Speech and Text Processing

- Dan Jurafsky
- James Martin

Chapter 2 : Regular Expressions, Text Normalization, Edit Distance

- Text Normalization
  - → tokenizing
  - → lemmatization
  - → Stemming
  - → Sentence segmentation

Exercises

(Q 2.1)

(a) `\b[a-zA-Z]+\b`

(b) `\b[a-z]*b\b`

(c) `\bb+(ab+)+\b`

(Q 2.2)

(a) `(.+)␣\1`

(b) `^[0-9]+.*[A-Za-z]*\.`

(c) `\bgrotto\b.*\braven\b |`
`\braven\b.*\bgrotto\b`

(Q 2.4)

	#	d	e	a	l
#	0	1	2	3	4
l	1	1	2	3	4
e	2	2	1	2	3
d	3	2	2	2	3
a	4	3	3	3	3

(Q 2.5)

	#	b	r	i	e	f
#	0	1	2	3	4	5
d	1	2	3	4	5	6
r	2	3	2	3	4	5
i	3	4	3	2	3	4
v	4	5	4	3	4	5
e	5	6	5	4	5	6

	#	d	i	v	e	r	s
#	0	1	2	3	4	5	6
d	1	0	1	2	3	4	5
r	2	1	2	3	4	5	6
i	3	2	1	2	3	4	5
v	4	3	2	1	2	3	4
e	5	4	3	2	1	2	3

Chapter 3 : N-gram Language Models

Exercises

(Q 3.1) $P(w_n \mid w_{n-2} w_{n-1}) = \dfrac{C(w_{n-2} w_{n-1} w_n)}{C(w_{n-2} w_{n-1})}$

<s><s> I am Sam </s>
<s><s> Sam I am </s>
<s><s> I do not like green eggs and ham </s>

$P(I \mid \langle s\rangle\langle s\rangle) = 2/3$
$P(am \mid \langle s\rangle I) = 1/2$

(Q 3.2) P(i want chinese food) = P(i | <s>) P(want | i) P(chinese | want) P(food | chinese) P(</s> | food)

$= 0.25 \times .33 \times 0\ 0065 \times .52 \times 0.68$

$= 0.0001896$

P(i want chinese food) = P(i | <s>) P(want | i) P(chinese | want) P(food | chinese) P(</s> | food)

$= .19 \times 0.21 \times 0.0029 \times 0.052 \times .4$

$= 0.00000 2406$

(Q 3.3) The unsmoothed probability is higher because the bigrams used in the sentences are very common and has probablities. However, in the smoothed case, their probablities are distributed among not-so-common bigrams which are not used in our test statement.

(Q 3.4)

	<s>	I	am	Sam	do	not	like	green	eggs	and	</s>
<s>	1	4	1	2	1	1	1	1	1	1	1
I	1	1	4	1	2	1	1	1	1	1	1
am	1	1	1	3	1	1	1	1	1	1	2
sam	1	2	1	1	1	1	1	1	1	1	4
do	"	1	"	"	"	2	"	"	"	"	1
not	"	"	"	"	"	1	2	"	"	"	"
like	"	"	"	"	"	"	1	2	"	"	"
green	"	"	"	"	"	"	"	1	2	"	"
eggs	"	"	"	"	"	"	"	"	1	2	"
and	"	"	"	2	"	"	"	"	"	1	"
</s>	"	"	"	1	"	"	"	"	"	"	"

$$P(\text{sam} \mid \text{am}) = \frac{C(\text{am sam})}{C(\text{am})} = \frac{3}{14} = .214$$

(Q 3.5)

	<s>	a	b
<s>	0	2	2
a	0	1	1
b	0	1	1

- $P(a\,a) = P(a \mid \langle s\rangle)\, P(a \mid a) = 0.5 \times 0.5 = 0.25$
  $P(a\,b) = P(a \mid \langle s\rangle)\, P(b \mid a) = 0.5 \times 0.5 = 0.25$
  $P(b\,b) = P(b \mid \langle s\rangle)\, P(b \mid b) = 0.5 \times 0.5 = 0.25$
  $P(b\,a) = P(b \mid \langle s\rangle)\, P(a \mid b) = 0.5 \times 0.5 = 0.25$

