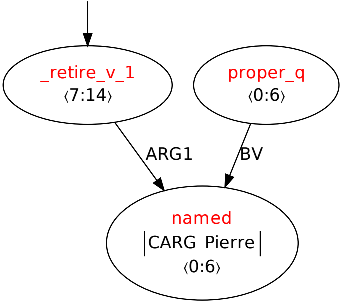


Pierre retired.

Different Types of Semantic Graph 'Atoms'

	DM	PSD	EDS	UCCA	AMR
Top Nodes	✓	✓	✓	✓	✓
Labeled Edges	✓	✓	✓	✓	✓
Node Labels	✓	✓	✓	✗	✓
Node Properties	✓	✓	✓	✗	✓
Node Anchoring	✓	✓	✓	✓	✗
Edge Attributes	✗	✗	✗	✓	✗

- Break down graphs into types of information: per-type and overall F_1;
- tops and (labeled) edges; labels,



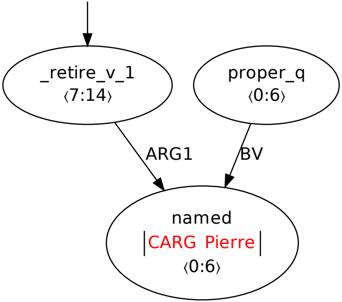
Pierre retired.

Different Types of Semantic Graph 'Atoms'

	DM	PSD	EDS	UCCA	AMR
Top Nodes	✓	✓	✓	✓	✓
Labeled Edges	✓	✓	✓	✓	✓
Node Labels	✓	✓	✓	✗	✓
Node Properties	✓	✓	✓	✗	✓
Node Anchoring	✓	✓	✓	✓	✗
Edge Attributes	✗	✗	✗	✓	✗

Cross-Framework Evaluation: MRP Graph Similarity

- Break down graphs into types of information: per-type and overall F_1;
- tops and (labeled) edges; labels, properties,



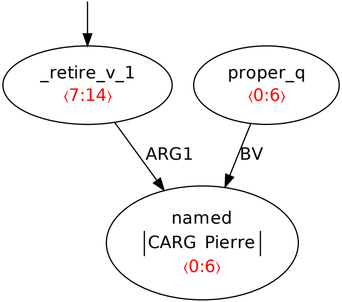
Pierre retired.

Different Types of Semantic Graph 'Atoms'

	DM	PSD	EDS	UCCA	AMR
Top Nodes	✓	✓	✓	✓	✓
Labeled Edges	✓	✓	✓	✓	✓
Node Labels	✓	✓	✓	✗	✓
Node Properties	✓	✓	✓	✗	✓
Node Anchoring	✓	✓	✓	✓	✗
Edge Attributes	✗	✗	✗	✓	✗

Cross-Framework Evaluation: MRP Graph Similarity

- Break down graphs into types of information: per-type and overall F_1;
- tops and (labeled) edges; labels, properties, anchors,



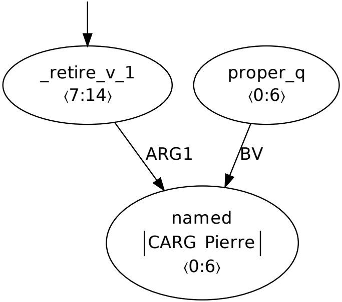
Pierre retired.

Different Types of Semantic Graph 'Atoms'

	DM	PSD	EDS	UCCA	AMR
Top Nodes	✓	✓	✓	✓	✓
Labeled Edges	✓	✓	✓	✓	✓
Node Labels	✓	✓	✓	✗	✓
Node Properties	✓	✓	✓	✗	✓
Node Anchoring	✓	✓	✓	✓	✗
Edge Attributes	✗	✗	✗	✓	✗

Cross-Framework Evaluation: MRP Graph Similarity

- Break down graphs into types of information: per-type and overall F_1;
- tops and (labeled) edges; labels, properties, anchors, and attributes;



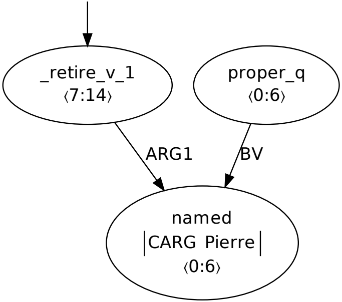
Pierre retired.

