Syntax-based Machine Translation using Multi Bottom-up Tree Transducers

Andreas Maletti
Fabienne Braune, Daniel Quernheim, Nina Seemann

Institute for Natural Language Processing
Universität Stuttgart, Germany

maletti@ims.uni-stuttgart.de

Uppsala — November 8, 2012

Overview

Machine translation

Translation

- Input:
  Official forecasts predicted just 3 percent, Bloomberg said.

- Reference:
  Offizielle Prognosen sind von nur 3 Prozent ausgegangen, meldete Bloomberg.
  [official] [forecasts] [are] [of] [only] [3 percent] [assumed] [reported] [Bloomberg]

- Our MBOT translator (untuned):
  offiziellen prognosen vorausgesagt nur 3 % bloomberg habe.
  [official] [forecasts] [*predicted] [only] [3 %] [Bloomberg] [*has]

- Google Translate (`translate.google.com`):
  Offizielle Prognosen vorausgesagt nur 3 Prozent, sagte Bloomberg.
  [official] [forecasts] [*predicted] [only] [3 percent] [said] [Bloomberg]

Machine translation

Translation

- Input:
  Official forecasts predicted just 3 percent, Bloomberg said.

- Reference:
  Offizielle Prognosen sind von nur 3 Prozent ausgegangen, meldete Bloomberg.
  [official] [forecasts] [are] [of] [only] [3 percent] [assumed] [reported] [Bloomberg]

- Our MBOT translator (untuned):
  offiziellen prognosen vorausgesagt nur 3 % bloomberg habe.
  [official] [forecasts] [*predicted] [only] [3 %] [Bloomberg] [*has]

- Google Translate (`translate.google.com`):
  Offizielle Prognosen vorausgesagt nur 3 Prozent, sagte Bloomberg.
  [official] [forecasts] [*predicted] [only] [3 percent] [said] [Bloomberg]

Machine translation

Translation

- Input:
  Official forecasts predicted just 3 percent, Bloomberg said.
- Reference:
  Offizielle Prognosen sind von nur 3 Prozent ausgegangen, meldete Bloomberg.
  [official] [forecasts] [are] [of] [only] [3 percent] [assumed] [reported] [Bloomberg]
- Our MBOT translator (untuned):
  offiziellen prognosen vorausgesagt nur 3 % bloomberg habe.
  [official] [forecasts] [*predicted] [only] [3 %] [Bloomberg] [*has]
- Google Translate (`translate.google.com`):
  Offizielle Prognosen vorausgesagt nur 3 Prozent, sagte Bloomberg.
  [official] [forecasts] [*predicted] [only] [3 percent] [said] [Bloomberg]

Machine translation

Translation

- Input:
  The ECB wants to hold inflation to under two percent, or somewhere in that vicinity.

- Reference:
  Die EZB ist bestrebt, die Inflationsrate unter zwei Prozent,
  [the] [ECB] [is] [desire] [the] [inflation rate] [below] [two percent]
  oder zumindest knapp an der Zwei-Prozent-Marke zu halten.
  [or] [at least] [close] [at] [the] [two percent mark] [to keep]

- Google Translate (`translate.google.com`):
  Die EZB will die Inflation zu halten unter zwei Prozent,
  [the] [ECB] [wants] [the] [inflation] [*to keep] [below] [two percent]
  oder irgendwo in der Nähe.
  [or] [somewhere] [in] [the] [vicinity]

Machine translation

Translation

- Input:
  The ECB wants to hold inflation to under two percent, or somewhere in that vicinity.

- Reference:
  Die EZB ist bestrebt, die Inflationsrate unter zwei Prozent,
  [the] [ECB] [is] [desire] [the] [inflation rate] [below] [two percent]
  oder zumindest knapp an der Zwei-Prozent-Marke zu halten.
  [or] [at least] [close] [at] [the] [two percent mark] [to keep]

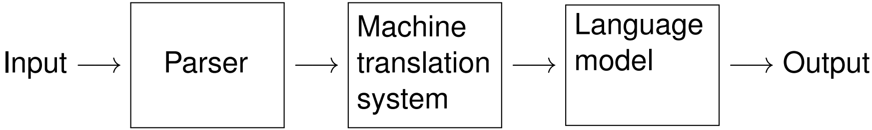
- Google Translate (`translate.google.com`):
  Die EZB will die Inflation zu halten unter zwei Prozent,
  [the] [ECB] [wants] [the] [inflation] [*to keep] [below] [two percent]
  oder irgendwo in der Nähe.
  [or] [somewhere] [in] [the] [vicinity]

Syntax-based machine translation

Remark

There is no universally accepted definition

Syntax-based systems



What do we have?

Input

- Parallel text (English and German) — EUROPARL
- Parsers — BITPAR, CHARNIAK, BERKELEY

Example

- "We must bear in mind the Community as a whole."
- "Wir müssen uns davor hüten, alles vergemeinschaften zu wollen."

What do we have?

Input

- Parallel text (English and German) EUROPARL
- Parsers BITPAR, CHARNIAK, BERKELEY

Example

- "We must bear in mind the Community as a whole."
- "Wir müssen uns davor hüten, alles vergemeinschaften zu wollen."

What do we have?

Input

- Parallel text (English and German) EUROPARL
- Parsers BITPAR, CHARNIAK, BERKELEY

Example

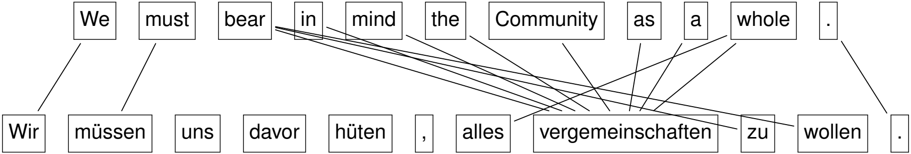
- “We must bear in mind the Community as a whole.”
- “Wir müssen uns davor hüten, alles vergemeinschaften zu wollen.”

