Part-of-speech tagging

Parts of speech, in English

- NN - noun (cat, John)
- VB - verb (walk, learn)
- JJ - adjective (fast, tall)
- RB - adverb (very, quickly)
- PRP - pronoun (her, they)
- IN - adposition/preposition (in, for)
- CC - conjunction (and, but)
- UH - interjection (whoa, lol)
- DT - article/determiner (a, the)
- RP - particle (off in “put off”, out in “get out”)
- TO - the infinitive "to"

‘to’ in I love to run
not ‘to’ in I love running to the gym

Label this

“The big barn toppled slowly to the ground.”

Label this, cont.

The	big	barn	toppled	slowly	to	the	ground
DT	JJ	NN	VB	RB	IN	DT	NN

Heuristic POS tagging - ambiguity

Many words can be multiple parts of speech.

e.g. “still” can be at least 4 parts of speech.

- noun - quietness “the still of the night”, apparatus for distillation "a whiskey still"
- adjective - not moving “be still now”
- verb - to make or become still "stilled the raging sea"
- adverb - up to and including the present “he’s still here”

Heuristic POS tagging - rules

Many "rules" govern arrangement of parts of speech:

- articles come right before nouns/adjectives
- adjectives come right before nouns, or act as objects
- pronouns replace noun/adjective sets
- adverbs come right before adjectives or other adverbs, or act as objects
- ...

Brill tagger

https://repository.upenn.edu/cgi/viewcontent.cgi?article=1193&context=ircs_reports

1. Assign most common POS from list

sign everything a preliminary state
and then, if you set up this set of rules and apply them iteratively,
then maybe you can correct things

- "still" is adverb

2. Iteratively correct with rules like *if DT JJ VB, change to DT NN VB*.

- e.g "the still was operational" → `DT JJ VB JJ` → `DT NN VB JJ`

Statistical POS tagging

"What is the probability of any given set of labels for a sequence of words?"

e.g. is "still flies" most likely to be labelled `JJ NN` ("The still flies slept.") or `RB VB` ("He still flies to Denver.")

Example manual POS inference

NN __ TO VB __ __ JJ NN

Bob likes to run though the Harry Dog

Bob likes to run slowly to grocery store

Example manual POS inference, cont.

NN __ TO VB __ __ JJ NN

Bob __ to run __ __ hairy dog

Bob likes to run with the hairy dog

NN VB TO VB IN DT JJ NN

Hidden Markov Model

Markov model:
current stage depends only on one previous state

A Markov model, but where the states are hidden.
Instead, a sequence of "observations" are visible,
that are stochastically related to the states.

observations are correlated with states, but are associated in a random way

Definitions

- $Q = (q_1, q_2, ..., q_N)$ a set of N states
-

$$A = \begin{bmatrix} a_{11} & ... & a_{1N} \\ ... & a_{ij} & ... \\ aN1 & ... & a_{NN} \end{bmatrix}$$

a transition probability matrix, each a_{ij} representing the probability of moving from state i to state j, s.t. $\sum_{j=1}^{N} a_{ij} = 1 \forall i$

Definitions, cont.

- $O = o_1 o_2 ... o_T$ a sequence of T observations, each one drawn from a vocabulary $V = (v_1, v_2, ..., v_V)$

  the probabilities of observations given states
- $B_{it} = b_i(o_t)$ a matrix of observation likelihoods, also called emission probabilities, each expressing the probability of an observation o_t being generated from a state q_i
- $\pi = \pi 1, \pi 2, ..., \pi N$ an initial probability distribution over states. π_i is the probability that the Markov chain will start in state i. Some states j may have $\pi_j = 0$, meaning that they cannot be initial states. Also, $\sum_{i=1}^{n} \pi_i = 1$

Markov assumption

$$P(q_i|q_1...q_{i-1}) = P(q_i|q_{i-1})$$

Output independence

the observation at each time depends only on the state at that time

$$P(o_t | q_1 \ldots q_t \ldots q_T, o_1 \ldots o_t \ldots o_T) = P(o_t | q_t)$$

our job: try to infer the most sequence of parts of speech that produce the tokens

A: probability of transitioning from one part of the speech to the next
B: drawing a specific token or word given a part of speech

HMM inference

As with the n-gram model, inferring A and B is pretty trivial.

Simply count occurrences (of transitions, observations) in a corpus.

Also consider unknown and rare tokens/events.

Penn Treebank

https://web.archive.org/web/19970614160127/http://www.cis.upenn.edu/~treebank/

https://catalog.ldc.upenn.edu/LDC99T42

HMM decoding

Given as input an HMM $\lambda = (A, B)$ and a sequence of observations $O = o_1, o_2, ..., o_T$, find the most probable sequence of states $Q = q_1 q_2 q_3 ... q_T$.

