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 dependency target representations used, and summarize the task setup, participating systems, and main results.

1 Background and Motivation

Syntactic dependency parsing has seen great advances in the past decade, but 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 semantically vacuous word classes unattached (with no incoming arcs). Thus, Task 18 at SemEval 2015, *Broad-Coverage Semantic Dependency Parsing* (SDP 2015),[1] seeks to stimulate the parsing community to move towards 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, bi-lexical semantic dependency graphs in all cases, for the WSJ sentence:

(1) A similar technique is almost impossible to apply to other crops, such as cotton, soybeans, and rice.

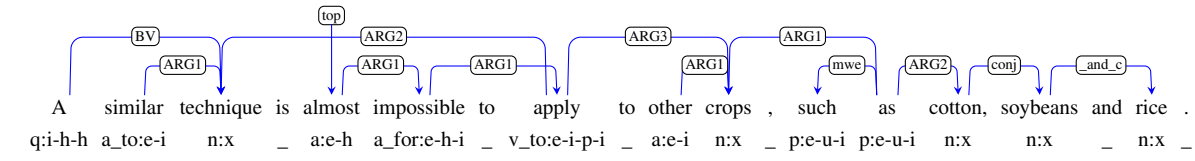
Semantically, *technique* arguably is dependent on the determiner (the quantificational locus), the modifier *similar*, and the predicate *apply*. Conversely, the predicative copula, infinitival *to*, and the vacuous preposition marking the deep object of *apply* can be argued to not have a semantic contribution of their own. Besides calling for node re-entrancies and partial connectivity, semantic dependency graphs may also exhibit higher degrees of non-projectivity than is typical of syntactic dependency trees.

Besides its relation to syntactic dependency parsing, the task also has some overlap with Semantic Role Labeling (SRL; Gildea & Jurafsky, 2002).[2] However, we require parsers to identify 'full-

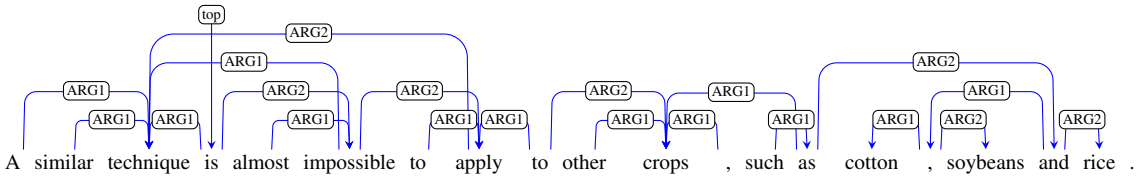
[1] See http://alt.qcri.org/semeval2015/task18/ for further technical details, information on how to obtain the data, and official results.

[2] In much previous SRL work, target representations typically draw on resources like PropBank and NomBank (Palmer et al., 2005; Meyers et al., 2004), which are limited to argument identification and labeling for verbal and nominal predicates. A plethora of semantic phenomena—for example negation

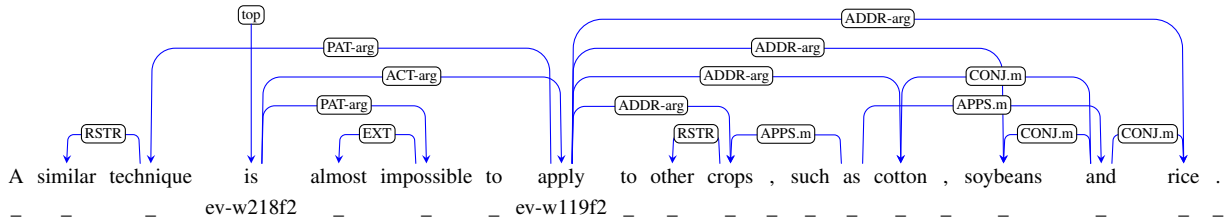
Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015), pages 915–926,
Denver, Colorado, June 4-5, 2015.



(a) DELPH-IN Minimal Recursion Semantics–derived bi-lexical dependencies (DM).



(b) Enju Predicate–Argument Structures (PAS).



(c) Parts of the tectogrammatical layer of the Prague Czech-English Dependency Treebank (PSD).

Figure 1: Sample semantic dependency graphs for Example (1).

sentence' semantic dependencies, i.e. compute a representation that integrates *all* content words in one structure. Finally, a third related area of much interest is often dubbed 'semantic parsing', which Kate and Wong (2010) define as "the task of mapping natural language sentences into complete formal meaning representations which a computer can execute for some domain-specific application." In contrast to much work in this tradition, our SDP target representations aim to be task- and domain-independent.

2 Target Representations

We use three distinct target representations for semantic dependencies. As is evident in our running example (Figure 1), showing what are called the DM, PAS, and PSD semantic dependencies, there are contentful differences among these annotations, and there is of course not one obvious (or even objective) truth. Advancing in-depth comparison of representations and underlying design decisions, in fact, is among the motivations for the SDP task series. Please see Oepen et al. (2014) and Miyao et al. (2014) for additional background.