Different Types of Semantic Graph 'Atoms'

	DM	PSD	EDS	UCCA	AMR
Top Nodes	✓	✓	✓	✓	✓
Labeled Edges	✓	✓	✓	✓	✓
Node Labels	✓	✓	✓	✗	✓
Node Properties	✓	✓	✓	✗	✓
Node Anchoring	✓	✓	✓	✓	✗
Edge Attributes	✗	✗	✗	✓	✗

Cross-Framework Evaluation: MRP Graph Similarity

- Break down graphs into types of information: per-type and overall F_1;
- tops and (labeled) edges; labels, properties, anchors, and attributes;
- requires node–node correspondences; search for overall maximum score;
- maximum common edge subgraph isomorphism (MCES) is NP-hard;



Pierre retired.

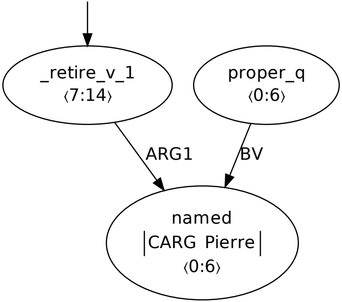
Different Types of Semantic Graph 'Atoms'

	DM	PSD	EDS	UCCA	AMR
Top Nodes	✓	✓	✓	✓	✓
Labeled Edges	✓	✓	✓	✓	✓
Node Labels	✓	✓	✓	✗	✓
Node Properties	✓	✓	✓	✗	✓
Node Anchoring	✓	✓	✓	✓	✗
Edge Attributes	✗	✗	✗	✓	✗

Cross-Framework Evaluation: MRP Graph Similarity

- Break down graphs into types of information: per-type and overall F_1;
- tops and (labeled) edges; labels, properties, anchors, and attributes;
- requires node–node correspondences; search for overall maximum score;
- maximum common edge subgraph isomorphism (MCES) is NP-hard;

→ smart initialization, scheduling, and pruning yield strong approximation.



Pierre retired.

Different Types of Semantic Graph 'Atoms'

	DM	PSD	EDS	UCCA	AMR
Top Nodes	✓	✓	✓	✓	✓
Labeled Edges	✓	✓	✓	✓	✓
Node Labels	✓	✓	✓	✗	✓
Node Properties	✓	✓	✓	✗	✓
Node Anchoring	✓	✓	✓	✓	✗
Edge Attributes	✗	✗	✗	✓	✗

Comparison to Top-Performing Data-Driven Parsers

		Tops			Labels			Properties			Anchors			Edges		
		P	R	F_1	P	R	F_1	P	R	F_1	P	R	F_1	P	R	F_1
DM	ERG	.92	.92	.918	.99	.99	**.987**	.96	.96	**.956**	.99	.99	**.994**	.91	.91	.912
	SJTU–NICT	.93	.93	**.933**	.95	.95	.949	.96	.95	.955	.99	.99	.993	.93	.92	.924
	HIT-SCIR	.93	.93	.926	.93	.93	.930	.95	.95	.953	.99	.99	.993	.93	.92	**.925**
	SUDA–Alibaba	.91	.91	.911	.90	.91	.903	.91	.92	.915	.97	.99	.982	.89	.91	.898
	Peking	.93	.93	.927	.92	.91	.915	.95	.94	.945	.99	.99	.991	.92	.92	.924
EDS	ERG	.90	.90	**.902**	.97	.96	**.965**	.96	.96	**.960**	.96	.96	**.963**	.93	.93	.929
	SUDA–Alibaba	.90	.90	.899	.91	.91	.912	.89	.91	.897	.95	.95	.949	.90	.90	.897
	HIT-SCIR	.88	.82	.852	.90	.89	.894	.89	.91	.895	.95	.94	.943	.89	.88	.888
	SJTU–NICT	.91	.85	.877	.93	.86	.894	.79	.76	.775	.97	.90	.934	.95	.82	.878
	Peking	.83	.83	.829	.95	.94	.946	.91	.96	.936	.96	.96	.961	.94	.93	**.933**

