Unsupervised Natural Language Parsing

Kewei Tu ShanghaiTech University
Yong Jiang Alibaba DAMO Academy
Wenjuan Han National University of Singapore
Yanpeng Zhao University of Edinburgh

Tutorial Overview

1.	Introduction	(Kewei)
2.	Generative Approaches	(Kewei, Yong)
3.	Discriminative Approaches	(Wenjuan)
4.	Special Topics	(Yanpeng)
5.	Summary	(Kewei)

2. Generative Approaches

Generative Approaches

A generative approach models the joint generation of sentence x and parse tree z.

- Main approach: learning a probabilistic generative grammar
  - Context-free grammar (CFG)
  - Dependency model with valence (DMV)
  - Other:
    - Tree substitution grammar (Bod, 2006a,b; Cohn et al., 2010; Blunsom & Cohn, 2010)
    - Combinatory categorial grammar (Bisk & Hockenmaier, 2012, 2013; Bisk et al., 2015)
- Other generative approaches
  - Constituent Context Model (Klein and Manning, 2002; Golland et al., 2012)
  - Language model with structural constraints (Shen et al., 2017; 2018)

The focus of part 2

To be discussed in part 4

Outline

- Structure Learning (Kewei)
- Parameter Learning (Yong)

Outline

- Structure Learning (Kewei)
- Parameter Learning (Yong)

Structure Learning

- Context-free grammar (CFG)
  - Σ: terminal symbols
  - N: nonterminal symbols
  - S: start symbol
  - R: production rules

Vocabulary of the language; no learning required

“Structure” of the grammar; must be learned

- Structure learning
  - Finding an optimal set of production rules
- Two classes of approaches
  - Heuristic approaches
  - Optimization-based approaches

Heuristic approaches

- Create nonterminals and production rules using heuristic criteria and rules
- No explicit learning objective

Three typical steps in a heuristic approach:

- Constituent filtering
- Nonterminal creation
- Reduce and repeat

Heuristic approaches - Constituent filtering

Goal: identify substrings in the training sentences that are likely constituents

Methods:

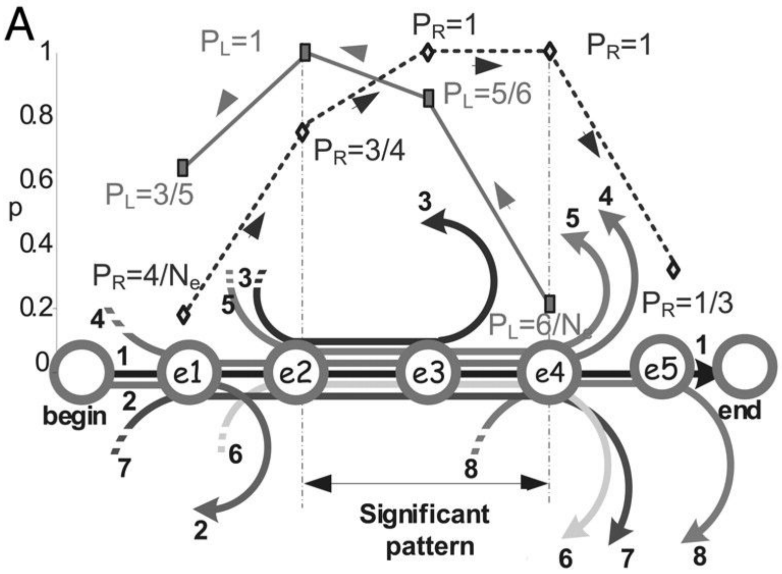
1. Frequency of a substring
2. Mutual information between the symbols occurring *before* and *after* a substring (Clark, 2001)
    - Constituents often have high MI

Heuristic approaches - Constituent filtering

Goal: identify substrings in the training sentences that are likely constituents

Methods:

3. Ratio of fan-through to fan-in
   (Solan et al., 2005)



Heuristic approaches - Nonterminal creation

Goal: create a new nonterminal representing a set of constituents

Methods:

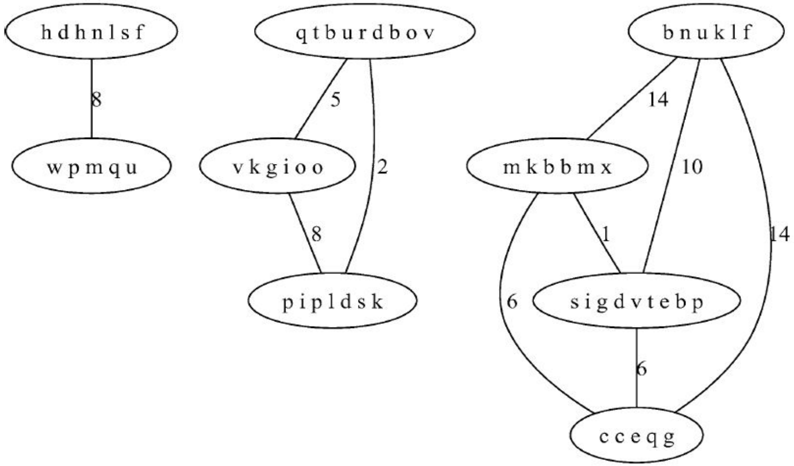
1. Substitutability heuristic (Adriaans et al., 2000; van Zaanen, 2000; Solan et al., 2005; Clark, 2007)
    - “Constituents of the same type can be replaced by each other” (Harris, 1951)
    - Create a nonterminal for substrings that appear in the same context (i.e., these substrings are substitutable)

Heuristic approaches - Nonterminal creation

Goal: create a new nonterminal representing a set of constituents

Methods:

1. Substitutability heuristic (Adriaans et al., 2000; van Zaanen, 2000; Solan et al., 2005; Clark, 2007)
   - Substitution-graph