DM: DELPH-IN MRS-Derived Bi-Lexical Dependencies These semantic dependency graphs originate in a manual re-annotation, dubbed DeepBank, of Sections 00–21 of the WSJ Corpus and of selected parts of the Brown Corpus with syntactico-semantic analyses of the LinGO English Resource Grammar (Flickinger, 2000; Flickinger et al., 2012). For this target representation, top nodes designate the highest-scoping (non-quantifier) predicate in the graph, e.g. the (scopal) adverb *almost* in Figure 1.[3]

PAS: Enju Predicate–Argument Structures The Enju Treebank and parser[4] are derived from the automatic HPSG-style annotation of the PTB (Miyao, 2006). Our PAS semantic dependency graphs are extracted from the Enju Treebank, without contentful conversion, and from the application of the same basic techniques to the Penn Chinese Treebank (CTB;

and other scopal embedding, comparatives, possessives, various types of modification, and even conjunction—often remain unanalyzed in SRL. Thus, its target representations are partial to a degree that can prohibit semantic downstream processing, for example inference-based techniques.

[3] However, non-scopal adverbs act as mere intersective modifiers, e.g. in a structure like *Abrams sang loudly*, the adverb is a predicate in DM, but the main verb nevertheless is the top node.

[4] See `http://kmcs.nii.ac.jp/enju/`.

Xue et al., 2005). Top nodes in this representation denote semantic heads.

PSD: Prague Semantic Dependencies The Prague Czech-English Dependency Treebank (PCEDT; Hajič et al., 2012)[5] is a set of parallel dependency trees over the WSJ texts from the PTB, and their Czech translations. Our PSD bi-lexical dependencies have been extracted from what is called the *tectogrammatical* annotation layer (t-trees). Top nodes are derived from t-tree roots; i.e. they mostly correspond to main verbs. In case of coordinate clauses, there are multiple top nodes per sentence.

3 Data Format

The SDP target representations can be characterized as labeled, directed graphs. Nodes are labeled with five pieces of information: word *form*, *lemma*, *part of speech*, a Boolean flag indicating whether the node represents a *top* predicate, and optional *frame* (or sense) information—for example the distinction between causative vs. inchoative predicates like *increase*. Edges are labeled with semantic relations that hold between source and target.

All data provided for the task uses a column-based file format that extends the format of the SDP 2014 task by a new `frame` column (thus making it a little more SRL-like). More details about the file format are available at the task website.

4 Data Sets

All three target representations for English are annotations of the same text, Sections 00–21 of the WSJ Corpus, as well as of a balanced sample of twenty files from the Brown Corpus (Francis & Kučera, 1982). For this task, we have synchronized these resources at the sentence and tokenization levels and excluded from the SDP 2015 training and testing data any sentences for which (a) one or more of the treebanks lacked a gold-standard analysis; (b) a one-to-one alignment of tokens could not be established across all three representations; or (c) at least one of the graphs was cyclic. Of the 43,746 sentences in these 22 first sections of WSJ text, DeepBank lacks analyses for some 11%, and the Enju Treebank has gaps for a little more than four percent.[6] Finally, 139 of the WSJ graphs obtained through the above conversions were cyclic. In total, we were left with 35,657 sentences (or 802,717 tokens; eight percent more than for SDP 2014[7]) as training data (Sections 00–20), 1,410 in-domain testing sentences (31,948 tokens) from WSJ Section 21, and 1,849 out-of-domain testing sentences (31,583 tokens) from the Brown Corpus.

Besides the additions of out-of-domain test data and frame (or sense) identifiers for English, another extension beyond the SDP 2014 task concerns the inclusion of additional languages, albeit only for select target representations. Our training data included an additional 31,113 Chinese sentences (649,036 tokens), taken from Release 7.0 of the CTB, for the PAS target representation, and 42,076 Czech sentences (985,302 tokens), drawing on the translations of the WSJ Corpus in PCEDT 2.0, for the PSD target representation. Additional out-of-domain Czech test data was drawn from the Prague Dependency Treebank 3.0 (PDT; Bejček et al., 2013). For these additional languages, the task comprised 1,670 sentences (38,397 tokens) of in-domain Chinese test data, and 1,670 sentences (38397 tokens) and 5,226 sentences (87,927 tokens) of in- and out-of-domain Czech data, respectively.

Quantitative Comparison As a first attempt at contrasting our three target representations, Table 1 shows some high-level statistics of the graphs comprising the training and testing data.[8] In terms of distinctions drawn in dependency labels (1), there are clear differences between the representations, with PSD appearing linguistically most fine-

[5] See `http://ufal.mff.cuni.cz/pcedt2.0/`.

[6] Additionally, some 500 sentences show tokenization mismatches, most owing to DeepBank correcting PTB idiosyncrasies like ⟨`G.m.b, H.`⟩, ⟨`S.p, A.`⟩, and ⟨`U.S., .`⟩, and introducing a few new ones (Fares et al., 2013).

[7] In comparison to the SDP 2014 data, our DM graphs were extracted from a newer, improved release of DeepBank (Version 1.1), and its conversion to bi-lexical dependencies was moderately revised to provide more systematic analyses of contracted negated auxiliaries and comparatives. At the same time, the extraction of PSD graphs from the PCEDT t-trees was refined to include edges representing grammatical coreference, e.g. re-entrancies introduced by control verbs.

[8] These statistics are obtained using the 'official' SDP toolkit. Our notions of singletons, roots, re-entrancies, and projectivity follow common graph terminology, but see Oepen et al. (2014) for formal definitions.