EUROPARL German-English parallel data:

- 1,920,209 parallel sentences
- 44,548,491 words in German
- 47,818,827 words in English

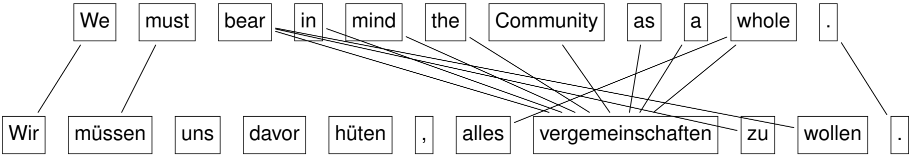
First step: Word Alignment

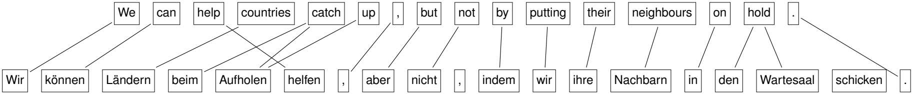
Alignments by GIZA++ [OCH, NEY '03]:



First step: Word Alignment

Alignments by GIZA++ [OCH, NEY '03]:

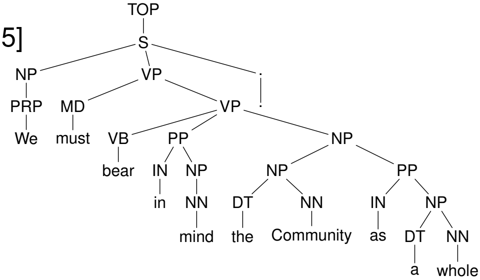




Second step: Parsing

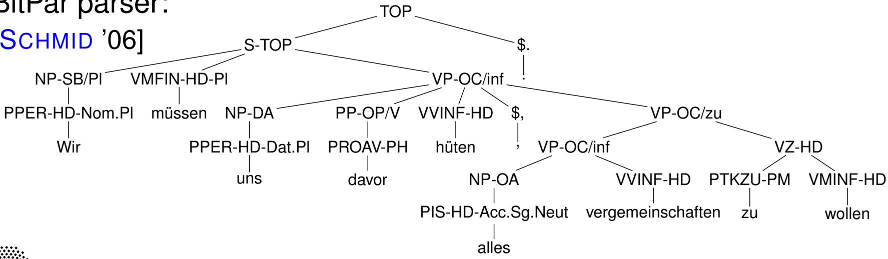
CHARNIAK parser:

[CHARNIAK, JOHNSON '05]



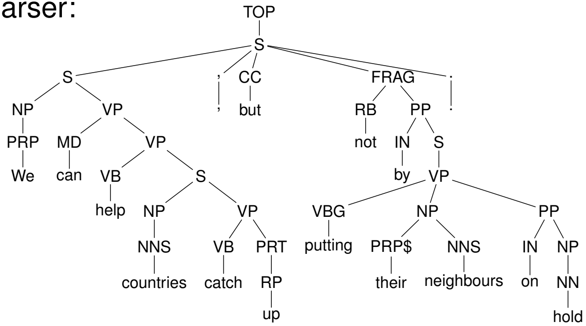
BitPar parser:

[SCHMID '06]

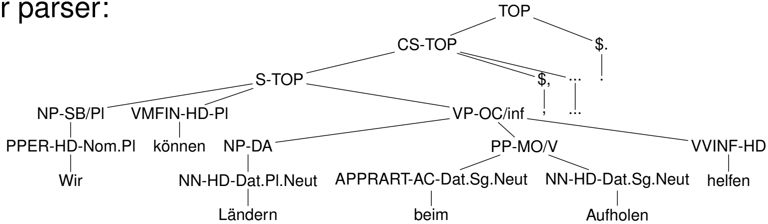


Second step: Parsing

CHARNIAK parser:



BitPar parser:



Equalizing examples

Input

Yugoslav President Voislav signed for Serbia.

و تولى التوقيع عن صربيا الرئيس اليوغوسلافي فويسلاف

<u>Transliteration:</u> w twlY AltwqyE En SrbyA Alr}ys AlywgwslAfy fwyslAf.

And then the matter was decided, and everything was put in place.

ف كان ان تم الحسم و وضعت الأمور في نصاب ها

<u>Transliteration:</u> f kAn An tm AlHsm w wDEt Al>mwr fy nSAb hA.

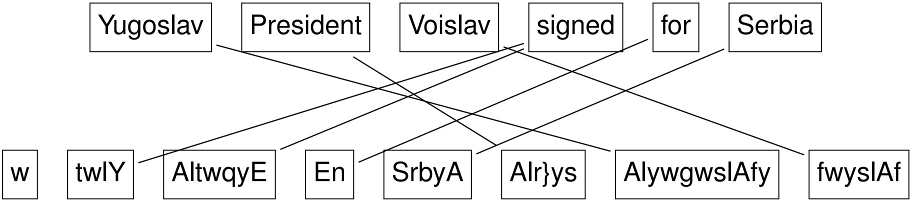
Below are the male and female winners in the different categories.

و هنا الأوائل و الأوليات في مختلف الفئات

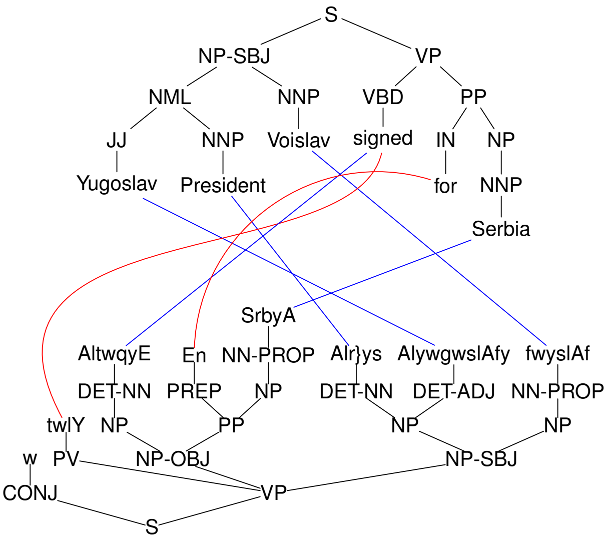
<u>Transliteration:</u> w hnA Al>wA}l w Al>wlyAt fy mxtlf Alf}At.