Heuristic approaches - Nonterminal creation

Goal: create a new nonterminal representing a set of constituents

Methods:

1. Substitutability heuristic (Adriaans et al., 2000; van Zaanen, 2000; Solan et al., 2005; Clark, 2007)

	John (.) tea	John (.) coffee	John (.) eating	John makes (.)	John likes (.)	John is (.)
makes	x	x				
likes	x	x	x			
is			x			
tea				x	x	
coffee				x	x	
eating					x	x

Heuristic approaches - Nonterminal creation

Goal: create a new nonterminal representing a set of constituents

Methods:

2. Biclustering (Adriaans et al., 2000)
    - Simultaneously group substrings and their contexts

	John (.) tea	John (.) coffee	John (.) eating	John makes (.)	John likes (.)	John is (.)
makes	x	x				
likes	x	x	x			
is			x			
tea				x	x	
coffee				x	x	
eating					x	x

Heuristic approaches - Nonterminal creation

Goal: create a new nonterminal representing a set of constituents

Methods:

3. Distributional clustering (Harris, 1954; Clark, 2001; Scicluna and de la Higuera,2014)
   - Cluster substrings based on their distributions over possible contexts
   - Based on co-occurrence frequencies, not just yes/no. Hence more robust.
   - Can be extended to biclustering (Tu&Honavar, 2008)

Heuristic approaches - Nonterminal creation

Goal: create a new nonterminal representing a set of constituents

Methods:

3. Distributional clustering (Harris, 1954; Clark, 2001; Scicluna and de la Higuera,2014)

Different contextual distributions

	John (.) tea	John (.) coffee	John (.) eating	John makes (.)	John likes (.)	John is (.)
makes	1	2				
likes	1	1	2			
is			1			
tea				1	1	
coffee				2	1	
eating					2	1

Distributional biclustering can be more robust (Tu&Honavar, 2008)

Heuristic approaches - Reduce & repeat

Once a nonterminal is created, along with a set of production rules, reduce the training sentences using the rules and then repeat the previous steps.

V → makes | likes

John makes tea .
John likes tea .
John makes coffee .
John likes coffee .

John V tea .
John V tea .
John V coffee .
John V coffee .

Optimization-based approaches

Optimizing an explicit objective function of the grammar structure by local search:

- Start with a trivial grammar
- Search with a set of structure-change operations

Optimization-based approaches

Objective functions

- Posterior probability (Stolcke and Omohundro, 1994; Chen, 1995; Tu&Honavar, 2008)

$$P(G|X) \propto \underline{P(X|G)}\,\underline{P(G)}$$

Likelihood, computed by parsing corpus *X* using grammar *G*.

Rule probabilities in *G* are either heuristically assigned or learned.

Prior probability. A typical choice is the *universal a priori probability*

$$P(G) = 2^{-L(G)}$$

L(*G*) is the description length of *G* in bits

Trade-off between data fitting and model complexity (generalizability)

Optimization-based approaches

Objective functions

- Description length (Langley&Stromsten, 2000)
  - Equivalent to posterior probability with the above prior
- Free energy (negative evidence lower bound) (Kurihara&Sato, 2006)

Optimization-based approaches

Start point of local search

1. Union of training sentences (Stolcke and Omohundro, 1994; Langley&Stromsten, 2000; Tu&Honavar, 2008)

Ex: S -> John makes tea
S -> John makes coffee
...
S -> John is eating

all the training sentences

➢ Perfect fitting of training data
➢ No generalizability

Optimization-based approaches

Start point of local search

2. Most general grammar (Chen, 1995; Kurihara&Sato, 2006)

Ex: S -> S X | X
X -> a | b | c | ...

all the terminals

- ➢ Bad fitting of training data
- ➢ Can generate any sentence

Optimization-based approaches

Structure-change operations

1. AND (a.k.a. composition, chunk) (Stolcke and Omohundro, 1994; Chen, 1995; Langley&Stromsten, 2000)

Add a new rule: A -> B C

a new symbol

existing symbols

Replace “BC” with A in the right-hand side of other rules

Ex: “X -> B C D” becomes “X -> A D”

Optimization-based approaches

Structure-change operations

2. OR (i.e., alternatives) (Chen, 1995)

Add a new rule: A -> B | C

a new symbol existing symbols

Replace B and C with A in the right-hand side of other rules

Ex: “X -> C D” becomes “X -> A D”

Optimization-based approaches

Structure-change operations

3. AND-OR (Tu&Honavar, 2008)

O_1 -> B_1 | B_2 | ...
Add new rules: A -> O_1 O_2 ... and O_2 -> C_1 | C_2 | ...
...

new symbols

existing symbols

Replace sequences "B_i C_j ..." with A in the right-hand side of other rules
Ex: "X -> B_2 C_1 ..." becomes "X -> A"

Optimization-based approaches

Structure-change operations

4. Merging two existing symbols (Stolcke and Omohundro, 1994; Langley&Stromsten, 2000; Kurihara&Sato, 2006)
5. Splitting an existing symbol to two and making copies of rules involving the symbol (Kurihara&Sato, 2006)
6. Deleting a rule (Kurihara&Sato, 2006)

Optimization-based approaches

Reevaluating the objective function after each structure-change operation

- A complete reevaluation is time-consuming
  - Requires re-parsing of all the training sentences
- Simple formulas may exist for computing the change of the objective function value
  - Only require computation on local changes
  - May be approximate

Structure Learning - Summary

- Goal: Finding an optimal set of production rules
- Two classes of approaches
  - Heuristic approaches
  - Optimization-based approaches

- Empirical results
  - Poor accuracies on real data, often below simple baselines 😟

Outline

- Structure Learning (Kewei)
- Parameter Learning (Yong)