Comparison to Top-Performing Data-Driven Parsers

		Tops			Labels			Properties			Anchors			Edges		
		P	R	F_1	P	R	F_1	P	R	F_1	P	R	F_1	P	R	F_1
DM	ERG	.92	.92	.918	.99	.99	**.987**	.96	.96	**.956**	.99	.99	**.994**	.91	.91	.912
	SJTU–NICT	.93	.93	**.933**	.95	.95	.949	.96	.95	.955	.99	.99	.993	.93	.92	.924
	HIT-SCIR	.93	.93	.926	.93	.93	.930	.95	.95	.953	.99	.99	.993	.93	.92	**.925**
	SUDA–Alibaba	.91	.91	.911	.90	.91	.903	.91	.92	.915	.97	.99	.982	.89	.91	.898
	Peking	.93	.93	.927	.92	.91	.915	.95	.94	.945	.99	.99	.991	.92	.92	.924
EDS	ERG	.90	.90	**.902**	.97	.96	**.965**	.96	.96	**.960**	.96	.96	**.963**	.93	.93	.929
	SUDA–Alibaba	.90	.90	.899	.91	.91	.912	.89	.91	.897	.95	.95	.949	.90	.90	.897
	HIT-SCIR	.88	.82	.852	.90	.89	.894	.89	.91	.895	.95	.94	.943	.89	.88	.888
	SJTU–NICT	.91	.85	.877	.93	.86	.894	.79	.76	.775	.97	.90	.934	.95	.82	.878
	Peking	.83	.83	.829	.95	.94	.946	.91	.96	.936	.96	.96	.961	.94	.93	**.933**

Comparison to Top-Performing Data-Driven Parsers

		Tops			Labels			Properties			Anchors			Edges		
		P	R	F_1	P	R	F_1	P	R	F_1	P	R	F_1	P	R	F_1
DM	ERG	.92	.92	.918	.99	.99	**.987**	.96	.96	**.956**	.99	.99	**.994**	.91	.91	.912
	SJTU–NICT	.93	.93	**.933**	.95	.95	.949	.96	.95	.955	.99	.99	.993	.93	.92	.924
	HIT-SCIR	.93	.93	.926	.93	.93	.930	.95	.95	.953	.99	.99	.993	.93	.92	**.925**
	SUDA–Alibaba	.91	.91	.911	.90	.91	.903	.91	.92	.915	.97	.99	.982	.89	.91	.898
	Peking	.93	.93	.927	.92	.91	.915	.95	.94	.945	.99	.99	.991	.92	.92	.924
EDS	ERG	.90	.90	**.902**	.97	.96	**.965**	.96	.96	**.960**	.96	.96	**.963**	.93	.93	.929
	SUDA–Alibaba	.90	.90	.899	.91	.91	.912	.89	.91	.897	.95	.95	.949	.90	.90	.897
	HIT-SCIR	.88	.82	.852	.90	.89	.894	.89	.91	.895	.95	.94	.943	.89	.88	.888
	SJTU–NICT	.91	.85	.877	.93	.86	.894	.79	.76	.775	.97	.90	.934	.95	.82	.878
	Peking	.83	.83	.829	.95	.94	.946	.91	.96	.936	.96	.96	.961	.94	.93	**.933**