Equalizing examples

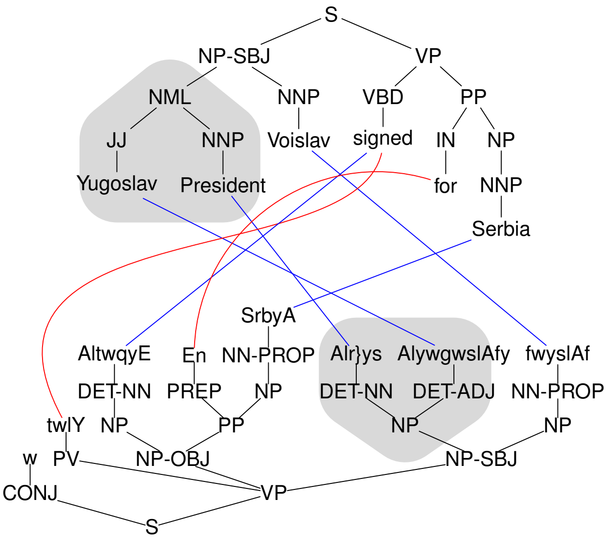
Alignment



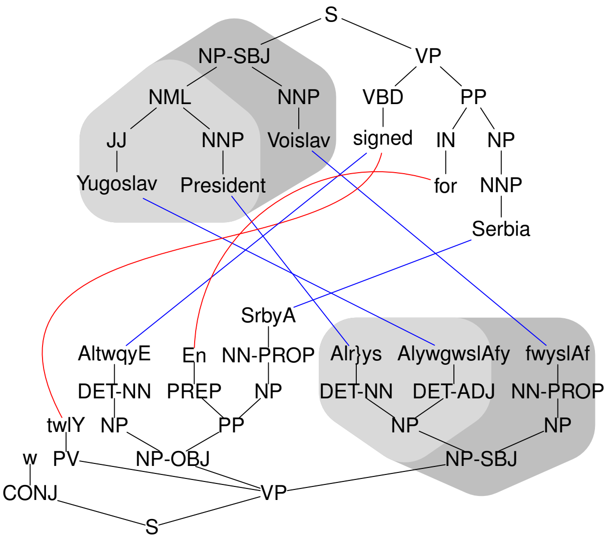
Rule extraction



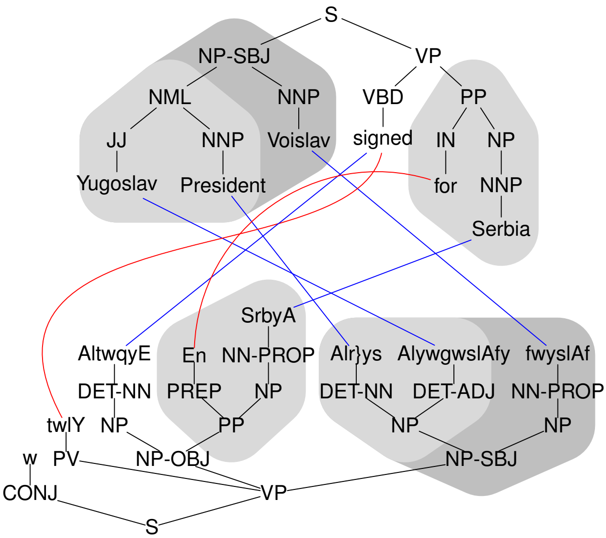
Rule extraction



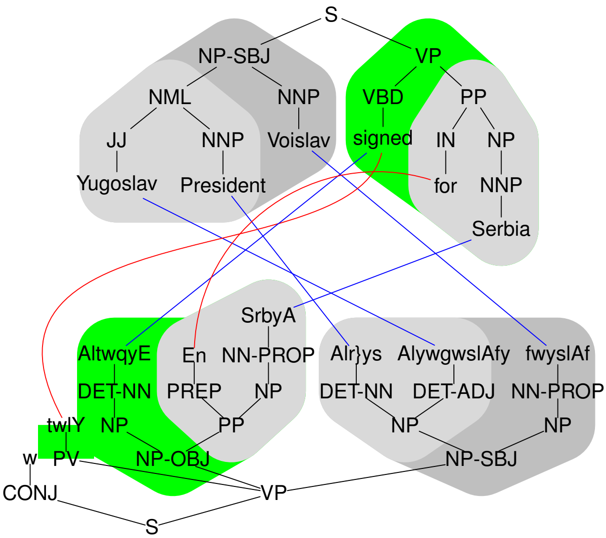
Rule extraction



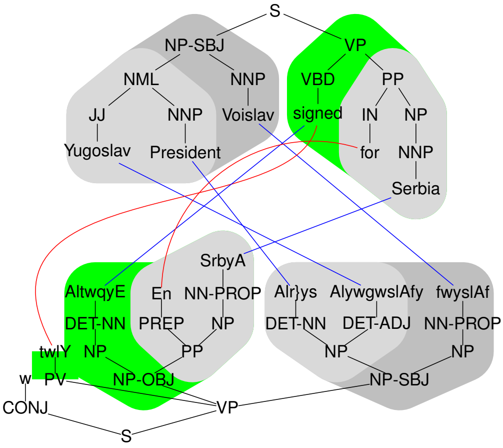
Rule extraction

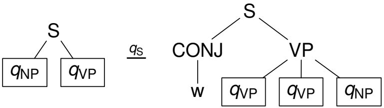


Rule extraction

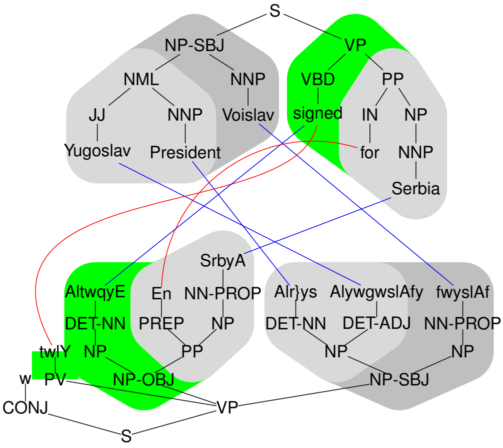


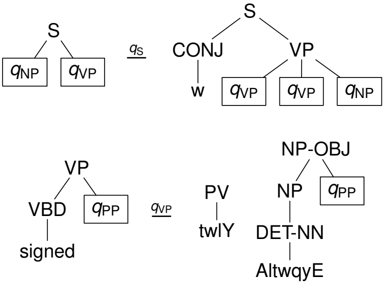
Rule extraction





Rule extraction





Roadmap

Syntax

Definition

Extended multi bottom-up tree transducer (XMBOT)

system (Q, Σ, I, R)

