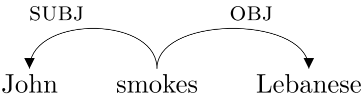
Using and teaching logic and machine learning for modeling cognitive processes

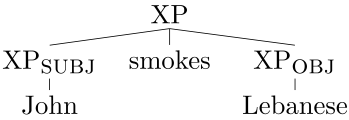
Anders Søgaard

Center for Language Technology
University of Copenhagen
Njalsgade 140–2
DK-2300 Copenhagen S
Email: soegaard@hum.ku.dk

November 25 2009

Dependency parsing





1	John	NP	2	SUBJ
2	smokes	V	0	ROOT
3	Lebanese	NP	2	OBJ

- The 1-best parsing problem for projective dependency grammars is in $\mathcal{O}(|G|n^3)$. Non-projective dependency parsing is NP-hard in general (e.g. by the Traveling Salesman Problem).
- Popular approximate parsing algorithms exist for both projective (deterministic transition-based; linear time) and non-projective dependency parsing (minimum spanning tree, $\mathcal{O}(|G|n^2)$.

Graph-based dependency parsing (MST)

Edmonds (1969) introduced a two-step $\mathcal{O}(|G|n^2)$ minimum spanning tree algorithm for edge-factored models:

(i) greedy head selection

(ii) cycle contraction.

Graph-based dependency parsing (MST)

Edmonds (1969) introduced a two-step $\mathcal{O}(|G|n^2)$ minimum spanning tree algorithm for edge-factored models:

(i) greedy head selection

(ii) cycle contraction.

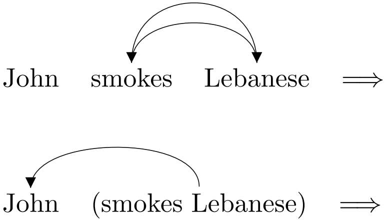
John smokes Lebanese $\Longrightarrow$

Graph-based dependency parsing (MST)

Edmonds (1969) introduced a two-step $\mathcal{O}(|G|n^2)$ minimum spanning tree algorithm for edge-factored models:

(i) greedy head selection

(ii) cycle contraction.

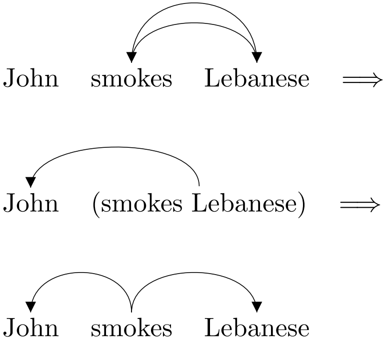


Graph-based dependency parsing (MST)

Edmonds (1969) introduced a two-step $\mathcal{O}(|G|n^2)$ minimum spanning tree algorithm for edge-factored models:

(i) greedy head selection

(ii) cycle contraction.



Transition-based dependency parsing

(i) $\dfrac{\textsc{Shift} \;\big|\; \ldots \qquad w \ldots}{\phantom{\textsc{Shift}} \;\big|\; \ldots w \qquad \ldots}$

Transition-based dependency parsing

(i) $\dfrac{\textsc{Shift} \mid \ldots \quad w \ldots}{\phantom{\textsc{Shift}} \mid \ldots w \quad \ldots}$

(ii) $\dfrac{\textsc{Reduce} \mid \ldots w \quad \ldots}{\phantom{\textsc{Reduce}} \mid \ldots \quad \ldots}$ iff $\exists v.v \to w$

Transition-based dependency parsing

(i) $\dfrac{\textsc{Shift} \mid \dots \quad w \dots}{\mid \dots w \quad \dots}$

(ii) $\dfrac{\textsc{Reduce} \mid \dots w \quad \dots}{\mid \dots \quad \dots \quad \text{iff} \quad \exists v. v \rightarrow w}$

(iii) $\dfrac{\textsc{Left-Arc} \mid \dots w \quad v \dots}{\mid \dots \quad v \dots \quad \text{add} \quad w \leftarrow v}$

Transition-based dependency parsing

(i)
$$\begin{array}{l|ll} \textsc{Shift} & \ldots & w\ldots \\ \hline & \ldots w & \ldots \end{array}$$

(ii)
$$\begin{array}{l|llll} \textsc{Reduce} & \ldots w & \ldots & & \\ \hline & \ldots & \ldots & \text{iff} & \exists v. v \rightarrow w \end{array}$$

