Using predicted semantic roles for DRS parsing

Tatiana Bladier[1], Gosse Minnema[2], Rik van Noord[2], Kilian Evang[1]

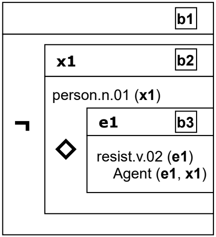
[1] Heinrich Heine University Düsseldorf
[2] University of Groningen

CSTFRS Workshop 2021
July 27, 2021

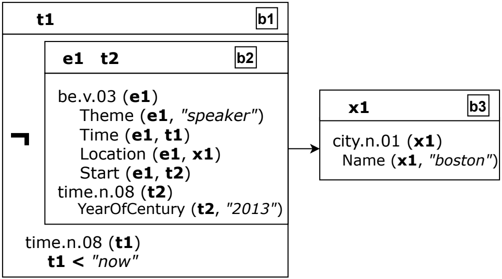
Discourse Representation Structures

- Meaning representations based on DRT, so contain scope!
- Goal: give structured meaning representation of input text

No one can resist.



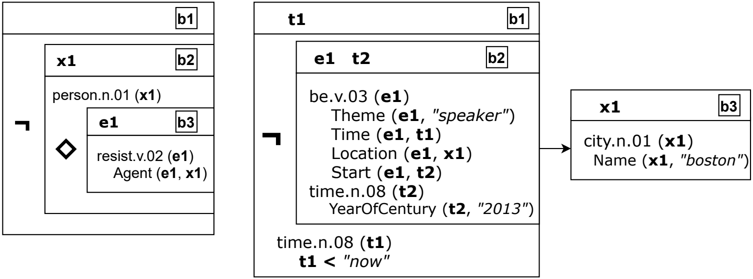
I haven't been to Boston since 2013.



DRS Parsing

- Main task: input text to DRS
- Lots of subtasks: Word Sense Disambiguation, Semantic Role Labeling, Named Entity Recognition, Negation Detection, Coreference Resolution, Presupposition Detection and many more

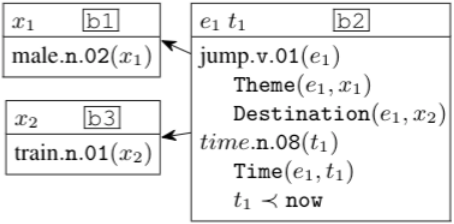
No one can resist. **I haven't been to Boston since 2013.**



DRS parsing: clause format

He jumped onto the train.

b1 REF x1
b1 PRESUPPOSITION b2
b1 male "n.02" x1
b2 REF e1
b2 REF t1
b2 TPR t1 "now"
b2 Theme e1 x1
b2 Time e1 t1
b2 jump "v.01" e1
b2 time "n.08" t1
b2 Destination e1 x2
b3 REF x2
b3 PRESUPPOSITION b2
b3 train "n.01" x2



Core Idea

- Current DRS parsers are trained end-to-end
- There exists a high quality SRL parser (He et al., 2018)
- Can we use this parser to improve DRS output?

Challenges

- DRSs do not follow standard SRL format

Token	*He*	*jumped*	*into*	*the*	*train*
PMB		Theme	Destination		
SRL: head	Theme	PRED			Destination
SRL: span	Theme	PRED	{ ←	Destination	→ }

Table 1: PMB-style versus standard SRL annotations.

- Semantic roles are carried by predicates instead of by arguments
- Prepositional and adverbial roles (e.g. *into the train*, *slowly*) are carried by the preposition or adverb itself, instead of by the verbal predicate they are associated to.

DRS-based conversion

```
b1 REF x1                  % He [0...2]
b1 PRESUPPOSITION b2       % He [0...2]
b1 male "n.02" x1          % He [0...2]
b2 REF e1                  % jumped [3...9]
b2 REF t1                  % jumped [3...9]
b2 TPR t1 "now"            % jumped [3...9]
b2 Theme e1 x1             % jumped [3...9]
b2 Time e1 t1              % jumped [3...9]
b2 jump "v.01" e1          % jumped [3...9]
b2 time "n.08" t1          % jumped [3...9]
b2 Destination e1 x2       % into [10...14]
b3 REF x2                  % the [15...18]
b3 PRESUPPOSITION b2       % the [15...18]
b3 train "n.01" x2         % train [19...24]
                           % . [24...25]
```

1) find predicate ("jumped")

2) find role filler (Theme → *x1*)

3) find introduction of filler (*x1* → "He")

4) result: predicate = "jumped", Theme = "he"