- Q finite set — *states*
- Σ alphabet — input/output symbols
- $I \subseteq Q$ — initial states
- R finite set of rules $\ell \xrightarrow{q} r_1 \cdots r_n$ — rules
  - $q \in Q$
  - linear $\ell \in T_\Sigma(Q)$ — each state occurs at most once
  - $r_1, \ldots, r_n \in T_\Sigma(\mathrm{var}(\ell))$ — only states from ℓ may occur

Syntax

Definition

Extended multi bottom-up tree transducer (XMBOT)
system (Q, Σ, I, R)

- Q finite set — *states*
- Σ alphabet — input/output symbols
- $I \subseteq Q$ — initial states
- R finite set of rules $\ell \xrightarrow{q} r_1 \cdots r_n$ — rules
  - $q \in Q$
  - linear $\ell \in T_\Sigma(Q)$ — each state occurs at most once
  - $r_1, \ldots, r_n \in T_\Sigma(\mathrm{var}(\ell))$ — only states from ℓ may occur

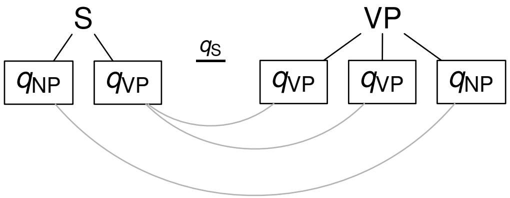
Syntax

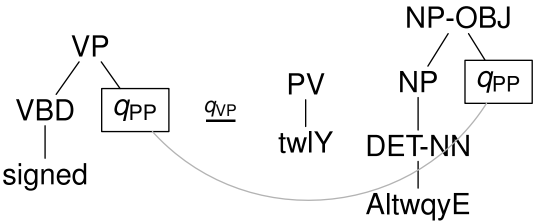
Definition

Extended multi bottom-up tree transducer (XMBOT)
system (Q, Σ, I, R)

- Q finite set — *states*
- Σ alphabet — input/output symbols
- $I \subseteq Q$ — initial states
- R finite set of rules $\ell \xrightarrow{q} r_1 \cdots r_n$ — rules
  - $q \in Q$
  - linear $\ell \in T_\Sigma(Q)$ — each state occurs at most once
  - $r_1, \ldots, r_n \in T_\Sigma(\text{var}(\ell))$ — only states from ℓ may occur

Extracted rules



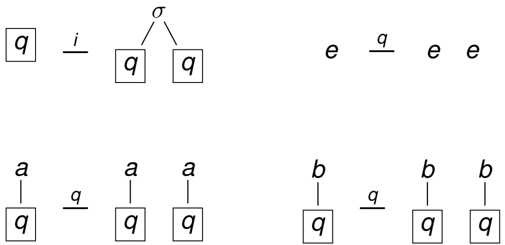


Illustrative example

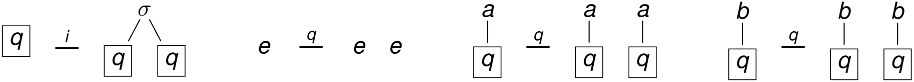
Example

XMBOT $(\{i, q\}, \Sigma, \{i\}, R)$

- $\Sigma = \{\sigma, a, b, e\}$
- R contains:



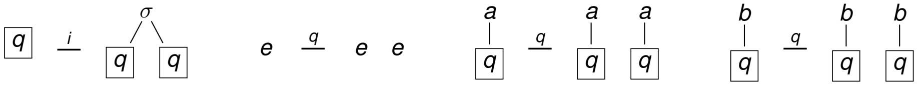
Synchronous derivation semantics



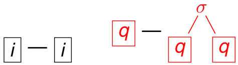
Example (Derivation)

i — i

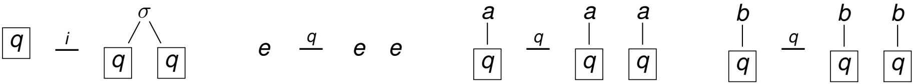
Synchronous derivation semantics



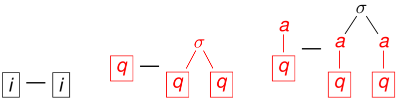
Example (Derivation)



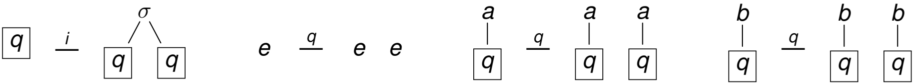
Synchronous derivation semantics



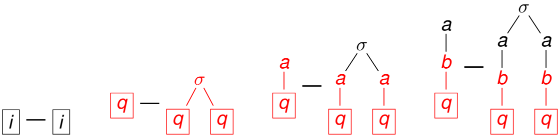
Example (Derivation)



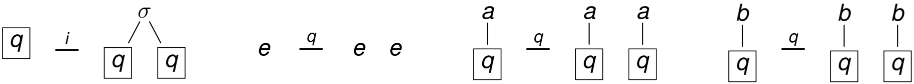
Synchronous derivation semantics



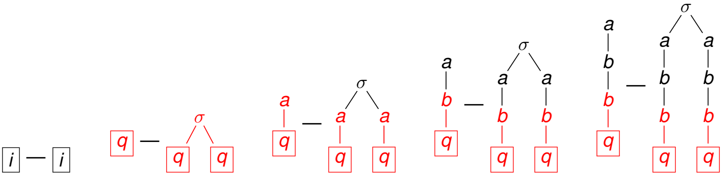
Example (Derivation)



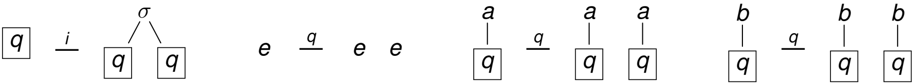
Synchronous derivation semantics



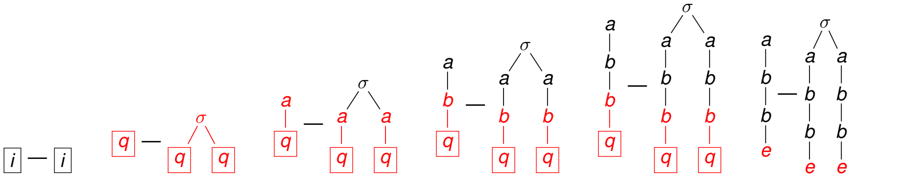
Example (Derivation)