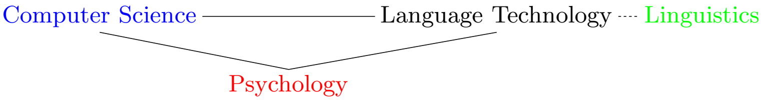
(iii)
$$\begin{array}{l|llll} \textsc{Left-Arc} & \ldots w & v\ldots & & \\ \hline & \ldots & v\ldots & \text{add} & w \leftarrow v \end{array}$$

(iv)
$$\begin{array}{l|llll} \textsc{Right-Arc} & \ldots v & w\ldots & & \\ \hline & \ldots v, w & \ldots & \text{add} & v \rightarrow w \\ & & & \text{iff} & \nexists w'. w' \rightarrow w \end{array}$$

How did John get to smoke Libanese?

SHIFT	...	John smokes Lebanese	
LEFT-ARC	..., John	smokes Lebanese	
SHIFT	...	smokes Lebanese	John $\leftarrow$ smokes
RIGHT-ARC	..., smokes	Lebanese	
REDUCE	... smokes, Lebanese	...	smokes $\longrightarrow$ Lebanese
ROOT	... smokes	...	

MA Programme in IT & Cognition



MA Programme in IT & Cognition

Computer Science —— Language Technology ···· Linguistics

Psychology

$$\begin{array}{ccccccc} 4 & & 4 & & 5 & & \\ 2007 & \rightarrow & 2008 & \rightarrow & 2009 & \rightarrow & 2010 \end{array}$$

MA Programme in IT & Cognition

Computer Science ——— Language Technology ···· Linguistics

Psychology

$$\begin{array}{ccccccc} & 4 & & 4 & & 5 & \\ 2007 & \rightarrow & 2008 & \rightarrow & 2009 & \rightarrow & 2010 \end{array}$$

1st	**2nd**	**3rd**	**4th**
RCS (F)	Form.Ling.	RCS(A)	Thesis
Logic	CP(F)	CP(A)	
Exp. mthd.	Stats	Adapt.Syst.	
Linguistics	LT(F)	LT(A)	
Progr(F/A)	HCI(F)	HCI(A)	

RCS(F) and RCS(A) are compulsary.

Student experiences 2007–9

	Good	Bad
Coherence		✓
Flexibility	✓	
Level	✓	
Social		✓
Thesis support		✓

Student experiences 2007–9

	Good	Bad
Coherence		✓
Flexibility	✓	
Level	✓	
Social		✓
Thesis support		✓

- Students never *stayed* at the university after class to work in groups.

Student experiences 2007–9

	Good	Bad
Coherence		✓
Flexibility	✓	
Level	✓	
Social		✓
Thesis support		✓

- Students never *stayed* at the university after class to work in groups.
- Students did not know much about each other.

Student experiences 2007–9

	Good	Bad
Coherence		✓
Flexibility	✓	
Level	✓	
Social		✓
Thesis support		✓

- Students never *stayed* at the university after class to work in groups.
- Students did not know much about each other.
- Several students did not have a thesis topic ready after having completed the first 90 ECTS.

Recent initiatives

Recent initiatives

- MENTORING, i.e. monthly interviews about:
  - $\longrightarrow$ coherence (1st year)
  - $\longrightarrow$ courses and exams
  - $\longrightarrow$ extra-curricular activities
  - $\longrightarrow$ thesis (primarily 2nd year)

Recent initiatives

- MENTORING, i.e. monthly interviews about:
  - → coherence (1st year)
  - → courses and exams
  - → extra-curricular activities
  - → thesis (primarily 2nd year)
- STUDENT GROUPS:
  a. ensemble-based part-of-speech tagging
  b. text prediction
  c. text classification
  d. model-checking for extensions of modal logic
  e. word alignment in translated text
- COLLABORATION:
  a. Center for Language Technology
  b. Mikroværkstedet, Lund University
  d. University of Tübingen (Germany)
  e. Copenhagen Business School

- EVENING LECTURES:
  - → J. Hansen (RUC): "Dynamic epistemic logic"
  - → P. Lindström (Lund, Sweden): "How children learn math"
  - → M. Haulrich (CBS): "Repair in transition-based parsing"
  - → R. Dekova (BAS, Bulgaria): "Lexical semantics and the mental lexicon"