		EN i-d			CS i-d	ZH i-d	EN o-o-d			CS o-o-d
		DM	PAS	PSD	PSD	PAS	DM	PAS	PSD	PSD
(1)	**# labels**	59	42	91	61	32	47	41	74	64
(2)	**% singletons**	22.97	4.38	35.76	28.91	0.11	25.40	5.84	39.11	29.04
(3)	**edge density**	0.96	1.02	1.01	1.03	0.98	0.95	1.02	0.99	1.00
(4)	**%$_g$ trees**	2.30	1.22	42.19	37.66	3.49	9.68	2.38	51.43	51.49
(5)	**%$_g$ noncrossing**	69.03	59.57	64.58	63.22	67.61	74.58	65.28	74.26	72.41
(6)	**%$_g$ projective**	2.91	1.64	41.92	38.32	12.89	8.82	3.46	54.35	53.02
(7)	**%$_g$ fragmented**	6.55	0.23	0.69	1.17	15.22	4.71	0.65	1.73	3.50
(8)	**%$_n$ reentrancies**	27.44	29.36	11.42	11.80	24.96	26.14	29.36	11.46	11.44
(9)	**%$_g$ topless**	0.31	0.02	–	0.04	6.92	1.41	–	–	0.02
(10)	**# top nodes**	0.9969	0.9998	1.1276	1.2242	0.9308	0.9859	1.0000	1.2645	1.2771
(11)	**%$_n$ non-top roots**	44.91	55.98	4.35	4.73	46.65	39.89	50.93	5.27	5.31
(12)	**# frames**	297	–	5426	–	–	172	–	1208	–
(13)	**%$_n$ frames**	13.52	–	16.77	–	–	15.79	–	19.50	–
(14)	**average treewidth**	1.30	1.72	1.61	1.66	1.35	1.31	1.69	1.50	1.49
(15)	**maximum treewidth**	3	3	7	6	3	3	3	5	5

Table 1: Contrastive high-level graph statistics across target representations, languages, and domains.

grained, and PAS showing the smallest label inventory. Unattached singleton nodes (2) in our setup correspond to tokens analyzed as semantically vacuous, which (as seen in Figure 1) include most punctuation marks in PSD and DM, but not PAS. Furthermore, PSD (unlike the other two) analyzes some high-frequency determiners as semantically vacuous. Conversely, PAS on average has more edges per (non-singleton) nodes than the other two (3), which likely reflects its approach to the analysis of functional words (see below).

Judging from both the percentage of actual trees (4), the proportions of noncrossing graphs (5), projective graphs (6), and the proportions of reentrant nodes (8), PSD is more 'tree-oriented' than the other two, which at least in part reflects its approach to the analysis of modifiers and determiners (again, see below). We view the small percentages of graphs without at least one top node (9) and of graphs with at least two non-singleton components that are not interconnected (7) as tentative indicators of general well-formedness. Intuitively, there should always be a 'top' predicate, and the whole graph should 'hang together'. Only DM exhibits non-trivial (if small) degrees of topless and fragmented graphs, which may indicate imperfections in DeepBank annotations or room for improvement in the conversion from full logical forms to bi-lexical dependencies, but possibly also exceptions to our intuitions about semantic dependency graphs.

	Directed			**Undirected**		
	DM	PAS	PSD	DM	PAS	PSD
DM	–	.6425	.2612	–	.6719	.5675
PAS	.6688	–	.2963	.6993	–	.5490
PSD	.2636	.2963	–	.5743	.5630	–

Table 2: Pairwise F_1 similarities, including punctuation (upper right diagonals) or not (lower left).

Frame or sense distinctions are a new property in SDP 2015 and currently are only available for the English DM and PSD data. Table 1 reveals a stark difference in granularity: DM limits itself to argument structure distinctions that are grammaticized, e.g. causative vs. inchoative contrasts or differences in the arity or coarse semantic typing of argument frames; PSD, on the other hand, draws on the much richer sense inventory of the EngValLex database (Cinková, 2006). Accordingly, the two target representations represent quite different challenges for the predicate disambiguation sub-task of SDP 2015.

Finally, in Table 2 we seek to quantify pairwise structural similarity between the three representations in terms of unlabeled dependency F_1 (dubbed UF in Section 5 below). We provide four variants of this metric, (a) taking into account the directionality of edges or not and (b) including edges involving punctuation marks or not. On this view, DM and PAS are structurally much closer to each other than either of the two is to PSD, even more so when discarding

punctuation. While relaxing the comparison to ignore edge directionality also increases similarity scores for this pair, the effect is much more pronounced when comparing either to PSD. This suggests that directionality of semantic dependencies is a major source of diversion between DM and PAS on the one hand, and PSD on the other hand.

Linguistic Comparison Among other aspects, Ivanova et al. (2012) categorize a range of syntactic and semantic dependency annotation schemes according to the role that functional elements take. In Figure 1 and the discussion of Table 1 above, we already observed that PAS differs from the other representations in integrating into the graph auxiliaries, the infinitival marker, the case-marking preposition introducing the argument of *apply* (*to*), and most punctuation marks;[9] while these (and other functional elements, e.g. complementizers) are analyzed as semantically vacuous in DM and PSD, they function as predicates in PAS, though do not always serve as 'local' top nodes (i.e. the semantic head of the corresponding sub-graph): For example, the infinitival marker in Figure 1 takes the verb as its argument, but the 'upstairs' predicate *impossible* links directly to the verb, rather than to the infinitival marker as an intermediate.

At the same time, DM and PAS pattern alike in their approach to modifiers, e.g. attributive adjectives, adverbs, and prepositional phrases. Unlike in PSD (or common syntactic dependency schemes), these are analyzed as semantic predicates and, thus, contribute to higher degrees of node reentrancy and non-top (structural) roots. Roughly the same holds for determiners, but here our PSD projection of Prague tectogrammatical trees onto bi-lexical dependencies leaves 'vanilla' articles (like *a* and *the*) as singleton nodes.

