Maria Paula de Alencar Viegas

Marco Natali

Artificial Intelligence 2019/2020

Assignment 7: Classification

1. Data Exploration and Preprocessing

We have implemented the panda library, using the read\_cvs(), count(), value\_counts(), and groupby() functions to analyze the provided cvs files. After inspecting the data, we were able to identify **41 POS tags** with the following frequency:

|  |  |
| --- | --- |
| POS | FREQUENCY |
| NN | 15316 |
| NNP | 13831 |
| IN | 12778 |
| DT | 10342 |
| JJ | 8285 |
| NNS | 7976 |
| . | 4989 |
| VBD | 4123 |
| , | 3528 |
| VBN | 3478 |
| CD | 2584 |
| VBZ | 2552 |
| CC | 2523 |
| VB | 2496 |
| TO | 2388 |
| RB | 2136 |
| VBG | 2020 |
| VBP | 1686 |
| PRP | 1358 |
| POS | 1185 |
| PRP$ | 934 |
| MD | 703 |
| WDT | 404 |
| `` | 362 |
| JJR | 314 |

|  |  |
| --- | --- |
| JJS | 309 |
| WP | 263 |
| NNPS | 234 |
| RP | 233 |
| WRB | 217 |
| $ | 130 |
| RBR | 123 |
| : | 105 |
| RRB | 69 |
| LRB | 69 |
| EX | 65 |
| RBS | 32 |
| ; | 20 |
| PDT | 16 |
| WP$ | 8 |
| UH | 4 |

The most common words within each data are

**POS WORD FREQUENCY**

UH Ah 1

WP$ whose 8

PDT all 9

; ; 20

RBS most 32

RBR earlier 42

EX there 44

LRB ( 68

RRB ) 68

: - 90

VBG saying 92

RP up 95

WRB when 119

$ $ 130

NNPS States 132

JJS least 153

JJ last 168

JJR more 184

WP who 200

WDT that 207

VB be 254

RB not 266

NNS people 267

VBN been 268

CD two 290

PR it 294

NN government 321

`` " 353

MD will 354

PRP$ his 357

NNP U.S. 448

VBD was 517

VBP have 517

VBZ has 781

POS 's 1144

CC and 2136

TO to 2387

IN in 2786

, , 3528

. . 4983

DT the 5481

1. Rule Based Tagging

This is the first approach for tagging. We have manually implemented the following rules for more simple tags:

|  |  |
| --- | --- |
| Words ending with | POS tag |
| ing | VBG |
| ed | VBD |
| es | VBZ |
| s | NNS |
| uld | MD |
| y | RB |
| ^-?[0-9]+(\.[0-9]+)?$ | CD |
| est | JJS |

If none of the rules is applied, we tag the word as NN, since noun is the most common classifications in the English grammar. We are able to use those rules with any word, not depending on our sample data.

The accuracy was 19.394%, a good enough percentual for manually implemented rules.

1. Evaluating Performance

To evaluate the performance of our predictions, we are using the sklearn and the nltk libraries. The nltk allow us to implement our rules and tag the words in the test. The sklearn has the function sklearn.metrics.accuracy\_score() that we used to measure the accuracy of our solution.

Later, to evaluate the Linear Model Classification, we used nltk.classify.accuracy() to measure accuracy.

1. Feature Extraction
2. Classification
3. Better Feature Encoding