Anish Sachdeva
DTU/2K16/MC/13

Class Fun

Date

Page No.

IT-425 Natural Language Processing
Class Test-2

Q1) For the given grammar, considering the given numbering of the rules construct top down parser for "The small dog ate".

Ans1)

| Stack | Input | Action |
| --- | --- | --- |
| S | The small dog ate $ | |
| VP NP | The small dog ate $ | S → VP NP (1) |
| V NP | The small dog ate $ | (3) VP → V |

No match, we backtrack.

| Stack | Input | Action |
| --- | --- | --- |
| VP NP | The small dog ate $ | |
| V NP NP | The small dog ate $ | (5) VP → V NP |

No match, we backtrack.

| Stack | Input | Action |
| --- | --- | --- |
| S | The small dog ate $ | |
| NP VP | The small dog ate $ | (2) S → NP VP |
| Det Adj NP VP | The small dog ate $ | (4) NP → Det Adj NP |
| The Adj NP VP | small dog ate $ | Det → The (Match) |
| The small NP VP | dog ate $ | Adj → small (Match) |
| The small Det Adj NP VP | dog ate $ | |

No match, we backtrack.

DTU/2K16/ME/113 Anish Sachdeva

| Matched | Stack | Input | Action |
|---|---|---|---|
| The small | NP VP | dog ate $ | |
| The small | Det N VP | dog ate $ | (6) NP → Det N |

There is no match, so we backtrack.

| Matched | Stack | Input | Action |
|---|---|---|---|
| The small | NP VP | dog ate $ | |

We have exhausted all 2 possibilities for NP, namely

(4) NP → Det Adj NP
(6) NP → Det N

We have also exhausted all possibilities for S as,

(1) S → VP NP
(2) S → NP VP

and hence, as our stack is not empty, the given sentence "The small dog ate" is not grammatically correct as per the given grammar.

Class Fun

Date
Page No.

Q2) For the bottom-up parser, explain the shift and reduction operations using one example each.

In bottom up parsing the parsing tree is constructed from the leaves to the root of the tree. It is also called a shift reduce parser.

The shift and reduce operations are as follows:-

i) Shift operation: The shift operation advances the input pointer to the next input symbol which is called the shifted symbol. This symbol is pushed onto the stack. The shifted symbol is treated as a single node of the syntax parse tree.

ii) Reduction operation: When a parser encounters a complete grammar using RHS and replaces it to LHS. This is known as the reduction step. This occurrs when the top of the stack contains a handle.

To reduce a pop function is performed on the stack which pops at the handle and replaces it with LHS non-terminal symbol

We take up the following grammar as an example :-

$S \rightarrow S + S$
$S \rightarrow S * S$
$S \rightarrow id$

We now perform shift-reduce parsing / Bottom up Parsing for the input string

id+id+id

| Stack | Input Buffer | Parsing Action |
|---|---|---|
| \$ | id+id+id \$ | Shift |
| \$id | +id+id \$ | Reduce S→id |
| \$ S | +id +id \$ | Shift |
| \$ S+ | id +id \$ | Shift |
| \$ S+id | +id \$ | Reduce by S→id |
| \$ S+S | +id \$ | Shift |
| \$ S+S+ | id \$ | Shift |
| \$ S+S+id | \$ | Reduced by S→id |
| \$ S+S+S | \$ | Reduced by S→id |
| \$ S+S | \$ | Reduced by S→S+S |
| \$ S | \$ | Accepted! |

Anish Sachdeva
DTU/2K16/MC/13



Q3) Explain the feature computation using a fictional word embedding (with values of your choice) for a sample sentence "the small dog ate" and 1D CNN. Assume a 3-dimensional word embedding.

We are given that each word is a 3-dimensional vector, so we assume the following values for our words :-

$$W = \begin{array}{c} \text{The} \\ \text{Small} \\ \text{dog} \\ \text{ate} \end{array} \begin{bmatrix} 0.1 & 0.7 & 0 \\ -0.2 & 0 & 0.5 \\ 0.3 & 0.4 & 0.4 \\ 0.8 & 0.1 & -0.3 \end{bmatrix}$$

We now take one kernel $f_1$ (which will be a $2 \times 3$ matrix, where we assume bigrams, hence 2 and our word embeddings have length 3)

$$f_1 = \begin{bmatrix} 0.4 & 0.5 & 0.2 \\ 0.1 & -0.3 & 0.8 \end{bmatrix}$$

Convoluting $W \oplus f_1$ we get

$$W \oplus f_1 = \begin{bmatrix} 0.77 \\ 0.25 \\ 0.21 \end{bmatrix}$$

From max pooling we get $= 0.77$

DTU/2K16/MC/13
Amish Sachdeva

We now take kernel 2 $f_2$ as

$$f_2 = \begin{bmatrix} 0.1 & 0.2 & 0.3 \\ 0.9 & -0.7 & 0 \end{bmatrix}$$

Convoluting, we get

$$W \oplus f_2 = \begin{bmatrix} -0.03 \\ 0.12 \\ 0.88 \end{bmatrix}$$

After maxpooling, we obtain = 0.88

So, $P_1 = 0.77$, $P_2 = 0.88$

We obtain the feature matrix as

$$\cancel{F = \frac{P_1}{P_2} = 0.}$$

$$F = \begin{bmatrix} P_1 \\ P_2 \end{bmatrix} = \begin{bmatrix} 0.77 \\ 0.88 \end{bmatrix}$$

This is the resulting feature matrix.