The analysis of coordination is distinct in the three representations, as also evident in Figure 1. By design, DM opts for what is often called the Mel'čukian analysis of coordinate structures (Mel'čuk, 1988), with a chain of dependencies rooted at the first conjunct (which is thus considered the head, 'standing in' for the structure at large); in the DM approach, coordinating conjunctions are not integrated with the graph but rather contribute different types of dependencies. In PAS, the final coordinating conjunction is the head of the structure and each coordinating conjunction (or intervening punctuation mark that acts like one) is a two-place predicate, taking left and right conjuncts as its arguments. Conversely, in PSD the last coordinating conjunction takes all conjuncts as its arguments (in case there is no overt conjunction, a punctuation mark is used instead); additional conjunctions or punctuation marks are not connected to the graph.[10]

A linguistic difference between our representations that highlights variable granularities of analysis and, relatedly, diverging views on the scope of the problem can be observed in Figure 2. Much noun phrase–internal structure is not made explicit in the PTB, and the Enju Treebank from which our PAS representation derives predates the bracketing work of Vadas and Curran (2007). In the four-way nominal compounding example of Figure 2, thus, PAS arrives at a strictly left-branching tree, and there is no attempt at interpreting semantic roles among the members of the compound either; PSD, on the other hand, annotates both the *actual* compound-internal bracketing and the assignment of roles, e.g. making *stock* the PAT(ient) of *investment*. In this spirit, the PSD annotations could be directly paraphrased along the lines of *plans by employees for investment in stocks.* In a middle position between the other two, DM disambiguates the bracketing but, by design, merely assigns an underspecified, construction-specific dependency type; its compound dependency, then, is to be interpreted as the most general type of dependency that can hold between the elements of this construction (i.e. to a first approximation either an argument role or a relation parallel to a preposition, as in the above paraphrase). The DM and PSD annotations of this specific example happen to diverge in their bracketing decisions, where the DM analysis corresponds to *[...] investments in stock for employees*, i.e. grouping

[9] In all formats, punctuation marks like dashes, colons, and sometimes commas can be contentful, i.e. at times occur as both predicates, arguments, and top nodes.

[10] As detailed by Miyao et al. (2014), individual conjuncts can be (and usually are) arguments of other predicates, whereas the topmost conjunction only has incoming edges in nested coordinate structures. Similarly, a 'shared' modifier of the coordinate structure as a whole would take as its argument the local top node of the coordination in DM or PAS (i.e. the first conjunct or final conjunction, respectively), whereas it would depend as an argument on all conjuncts in PSD.

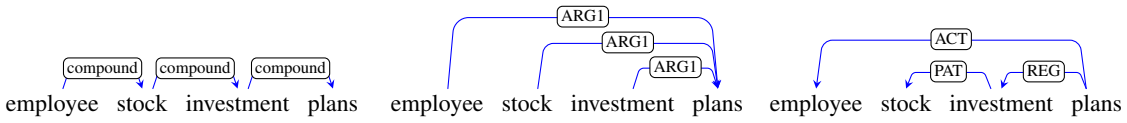


Figure 2: Analysis of nominal compounding in DM, PAS, and PSD, respectively .

the concept *employee stock* (in contrast to 'common stock').

Without context and expert knowledge, these decisions are hard to call, and indeed there has been much previous work seeking to identify and annotate the relations that hold between members of a nominal compound (see Nakov, 2013, for a recent overview). To what degree the bracketing and role disambiguation in this example are determined by the linguistic signal (rather than by context and world knowledge, say) can be debated, and thus the observed differences among our representations in this example relate to the classic contrast between 'sentence' (or 'conventional') meaning, on the one hand, and 'speaker' (or 'occasion') meaning, on the other hand (Quine, 1960; Grice, 1968; Bender et al., 2015). In turn, we acknowledge different plausible points of view about which level of semantic representation should be the target representation for data-driven *parsing* (i.e. structural analysis guided by the grammatical system), and which refinements like the above could be construed as part of a subsequent task of *interpretation*.

5 Task Setup

English training data for the task, providing all columns in the file format sketched in Section 3 above, together with a first version of the SDP toolkit—including graph input, basic statistics, and scoring—were released to candidate participants in early August 2014. In mid-November, cross-lingual training data, a minor update to the English data, and optional syntactic 'companion' analyses (see below) were provided. Anytime between mid-December 2014 and mid-January 2015, participants could request an input-only version of the test data, with just columns (1) to (4) pre-filled; participants then had six days to run their systems on these inputs, fill in columns (5), (6), (7), and upwards, and submit their results (from up to two different runs) for scoring. Upon completion of the testing phase, we have shared the gold-standard test data, official scores, and system results for all submissions with participants and are currently preparing all data for general release through the Linguistic Data Consortium.

Evaluation Systems participating in the task were evaluated based on the accuracy with which they can produce semantic dependency graphs for previously unseen text, measured relative to the gold-standard testing data. For comparability with SDP 2014, the primary measures for this evaluation were labeled and unlabeled precision and recall with respect to predicted dependencies (predicate–role–argument triples) and labeled and unlabeled exact match with respect to complete graphs. In both contexts, identification of the top node(s) of a graph was considered as the identification of additional, 'virtual' dependencies from an artificial root node (at position 0). Below we abbreviate these metrics as (a) labeled precision, recall, and F_1: LP, LR, LF; (b) unlabeled precision, recall, and F_1: UP, UR, UF; and (c) labeled and unlabeled exact match: LM, UM.

