Elements of Language Processing and Learning
Lab assignment report
Stage 2: Adding support of Unary Rules in the CKY ALgorithm

Benno Kruit, 10576223
Sara Veldhoen, 10545298

December 4, 2013

The CKY algorithm was originally implemented for binary rules only. We extended the code to support unary rules.

1 Implementation

Grammar

- We created a general class `Rule` to replace the `BinaryRule` and `UnaryRule` classes. Occurrences of `Rule` can be either binary or unary rules. The method `getChildren()` returns an array of Strings. This enables other functions to iterate over the children in a more general way.
- We extended the class `Grammar` with `UnaryClosedGrammar`, which is a grammar that computes and stores the unary closure of all unary rules after instantiation. The methods `getUnaryRulesByParent(String)` and `getUnaryRulesByChild` are guaranteed to return the unary closure of unary rules.

Parsing

- The Parsing algorithm itself is run with the function `getBestParse(List<String>)`. The pseudocode of the algorithm is as follows:

```
match preterminal productions
for max = 1 : 1 : n
    for min = max − 1 : −1 : 0
        foreach C ∈ syntacticCategories
          foreach R ∈ binaryrules
            for mid = min + 1 : 1 : max − 1
                match binary rule, keep the best scoring rule and its midpoint
            end
          end
          store the best rule and midpoint in the chart
        end
        foreach C ∈ syntacticCategories
          foreach R ∈ unaryclosure
            match unary rule, keep the best scoring rule
          end
          if (bestScoringUnary > bestStoredInChart)
            replace this position in the chart with the unary rule
          end
        end
    end
end
```

- Traverse backpointers

2 Results