Synchronous derivation semantics



Example (Derivation)



Synchronous derivation semantics

Definition

XMBOT $M = (Q, \Sigma, I, R)$

$$\tau_M = \{(t, u) \in T_\Sigma \times T_\Sigma \mid \exists q \in I \colon (q, q) \Rightarrow_M^* (t, u)\}$$

Synchronous derivation semantics

Definition

XMBOT $M = (Q, \Sigma, I, R)$

$$\tau_M = \{(t, u) \in T_\Sigma \times T_\Sigma \mid \exists q \in I\colon (q, q) \Rightarrow_M^* (t, u)\}$$

Example

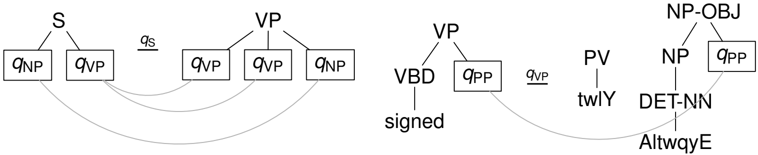
It computes $\{(t, \sigma(t, t)) \mid t \in T_\Sigma\}$

Roadmap

Input and output language

Input side

Let us look only at the input side (lhs)



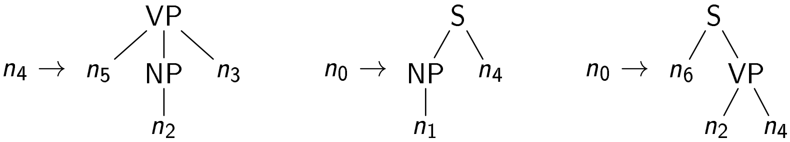
Rewrite

Instead of $t \overset{q}{\longrightarrow}$ we write $q \rightarrow t$

$$q_S \rightarrow \text{S}(q_{NP}, q_{VP}) \qquad q_{VP} \rightarrow \text{VP}(\text{VBD}(\text{signed}), q_{PP})$$

Regular tree grammar

Example (Rules)

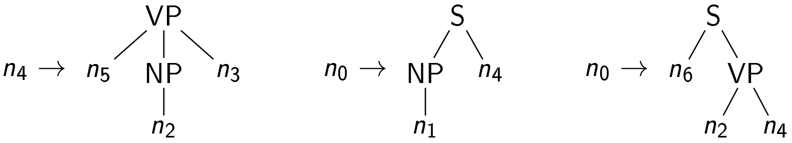


Example (Derivation)

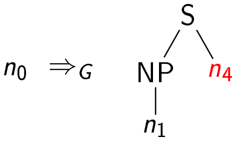
n_0

Regular tree grammar

Example (Rules)

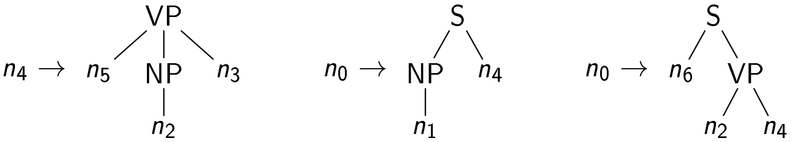


Example (Derivation)

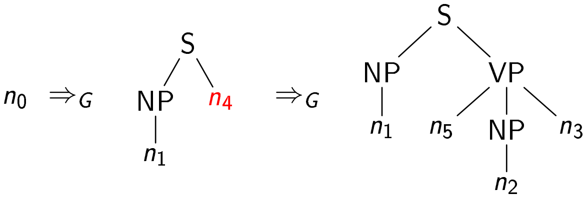


Regular tree grammar

Example (Rules)



Example (Derivation)



Regular tree grammar

Definition

The tree languages recognized by regular tree grammars are the regular tree languages

Theorem (Input side)

The input language of each XMBOT is regular

Theorem (Output side)

The output (string) language of each XMBOT is an LCFRS

Regular tree grammar

Definition

The tree languages recognized by regular tree grammars are the regular tree languages

Theorem (Input side)

The input language of each XMBOT is regular

Theorem (Output side)

The output (string) language of each XMBOT is an LCFRS

Regular tree grammar

Definition

The tree languages recognized by regular tree grammars are the regular tree languages

Theorem (Input side)

The input language of each XMBOT is regular

Theorem (Output side)

The output (string) language of each XMBOT is an LCFRS

Prioritizing rules

Observation

- not all rules are equally useful
- determine and set priorities

What we do

- count the number of times a rule is extracted
- normalize to obtain a probability

Not clever, but it is what most people do!

Prioritizing rules

Observation

- not all rules are equally useful
- determine and set priorities

What we do

- count the number of times a rule is extracted
- normalize to obtain a probability

Not clever, but it is what most people do!

Prioritizing rules

Observation

- not all rules are equally useful
- determine and set priorities

What we do

- count the number of times a rule is extracted
- normalize to obtain a probability

Not clever, but it is what most people do!

Prioritizing rules

Observation

- not all rules are equally useful
- determine and set priorities

What we do

- count the number of times a rule is extracted
- normalize to obtain a probability

Not clever, but it is what most people do!