The 'official' ranking of participating systems is determined based on the arithmetic mean of the labeled dependency F_1 scores (i.e. the geometric mean of labeled precision and labeled recall) on the three target representations (DM, PAS, and PSD). Thus, to be competitive in the overall ranking, a system had to submit semantic dependencies for all three target representations.

In addition to these metrics, we apply two additional metrics that aim to capture fragments of semantics that are 'larger' than individual dependencies but 'smaller' than the semantic dependency graph for the complete sentence, viz. what we call (a) *complete predications* and (b) *semantic frames*. A complete predication is comprised of the set of all core arguments to one predicate, which for the DM and PAS target representations corresponds to all outgoing dependency edges, and for the PSD target representation to only those outgoing dependencies marked by an '`-arg`' suffix on the edge label. Pushing the units of evaluation one step further towards inter-

		DM				PAS				PSD			
	LF	LF	LP	LR	FF	LF	LP	LR	PF	LF	LP	LR	FF
Turku✧	86.81	88.29	89.52	87.09	58.39	95.58	95.94	95.21	87.99	76.57	78.24	74.97	56.85
Lisbon*	86.23	89.44	90.52	88.39	00.20	91.67	92.45	90.90	84.18	77.58	79.88	75.41	00.06
Peking	*85.33*	*89.09*	90.93	87.32	63.08	*91.26*	92.90	89.67	79.08	75.66	78.60	72.93	49.95
Lisbon	85.15	88.21	89.84	86.64	00.15	90.88	91.87	89.92	81.74	*76.36*	78.62	74.23	00.03
Riga	84.00	87.90	88.57	87.24	58.12	90.75	91.50	90.02	80.03	73.34	75.25	71.52	52.54
Turku*	83.47	86.17	87.80	84.60	54.67	90.62	91.38	89.87	80.60	73.63	76.10	71.32	53.20
Minsk	80.74	84.13	86.28	82.09	54.24	85.24	87.28	83.28	64.66	72.84	74.65	71.13	51.63
In-House*	61.61	92.80	92.85	92.75	83.79	92.03	92.07	91.99	87.24	–	–	–	–

		DM				PAS				PSD			
	LF	LF	LP	LR	FF	LF	LP	LR	PF	LF	LP	LR	FF
Turku✧	83.50	82.11	84.26	80.07	42.89	92.92	93.52	92.33	83.80	75.47	77.77	73.31	42.37
Lisbon*	82.53	83.77	85.79	81.84	00.35	87.63	88.88	86.41	80.19	76.18	80.12	72.61	02.25
Lisbon	*81.15*	81.75	84.81	78.90	00.27	86.88	88.52	85.30	78.47	*74.82*	78.68	71.31	02.09
Peking	80.78	*81.84*	84.29	79.53	47.49	*87.23*	89.47	85.10	74.75	73.28	77.36	69.61	34.28
Riga	79.23	80.69	81.69	79.72	41.88	86.63	87.56	85.72	76.26	70.37	73.23	67.71	40.76
Turku*	78.85	79.01	81.54	76.63	39.15	85.95	86.95	84.98	76.38	71.59	74.92	68.55	38.75
Minsk	75.79	77.24	80.24	74.46	42.18	80.44	83.07	77.96	62.00	69.68	72.26	67.27	41.25
In-House*	59.24	89.69	89.80	89.58	76.39	88.03	88.10	87.96	81.69	–	–	–	–

Table 3: Results of the gold track (marked ✧), open track (marked *) and closed track (unmarked) submissions for the English in-domain (top) and out-of-domain (bottom) data. For each system, the second column (LF) indicates the averaged LF score across all representations, used to rank the systems. The best *closed track* scores are highlighted in italices.

	LF	LP	LR	PF
Peking	*83.43*	84.75	82.15	66.09
Riga	82.47	83.12	81.84	66.05
Lisbon	82.02	83.81	80.31	66.05
Turku*	79.64	80.81	78.51	62.04
Minsk	77.68	79.27	76.15	58.23

	LF	LP	LR	PF
Lisbon	*79.33*	83.52	75.54	55.91
Peking	78.45	83.61	73.89	55.36
Riga	75.34	78.77	72.19	50.90
Turku*	75.30	77.53	73.20	54.26

	LF	LP	LR	PF
Peking	*64.37*	69.41	60.02	48.82
Turku*	63.70	65.11	62.35	51.04
Lisbon	63.50	67.94	59.61	43.10
Riga	61.32	64.50	58.44	44.34

Table 4: Results of the open (Turku) and closed (other teams) tracks for the Chinese in-domain (left) and Czech in- (center) and out-of-domain (right) data. The systems are ranked according to their LF scores.

pretation, a semantic frame is comprised of a complete predication combined with the frame (or sense) identifier of its predicate. Both complete-predicate and semantic-frame evaluation are restricted to predicates corresponding to verbal parts of speech (as determined by the gold-standard part of speech), and semantic frames are further restricted to those target representations for which frame or sense information is available in our data (English DM and PSD). As with the other metrics, we score precision, recall, and F_1, which we abbreviate as PP, PR, and PF for complete predications, and FP, FR, and FF for semantic frames.