$P(s \in \{a,b\}^2) = 1$

(Q3.6) $P(w_3 | w_1 w_2) = \dfrac{C(w_1 w_2 w_3) + 1}{\sum_{w} (C(w_1 w_2 w) + 1)}$

$$= \frac{C(w_1 w_2 w_3) + 1}{C(w_1 w_2) + V}$$

(Q3.7) = $P(sam | am) = \lambda_1 P(Sam) + \lambda_2 (Sam|am)$

$$= \frac{1}{\cancel{2}} \times \frac{42}{25} + \frac{1}{\cancel{2}} \times \frac{\cancel{2}}{3}$$

$$= \frac{2}{25} + \frac{1}{3} = 0.41$$

(Q3.12) $PP(W) = \sqrt[N]{\prod_{i=1}^{N} \frac{1}{P(w_i)}}$

$P(0) = \frac{91}{100}$

$P(1) = P(2) \ldots P(9) = \frac{1}{100}$

$$= \sqrt[10]{\frac{(100)^{10}}{(91)^{9}}}$$

$$= 1.726$$

Chapter 4: Naive Bayes and Sentiment Classification

(Q4.1) S = "I always like foreign films"

$$P(neg|S) = \frac{P(S|neg)\,P(neg)}{P(S)}$$

$$P(pos|S) = \frac{P(S|pos)\,P(pos)}{P(S)}$$

ignore common base.

- $p(neg|S) = (0.16 \times 0.06^2 \times 0.15 \times 0.11) \times 0.5$

  $= 0.000004752$

- $P(pos|S) = (0.09 \times 0.07 \times 0.29 \times 0.04 \times 0.08) \times 0.5$

  $= 0.000002923$

The naive bayes will assign "neg" class to the sentence because $P(neg|S) > P(pos|S)$

(Q4.2) $P(\text{comedy}) = 2/5$ $\quad |V| = 7$

$P(\text{action}) = 3/5$

$$P(\text{fast} \mid \text{comedy}) = \frac{\text{count}(\text{fast}, \text{comedy}) + 1}{\sum_{w \in V} (\text{count}(w, \text{comedy}) + 1)}$$

$$= \frac{2}{9+7} = \frac{2}{16}$$

$$P(\text{fast} \mid \text{action}) = \frac{3}{11+7} = \frac{3}{18}$$

$P(\text{couple} \mid \text{comedy}) = \frac{3}{16}$ $\quad P(\text{shoot} \mid \text{comedy}) = \frac{1}{16}$

$P(\text{couple} \mid \text{action}) = \frac{1}{18}$ $\quad P(\text{shoot} \mid \text{action}) = \frac{5}{18}$

$P(\text{fly} \mid \text{comedy}) = \frac{2}{16}$

$P(\text{fly} \mid \text{action}) = \frac{2}{18}$

$$P(\text{comedy} \mid D) = P(D \mid \text{comedy})\, P(\text{comedy})$$

$$= \frac{2}{16} \times \frac{3}{16} \times \frac{2}{16} \times \frac{1}{16} \times \frac{2}{5}$$

$$= 0.0000732$$

$$P(\text{action} \mid D) = 3/18 \times 1/18 \times 2/18 \times 5/18 \times 3/5$$

$$= 0.000171$$

D will be classified as "action".

(Q 4.3) • Binarized naive Bayes

$P(neg) = 0.6 \qquad P(pos) = 0.4$

$P(good \mid neg) = 3/9 \qquad P(good \mid pos) = 2/7$

$P(poor \mid neg) = 4/9 \qquad P(poor \mid pos) = 2/7$

$P(great \mid neg) = 2/9 \qquad P(great \mid pos) = 3/7$

$$P(neg \mid D) = \frac{3}{9} \times \frac{4}{9} \times \frac{2}{9} \times 0.6 = 0.0197$$

$$P(pos \mid D) = \frac{2}{7} \times \frac{2}{7} \times \frac{3}{7} \times 0.4 = 0.0139$$

Classified as "neg" by BNB.