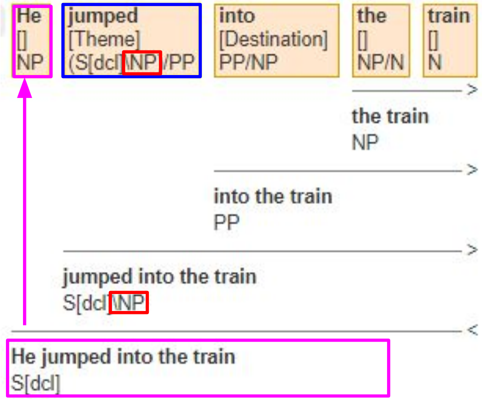
CCG-based conversion

Idea:

- Match semantic roles with syntactic roles from CCG type signature
- Track these roles through the derivation tree until they are resolved
- Find the spans/heads corresponding to the roles

CCG-based conversion

Example:



1) find predicate ("jumped")

2) find & trace CCG argument for role (*Theme* → "\NP")

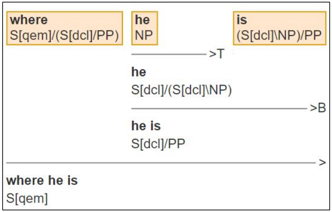
3) find derivation step where role is resolved

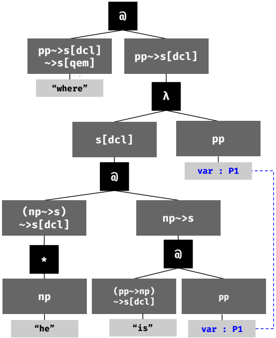
4) result: predicate = "jumped", Theme = "he"

CCG-based conversion

Some more details:

- Simplify the CCG trees with `LangPro` (Abzianidze, EMNLP 2017)
  - Remove directionality
  - Composition becomes λ-abstraction





CCG-based conversion

Some more details:

- Simplify the CCG trees with `LangPro` (Abzianidze, EMNLP 2017)
  - Remove directionality
  - Composition becomes λ-abstraction
- Separate procedures for arguments and PPs/adjuncts
  - PP: syntactic role on verb, semantic role on P head:
    jumped : $PP \rightarrow NP \rightarrow S$, *into* : $NP \rightarrow PP$
    He jumped$_{[Theme]}$ *into*$_{[Destination]}$ *the train*
  - Adverbial roles on ADV:
    quickly : $(NP \rightarrow S) \rightarrow (NP \rightarrow) S$
    She ran$_{[Theme]}$ *quickly*$_{[Manner]}$

Comparison of both conversion algorithms

- 68% of docs match exactly, and 82% differ by at most one role.
- Mismatches between syntax and semantics.
- CCG-based conversion gives more intuitive results on co-referential NPs, e.g. *she handed* him_1 *the money that she owed* him_2.
- CCG-based conversion allows for a better resolution of hearer and speaker: *I don't remember your name*.
- CCG-based conversion has difficulties with light verb constructions where the semantics of the main verb and the light verb interact: *he had his wallet stolen*.
- Semantic and syntactic head, e.g. *all of the town*, *a kilo of plums*.

SRL Predictions

- Graph-based end-to-end coreference resolution system by He et al. (2018)
- This syntax-agnostic SRL model jointly predicts predicates, role fillers, and role labels.

- Train, dev, test split: 6620, 885, and 898 documents (PMB 3.0.0)

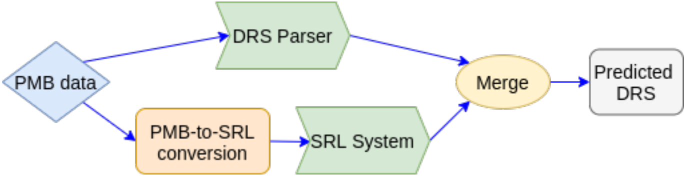
Conversion algorithm	F1 labeled SRL	F1 predicate identification
DRS + GloVE	81.64	97.69
DRS + ELMo	86.27	98.91
CCG + GloVE	82.89	97.43
CCG + ELMo	**86.88**	**99.14**

Table 2: SRL results on the PMB test set.

Baseline Parsers

- **E19**: lexically anchored, transition-based (Evang, 2019)
- **N18**: seq2seq (van Noord et al., 2018)
- **N19**: + linguistic features (van Noord et al., 2019)
- **N20**: + BERT and character representations (van Noord et al., 2020)

Merging System Outputs



- Combine predictions of parser and SRL system
- Idea: match SRL predictions to clauses in parser output. If the role label is different, change the clause.