Closed vs. Open vs. Gold Tracks Much like in 2014, the task distinguished a *closed* track and an *open* track, where systems in the closed track could only be trained on the gold-standard semantic dependencies distributed for the task. Systems in the open track, on the other hand, could use additional resources, such as a syntactic parser, for example—provided that they make sure to not use any tools or resources that encompass knowledge of the gold-standard syntactic or semantic analyses of

the SDP 2015 test data.[11] To simplify participation in the open track, the organizers prepared ready-to-use 'companion' syntactic analyses, sentence- and token-aligned to the SDP data, in the form of Stanford Basic syntactic dependencies (de Marneffe et al., 2006) produced by the parser of Bohnet and Nivre (2012).

Finally, to more directly gauge the the contributions of syntactic structure on the semantic dependency parsing problem, an idealized *gold* track was introduced in SDP 2015. For this track, gold-standard syntactic companion files were provided in a varity of formats, viz. (a) Stanford Basic dependencies, derived from the PTB, (b) HPSG syntactic dependencies in the form called DM by Ivanova et al. (2012), derived from DeepBank, and (c) HPSG syntactic dependencies derived from the Enju Treebank.

6 Submissions and Results

From almost 40 teams who had registered for the task, twelve teams obtained the test data, and test runs were submitted for six systems—including one 'inofficial' submission by a sub-set of the task organizers (Miyao et al., 2014). Each team submitted up to two test runs per track. In total, there were seven runs submitted to the English closed track, five to the open track and two to the gold track; seven runs were submitted to the Chinese closed track, two to the open track; and five runs submitted to the Czech closed track, two to the open track. One team submitted only to the open and gold tracks, three teams submitted only to the closed track, one team submitted to open and closed tracks in English but only to the closed tracks in the other two languages. The main results are summarized and ranked in Tables 3 and 4. The ranking is based on the average $\overline{\text{LF}}$ score across all three target representations. Besides LF, LP and LR we also indicate the F_1 score of prediction of semantic frames (FF), or, where frame (or sense) identifiers are not available, of complete predications (PF). In cases where a team submitted two runs to a track, only the highest-ranked score is included in the table.

In the English closed track, the average LF scores across target representations range from 85.33 to 80.74. Comparing the results for different target representations, the average LF scores across systems are 89.13 for PAS, 87.09 for DM, and 74.24 for PSD. The scores for semantic frames show a much larger variation across representations and systems.[12]

The Lisbon team is the only one that submitted to both the open and the closed tracks; with the additional resources allowed in the open track, they were able to improve over all closed-track submissions. Similarly, the perfect Stanford dependencies in the gold track helped the Turku team a lot in PAS and somewhat in DM and PSD; interestingly, they did not obtain the best results in the latter two representations, but their cross-representation average was still the best. The In-House system is ranked low because its submission was incomplete (no of-the-shelf parser for PSD being available); however, for DM and PAS they yielded the best open-track scores.

We see very similar trends for the out-of-domain data, though the scores are a few points lower.

Chinese PAS seems to be more difficult than English (cross-system average LF being 81.05, as opposed to English 90.07). The Czech and English in-domain data are actually parallel translations and the Czech PSD average LF is slightly higher (77.11, as opposed to English 74.90). The Turku open-track system shined in the Czech out-of-domain data, presumably because the additional dependency parser they used was trained on data from the target domain.

7 Overview of Approaches

Table 5 shows a summary of the tracks in which each submitted system participated, and Table 6 shows an overview of approaches and additionally used resources. All the teams except In-House submitted results for cross-lingual data (Czech and Chinese). Teams except Lisbon also tackled with predicate disambiguation. Only Turku participated in the Gold track.

The submitted teams explored a variety of approaches. Riga and Peking relied on the graph-to-tree transformation of Du et al. (2014) as a basis. This method converts semantic dependency graphs into tree structures. Training data of semantic dependency

[11] This restriction implies that typical off-the-shelf syntactic parsers have to be re-trained, as many data-driven parsers for English include WSJ Section 21 in their default training data.

[12] Please see the task web page at the address indicated above for full labeled and unlabeled scores.

Team	Closed	Open	Cross-Lingual	Predicate Disambiguation	Gold
In-House		✓		✓	
Lisbon	✓	✓	✓		
Minsk	✓		✓	✓	
Peking	✓		✓	✓	
Riga	✓		✓	✓	
Turku		✓	✓	✓	✓

Table 5: Summary of tracks in which submitted systems participated

Team	Approach	Resources
In-House	grammar-based parsing (Miyao et al., 2014)	ERG & Enju
Lisbon	graph parsing with dual decomposition (Martins & Almeida, 2014)	companion
Minsk	transition-based dependency graph parsing in the spirit of Titov et al. (2009)	—
Peking	(Du et al., 2014) extended with weighted tree approximation, parser ensemble	—
Riga	(Du et al., 2014)'s graph-to-tree transformation, Mate, C6.0, parser ensemble	—
Turku	sequence labeling for argument detection for each predicate, SVM classifiers for top node recognition and sense prediction	companion

Table 6: Overview of approaches and additional resources used (if any).

graphs are converted into tree structures, and well-established parsing methods for tree structures are applied to converted structures. In run-time, the tree parser is applied, and predicted trees are converted back into graph structures. Labels of tree edges encode additional information to recover original graph structures. This idea was applied in Du et al. (2014) and contributed to their best-performing system in the 2014 SDP task.