Comparison to Top-Performing Data-Driven Parsers

		Tops			Labels			Properties			Anchors			Edges		
		P	R	F_1	P	R	F_1	P	R	F_1	P	R	F_1	P	R	F_1
DM	ERG	.92	.92	.918	.99	.99	**.987**	.96	.96	**.956**	.99	.99	**.994**	.91	.91	.912
	SJTU–NICT	.93	.93	**.933**	.95	.95	.949	.96	.95	.955	.99	.99	.993	.93	.92	.924
	HIT-SCIR	.93	.93	.926	.93	.93	.930	.95	.95	.953	.99	.99	.993	.93	.92	**.925**
	SUDA–Alibaba	.91	.91	.911	.90	.91	.903	.91	.92	.915	.97	.99	.982	.89	.91	.898
	Peking	.93	.93	.927	.92	.91	.915	.95	.94	.945	.99	.99	.991	.92	.92	.924
EDS	ERG	.90	.90	**.902**	.97	.96	**.965**	.96	.96	**.960**	.96	.96	**.963**	.93	.93	.929
	SUDA–Alibaba	.90	.90	.899	.91	.91	.912	.89	.91	.897	.95	.95	.949	.90	.90	.897
	HIT-SCIR	.88	.82	.852	.90	.89	.894	.89	.91	.895	.95	.94	.943	.89	.88	.888
	SJTU–NICT	.91	.85	.877	.93	.86	.894	.79	.76	.775	.97	.90	.934	.95	.82	.878
	Peking	.83	.83	.829	.95	.94	.946	.91	.96	.936	.96	.96	.961	.94	.93	**.933**

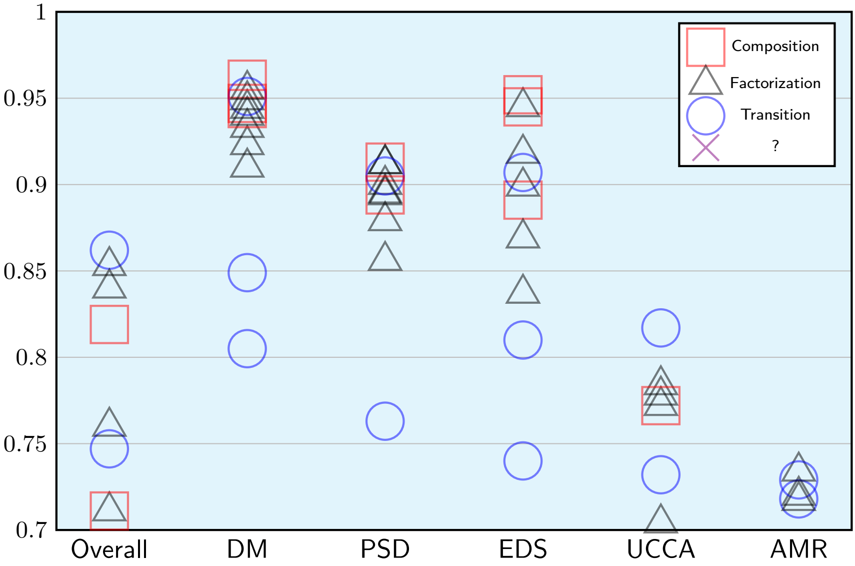
Comparison to Top-Performing Data-Driven Parsers

		Tops			Labels			Properties			Anchors			Edges		
		P	R	F_1	P	R	F_1	P	R	F_1	P	R	F_1	P	R	F_1
DM	ERG	.92	.92	.918	.99	.99	**.987**	.96	.96	**.956**	.99	.99	**.994**	.91	.91	.912
	SJTU–NICT	.93	.93	**.933**	.95	.95	.949	.96	.95	.955	.99	.99	.993	.93	.92	.924
	HIT-SCIR	.93	.93	.926	.93	.93	.930	.95	.95	.953	.99	.99	.993	.93	.92	**.925**
	SUDA–Alibaba	.91	.91	.911	.90	.91	.903	.91	.92	.915	.97	.99	.982	.89	.91	.898
	Peking	.93	.93	.927	.92	.91	.915	.95	.94	.945	.99	.99	.991	.92	.92	.924
EDS	ERG	.90	.90	**.902**	.97	.96	**.965**	.96	.96	**.960**	.96	.96	**.963**	.93	.93	.929
	SUDA–Alibaba	.90	.90	.899	.91	.91	.912	.89	.91	.897	.95	.95	.949	.90	.90	.897
	HIT-SCIR	.88	.82	.852	.90	.89	.894	.89	.91	.895	.95	.94	.943	.89	.88	.888
	SJTU–NICT	.91	.85	.877	.93	.86	.894	.79	.76	.775	.97	.90	.934	.95	.82	.878
	Peking	.83	.83	.829	.95	.94	.946	.91	.96	.936	.96	.96	.961	.94	.93	**.933**

