Syntactic Formalisms for Parsing Natural Languages

Aleš Horák, Miloš Jakubíček, Vojtěch Kovář
(based on slides by Juyeon Kang)

ia161@nlp.fi.muni.cz

Autumn 2013

Parsing Evaluation

Parsing Results

- usually some complex (i.e. non-scalar) structure, mostly a tree or a graph-like structure
- crucial question: how to measure the “goodness” of the result?

Extrinsic vs. Intrinsic Evaluation

- Intrinsic
  - by comparing to a “gold”, i.e. correct, representation
- Extrinsic
  - by exploiting the result in a 3rd party task and evaluating its results
- Which is better?

Intrinsic Evaluation – Phrase-Structure Syntax

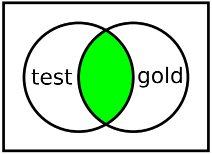
- i.e. compare two phrase-structure trees and tell a number
- PARSEVAL metric
- LAA (Leaf-ancestor assessment) metric

PARSEVAL metric

- basic idea: penalize crossing brackets in the tree
- i.e. compare all constituents in the test tree to the gold tree
- $\Rightarrow$ parsing viewed as classification problem

Precision, recall

- for classification problems in NLP, the standard evaluation is by means of precision and recall



$$\text{precision} = \frac{|\text{test} \cap \text{gold}|}{|\text{test}|} \quad \text{recall} = \frac{|\text{test} \cap \text{gold}|}{|\text{gold}|}$$

- two numbers, we just want to have one – F-score

$$\text{F}_1 \text{ score} = \frac{2 \cdot \text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$$

F-score

- also F-measure
- general form: F_β score

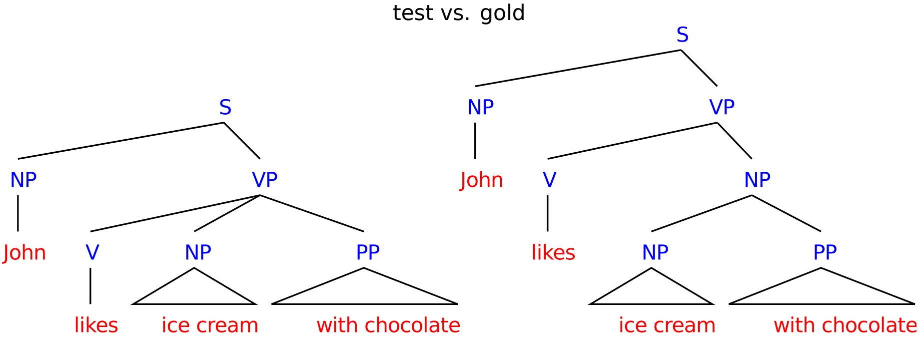
$$F_\beta \text{ score} = (1+\beta^2) \cdot \frac{\text{precision} \cdot \text{recall}}{(\beta^2 + \text{precision}) + \text{recall}}$$

- special case of $\beta = 1$ corresponds to the harmonic mean of precision and recall
- β can be used for favouring precision over recall (for $\beta < 1$) or vice versa (for $\beta > 1$)

PARSEVAL metric

- basic idea: penalize crossing brackets in the tree
- i.e. compare all constituents in the test tree to the gold tree
- ⇒ parsing viewed as classification problem
- ⇒ F-score on correct bracketings/constituents
- might even disregard non-terminal names
- sort of standardized tool available: the `evalb` script at `http://nlp.cs.nyu.edu/evalb/`

PARSEVAL metric – example

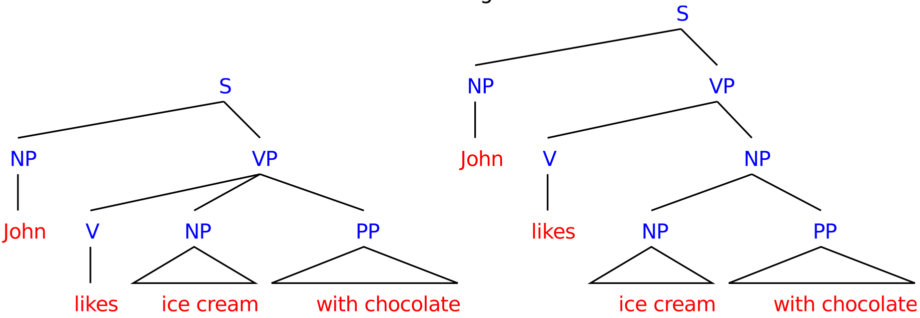



```
test:[S [NP John][VP [V likes][NP ice cream] [PP with chocolate]]]
gold:[S [NP John][VP [V likes][NP [NP ice cream] [PP with chocolate]]]]
```