- Evening lectures:
  - → J. Hansen (RUC): "Dynamic epistemic logic"
  - → P. Lindström (Lund, Sweden): "How children learn math"
  - → M. Haulrich (CBS): "Repair in transition-based parsing"
  - → R. Dekova (BAS, Bulgaria): "Lexical semantics and the mental lexicon"
- Reading groups, workshops, etc.:
  - → CBS-RG Machine Learning
  - → CBS-RG Natural Language Processing
  - → Linguistic Circle of Copenhagen
  - → ACL'10 (Uppsala, Sweden)
  - → ESSLLI'10

Challenge: 90 ECTS and a diverse group of students

1. Students *do* stay at the university after class.
2. The student groups “average out” the students.
3. Student groups are also a chance for excellent students to excel.
4. Finally, however, we synchronized our courses a bit:

Challenge: 90 ECTS and a diverse group of students

1. Students *do* stay at the university after class.
2. The student groups “average out” the students.
3. Student groups are also a chance for excellent students to excel.
4. Finally, however, we synchronized our courses a bit:
    (a) RCS(F) and RCS(A) model topics introduced in CP(F).

Challenge: 90 ECTS and a diverse group of students

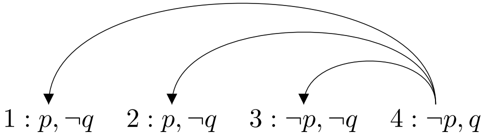
1. Students *do* stay at the university after class.
2. The student groups "average out" the students.
3. Student groups are also a chance for excellent students to excel.
4. Finally, however, we synchronized our courses a bit:
   (a) RCS(F) and RCS(A) model topics introduced in CP(F).
   (b) Topics from (a) are reused in other courses (Logic, LT(F), etc.).

Challenge: 90 ECTS and a diverse group of students

1. Students *do* stay at the university after class.
2. The student groups "average out" the students.
3. Student groups are also a chance for excellent students to excel.
4. Finally, however, we synchronized our courses a bit:
   (a) RCS(F) and RCS(A) model topics introduced in CP(F).
   (b) Topics from (a) are reused in other courses (Logic, LT(F), etc.).
   (c) All course exercises are in Python/Orange, also used in the student groups.

Logic for dependency parsing

In the dependency graph:



- the formula $\langle \prec \rangle \langle \leftarrow \rangle q$, i.e. the current node precedes a node whose syntactic head is in the denotation of q, evaluates as true in nodes 1 and 2.
- the formula $\langle \leftarrow \cap \prec ; \leftarrow \rangle \top$ is not true in any node.

Logic for dependency parsing (cont'd)

- A modal logic for dependency parsing was first introduced in Bröker (1997).

Logic for dependency parsing (cont'd)

- A modal logic for dependency parsing was first introduced in Bröker (1997).
- Kepser (2008) uses monadic second order logic to query dependency treebanks.

Logic for dependency parsing (cont'd)

- A modal logic for dependency parsing was first introduced in Bröker (1997).
- Kepser (2008) uses monadic second order logic to query dependency treebanks.
- Søgaard (2009) uses hybrid logic and crude repair to improve accuracy of dependency parsers.

Logic for dependency parsing (cont'd)

- A modal logic for dependency parsing was first introduced in Bröker (1997).
- Kepser (2008) uses monadic second order logic to query dependency treebanks.
- Søgaard (2009) uses hybrid logic and crude repair to improve accuracy of dependency parsers.
- Modal logics for other parsing formalisms are presented in Keller (1993), Søgaard (2007) and Søgaard and Lange (2009).

Machine learning for dependency parsing

- Dependency parsing is typically cast as supervised learning.

Machine learning for dependency parsing

- Dependency parsing is typically cast as supervised learning.
- Sufficient labeled data exists for a wide variety of languages.
  - The CONLL-X Shared Task used datasets from 12 languages.
  - The CONLL 2007 Shared Task used datasets from 10 languages (with three repeats).
  - Labeled data exists for other languages, incl. Hebrew, Latin, Romanian, Thai.

Machine learning for dependency parsing

- Dependency parsing is typically cast as supervised learning.
- Sufficient labeled data exists for a wide variety of languages.
  - The CONLL-X Shared Task used datasets from 12 languages.
  - The CONLL 2007 Shared Task used datasets from 10 languages (with three repeats).
  - Labeled data exists for other languages, incl. Hebrew, Latin, Romanian, Thai.
- The CONLL format and evaluation procedure are standard in the community.