Example

Markov model: what's the weather tomorrow given today's weather

A:

	rain	sun
rain	0.6	0.4
sun	0.2	0.8

HMM: infer the weather based on whether your neighbor is carrying an umbrella while she is walking her dog

umbrella?

B:

	yes	no
rain	0.6	0.4
sun	0.2	0.8

if it's raining today, there's a 60% chance the neighbor will be carrying her umbrella
sunny today is only 20% chance of carrying the umbrella

$$\pi = [0.5, 0.5]$$

Example, cont.

Y for umbrella and N for no umbrella

the probability of on three sunny days having observed no umbrella three times

observations: $\mathbf{o} = NNYNYYY$

our job: what sequence of states(what weather over the past week), mostly led to that sequence of observations

option 1: $\mathbf{q_1} = ssrsrrr$

$$
\begin{aligned}
p(\mathbf{q_1}|\mathbf{o}, A, B) &= p(s)p(s|s)p(r|s)p(s|r)p(r|s)p(r|r)p(r|r)p(N|s)^3 p(Y|r)^4 \\
&= p(s)p(s|s)p(r|s)^2 p(s|r)p(r|r)^2 p(N|s)^3 p(Y|r)^4 \\
&= 1.53e-4
\end{aligned}
$$

we observed an umbrella on the four rainy days(the four days that we think might have been ringing)

option 2: $\mathbf{q_2} = ssssrrr$

$$
\begin{aligned}
p(\mathbf{q_2}|\mathbf{o}, A, B) &= p(s)p(s|s)p(s|s)p(s|s)p(r|s)p(r|r)p(r|r)p(N|s)^3 \underline{p(Y|s)}p(Y|r)^3 \\
&= p(s)p(s|s)^3 p(r|s)p(r|r)^2 p(N|s)^3 p(Y|s)p(Y|r)^3 \\
&= 4.08e-4
\end{aligned}
$$

unlike option1 this time an umbrella o sunny day

why is higher? because the probability of the weather staying the same will be higher than the probability of the weather changing

when we map this on the part of speechwriting, the states are invisible the observation is the part that we can see

Viterbi (1)

We could exhaustively compute the probability of each state sequence...
or we could be smarter.

- minimum edit distance: find the cheapest sequence of edits from ("", "") to (word1, word2).
- the Viterbi algorithm: find the most probable sequence of states from time 1 to time T.

We can use dynamic programming!

Viterbi (2)

NN VB KB
cows walk slowly

what's the probability of starting with a noun * the probability of choosing cows

$$v_0(j) = \pi_j b_j(o_0)$$

the maximum probability of arriving in state verb

$$v_t(j) = \max_{i=1}^{N} v_{t-1}(i) a_{ij} b_j(o_t)$$

we end in state verb given that we observed cows walk

there are tons of ways to get to that verb state (come from a noun/verb/adverb)
what's the probability that we are in that state * the probability that we transition from that state to the current state

Viterbi example

log(A):

	rain	sun
rain	-0.5	-0.9
sun	-1.6	-0.2

umbrella?

log(B):

	yes	no
rain	-0.5	-0.9
sun	-1.6	-0.2

$\log \pi = [-0.7, -0.7]$

$$v'_0(s) = \pi'_s + b'_s(o_0) = -0.7 + b'_s(o_0)$$

$$v'_0(r) = \pi'_r + b'_r(o_0) = -0.7 + b'_r(o_0)$$

$$\begin{aligned} v'_t(s) &= b'_s(o_t) + \max[v'_{t-1}(s) + a'_{ss}, v'_{t-1}(r) + a'_{rs}] \\ &= b'_s(o_t) + \max[v'_{t-1}(s) - 0.2, v'_{t-1}(r) - 0.9] \end{aligned}$$

$$\begin{aligned} v'_t(r) &= b'_r(o_t) + \max[v'_{t-1}(s) + a'_{sr}, v'_{t-1}(r) + a'_{rr}] \\ &= b'_r(o_t) + \max[v'_{t-1}(s) - 1.6, v'_{t-1}(r) - 0.5] \end{aligned}$$

```
     N     N     Y     N     Y     Y     Y
s  -0.9  -1.3  -3.1  -3.5  -5.3  -7.1  -8.9
r  -1.6  -3.0  -3.4  -4.8  -5.6  -6.6  -7.6
```

```
     N     N     Y     N     Y     Y     Y
s    x     ←     ←     ←     ←     ←     ←
r    x     ←     ↖     ←     ↖     ←     ←
```