In addition to applying the Mate parser to the tree-transformed data of Du et al. (2014), Riga developed a high-precision but low-recall semantic parser. This method applies a decision tree classifier (C6.0) to edge detection. C6.0 learns patterns of semantic dependencies, which means it outputs highly reliable prediction when a learned pattern applies, while in most cases it cannot produce any predictions. These two types of parsers are finally combined by parser ensemble. They also applied C6.0 to frame (or sense) label prediction for DM and PSD. Graph parsing and frame prediction are performed independently.

Peking proposed a novel method for graph-to-tree transformation, namely weighted tree approximation. The intuition behind this method is that the core part of graph-to-tree transformation is the extraction of an essential tree-forming subset of edges from semantic dependency graphs, but it is not trivial to determine a reasonable subset. Therefore, the idea of weighted tree approximation is to define an edge score to quantify importance of each edge, and extract tree-forming edges that maximizes the sum of edge scores globally. After defining edge scores, tree-forming edges with optimal scores can be extracted by applying decoding methods like maximum spanning tree and the Eisner algorithm. They applied this method as well as the previous method proposed in Du et al. (2014) with several variations on encoding edge labels, finally obtaining nine tree parsers. In the final submission, outputs from these parsers are combined by the parser ensemble technique. For predicate disambiguation, they independently applied a sequence labeling technique.

Turku took a completely different approach. They consider each predicate separately, and apply sequence labeling for each predicate individually, to recognize arguments of the target predicate. That is, the task is reduced to assign each word an argument tag (e.g. ARG1) or a negative 'pseudo-'label indicating it is not an argument of the target predicate. Outputs from sequence labeling for each predicate are combined to derive final semantic dependencies. Top node recognition and frame label prediction are performed separately. Turku is the only team who participated in the Gold track; they used gold syntactic dependencies as features for sequence labeling.

Lisbon and In-House applied their parsers from

SDP 2014 without substantive changes. The Lisbon parser (*TurboSemanticParser*) computes globally optimal semantic dependencies using rich second order features on semantic dependencies, such as siblings and grand parents. This optimization is impractical in general, but they achieve tractable parsing time by applying dual decomposition. In-House uses deep parsers with specifically developed linguistically motivated grammars, namely the LinGO English Resource Grammar and the Enju grammar. As described in Section 2, these same grammars were used for deriving the training and test data sets of this task, i.e. these components of the In-House ensemble exclusively support the DM and PAS target representations, respectively.

Peking and Lisbon tend to attain high scores in their participated tracks in LF. Riga ranked third in LF in the closed tracks (both in-domain and out-of-domain), while it achieved higher scores than others in FF. This might be due to high-precision rules obtained by their model, although this does not apply in the cross-lingual track. The Turku results in the gold track achieved considerably higher scores, which indicate that better syntactic parsing will help improve semantic dependency parsing.[13] It is difficult to describe a tendency in the out-of-domain track; all the systems scree three to five points lower scores than the in-domain track, indicating that domain variation is still a significant challenge in semantic dependency parsing.

8 Conclusion

We have described the motivation, design, and outcomes of the SDP 2015 task on semantic dependency parsing, i.e. retrieving bi-lexical predicate–argument relations between all content words within an English sentence. We have converted to a common format three existing annotations (DM, PAS, and PSD) over the same text and have put this to use in training and testing data-driven semantic dependency parsers. In contrast to SDP 2014 the task was extended by cross-domain testing and evaluation at the level of 'complete' predications and semantic frame (or sense) disambiguation. Furthermore, we provided comparable annotations of Czech and Chinese texts to enable cross-linguistic comparison. To start further probing of the role of syntax in the recovery of predicate–argument relations, we added a third (idealized) 'gold' track, where syntactic dependencies are provided directly from available syntactic annotations of the underlying treebanks.

[13] The SDP 2014 and 2015 task setups, however, somewhat artificially constrain the possible contributions of syntactic analysis, as all training and testing data (even in the closed track) includes high-quality parts of speech and lemmata.

Acknowledgements

We are grateful to Angelina Ivanova for help in DM data preparation and contrastive analysis, to Željko Agić and Bernd Bohnet for consultation and assistance in preparing our companion parses, to the Linguistic Data Consortium (LDC) for support in distributing the SDP data to participants, as well as to Emily M. Bender and two anonymous reviewers for feedback on an earlier version of this manuscript. We warmly thank the general SemEval 2015 chairs, Preslav Nakov and Torsten Zesch, for always being role-model organizers, equipped with an outstanding balance of structure, flexibility, and community spirit. Data preparation was supported through the ABEL high-performance computing facilities at the University of Oslo, and we acknowledge the Scientific Computing staff at UiO, the Norwegian Metacenter for Computational Science, and the Norwegian taxpayers. Part of the work was supported by the grants 15-10472S, GP13-03351P and 15-20031S of the Czech Science Foundation, and by the infrastructural funding by the Ministry of Education, Youth and Sports of the Czech Republic (LM2010013).

References

Bejček, E., Hajičová, E., Hajič, J., Jínová, P., Kettnerová, V., Kolářová, V., ... Zikánová, Š. (2013). *Prague dependency treebank 3.0.* Retrieved from `http://hdl.handle.net/11858/00-097C-0000-0023-1AAF-3`

Bender, E. M., Flickinger, D., Oepen, S., Packard, W., & Copestake, A. (2015). Layers of interpretation. On grammar and compositionality. In *Proceedings of the 11th International Conference on Computational Semantics.* London, UK.

Bohnet, B., & Nivre, J. (2012). A transition-based system for joint part-of-speech tagging and labeled non-projective dependency parsing. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Conference on Natural Language Learning* (p. 1455 – 1465). Jeju Island, Korea.