• Multinomial Naive Bayes

$P(good \mid pos) = 4/12 \qquad P(good \mid neg) = 3/17$

$P(poor \mid pos) = 2/12 \qquad P(poor \mid neg) = 11/17$

$P(great \mid pos) = 6/12 \qquad P(great \mid neg) = 3/17$

$$P(pos \mid D) = \left(\frac{4}{12}\right)^2 \times \frac{2}{12} \times \frac{6}{12} \times 0.6$$

$$= 0.0055$$

$$P(neg \mid D) = \left(\frac{3}{17}\right)^2 \times \left(\frac{11}{17}\right) \times \left(\frac{3}{17}\right) \times 0.4$$

$$= 0.0014$$

Classified as "pos" by MNB.
Both models disagree.

Chapter 10:

CQ 10.1) REF: witnessthe past (18 unigram, 17 bigrams)

HYP2: pastwitness (11 unig., 10 bigrams)

Unigrams that match: pastwitness (11 unigrams)

bigrams that match: pa as st wi it tn ne es ss (9 bigrams)

Unigram P = $11/11$ Unigram R: $11/18$

Bigram P = $9/10$ Bigram R: $9/17$

$$\text{chrP} = (11/11 + 9/10)/2 = .95$$

$$\text{chrR} = (11/18 + 9/17)/2 = 0.57$$

$$\text{Chrf}_{2,2} = 5\,\frac{\text{chrP} * \text{chrR}}{4\,\text{chrP} + \text{chrR}} = \frac{5 \times .95 \times 0.57}{4 \times .95 + 0.57}$$

$$= 0.6199$$

Chapter 12: Constituency Grammars

(Q 12.1)

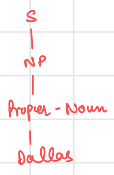
Grammar used:

S → NP | VP | NP VP | NP PP
NP → Pronoun | Proper-Noun | Det Nominal | Nominal | Adj
Nominal → Nominal N | Adj N | N
VP → V | V NP | V NP PP | V PP
PP → Preposition NP

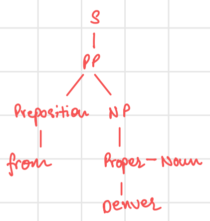
Lexicals used:

Adj → early | all | one-way | any
N → p.m | flights | redeye | fare | delays | five
Det → a
V → arriving
Preposition → from | in | after | on
Proper-Noun → Denver | Dallas | Washington | Thursday