EM training

Theorem

Derivations of XMBOT are regular (even in the weighted case)

EM training

- given translation pair (w_1, w_2)
- input- and output restrict to w_1 and w_2 parsing
- build derivations
- compute relative "usefulness" of each rule
- move to the next training sentence (and start anew)

EM training

Theorem

Derivations of XMBOT are regular (even in the weighted case)

EM training

- given translation pair (w_1, w_2)
- input- and output restrict to w_1 and w_2 parsing
- build derivations
- compute relative “usefulness” of each rule
- move to the next training sentence (and start anew)

EM training

Theorem

Derivations of XMBOT are regular (even in the weighted case)

EM training

- given translation pair (w_1, w_2)
- input- and output restrict to w_1 and w_2 — parsing
- build derivations
- compute relative "usefulness" of each rule
- move to the next training sentence (and start anew)

EM training

Theorem

Derivations of XMBOT are regular (even in the weighted case)

EM training

- given translation pair (w_1, w_2)
- input- and output restrict to w_1 and w_2 — parsing
- build derivations
- compute relative “usefulness” of each rule
- move to the next training sentence (and start anew)

EM training

Theorem

Derivations of XMBOT are regular (even in the weighted case)

EM training

- given translation pair (w_1, w_2)
- input- and output restrict to w_1 and w_2 parsing
- build derivations
- compute relative “usefulness” of each rule
- move to the next training sentence (and start anew)

EM training

Theorem

Derivations of XMBOT are regular (even in the weighted case)

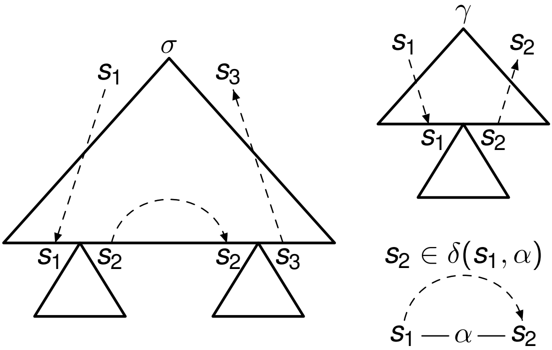
EM training

- given translation pair (w_1, w_2)
- input- and output restrict to w_1 and w_2 parsing
- build derivations
- compute relative “usefulness” of each rule
- move to the next training sentence (and start anew)

Input/output restriction

Definition

Input restriction restricts the string language of the domain of an XMBOT to a regular language



Input/output restriction

Theorem

Restricting the ... by FSA A is ...

device	input	output
XMBOT M	$\mathcal{O}(\lvert M\rvert \cdot \lvert A\rvert^3)$	$\mathcal{O}(\lvert M\rvert \cdot \lvert A\rvert^x)$
XTOP M	$\mathcal{O}(\lvert M\rvert \cdot \lvert A\rvert^y)$	$\mathcal{O}(\lvert M\rvert \cdot \lvert A\rvert^y)$

with $x = 2\,\mathrm{rk}(M) + 2$ *and* $y = 2\,\mathrm{rk}(M) + 5$

But alas

We cannot handle it yet.

Input/output restriction

Theorem

Restricting the … by FSA A is …

device	input	output
XMBOT M	$\mathcal{O}(\lvert M\rvert \cdot \lvert A\rvert^3)$	$\mathcal{O}(\lvert M\rvert \cdot \lvert A\rvert^x)$
XTOP M	$\mathcal{O}(\lvert M\rvert \cdot \lvert A\rvert^y)$	$\mathcal{O}(\lvert M\rvert \cdot \lvert A\rvert^y)$

with $x = 2\,\mathrm{rk}(M) + 2$ *and* $y = 2\,\mathrm{rk}(M) + 5$

But alas

We cannot handle it yet.

High-performance regular tree grammar toolkit

Development

- in cooperation with THOMAS HANNEFORTH (Potsdam, Germany)
- basic operations work
- determinization and minimization work

Current use

- used to represent trees during rule extraction
- EM training
- many more uses planned

High-performance regular tree grammar toolkit

Development

- in cooperation with THOMAS HANNEFORTH (Potsdam, Germany)
- basic operations work
- determinization and minimization work

Current use

- used to represent trees during rule extraction
- EM training
- many more uses planned

High-performance regular tree grammar toolkit

Development

- in cooperation with THOMAS HANNEFORTH (Potsdam, Germany)
- basic operations work
- determinization and minimization work

Current use

- used to represent trees during rule extraction
- EM training
- many more uses planned

High-performance regular tree grammar toolkit

Development

- in cooperation with THOMAS HANNEFORTH (Potsdam, Germany)
- basic operations work
- determinization and minimization work

Current use

- used to represent trees during rule extraction
- EM training
- many more uses planned

High-performance regular tree grammar toolkit

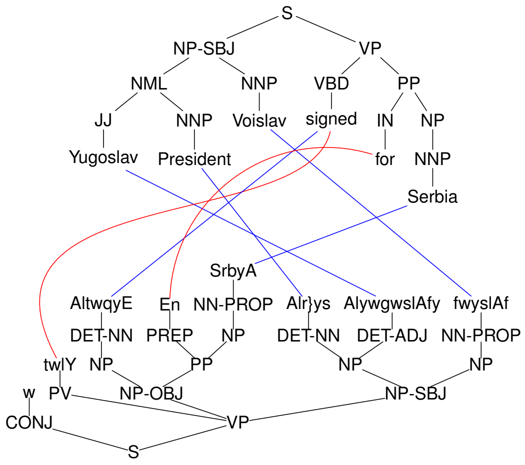
Development

- in cooperation with THOMAS HANNEFORTH (Potsdam, Germany)
- basic operations work
- determinization and minimization work

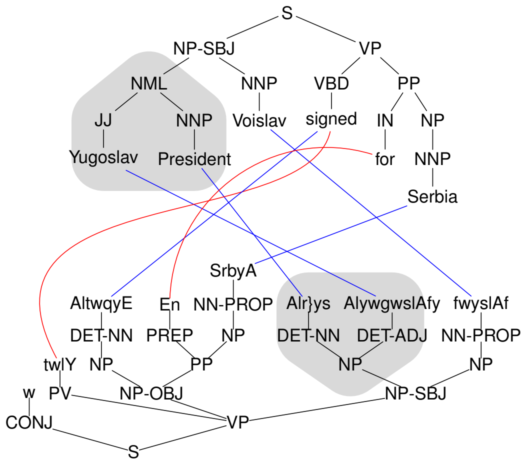
Current use

- used to represent trees during rule extraction
- EM training
- many more uses planned

