Constrained decoding for text-level discourse parsing

by Muller, Afantenos, Denis, Asher

Max Depenbrock

Project Seminar "Discourse Parsing"

13.11.2017

Table of Contents

Introduction

Main Points of the Paper

- Discourse structure on text-level
- Informed search
- Generating discourse structures with respect to constraints

Text-level discourse structure prediction

Most work so far

- Relation labeling only between 2 DUs
- Structure prediction only at sentence level

Informed search

Last week

- HILDA: Greedy DFS
- Solution might not be optimal

In this paper

- Informed search: A*
- Guaranteed to find optimal solution...
- ...given a good heuristic

Informed search (cont'd)

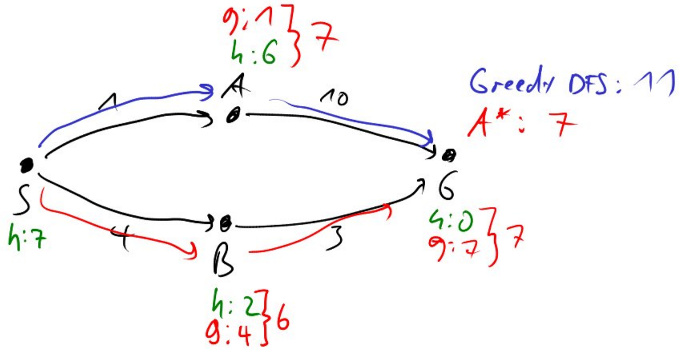


Figure: Greedy search vs. informed search

The A* Algorithm

```
Input: Initial state S_0
queue ← {S_0};
while queue is not empty do
    current ← removeBest(queue);
    if current is solution then
        return current
    else
        newStates ← generate(current) ;
        queue = queue ∪ newStates
    end
end
```

Algorithm 1: Structure of A*

Constraints

Right Frontier Constraint (RFC)

- New DUs can only be attached to
  - the current node
  - nodes which the current node is subordinated to

- Difficult to integrate in current solutions
- A* makes integration easy

Constraints (cont'd)

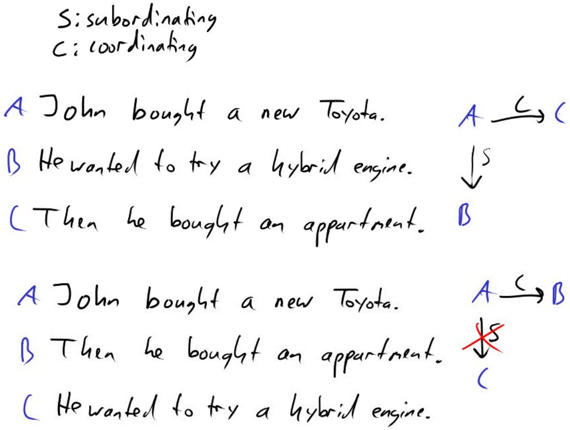


Figure: Illustration of RFC

Corpus

Available Corpora

PTDB

- Very simple structure
- only adjacent DUs

RST-DT

- Recursive but enforcing adjacency

DISCOR / ANNODIS

- SDRT
- directed, acyclic graphs, imposing RFC

GraphBank

- general graphs

A closer look at ANNODIS

- Created in 2012
- French news and Wikipedia articles
- Annotated in SDRT

Statistics

- 18 types of relations
- 86 texts
- 3188 EDUs
- 1395 CDUs
- 3355 relations

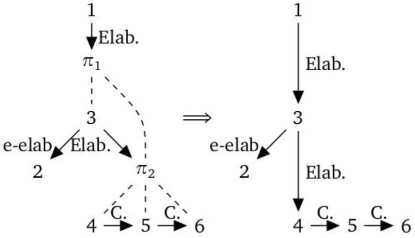
From SDRT to dependency graphs

SDRT is too complex.

Idea

- Collapse CDUs
- Replace them by their *recursive heads*
- Recursive head: highest DU in subgraph
- If that is a CDU: recursion

[Principes de la sélection naturelle.]_1 [La théorie de la sélection naturelle [telle qu'elle a été initialement décrite par Charles Darwin,]_2 repose sur trois principes:]_3 [1. le principe de variation]_4 [2. le principe d'adaptation]_5 [3. le principe d'hérédité]_6



Local Models

The basic Idea

2 locally trained probabilistic classifiers

- one to predict attachment site
- one to predict discourse relation for attached pairs

2 different training models

- Naive Bayes
- Logistic Regression / MaxEnt

The 2 classifiers will be combined during decoding phase

Features (selection)

attachment and labeling

- First EDU of the paragraph / sentence?
- number of tokens in EDU
- number of EDUs between source and target
- source embedded in target? (and vice versa)

attachment

- presence of discourse marker
- embedded in other EDU?

labeling

- presence of verb
- presence of negation
- tense agreement between head verbs of source / target

Evaluation

2 sets of relations

- 18 relations (full set)
- 4 relation groups (coarse grain)

2 windows for relations to be attached

- all relations (full)
- relations with distance 5 (w5)

Evaluation (cont'd)

	MaxEnt	NB	Majority
w5 (18 relations)	**44.8**	34.7	19.1
full (18 relations)	43.3	32.9	19.7
w5 (4 relations)	**65.5**	62.1	51.2
full (4 relations)	63.6	60.1	50.1

	MaxEnt	NB
w5	**67.4**	61.1
full	63.5	51.3

Figure: labeling (accuracy, left) and attachment (F1 score, right)

- MaxEnt best model for both cases
- resampling increases performance

Parsing Experiments

Baselines

Baselines

Last

- Always attach current DU to previous one

Greedy

- Local greedy approach
- For all pairs of adjacent DUs select the one with highest prob.

