Parsing Algorithms Based on Tree Automata

Andreas Maletti[1] and Giorgio Satta[2]

[1] Universitat Rovira i Virgili, Tarragona, Spain
[2] University of Padua, Italy

andreas.maletti@urv.cat

Paris — October 7, 2009

Parsing and CFG

Example (Context-free grammar)

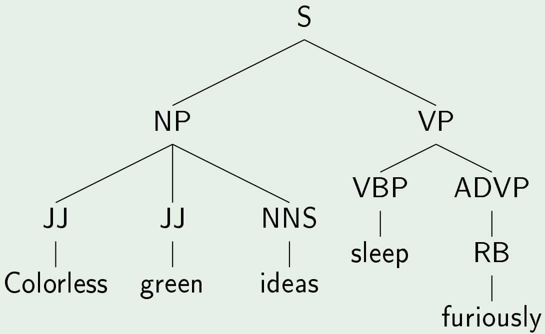
$$S \rightarrow NP\ VP \qquad NP \rightarrow JJ\ JJ\ NNS$$
$$VP \rightarrow VBP\ ADVP \qquad ADVP \rightarrow RB$$
$$JJ \rightarrow \textit{Colorless} \qquad JJ \rightarrow \textit{green}$$
$$NNS \rightarrow \textit{ideas} \qquad VBP \rightarrow \textit{sleep}$$
$$RB \rightarrow \textit{furiously}$$

Derivation

$S \rightarrow^*$ Colorless green ideas sleep furiously

Parse tree

Example

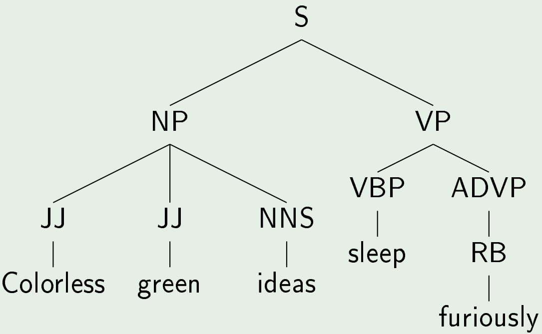


Remark

We are interested in the parse tree, not just whether $S \longrightarrow^* w$!

Parse tree

Example



Remark

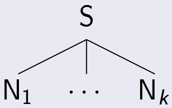
We are interested in the parse tree, not just whether $\mathrm{S} \longrightarrow^* w$!

Local tree language

Definition

A local tree grammar is a grammar with rules of the form

$S \rightarrow$



The such generated languages are the local tree languages.

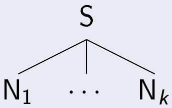
Theorem

The derivations of a context-free grammar are a local tree language.

Local tree language

Definition

A local tree grammar is a grammar with rules of the form

$S \rightarrow$ 

The such generated languages are the local tree languages.

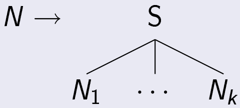
Theorem

The derivations of a context-free grammar are a local tree language.

Regular tree language

Definition

A regular tree grammar is a grammar with rules of the form



The such generated languages are the regular tree languages.

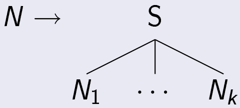
Remark

Regular tree grammars are local tree grammars with hidden states.

Regular tree language

Definition

A regular tree grammar is a grammar with rules of the form



The such generated languages are the regular tree languages.

Remark

Regular tree grammars are local tree grammars with hidden states.

Back to parsing

Observation

Most CFG-parsers are regular tree grammars (+ control) because

- they are based on a CFG ($\rightarrow$ local tree grammar) and
- have hidden states (or features)

Alternative

The features can be made explicit in the parse tree structure.

Back to parsing

Observation

Most CFG-parsers are regular tree grammars (+ control) because

- they are based on a CFG ($\rightarrow$ local tree grammar) and
- have hidden states (or features)

Alternative

The features can be made explicit in the parse tree structure.

Regular restriction

Theorem [Bar-Hillel et al '64]

The intersection of a context-free language with a regular language is again context-free.

Regular restriction — Trivial approach

Theorem

For every regular language L, the set of all trees, whose yield is in L, is regular.

Theorem

The intersection of two regular tree languages is regular.

Theorem

For every regular tree language the restriction to a regular language of yields is again regular.

Regular restriction — Trivial approach

Theorem

For every regular language L, the set of all trees, whose yield is in L, is regular.

Theorem

The intersection of two regular tree languages is regular.

Theorem

For every regular tree language the restriction to a regular language of yields is again regular.

Regular restriction — Trivial approach

Theorem

For every regular language L, the set of all trees, whose yield is in L, is regular.

Theorem

The intersection of two regular tree languages is regular.

Theorem

For every regular tree language the restriction to a regular language of yields is again regular.

The End?

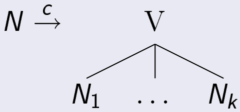
Thank you for your attention!