precision = 6/6 = 1.0, recall = 6/7 = 0.86, F-score = 0.92

PARSEVAL metric

test vs. gold



`test:[S [NP John][VP [V likes][NP ice cream] [PP with chocolate]]]`

`gold:[S [NP John][VP [V likes][NP [NP ice cream] [PP with chocolate]]]]`

precision = 6/6 = 1.0, recall = 6/7 = 0.86, F-score = 0.92

PARSEVAL metric

- often subject to criticism (see e.g. Sampson, 2000)
- Sampson proposed another metric, the leaf-ancestor assessment (LAA)

LAA metric

- basic idea: for each leaf (word), compare the path to the root of the tree, compute the edit distance between both paths, finally take the average of all words
- in the previous example, the paths (lineages) are:
  - (John) NP S vs. (John) NP S
  - (likes) V VP S vs. (likes) V VP S
  - (ice cream) NP VP S vs. (ice cream) NP NP VP S
  - (with chocolate) PP VP S vs. (with chocolate) PP NP VP S

Intrinsic Evaluation – Dependency Syntax

- much easier
- just precision, labeled or unlabeled (as the number of correct dependencies)

Intrinsic Evaluation – Building Treebanks

- treebank = a syntactically annotated text corpus
- manual annotation according to some guidelines
- from the evaluation point of view: inter-annotator agreement (IAA) is a crucial property

Measuring IAA

- naïve approach: count how many times people agreed on
- problem: it does not account for agreement by chance

Chance-corrected coefficients for IAA

- S (Benett, Alpert and Goldstein, 1954)
- π (Scott, 1955)
- κ (Cohen, 1960)
- (there is lot of terminology confusion, we follow Ron Artstein, Massimo Poesio: Inter-coder Agreement for Computational Linguistics, 2008)
- A_o – observed agreement
- A_e – expected (chance) agreement
- for all coefficients, they compute:

$$S, \pi, \kappa = \frac{A_o - A_e}{1 - A_e}$$

Chance-corrected coefficients for IAA

- *S* (Benett, Alpert and Goldstein, 1954)
  - assumes that all categories and all annotators have uniform probability distribution
- π (Scott, 1955)
  - assumes that different categories have different distributions shared across annotators
- κ (Cohen, 1960)
  - assumes that different categories and different annotators have different distributions
- devised for 2 annotators, various modifications for more than 2 annotators available

Intrinsic Evaluation – Conclusions

- generally not easy
- builds on the assumption of having THE correct parse
- there is evidence that it does not correlate with extrinsic evaluation, i.e. how good the tool is for some particular job

Extrinsic Evaluation

- = evaluation on a particular task/application
- advantages: measures direct fitness for that task
- disadvantages: may not generalize for other tasks

- leads to crucial question: what can be parsing used for?

What can parsing be used for?

- in theory, (full) parsing is suitable/appropriate/necessary for many NLP tasks
- practically it turns out to be:
  - often not accurate enough
  - often too complicated to exploit
  - sometimes just an overkill compared to shallow parsing or yet simpler approaches

What can parsing be used for?

- in theory, (full) parsing is suitable/appropriate/necessary for many NLP tasks
    - information extraction
    - information retrieval
    - machine translation
    - corpus linguistics
    - computer lexicography
    - question answering
    - ...

Where is parsing actually used now?

- prototype systems
- academia work
- production systems ???

What to evaluate parsing on

Sample (more or less well defined) applications

- (partial) morphological disambiguation
- text correcting systems
- word sketches
- phrase extraction
- simple treebank of high IAA