

Keyword in context

```

-----
|                               | A       | corpus can't describe a natural
|                               | Nature | abhors  | a vacuum.
|                               | All's  | well that ends well.
|                               | A corpus | can't   | describe a natural language en
| You shall know a word by the | company | it keeps.
|                               | A       | corpus  | can't describe a natural langu
|                               | No     | corpus  | is ever too large.
|                               | Corpus | linguists study real language.
| e people live in New York than | Dayton | Ohio.
|                               | A corpus can't | describe | a natural language entirely.
|                               | Other linguists just | dream   | up wild and impossible sentenc
|                               | All's well that | ends    | well.
| 't describe a natural language | entirely. |
|                               | No corpus is | ever    | too large.
|                               | Every      | man has a price ...

```

LaTeX as display engine

Prefix	Word	Suffix
	A	corpus can't describe a natura
Nature	abhors	a vacuum.
	All's	well that ends well.
A corpus	can't	describe a natural language en
You shall know a word by the	company	it keeps.
A	corpus	can't describe a natural langu
No	corpus	is ever too large.
	Corpus	linguists study real language.
e people live in New York than	Dayton	Ohio.
A corpus can't	describe	a natural language entirely.
Other linguists just	dream	up wild and impossible sentenc
All's well that	ends	well.
't describe a natural language	entirely.	
	...	

Rotate

```
#!/bin/env python
# rotate.py
import fileinput
import string

for line in fileinput.input():
    line = string.rstrip(line)
    for i in range(len(line)):
        if line[i] == " ":
            print line[i:] + "\t" + line[:i]

Every man has a price.
price. Every man has a
a price. Every man has
has a price. Every man
man has a price. Every
Nature abhors a vacuum.
```

Sorting

```
cat text | rotate.py | sort -f | unrotate.py
```

Unrotate

```
#!/bin/env python
# unrotate.py
import fileinput
import string

width = 30
for line in fileinput.input():
    fields = string.split(string.rstrip(line),'\t')
    size = len(fields[1]) - width + 1
    print ("% " + 'width' + "s %s" ) \
          % (fields[1][size:],fields[0][:width])
```

```
rotate.awk example | sort -f | unrotate.awk | sed 20q

                A corpus can't describe a natu
A corpus can't describe a natural language entirely.
    Every man has a price.
    Nature abhors a vacuum.
    You shall know a word by the company it keeps
    Nature abhors a vacuum.
                All's well that ends well.
r linguists just dream up wild and impossible sentences.
    You shall know a word by the company it keeps.
    A corpus can't describe a natural langu
You shall know a word by the company it keeps.
                A corpus can't describe a natura
    No corpus is ever too large.
    Corpus linguists study real la
e people live in New York than Dayton Ohio.
    A corpus can't describe a natural language en
    Other linguists just dream up wild and impossible s
    All's well that ends well.
't describe a natural language entirely.
    No corpus is ever too large.
```

Advantages of KWIC format

- Displays real examples.
- Focusses on usages of the keyword.
- Differences easily visible.
- Exposes relationships between words.

Disadvantages of KWIC format

- Can produce too much information (but more selectivity is easy to arrange)
- Limited window to left and right (no good solution other than big screens)
- No quantitative component. (subject of this lecture)

Why to count

- One event on its own doesn't tell us much.
- Five hundred events might be more convincing.
- Some patterns are surprising; others might have arisen by chance.
- Statistics builds (mathematical) models of which patterns are likely to happen by chance.
- The idea is to find patterns which are very unlikely to be accidental. To do this, you compare them with the model.
- The more you count, the more certain you get.

Patterns of word usage

We are interested in finding patterns in word usage. *Collocations* are pairs of words that seem closely related (M & S, p 183ff for detail). For now, won't define them. The first step is to count something.

Could count any of:

- Bigrams.
- Words occurring within five words of each other.
- Translation pairs
- Phrases.
- ...

For the purpose of illustration, we choose bigrams. But the principle is the same for the others.

Frequency-based search

- Count bigrams.
- Raw frequency gives “of the” etc.
- Justeson and Katz filter by part-of-speech (much better).
- Simple statistical measure (frequency) + Linguistic insight (p-o-s matters).

Long distance collocations

M & S 5.2

- Tabulate offsets between (say) “hundreds” and “dollars”.
- Summarize table with mean and variance.
- By looking at the summary stats, can we infer anything useful about the words?

Contingency tables

We count the bigrams in “A Case of Identity”. We can cut up the outcomes into four distinct possibilities.

- We get **Sherlock** then **Holmes**.
- We get **Sherlock** then some other word.
- We get some other word then **Holmes**.
- We get two words, first not **Sherlock**, second not **Holmes**.

This covers all the possibilities (contingencies). You can do the same sort of counting for every pair of words in the corpus.

A contingency table

	Word B holmes	Not Word B	Total
Word A sherlock	$ AB = 7$	$ A\neg B = 0$	$ A = 7$
Not Word A	$ AB = 39$	$ \neg A\neg B = 7059$	$ \neg A = 7098$
Total	$ B = 46$	$ \neg B = 7059$	7105

Equivalent to:

sherlock holmes 7 0 39 7059

which is easier to process with a computer.

Which are the interesting pairs?