Cinková, S. (2006). From PropBank to EngValLex. Adapting the PropBank lexicon to the valency theory of the Functional Generative Description. In *Proceedings of the 5th International Conference on Language Resources and Evaluation.* Genoa, Italy.

de Marneffe, M.-C., MacCartney, B., & Manning, C. D. (2006). Generating typed dependency parses from phrase structure parses. In *Proceedings of the 5th International Conference on Language Resources and Evaluation* (p. 449 – 454). Genoa, Italy.

Du, Y., Zhang, F., Sun, W., & Wan, X. (2014). Peking: Profiling syntactic tree parsing techniques for semantic graph parsing. In *Proceedings of the 8th international workshop on semantic evaluation (semeval 2014).*

Fares, M., Oepen, S., & Zhang, Y. (2013). Machine learning for high-quality tokenization. Replicating variable tokenization schemes. In *Computational linguistics and intelligent text processing* (p. 231 – 244). Springer.

Flickinger, D. (2000). On building a more efficient grammar by exploiting types. *Natural Language Engineering*, *6 (1)*, 15 – 28.

Flickinger, D., Zhang, Y., & Kordoni, V. (2012). DeepBank. A dynamically annotated treebank of the Wall Street Journal. In *Proceedings of the 11th International Workshop on Treebanks and Linguistic Theories* (p. 85 – 96). Lisbon, Portugal: Edições Colibri.

Francis, W. N., & Kučera, H. (1982). *Frequency analysis of English usage. Lexicon and grammar*. New York, USA: Houghton Mifflin.

Gildea, D., & Jurafsky, D. (2002). Automatic labeling of semantic roles. *Computational Linguistics*, *28*, 245 – 288.

Grice, H. P. (1968). Utterer's meaning, sentence-meaning, and word-meaning. *Foundations of Language*, *4*(3), 225 – 242.

Hajič, J., Hajičová, E., Panevová, J., Sgall, P., Bojar, O., Cinková, S., ... Žabokrtský, Z. (2012). Announcing Prague Czech-English Dependency Treebank 2.0. In *Proceedings of the 8th International Conference on Language Resources and Evaluation* (p. 3153 – 3160). Istanbul, Turkey.

Ivanova, A., Oepen, S., Øvrelid, L., & Flickinger, D. (2012). Who did what to whom? A contrastive study of syntacto-semantic dependencies. In *Proceedings of the Sixth Linguistic Annotation Workshop* (p. 2 – 11). Jeju, Republic of Korea.

Kate, R. J., & Wong, Y. W. (2010). Semantic parsing. The task, the state of the art and the future. In *Tutorial abstracts of the 20th Meeting of the Association for Computational Linguistics* (p. 6). Uppsala, Sweden.

Marcus, M., Santorini, B., & Marcinkiewicz, M. A. (1993). Building a large annotated corpora of English. The Penn Treebank. *Computational Linguistics*, *19*, 313 – 330.

Martins, A. F. T., & Almeida, M. S. C. (2014). Priberam: A turbo semantic parser with second order features. In *In proceedings of the 8th international workshop on semantic evaluation (semeval 2014).*

Mel'čuk, I. (1988). *Dependency syntax. Theory and practice*. Albany, NY, USA: SUNY Press.

Meyers, A., Reeves, R., Macleod, C., Szekely, R., Zielinska, V., Young, B., & Grishman, R. (2004). Annotating noun argument structure for NomBank. In *Proceedings of the 4th International Conference on Language Resources and Evaluation* (p. 803 – 806). Lisbon, Portugal.

Miyao, Y. (2006). *From linguistic theory to syntactic analysis. Corpus-oriented grammar development and feature forest model*. Unpublished doctoral dissertation, University of Tokyo, Tokyo, Japan.

Miyao, Y., Oepen, S., & Zeman, D. (2014). In-House. An ensemble of pre-existing off-the-shelf parsers. In *Proceedings of the 8th International Workshop on Semantic*

Evaluation (p. 63 – 72). Dublin, Ireland.

Nakov, P. (2013). On the interpretation of noun compounds: Syntax, semantics, and entailment. *Natural Language Engineering*, *19*(3), 291 – 330.

Oepen, S., Kuhlmann, M., Miyao, Y., Zeman, D., Flickinger, D., Hajič, J., ... Zhang, Y. (2014). SemEval 2014 Task 8. Broad-coverage semantic dependency parsing. In *Proceedings of the 8th International Workshop on Semantic Evaluation.* Dublin, Ireland.

Palmer, M., Gildea, D., & Kingsbury, P. (2005). The Proposition Bank. A corpus annotated with semantic roles. *Computational Linguistics*, *31*(1), 71 – 106.

Quine, W. V. O. (1960). *Word and object*. Cambridge, MA, USA: MIT Press.

Titov, I., Henderson, J., Merlo, P., & Musillo, G. (2009). Online graph planarisation for synchronous parsing of semantic and syntactic dependencies. In *Proceedings of the 21st International Joint Conference on Artificial Intelligence.* Pasadena, CA, USA.

Vadas, D., & Curran, J. (2007). Adding Noun Phrase Structure to the Penn Treebank. In *Proceedings of the 45th Meeting of the Association for Computational Linguistics* (p. 240 – 247). Prague, Czech Republic.

Xue, N., Xia, F., Chiou, F.-D., & Palmer, M. (2005). The Penn Chinese TreeBank. Phrase structure annotation of a large corpus. *Natural Language Engineering*, *11*, 207 – 238.