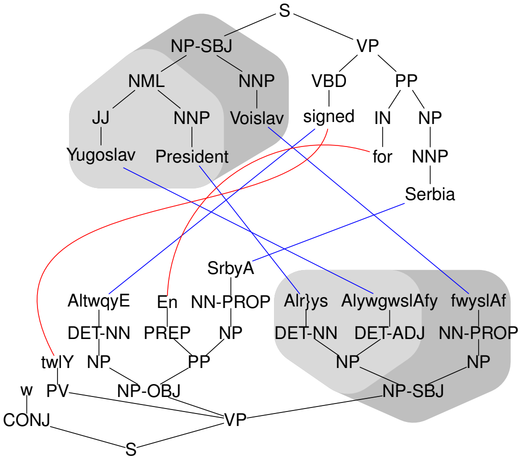
Rule extraction



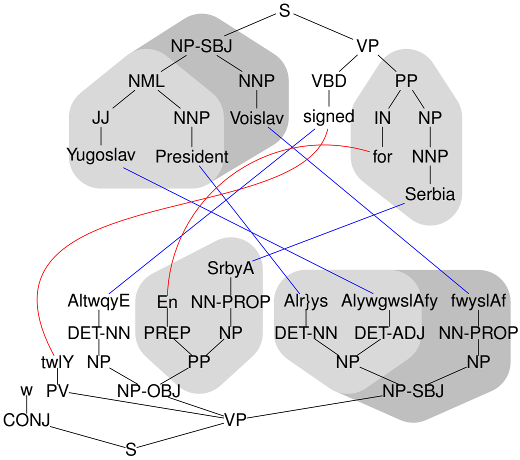
Rule extraction



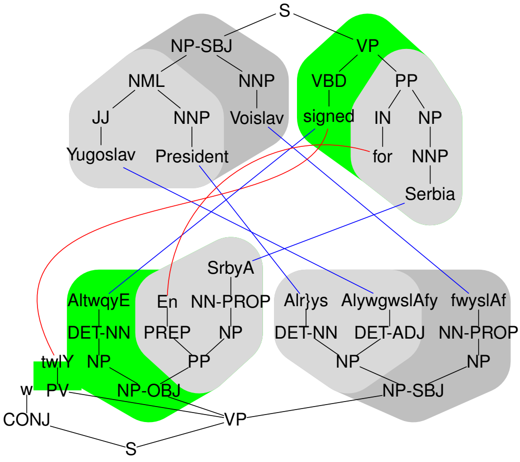
Rule extraction



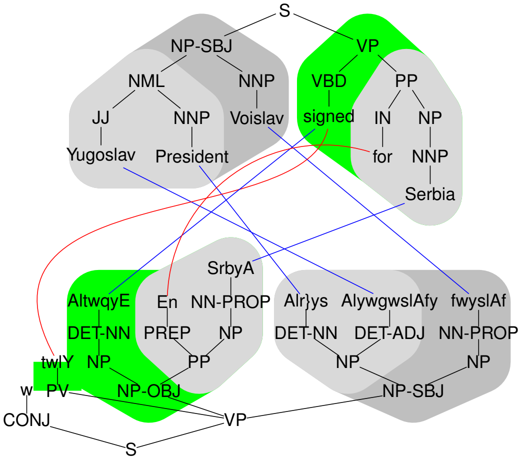
Rule extraction



Rule extraction



Rule extraction



There are sometimes millions of extractable rules!

Fantasy 1

Rules are trees

Why not represent the rules with regular tree grammars?

- How?
- Is it efficient?
- Can operations be used on this representation?
- . . .
- Does it help?

Fantasy 1

Rules are trees

Why not represent the rules with regular tree grammars?

- How?
- Is it efficient?
- Can operations be used on this representation?
- ...
- Does it help?

Parser-Translator interface

Syntax-based systems

Input ⟶ Parser ⟶ Machine translation system ⟶ Language model ⟶ Output

Remarks

- Parser can deliver *n*-best lists (instead of just 1-best)
- or even all parses (with scores)
- Theoretically XMBOT can handle any RTG as input
- Practically even explicit *n*-best lists are inefficient

Parser-Translator interface

Syntax-based systems

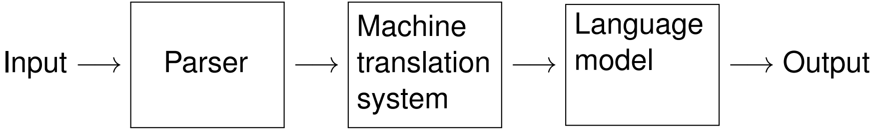
Input ⟶ Parser ⟶ Machine translation system ⟶ Language model ⟶ Output

Remarks

- Parser can deliver *n*-best lists (instead of just 1-best)
- or even all parses (with scores)
- Theoretically XMBOT can handle any RTG as input
- Practically even explicit *n*-best lists are inefficient

Parser-Translator interface

Syntax-based systems

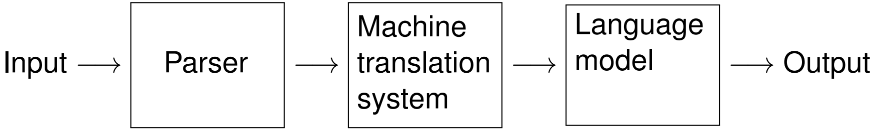


Remarks

- Parser can deliver *n*-best lists (instead of just 1-best)
- or even all parses (with scores)
- Theoretically XMBOT can handle any RTG as input
- Practically even explicit *n*-best lists are inefficient

Parser-Translator interface

Syntax-based systems



Remarks

- Parser can deliver *n*-best lists (instead of just 1-best)
- or even all parses (with scores)
- Theoretically XMBOT can handle any RTG as input
- Practically even explicit *n*-best lists are inefficient

Fantasy 2

Use lattices

- Get parser to output RTG
- Handle RTG as input to XMBOT

⇝ tight integration of parser and XMBOT

- XMBOT and LM already have this tight integration

Fantasy 2

Use lattices

- Get parser to output RTG
- Handle RTG as input to XMBOT

$\rightsquigarrow$ tight integration of parser and XMBOT

- XMBOT and LM already have this tight integration

Roadmap

XMBOT in machine translation

Moses [Koehn et al. '07]

- framework for statistical MT
- implementations for many standard tasks (alignment, lexical scores, language model, BLEU scoring)
- supports syntax-based MT

We added

- XMBOT rule support
- XMBOT chart decoder
- adjusted language model calls

XMBOT in machine translation

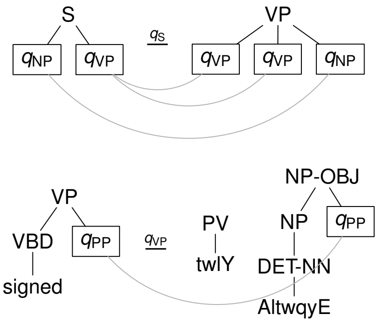
Moses [Koehn et al. '07]