A*

Cost of Edges

$$cost((u,v)) =$$
$$-\log(P(attach(u,v) = True) \cdot \arg\max_R P(R|attach(u,v) = True))$$

- vertices u, v, relation R
- -log() to convert probabilities into costs
- A* needs positive weights
- taking attachment and labeling decisions jointly

Cost of Edges (cont'd)

$$\log(0{,}05) < \log(0{,}5)$$

$$-\log(0{,}05) > -\log(0{,}5)$$

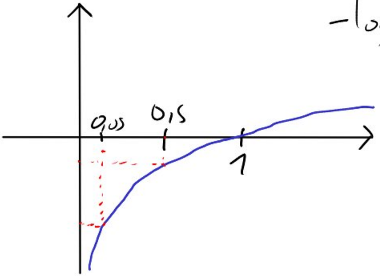


Figure: Converting probabilities to costs

Heuristic

h_best

- best cost of attaching current unit to some unit already attached

h_avg

- high variance in costs
- average of attachments to every remaining node
- potentially not admissible

h_avg was chosen due to performance reasons

Node generation / Constraints

State S consisting of $< V, E, RF >$

- V: List of DUs to be attached
- E: Set of edges
- RF: List of accesible nodes according to RFC

Input: State S
resultlist $= \varnothing$;
foreach *node n in RF* **do**
  Create new state S_n with added edge $(n, head(S.V))$ and modified V, RF;
  resultlist $\leftarrow$ resultlist $\cup$ S_n;
end
return *resultlist*

Chu-Liu-Edmonds-Algorithm

Chu-Liu-Edmonds Algorithm (CLE)

- replacement of CDUs with recursive heads similar to *non-projective trees with directed arcs*
- suitable for *Maximum Spanning Tree* Approach (MST, c.f. McDonald et al (2005))
- MST: Tree with maximum edge scores
- CLE generates MSTs from complete graphs
- Idea: Calculate attachment / labeling probabilities with NB / MaxEnt; then use CLE
- Difficult to impose constraints

Results

Unlabeled structures

Training model	Naive Bayes			Maxent			∅
Decoding method	greedy	MST	A*	greedy	MST	A*	Last
attachment alone (w5)	61.2	65.7	**66.2**	62.1	65.7	65.7	62.4
attachment alone	58.5	62.0	62.1	62.2	*65.7*	*65.7*	62.4
joint/unlabelled (w5)	59.7	61.7	64.8	62.2	65.1	*65.3*	62.4
joint/unlabelled	57.9	57.0	59.6	62.3	65.1	*65.4*	62.4

Figure: F1 Scores for unlabeled structures

- A* and MST differ from all other methods
- predicting relations does not improve attachment

Complete, labeled structures

Training model		Naive Bayes			Maxent			
Decoding method		greedy	MST	A*	greedy	last	MST	A*
joint(w5)	4 rels	38.9	29.3	41.7	42.2	42.2	31.6	*44.1*
joint	4 rels	38.7	26.7	39.6	44.6	44.5	30.0	**46.8**
pipe-line(w5)	4 rels	39.5	42.1	42.5	42.1	42.2	*44.3*	*44.3*
pipe-line	4 rels	38.7	40.8	40.8	44.5	44.5	**46.8**	**46.8**
joint(w5)	18 rels	22.0	8.2	23.7	28.7	28.6	4.8	*30.1*
joint	18 rels	23.4	4.1	24.0	34.2	34.1	5.4	**36.1**
pipe-line(w5)	18 rels	22.5	24.0	24.5	28.7	28.6	*30.2*	*30.2*
pipe-line	18 rels	23.9	24.7	24.8	34.0	34.1	**36.1**	**36.1**

Figure: F1 Scores for labeled structures

- A* and MST are still best methods
- pipe-lining performs better than joint-encoding
- pruning (w5) does not lead to significant improvement

Conclusion

Conclusions

- Comparable to similar approaches
  - labeled: 30-40%
  - unlabeled: 60-70%
- Global encoding improvments are significant over baselines
- Predictions respect the desired properties of discourse structures

Sources

- Muller, P., Afantenos, S., Denis, P., & Asher, N. (2012). Constrained decoding for text-level discourse parsing. Proceedings of COLING, 3(December 2012), 1883–1900.
- McDonald, R., Pereira, F., Ribarov, K., & Hajič, J. (2005). Non-projective dependency parsing using spanning tree algorithms. In Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing - HLT '05 (pp. 523–530). Morristown, NJ, USA: Association for Computational Linguistics.