Maybe the frequent ones:

of the 41 126 309 6629

in the 24 82 326 6673

that i 23 111 137 6834

it is 22 97 58 6928

to the 22 169 328 6586

it was 21 98 90 6896

at the 21 29 329 6726

said holmes 18 27 28 7032

hosmer angel 17 6 4 7078

i have 15 145 32 6913

Can't tell difference between strong pairs made of rare words and weak pairs of common words.

The *t* test

- M & S section 5.3.1, 5.3.2
- If you don't know t-tests already, skip this.

Word confetti

- Imagine a cup of word confetti made by cutting up a copy of “A Case of Identity”
- Pick words out one at a time. Note them and put them back.
- The *probability* of `sherlock` is $p(\text{sherlock}) = 7/7105 = 0.0009854$. The fraction of time you *expect* to see it if you draw one word.
- $p(\text{holmes}) = 46/7105 = 0.00637$.
- In fact we see it 150 times more often than that, so language is more interesting than word confetti.
- And `Sherlock Holmes` is part of what makes it so.

Bigram probabilities

- Each word token in the document gets to be first in a bigram once, so the number of bigrams is 7105 too. (or 7104? 7106?).
- $p(\text{sherlock}, \text{holmes}) = 7/7105 = 0.0009854$
- $p(\text{sherlock}, \neg\text{holmes}) = 0/7105 = 0.0$
- $p(\neg\text{sherlock}, \text{holmes}) = 39/7105 = 0.0055$
- $p(\neg\text{sherlock}, \neg\text{holmes}) = 7059/7105 = 0.9935$
- We lay these out in a table. Note the marginal totals.

	holmes	¬holmes	Total
sherlock	0.00099	0.0	0.00099
¬sherlock	0.0055	0.9935	0.9990
Total	0.0064	0.9935	1.0

Assume word confetti

- If it were word confetti, we could assume that the probability of second word is unaffected by probability of first word.
- In the table, we can do this by multiplying marginal probabilities.
- This gets the probabilities if you assume word confetti.

	holmes	¬holmes	Total
sherlock	0.00647×0.00099	0.9935×0.00099	0.00099
¬sherlock	0.00647×0.9990	0.9935×0.9990	0.9990
Total	0.00647	0.9935	1.0

Expected frequencies from probabilities

- Multiply everything by 7105

Expected frequencies (given confetti)

	holmes	¬holmes	Total
sherlock	0.05	6.95	7
¬sherlock	45.5	7052.05	7098
Total	46	7059	7105

Deviations from expectation

	holmes	-holmes	Total
sherlock	7 - 0.05	0 - 6.95	-
¬sherlock	39 - 45.94	7059 - 7052.06	-
Total	-	-	-

and there is a statistic called χ^2 which is made from these differences. (M & S 5.3.3)

$$\chi^2 = \sum \frac{(f_o - f_e)^2}{f_e}$$

This will be big when we are *not* dealing with word confetti.

Contributions to χ^2

	holmes	-holmes	Total
sherlock	$\frac{(7-0.05)^2}{0.05}$	$\frac{(0-6.95)^2}{6.95}$	-
¬sherlock	$\frac{(39-45.94)^2}{45.94}$	$\frac{(7059-7052.06)^2}{7052.06}$	-
Total	-	-	-

Summing these

$$\begin{aligned} \chi^2 &= \frac{(7-0.05)^2}{0.05} + \frac{(0-6.95)^2}{6.95} \\ &\quad + \frac{(39-45.94)^2}{45.94} + \frac{(7059-7052.06)^2}{7052.06} \\ &= 966.05 + 6.95 + 1.048 + 0.006 \\ &= 974.05 \end{aligned}$$

which you can look up in the table and find to be unlikely by chance.

So what happens?

```
7105.00 zealand stock 1 0 0 7104
7105.00 wreaths spinning 1 0 0 7104
7105.00 wild clatter 1 0 0 7104
7105.00 whoso snatches 1 0 0 7104
7105.00 westhouse marbank 2 0 0 7103
7105.00 wel comed 1 0 0 7104
7105.00 wash linen 1 0 0 7104
```

...

Oops.

For technical reasons you have to have a situation where $f_e > 5$ or χ^2 might produce nonsense. It over-emphasises the significance of rare events. In general, statistical tests are like this.

Binomial Likelihood ratio

Dunning (CL 19 (1) pp 61–74, 1993) M & S 5.3.4

Based on a different set of assumptions:

- Tests hypothesis that the $|A|$ and the $|\neg A|$ rows come from the same binomial distribution.
- Less sensitive to rare events.
- Might still (like any test) mislead sometimes.

Results of likelihood ratio

192.31 hosmer angel 17 6 4 7078
129.23 said holmes 18 27 28 7032
118.73 mr windibank 14 36 6 7049
101.18 mr hosmer 13 37 10 7045
91.71 it is 22 97 58 6928
86.90 mr holmes 14 36 32 7023
76.86 of the 41 126 309 6629
73.10 there is 13 22 67 7003
71.68 sherlock holmes 7 0 39 7059
70.48 it was 21 98 90 6896

Binomial likelihood ratio is not infallible

- For even smaller samples can use yet other tests.
- You need to know your tests and use them wisely.
- cf. Mutual information M & S 5.4

Words and documents

There *ought* to be a difference between things which are frequent in all documents (e.g. of **the**) and those which are frequent in some only (e.g. **sherlock holmes**).

- The binomial model, and its relative the Poisson distribution don't take account of the "burstiness" of words.
- The negative binomial does, used by Church to find "interesting words" and by Mosteller and Wallace to discriminate authorship