Token-based Merging

Parser prediction:

```
b1 REF x1                  % He
b1 PRESUPPOSITION b2       % He
b1 male "n.02" x1          % He
b2 REF e1                  % jumped
b2 Agent e1 x1             % jumped
b2 jump "v.01" e1          % jumped
b2 Destination e1 x2       % into
b3 REF x2                  % the
b3 PRESUPPOSITION b2       % the
b3 train "n.01" x2         % train
```

SRL prediction:

$\langle\langle$ jumped ,Theme, He $\rangle$, $\langle\langle$ jumped , *Destination*, train $\rangle\rangle$

Concept-based Merging

- Token-based merging only works for anchored DRS parsers
- Seq2seq output is not aligned to input tokens
- For N18, N19 and N20 we recover the anchoring heuristically
- Create dictionary of token-concept alignments from training set

  dic (jump) = ⟨ jump, jumps, jumped ⟩
  dic (male) = ⟨ he, him, his, Tom, John, Napoleon, ... ⟩
  dic (large) = ⟨ large, largest ⟩
  dic (conductor) = ⟨ conductor ⟩
  ...

- Lemmatize tokens using SpaCy

Concept-based Merging

Parser prediction:

```
b1 REF x1
b1 PRESUPPOSITION b2
b1 male "n.02" x1          % ⟨ he, him, his, Tom, ... ⟩
b2 REF e1
b2 Agent e1 x1
b2 jump "v.01" e1          % ⟨ jump, jumps, jumped ⟩
b2 Destination e1 x2
b3 REF x2
b3 PRESUPPOSITION b2
b3 train "n.01" x2         % ⟨ train, trains, trained ⟩
```

SRL prediction:

⟨⟨jumped,Theme,He⟩, ⟨⟨jumped, *Destination*, train⟩⟩

Restrictions

- Heuristic restrictions to role replacements to prevent some types of false matches:
  1. Do not replace the special roles `Time` and `Name`.
  2. Do not replace with roles that were predicted by the SRL system with $< .5$ precision on the dev data.
  3. Do not do substitutions that would lead to duplicate roles.
  4. In concept-based merging, do not match with the special concepts `person`, `be`, and `entity`.

Results

System	SRL	E19	E19	N18	N19	N20
Merging	-	tok	con	con	con	con
Baseline	-	81.4	81.4	84.9	88.7	89.3
DRS + GloVE	81.6	+0.3	+0.2	+0.4	+0.2	-0.1
DRS + ELMo	86.3	**+0.4**	**+0.4**	**+0.5**	**+0.3**	+0.0
CCG + GloVE	83.0	+0.3	+0.2	+0.4	+0.1	-0.1
CCG + ELMo	**87.0**	**+0.4**	+0.3	+0.4	+0.2	+0.0
DRS bound	100	+ 1.3	+1.2	+1.2	+1.1	+0.7
CCG bound	100	+ 1.2	+1.1	+1.2	+1.1	+0.8

Conclusions

- Our approach is useful especially with parsers such as E19 which do not reach state-of-the-art accuracy but may have other advantages such as smaller models or lexical anchoring.
- Flexibility: our approach can be applied on top of any DRS parsing model without having to alter or retrain the model itself.

Future Work

- Experiment with prediction of nominal and adjectival predicates along with their semantic roles.
- Reconstruct and predict full spans of semantic roles.
- Carry out parsing experiments with further languages in the PMB, including Dutch, German, and Italian.
- Improve the SRL predictions by enforcing coherence of predicted predicates and corresponding semantic roles.

References I

Evang, K. (2019). Transition-based DRS parsing using stack-LSTMs. In *Proceedings of the IWCS Shared Task on Semantic Parsing*, Gothenburg, Sweden. Association for Computational Linguistics.

He, L., Lee, K., Levy, O., and Zettlemoyer, L. (2018). Jointly predicting predicates and arguments in neural semantic role labeling. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 364–369, Melbourne, Australia. Association for Computational Linguistics.

van Noord, R., Abzianidze, L., Toral, A., and Bos, J. (2018). Exploring neural methods for parsing discourse representation structures. *Transactions of the Association for Computational Linguistics*, 6:619–633.

van Noord, R., Toral, A., and Bos, J. (2019). Linguistic information in neural semantic parsing with multiple encoders. In *Proceedings of the 13th International Conference on Computational Semantics - Short Papers*, pages 24–31, Gothenburg, Sweden. Association for Computational Linguistics.

van Noord, R., Toral, A., and Bos, J. (2020). Character-level representations improve DRS-based semantic parsing even in the age of BERT. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4587–4603, Online. Association for Computational Linguistics.