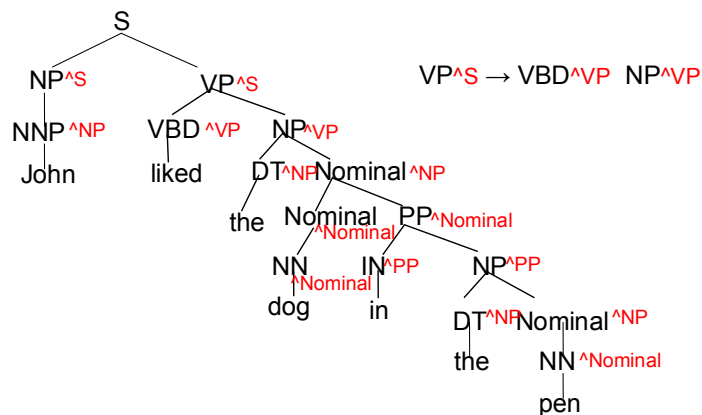# Splitting Non-Terminals

- To provide more contextual information, non-terminals can be split into multiple new non-terminals based on their parent in the parse tree using parent annotation.
  - A subject NP becomes NP^S since its parent node is an S.
  - An object NP becomes NP^VP since its parent node is a VP

44

Parent Annotation Example

VP^S → VBD^VP NP^VP

45

Split and Merge

- Non-terminal splitting greatly increases the size of the grammar and the number of parameters that need to be learned from limited training data.

- Best approach is to only split non-terminals when it improves the accuracy of the grammar.

- May also help to merge some non-terminals to remove some un-helpful distinctions and learn more accurate parameters for the merged productions.

- Method: Heuristically search for a combination of splits and merges that produces a grammar that maximizes the likelihood of the training treebank.

46

Human Parsing

- Computational parsers can be used to predict human reading time as measured by tracking the time taken to read each word in a sentence.

- Psycholinguistic studies show that words that are more probable given the preceding lexical and syntactic context are read faster.

47

Garden Path Sentences

- People are confused by sentences that seem to have a particular syntactic structure but then suddenly violate this structure, so the listener is "lead down the garden path".
  - The horse raced past the barn fell.
    - vs. The horse raced past the barn broke his leg.
  - The complex houses married students.
  - The old man the sea.
  - While Anna dressed the baby spit up on the bed.

48

Statistical Parsing Conclusions

- Statistical models such as PCFGs allow for probabilistic resolution of ambiguities.
- PCFGs can be easily learned from treebanks.
- Lexicalization and non-terminal splitting are required to effectively resolve many ambiguities.
- Current statistical parsers are quite accurate but not yet at the level of human-expert agreement.

49

Try at Home

- http://nlp.cs.berkeley.edu/software.shtml
  - http://tomato.banatao.berkeley.edu:8080/parser/parser.html

- https://nlp.stanford.edu/software/lex-parser.shtml
  - http://nlp.stanford.edu:8080/parser/