Exercise from Lect. 1, RCS(F): Naive Bayes

Features: $\textsc{pos}(w')$. Class: $\textsc{pos}(w)$. Labeled data:

1	John	NP	2
2	drives	V	0
3	cars	NP	2

1	John	NP	3
2	is	V	3
3	fast	ADJ	0

1	John	NP	2
2	walks	V	0
3	and	CONJ	2
4	talks	V	2
5	fast	ADJ	2

$P(\leftarrow \textsc{Root})$	3/11	$P(\leftarrow \text{V})$	6/11	$P(\leftarrow \text{ADJ})$	2/11

$d = \text{NP}$		$d = \text{V}$		$d = \text{ADJ}$	
$P(d\| \leftarrow \textsc{Root})$	0/3	$P(d\| \leftarrow \textsc{Root})$	2/3	$P(d\| \leftarrow \textsc{Root})$	1/3
$P(d\| \leftarrow \text{V})$	3/6	$P(d\| \leftarrow \text{V})$	1/6	$P(d\| \leftarrow \text{V})$	1/6
$P(d\| \leftarrow \text{ADJ})$	1/2	$P(d\| \leftarrow \text{ADJ})$	1/2	$P(d\| \leftarrow \text{ADJ})$	0/2
0.09: V		0.19 : $\textsc{Root}$		0.09:$\textsc{Root}$/V	

If single-rooted:

1	John	2
2	drives	0
3	fast	2

Dependency parsing, now

	Algorithm	Learner	Complexity
MaltParser	Transition-based	SVM	$\mathcal{O}(\|G\|n)$
MSTParser	Graph-based	MIRA	$\mathcal{O}(\|G\|n^2)$*

*Faster on average, since models are smaller.

Dependency parsing, now

	Algorithm	Learner	Complexity
MaltParser	Transition-based	SVM	$\mathcal{O}(\|G\|n)$
MSTParser	Graph-based	MIRA	$\mathcal{O}(\|G\|n^2)$*

*Faster on average, since models are smaller.

- Nivre and McDonald (2008) use *stacking* to combine the strengths of the two parsers.

Dependency parsing, now

	Algorithm	Learner	Complexity
MaltParser	Transition-based	SVM	$\mathcal{O}(\|G\|n)$
MSTParser	Graph-based	MIRA	$\mathcal{O}(\|G\|n^2)$*

*Faster on average, since models are smaller.

- Nivre and McDonald (2008) use *stacking* to combine the strengths of the two parsers.
- Martins et al. (2008) use *recursive stacking* to obtain previously best reported results.

Dependency parsing, now

	Algorithm	Learner	Complexity
MaltParser	Transition-based	SVM	$\mathcal{O}(\|G\|n)$
MSTParser	Graph-based	MIRA	$\mathcal{O}(\|G\|n^2)$*

*Faster on average, since models are smaller.

- Nivre and McDonald (2008) use *stacking* to combine the strengths of the two parsers.
- Martins et al. (2008) use *recursive stacking* to obtain previously best reported results.
- Semisupervised methods have also been used to boost state-of-the-art (Koo et al., 2008; Sagae and Gordon, 2009; Suzuki et al., 2009).

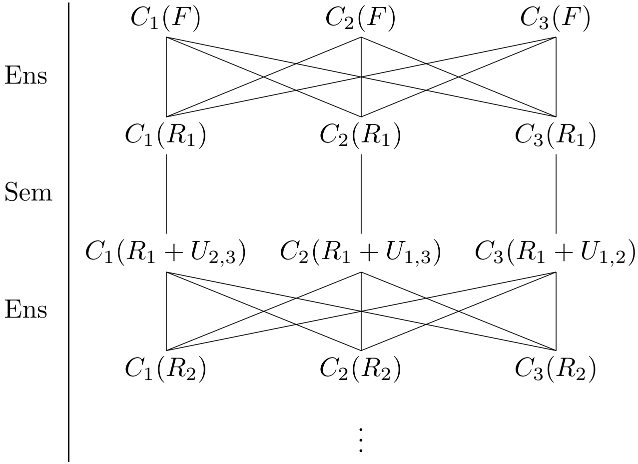
Dependency parsing, now

	Algorithm	Learner	Complexity
MaltParser	Transition-based	SVM	$\mathcal{O}(\|G\|n)$
MSTParser	Graph-based	MIRA	$\mathcal{O}(\|G\|n^2)$*

*Faster on average, since models are smaller.