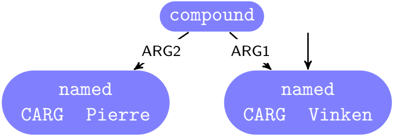
High-Level Overview of Submissions

Teams	DM	PSD	EDS	UCCA	AMR	MTL	Approach
ERG$^{\nparallel\S\dagger}$	✓	✗	✓	✗	✗	✗	Composition
TUPA§†	✓	✓	✓	✓	✓	✗/✓	Transition
HIT-SCIR	✓	✓	✓	✓	✓	✗	Transition
SJTU–NICT	✓	✓	✓	✓	✓	✗	Factorization
SUDA–Alibaba	✓	✓	✓	✓	✓	(✓)	Factorization
Saarland	✓	✓	✓	✓	✓	✗	Composition
Hitachi	✓	✓	✓	✓	✓	(✓)	Factorization
ÚFAL MRPipe	✓	✓	✓	✓	✓	✗	Transition
ShanghaiTech	✓	✓	✓	✗	✓	✗	Factorization
Amazon	✓	✓	✗	✗	✓	✗	Factorization
JBNU	✓	✓	✗	✓	✗	✗	Factorization
SJTU	✓	✓	✓	✓	✓	✓	Transition
ÚFAL–Oslo	✓	✓	✓	✓	✓	✗	Transition
HKUST	✓	✓	✗	✓	✗	?	
Bocharov	✗	✗	✗	✗	✓	?	
Peking$^{\nparallel}$	✓	✓	✓	✓	✗	✗	Factorization
CUHK§	✓	✓	✓	✓	✓	✓	Transition
Anonymous§	✗	✓	✗	✗	✗	?	

Score Distributions: Top Systems

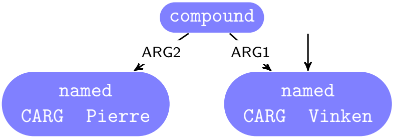


EDSs are 'Radically Compositional'



Pierre Vinken

EDSs are ‘Radically Compositional’



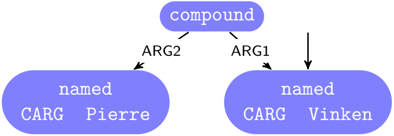
Pierre Vinken

Named Entities

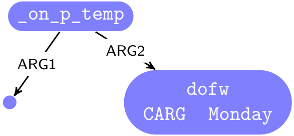
- Underspecified structure in names;
- few, lexically determined sub-types.

Michelle and Barack Obama

EDSs are 'Radically Compositional'



Pierre Vinken



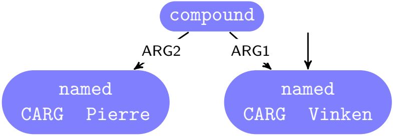
on Monday

Named Entities

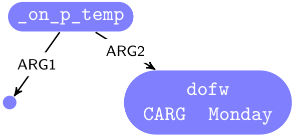
- Underspecified structure in names;
- few, lexically determined sub-types.

Michelle and Barack Obama

EDSs are ‘Radically Compositional’



Pierre Vinken



on Monday

Named Entities

- Underspecified structure in names;
- few, lexically determined sub-types.