Contents

Weighted tree grammar

Definition

A weighted tree grammar is a structure $(\mathcal{N}, \Sigma, \mathbb{K}, N_0, R)$ with rules of the form

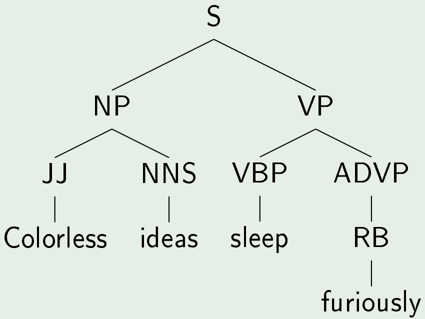


where

- $N, N_1, \ldots, N_k \in \mathcal{N}$ are nonterminals
- $c \in \mathbb{K}$ is a weight (taken from a semiring)
- $\mathrm{V} \in \Sigma$ is a terminal symbol

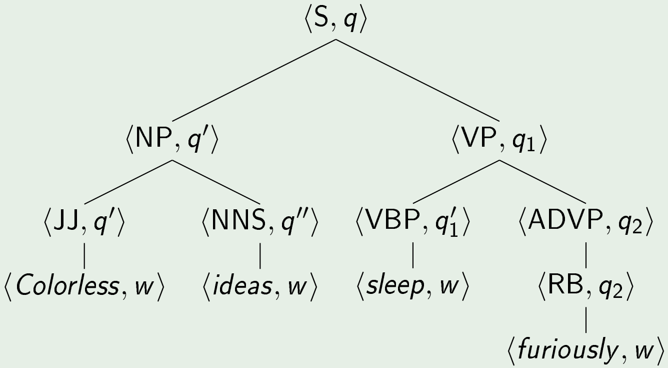
Runs

Example (Input tree)



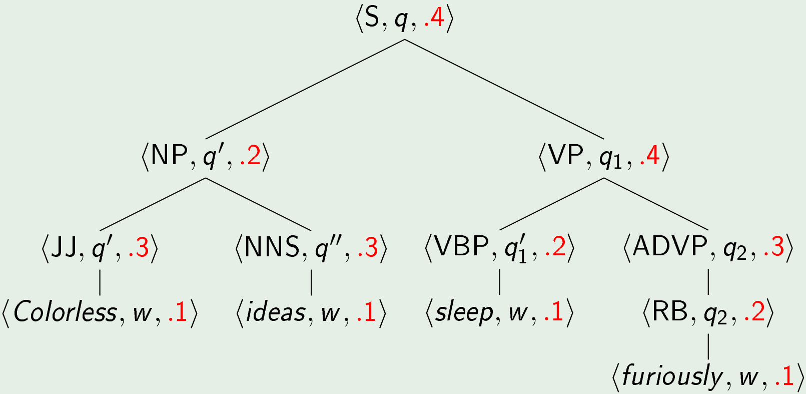
Runs

Example (Run)



Runs

Example (Run with weights)



Weight of a run

Definition

The weight of a run is obtained by multiplying the weights in it.

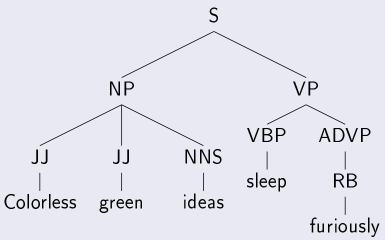
Definition

The weight of an input tree is obtained by adding the weights of all runs on it.

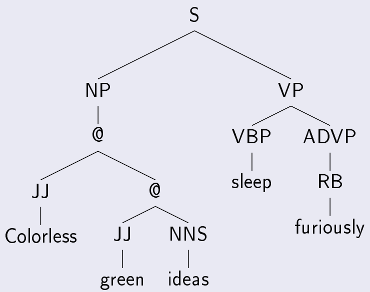
Contents

Binarization

Input tree



Binarized tree

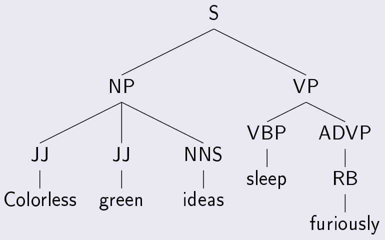


Theorem

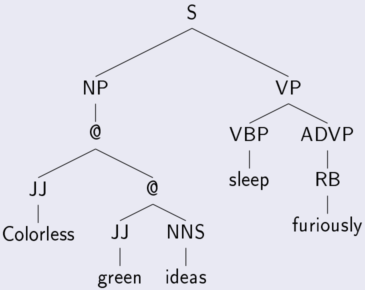
A tree language is regular if and only if its binarization is regular (also holds in the weighted setting).