- framework for statistical MT
- implementations for many standard tasks (alignment, lexical scores, language model, BLEU scoring)
- supports syntax-based MT

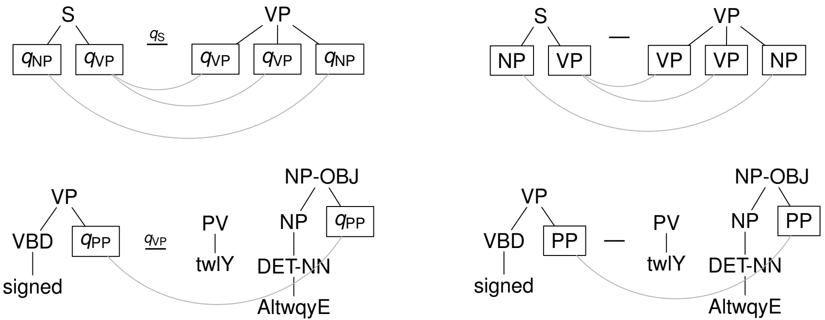
We added

- XMBOT rule support
- XMBOT chart decoder
- adjusted language model calls

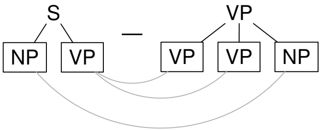
XMBOT rule encoding



XMBOT rule encoding

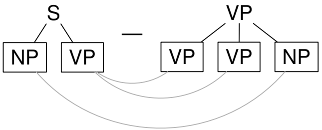


XMBOT rule encoding

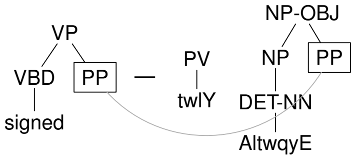



```
S(NP,VP) ||| VP(VP,VP,NP) ||| S ||| VP ||| 0-2 1-0 1-1 ||| ...
```

XMBOT rule encoding



`S(NP,VP) ||| VP(VP,VP,NP) ||| S ||| VP ||| 0-2 1-0 1-1 ||| ...`



`VP(VBD(signed),PP) ||| PV(twlY) || NP-OBJ(NP(DET-NN(AltwqyE)),PP) |||`

`VP ||| PV NP-OBJ ||| || 0-0 ||| ...`

XMBOT decoder

FABIENNE BRAUNE

- CYK-like chart parser
- only forward application (backward planned)
- supports all standard features
- integrated cube pruning with language model

Notes

- reasonably fast
- generated the examples in Motivation (with only translation weights)
- we are still working on it

XMBOT decoder

Fabienne Braune

- CYK-like chart parser
- only forward application (backward planned)
- supports all standard features
- integrated cube pruning with language model

Notes

- reasonably fast
- generated the examples in Motivation
  (with only translation weights)
- we are still working on it

XMBOT decoder

Fabienne Braune

- CYK-like chart parser
- only forward application (backward planned)
- supports all standard features
- integrated cube pruning with language model

Notes

- reasonably fast
- generated the examples in Motivation (with only translation weights)
- we are still working on it

XMBOT external tools

Nina Seemann

- rule extraction
- input/output restriction
- EM training
- conversion tools, pipeline scripts, . . .

Notes

- in Python (not inside Moses)
- computationally quite expensive
- variants for reduced POS-tags

XMBOT external tools

Nina Seemann

- rule extraction
- input/output restriction
- EM training
- conversion tools, pipeline scripts, . . .

Notes

- in Python (not inside Moses)
- computationally quite expensive
- variants for reduced POS-tags

References

- AHO, ULLMAN: *The theory of parsing, translation, and compiling.* Prentice Hall. 1972
- ARNOLD, DAUCHET: Morphismes et bimorphismes d'arbres. *Theoret. Comput. Sci.* 20(1):33–93, 1982
- CHARNIAK, JOHNSON: Coarse-to-fine *n*-best parsing and MaxEnt discriminative reranking. In *ACL* 2005
- DENERO, PAULS, KLEIN: Asynchronous binarization for synchronous grammars. In *ACL* 2009
- ENGELFRIET: Bottom-up and top-down tree transformations — a comparison. *Math. Systems Theory* 9(3), 1975
- ENGELFRIET, MANETH: Macro tree translations of linear size increase are MSO definable. *SIAM J. Comput.* 32(4):950–1006, 2003
- ENGELFRIET, LILIN, MALETTI: Extended multi bottom-up tree transducers — composition and decomposition. *Acta Inf.* 46(8):561–590, 2009
- GALLEY, HOPKINS, KNIGHT, MARCU: What's in a translation rule? In *HLT-NAACL* 2004
- GRAEHL, KNIGHT, MAY: Training tree transducers. *Comput. Linguist.* 34(3):391–427, 2008
- GILDEA: On the string translations produced by multi bottom-up tree transducers. *Comput. Linguist.*, 2012 (to appear)
- KOEHN, HOANG, BIRCH, CALLISON-BURCH, FEDERICO, BERTOLDI, COWAN, SHEN, MORAN, ZENS, DYER, BOJAR, CONSTANTIN, HERBST: MOSES: open source toolkit for statistical machine translation. In ACL 2007
- MALETTI, SATTA: Parsing and translation algorithms based on weighted extended tree transducers. In *ATANLP* 2010
- MALETTI: An alternative to synchronous tree substitution grammars. *J. Nat. Lang. Engrg.* 17(2):221–242, 2011
- MALETTI: How to train your multi bottom-up tree transducer. In *ACL* 2011
- MALETTI: Every sensible extended top-down tree transducer is a multi bottom-up tree transducer. In *HLT-NAACL* 2012
- MAY, KNIGHT: TIBURON — a weighted tree automata toolkit. In *CIAA* 2006
- OCH, NEY: A systematic comparison of various statistical alignment models. *Comput. Linguist.* 29(1):19–51, 2003
- SCHMID: Trace prediction and recovery with unlexicalized PCFGs and slash features. In *COLING-ACL* 2006
- ZHANG, HUANG, GILDEA, KNIGHT: Synchronous binarization for machine translation. In *HLT-NAACL* 2006