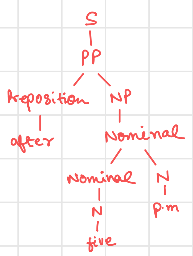
(a) Dallas



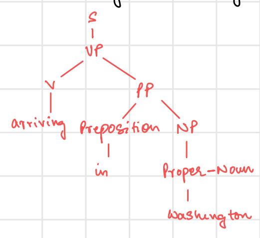
(b) from Denver



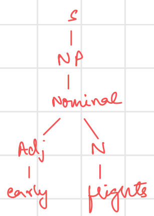
(c) after five p.m



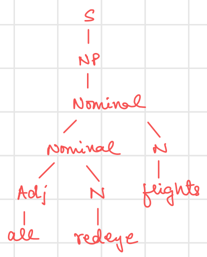
(d) arriving in washington



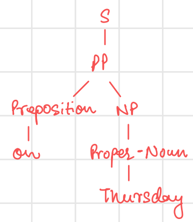
(e) early flights



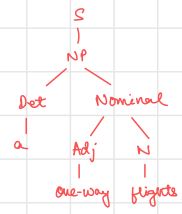
(f) all redeye flights



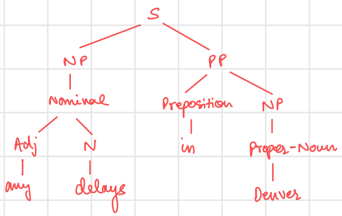
(g) on Thursday



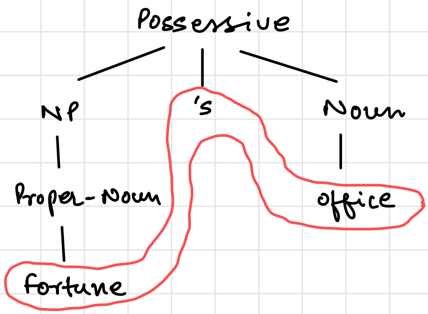
(h) a one-way -flight

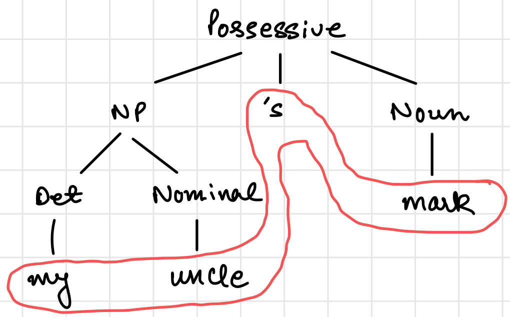


(i) any delays in denver



(Q 12.6) Possessive $\rightarrow$ NP 's Noun





(Q 12.8)

- Step 1: Remove all terminals from right hand side if they occur with a non-terminal.

Example:

$$A \rightarrow bC \implies \begin{array}{l} A \rightarrow BC \\ B \rightarrow b \end{array}$$

- Step 2: on rules with more than 2 non-terminals on RHS, create new rules and substitute.

Example:

$$A \rightarrow BCD \implies \begin{array}{l} A \rightarrow BE \\ E \rightarrow CD \end{array}$$

chapter 13 : Constituency Parsing

(§ 13.3)

function CKY-PARSE (words, grammar) returns table

for j ← from 1 to LENGTH (words) do
    for all {A | A → words[j] ∈ grammar}
        table[j-1, j] ← CHECK-UNITS(table[j-1, j] ∪ A)

    for i ← from j-2 downto 0 do
        for k ← i+1 to j-1 do
            for all {A | A → BC ∈ grammar and B ∈ table[i,k] and C ∈ table[k,j]}
                table[i,j] ← CHECK-UNITS(table[i,j] ∪ A)

function CHECK-UNITS (cell) return updated_cell

updated_cell = cell
for i ← from 1 to LENGTH (cell)
    B ← cell[i]
    for all {A | $A \overset{*}{\Rightarrow} B$ ∈ grammar}
        updated_cell ← cell ∪ A

S → NP VP
S → X1 VP
X1 → Aux NP
S → VP
NP → Pronoun
NP → Proper-Noun
NP → Det Nominal
Nominal → Noun
Nominal → Nominal Noun
Nominal → Nominal PP
VP → Verb
VP → Verb NP
VP → X2 PP
X2 → Verb NP
VP → Verb PP
VP → VP PP
PP → Preposition NP
Det → that | this | the | a
Noun → book | flight | meal | money
Verb → book | include | prefer
Pronoun → I | she | me
Proper-noun → Houston | NWA
Aux → does
Preposition → from | to | on | near | through

Book	the	flight	through	Houston
Noun, Verb ↓ ↓ Nominal VP, S 0,1	0,2	VP, X_2 ↓ S 0,3	0,4	VP VP X_2 ↓ ↓ S S 0,5
	Det 1,2	NP 1,3	1,4	NP 1,5
		Noun ↓ Nominal 2,3	2,4	Nominal 2,5
			Prep 3,4	PP 3,5
				Proper-Noun ↓ NP 4,5

Fig: Running above Algo.

(Q13.4) Partial parsing is mostly used in information extracal systems and its main advantage is the accelerated processing speed. Since you are only parsing chunks of data instead of each word, partial parsing is much faster. Also, as mentioned in section 13.6, eliminating post-head modifiers obviates the need to resolve attachment ambiguities.

But a major disadvantage is probably the fact that you end up loosing a lot of valuable information.

(Q13.5) We can do the following things:

(i) In case of spelling mistakes, we can flag the incorrect words and use string matching algorithm to correct the word.

(ii) or, we can learn the n-gram probabilities, and use them to predict the closest word to replace the incorrect word.

(iii) We can extend (ii) to produce a set of candidate sentences, and choose one with the highest probability.