Binarization

Input tree



Binarized tree



Theorem

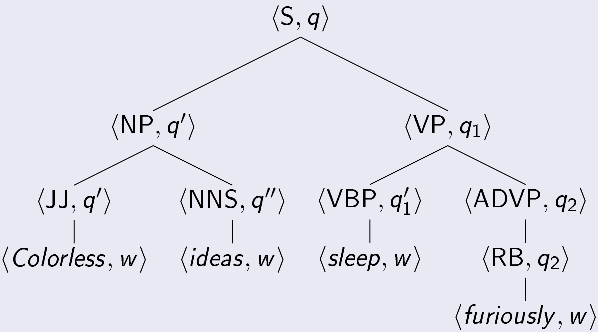
A tree language is regular if and only if its binarization is regular (also holds in the weighted setting).

Individual runs

Run on the yield

($\color{red}{p}$) Colorless ($\color{red}{p_1}$) ideas ($\color{red}{p_2}$) sleep ($\color{red}{p_3}$) furiously ($\color{red}{p'}$)

Run on the input tree

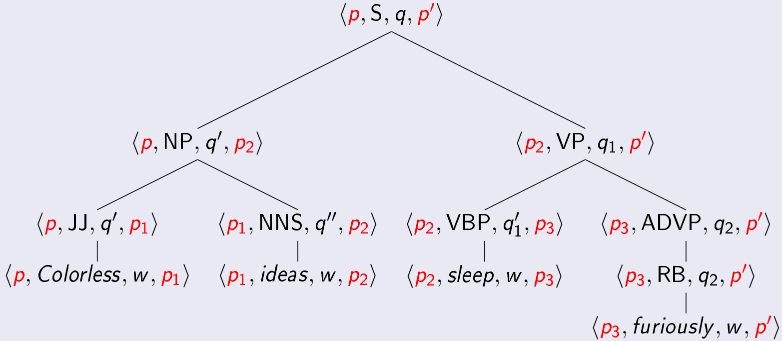


Bar-Hillel construction

Run on the yield

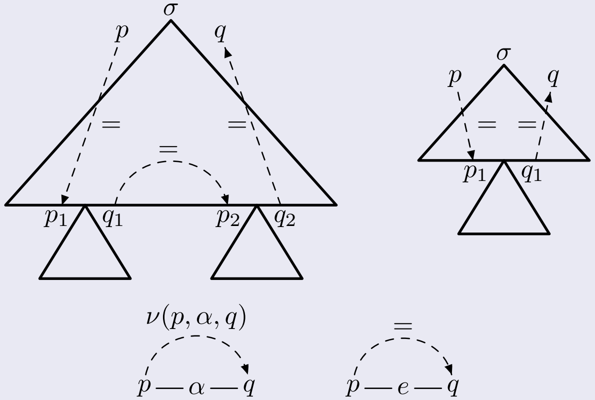
(p) Colorless (p_1) ideas (p_2) sleep (p_3) furiously (p')

Composite run



Bar-Hillel construction (cont'd)

Illustration



Bar-Hillel construction (cont'd)

Theorem

The regular restriction of a regular tree language is regular (also in the weighted setting).

Remark

Complexity: $O(mn^3)$

- m: size of the regular tree grammar
- n: size of the regular grammar (or input string)

Conclusion

We can parse with regular tree grammars in $O(mn^3)$.

Contents

Further topics — Probability mass

Definition

The probability mass of a nonterminal is the sum of the weights of runs with that nonterminal at the root.

Algorithm

The probability mass of a state can be computed using fix-point iteration.

Further topics — Probability mass

Definition

The probability mass of a nonterminal is the sum of the weights of runs with that nonterminal at the root.

Algorithm

The probability mass of a state can be computed using fix-point iteration.

Further topics — Normalization

Definition

A weighted regular tree grammar is

- convergent if the sum of weights assigned to trees are (uniformly) bounded
- proper if the probabilities of rules for one nonterminal add to 1
- consistent if it computes a probability distribution.

Theorem

For every convergent grammar, there exist a scaled proper and consistent grammar.

Further topics — Normalization

Definition

A weighted regular tree grammar is

- convergent if the sum of weights assigned to trees are (uniformly) bounded
- proper if the probabilities of rules for one nonterminal add to 1
- consistent if it computes a probability distribution.

Theorem

For every convergent grammar, there exist a scaled proper and consistent grammar.

Further topics — Best parse

Theorem

For an unambiguous grammar we can compute the best parse in linear time.

Proof.

Using a variant of Knuth's algorithm. □

References

- Bar-Hillel, Perles, Shamir: *On formal properties of simple phrase structure grammars.* Language and Information, 1964
- Berstel, Reutenauer: Recognizable formal power series on trees. *Theoret. Comput. Sci.* 18, p. 115–148, 1982
- Borchardt: The theory of recognizable tree series. Ph.D. thesis TU Dresden, 2005
- Gécseg, Steinby: *Tree Automata*. Akadémiai Kiadó, Budapest 1984
- Nederhof, Satta: Probabilistic parsing as intersection. In IWPT 2003

Thank you for your attention!