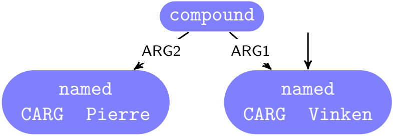
Michelle and Barack Obama

Prepositions (and Similar)

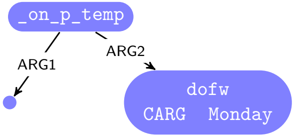
- Predicates: distinct two-place relation;
- specialized sub-senses as appropriate.

before and during the meeting

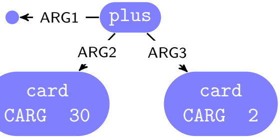
EDSs are ‘Radically Compositional’



Pierre Vinken



on Monday



thirty-two

Named Entities

- Underspecified structure in names;
- few, lexically determined sub-types.

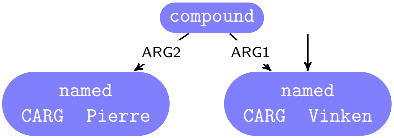
Michelle and Barack Obama

Prepositions (and Similar)

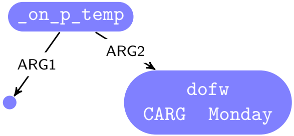
- Predicates: distinct two-place relation;
- specialized sub-senses as appropriate.

before and during the meeting

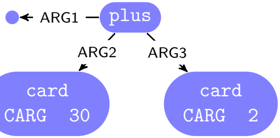
EDSs are 'Radically Compositional'



Pierre Vinken



on Monday



thirty-two

Named Entities

- Underspecified structure in names;
- few, lexically determined sub-types.

Michelle and Barack Obama

Prepositions (and Similar)

- Predicates: distinct two-place relation;
- specialized sub-senses as appropriate.

before and during the meeting

Literal Numbers

- syntax yields arithmetic expressions;
- trivial 'downstream' normalization.

ten to twenty thousand

References I

Omri Abend & Ari Rappoport. 2013. Universal Conceptual Cognitive Annotation (UCCA). In Proceedings of the 51th Meeting of the Association for Computational Linguistics, pages 228–238, Sofia, Bulgaria.

Laura Banarescu, Claire Bonial, Shu Cai, Madalina Georgescu, Kira Griffitt, Ulf Hermjakob, Kevin Knight, Philipp Koehn, Martha Palmer, & Nathan Schneider. 2013. Abstract Meaning Representation for sembanking. In Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse, pages 178–186, Sofia, Bulgaria.

Ann Copestake. 2009. Slacker semantics. Why superficiality, dependency and avoidance of commitment can be the right way to go. In Proceedings of the 12th Meeting of the European Chapter of the Association for Computational Linguistics, pages 1–9, Athens, Greece.

Ann Copestake, Dan Flickinger, Carl Pollard, & Ivan A. Sag. 2005. Minimal Recursion Semantics. An introduction. Research on Language and Computation, 3(4):281–332.

References II

Jan Hajič, Eva Hajičová, Jarmila Panevová, Petr Sgall, Ondřej Bojar, Silvie Cinková, Eva Fučíková, Marie Mikulová, Petr Pajas, Jan Popelka, Jiří Semecký, Jana Šindlerová, Jan Štěpánek, Josef Toman, Zdeňka Urešová, & Zdeněk Žabokrtský. 2012. Announcing Prague Czech-English Dependency Treebank 2.0. In Proceedings of the 8th International Conference on Language Resources and Evaluation, pages 3153 – 3160, Istanbul, Turkey.

Angelina Ivanova, Stephan Oepen, Lilja Øvrelid, & Dan Flickinger. 2012. Who did what to whom? A contrastive study of syntacto-semantic dependencies. In Proceedings of the 6th Linguistic Annotation Workshop, pages 2 – 11, Jeju, Republic of Korea.

Marco Kuhlmann & Stephan Oepen. 2016. Towards a catalogue of linguistic graph banks. Computational Linguistics, 42(4):819 – 827.

Stephan Oepen & Jan Tore Lønning. 2006. Discriminant-based MRS banking. In Proceedings of the 5th International Conference on Language Resources and Evaluation, pages 1250 – 1255, Genoa, Italy.

References III

Petr Sgall, Eva Hajičová, & Jarmila Panevová. 1986. The Meaning of the Sentence and Its Semantic and Pragmatic Aspects. D. Reidel Publishing Company, Dordrecht, The Netherlands.