**Fine vs Coarse POS Tagging**

**Part 1: Fine-grained POS Tagging**

Following table gives the accuracy achieved.

| Tag | Accuracy% |
|---|---|
| `` | 100 |
| $ | 100 |
| , | 100 |
| . | 100 |
| -LRB- | 100 |
| : | 100 |
| EX | 100 |
| WP$ | 100 |
| -RRB- | 100 |
| '' | 99 |
| DT | 99 |
| CC | 99 |
| TO | 98 |
| PRP | 97 |
| WDT | 96 |
| NN | 96 |
| MD | 95 |
| POS | 94 |
| WP | 94 |
| WRB | 93 |
| IN | 93 |
| -NONE- | 89 |
| RBS | 75 |
| VBZ | 70 |
| RB | 69 |
| PRP$ | 67 |
| VBP | 67 |
| VBD | 63 |
| CD | 63 |
| JJS | 62 |
| RP | 59 |
| RBR | 50 |
| VB | 46 |
| JJ | 40 |
| NNP | 40 |
| JJR | 40 |
| VBG | 36 |
| VBN | 34 |
| NNS | 33 |
| NNPS | 2 |
| FW | 0 |
| PDT | 0 |
| SYM | 0 |
| UH | 0 |
| Overall | 76.23 |

**Observations made:**

1. The tags "``" and "$" among many others have a 100% accuracy. The reason for this is that there is no ambiguity with respect to these tags as they always tag to the very same literals in training sentences.
2. This is true even when a bigram tagger is used as these tags remain the same irrespective of the tag of the previous word in the sentence.
3. The tags "UH" and "SYM" which correspond to tokens "OK" and "&" have showed 0% accuracy. The reason being these tokens and hence the tags never appeared in the trained data.

Confusion matrix for fine-grained POS tagging:

| * | JJ | NN | NNP | NNPS | RB | RP | IN | VB | VBD | VBN | VBP |
|---|---|---|---|---|---|---|---|---|---|---|---|
| JJ | 323 | 436 | 2 | 0 | 13 | 0 | 2 | 1 | 0 | 8 | 1 |
| NN | 12 | 1535 | 4 | 0 | 1 | 0 | 0 | 31 | 0 | 1 | 6 |
| NNP | 7 | 714 | 496 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| NNPS | 0 | 32 | 4 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RB | 5 | 73 | 0 | 0 | 228 | 3 | 11 | 1 | 0 | 0 | 0 |
| RP | 0 | 1 | 0 | 0 | 5 | 22 | 9 | 0 | 0 | 0 | 0 |
| IN | 2 | 16 | 0 | 0 | 12 | 3 | 1169 | 0 | 0 | 0 | 0 |
| VB | 2 | 130 | 0 | 0 | 0 | 0 | 0 | 143 | 0 | 2 | 31 |
| VBD | 0 | 113 | 0 | 0 | 0 | 0 | 0 | 1 | 271 | 41 | 0 |
| VBN | 0 | 132 | 0 | 0 | 0 | 0 | 0 | 1 | 23 | 82 | 0 |
| VBP | 0 | 41 | 0 | 0 | 0 | 0 | 1 | 12 | 0 | 1 | 114 |

**Part 2: Coarse-grained POS Tagging**

Following table gives the accuracy achieved in method A.

| Tag | Accuracy% |
|---|---|
| SNN | 98 |
| MISC | 96 |
| SRB | 68 |
| SVB | 61 |
| SJJ | 41 |
| Overall | 87.89 |

Following table gives the accuracy achieved in method B.

| Tag | Accuracy% |
|---|---|
| SNN | 98 |
| MISC | 96 |
| SRB | 66 |
| SVB | 60 |
| SJJ | 41 |
| Overall | 87.67 |

Observations made:

1. I feel method A should relatively perform better. The reason being, the accuracy of any POS tagging model depends on the training data. And fine-grained tagged data would capture more context which is lost if it were coarse-gained.
2. Since in method A, the tagger is trained on fine-grained data as opposed to method B's coarse-grained data, method-A should relatively perform better

Confusion matrix for method A:

| * | SNN | MISC | SRB | SVB | SJJ |
|---|---|---|---|---|---|
| SNN | 3920 | 2 | 1 | 45 | 19 |
| MISC | 209 | 5551 | 18 | 0 | 2 |
| SRB | 74 | 18 | 240 | 1 | 18 |
| SVB | 621 | 6 | 0 | 1011 | 2 |
| SJJ | 452 | 5 | 21 | 11 | 347 |

Confusion matrix for method B:

| * | SNN | MISC | SRB | SVB | SJJ |
|---|---|---|---|---|---|
| SNN | 3912 | 6 | 1 | 49 | 19 |
| MISC | 208 | 5550 | 20 | 0 | 2 |
| SRB | 75 | 22 | 234 | 0 | 20 |
| SVB | 627 | 12 | 0 | 998 | 3 |
| SJJ | 451 | 5 | 23 | 10 | 347 |