- Nivre and McDonald (2008) use *stacking* to combine the strengths of the two parsers.
- Martins et al. (2008) use *recursive stacking* to obtain previously best reported results.
- Semisupervised methods have also been used to boost state-of-the-art (Koo et al., 2008; Sagae and Gordon, 2009; Suzuki et al., 2009).
- Søgaard (t.a.) *combines* ensemble-based and semisupervised methods to obtain best reported results.

Søgaard (t.a.)


Ens
Sem
Ens
$C_1(F)$ $C_2(F)$ $C_3(F)$
$C_1(R_1)$ $C_2(R_1)$ $C_3(R_1)$
$C_1(R_1 + U_{2,3})$ $C_2(R_1 + U_{1,3})$ $C_3(R_1 + U_{1,2})$
$C_1(R_2)$ $C_2(R_2)$ $C_3(R_2)$
$\vdots$

CONLL-X datasets

	C06	Mar08	Ours
Arabic	66.91	69.12	**70.12**
Danish	84.79	**86.79**	86.47
Dutch	79.19	81.51	**81.87**
German	87.34	88.68	**89.08**
Japanese	91.65	91.61	**92.28**
Portuguese	76.60	88.30	**88.76**
Slovene	76.12	76.72	**77.98**
Spanish	82.25	83.73	**84.67**
Swedish	84.58	85.16	**85.92**
Turkish	65.68	65.21	**67.42**

Logic and machine learning? (2do-list)

Checked items:

- Model-checking is used to verify labeled data.

Non-checked items:

Logic and machine learning? (2do-list)

Checked items:

- Model-checking is used to verify labeled data.
- Crude repair is used to improve parsing quality.

Non-checked items:

Logic and machine learning? (2do-list)

Checked items:

- Model-checking is used to verify labeled data.
- Crude repair is used to improve parsing quality.
- Finally, logic is used to study the properties of linguistic theories (Blackburn and Spaan, 1993; Søgaard, 2007).

Non-checked items:

Logic and machine learning? (2do-list)

Checked items:

- Model-checking is used to verify labeled data.
- Crude repair is used to improve parsing quality.
- Finally, logic is used to study the properties of linguistic theories (Blackburn and Spaan, 1993; Søgaard, 2007).

Non-checked items:

- Transition-based dependency parsing in Cross-style probabilistic modal logic.

Logic and machine learning? (2do-list)

Checked items:

- Model-checking is used to verify labeled data.
- Crude repair is used to improve parsing quality.
- Finally, logic is used to study the properties of linguistic theories (Blackburn and Spaan, 1993; Søgaard, 2007).

Non-checked items:

- Transition-based dependency parsing in Cross-style probabilistic modal logic.
- Learning crude repair functions.

Logic and machine learning? (2do-list)

Checked items:

- Model-checking is used to verify labeled data.
- Crude repair is used to improve parsing quality.
- Finally, logic is used to study the properties of linguistic theories (Blackburn and Spaan, 1993; Søgaard, 2007).

Non-checked items:

- Transition-based dependency parsing in Cross-style probabilistic modal logic.
- Learning crude repair functions.
- Modal characterizations of mildly non-projective dependency grammars.

Logic and machine learning? (2do-list)

Checked items:

- Model-checking is used to verify labeled data.
- Crude repair is used to improve parsing quality.
- Finally, logic is used to study the properties of linguistic theories (Blackburn and Spaan, 1993; Søgaard, 2007).

Non-checked items:

- Transition-based dependency parsing in Cross-style probabilistic modal logic.
- Learning crude repair functions.
- Modal characterizations of mildly non-projective dependency grammars.
- Model-checking polyadic dynamic logic.

Data-driven dependency parsing in collaborative research projects at CST

- QUESTION ANSWERING:
  - MOSES (university websites); led by Patrizia Paggio.
  - ESICT (patient diagnosis); led by Bente Maegaard.
- MACHINE TRANSLATION:
  - ESSMT (practical); led by me.
  - EMCOTT (theoretical; under review); led by Jürgen Wedekind and me.