# Today's Goals

- Know when (and when not) to use **Regular expressions**

- Learn about forms of **TF-IDF** and its possibilities

- Understand how information can be quantified with **entropy**

- Understand **PMI** and see why it can help find collocations

Bocconi

# Flexible Matches:
# Regular Expressions

Bocconi

# The promise…

# Is it an (Email) Address?

- notMyFault@webmail.com ✔

- smithie123@gmx ✗

- Free stuff@unibocconi.it ✗

- mark_my_words@hotmail;com ✗

- truthOrDare@webmail.in ✔

- look@me@twitter.com ✗

- how2GetAnts@aol.dfdsfgfdsgfd ✗

| NAME | @ | DOMAIN | . | CODE |

*Bocconi*

# Simple Matching

| sequence | Matches |
|---|---|
| e | any single occurrence of e |
| at | **at**, r**at**, m**at**, s**at**, c**at**, **at**tack, **at**tention, l**at**er |

Bocconi

# Quantifiers

| | Means | Example | Matches |
|---|---|---|---|
| * | 0 or more | `cooo*l` | `cool, coool` |
| + | 1 or more | `hello+` | `hello, helloo, hellooooooo` |
| ? | 0 or 1 | `fr?og` | `fog, frog` |

Bocconi

# Special Characters

| | Means | Example | Matches |
|---|---|---|---|
| . | any single character | `.el` | `eel, Nel, gel` |
| \n | newline character (line break) | `\n+` | One or more line breaks |
| \t | a tab stop | `\t+` | One or more tabs |
| \d | a single digit [0-9] | `B\d` | `B0, B1, ..., B9` |
| \D | a non-digit | `\D.t` | `' t, But, eat` |
| \w | any alphanumberic character | `\w\w\w` | `Top, WOO, ash, bee, ...` |
| \W | non-alphanumberic character | | |
| \s | a whitespace character | | |
| \S | a non-whitespace character | | |
| \ | "Escapes" special characters to match them | `.+`<br>`\.com` | `abc.com, united.com` |
| ^ | the beginning of the input string | `^...` | First word in line |
| $ | the end of the input string | `^\n$` | Empty line |

8

# Classes

| | Means | Example | Matches |
|---|---|---|---|
| **[abc]** | Match any of a, b, c | `[bcrms]at` | `bat, cat, rat, mat, sat` |
| **[^abc]** | Match anything BUT a, b, c | `te[^ ]+s` | `tens, tests, teens, texts, terrors...` |
| **[a-z]** | Match any lowercase character | `[a-z][a-z]t` | `act, ant, not, ... wit` |
| **[A-Z]** | Match any uppercase character | `[A-Z]...` | `Ahab, Brit, In a, ..., York` |
| **[0-9]** | Match any digit | `DIN A[0-9]` | `DIN A0, DIN A1, ..., DIN A9` |

**Bocconi**

# Groups

| | Means | Example | Matches |
|---|---|---|---|
| **(abc)** | Match abc | `.(ar).` | `hard, cart, fare, …` |
| **(ab\|c)** | Match ab OR c | `(ab\|c)ate` | `abate, cate` |

# Matching Addresses

| NAME | @ | DOMAIN | . | CODE |
|------|---|--------|---|------|

`^[A-Za-z0-9_\.-]+@[A-Za-z0-9_\.-]+\.[A-Za-z0-9_][A-Za-z0-9_]+$`
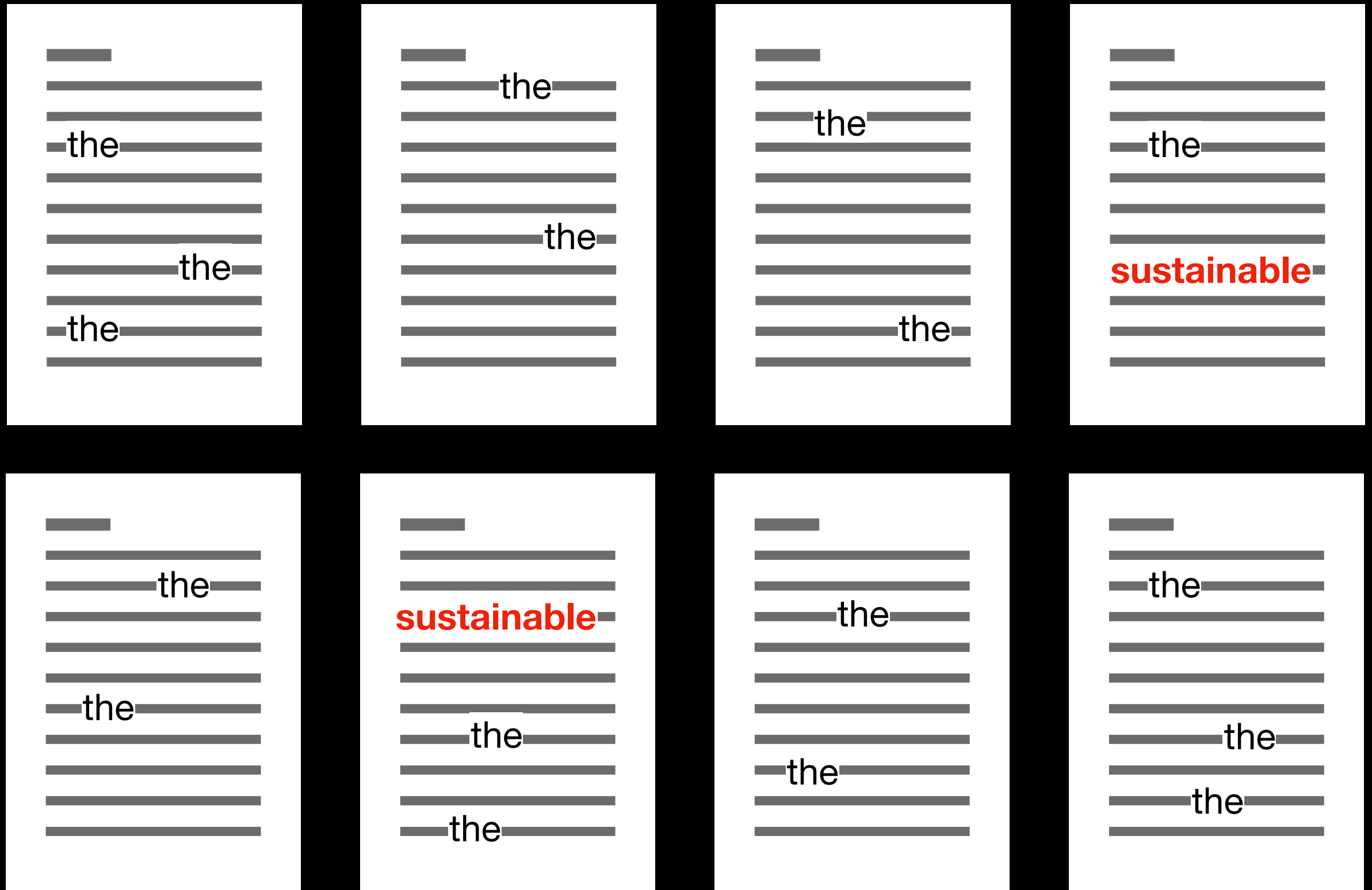
# A (W|w)ord of [Ww]arning

# RegEx as Automata
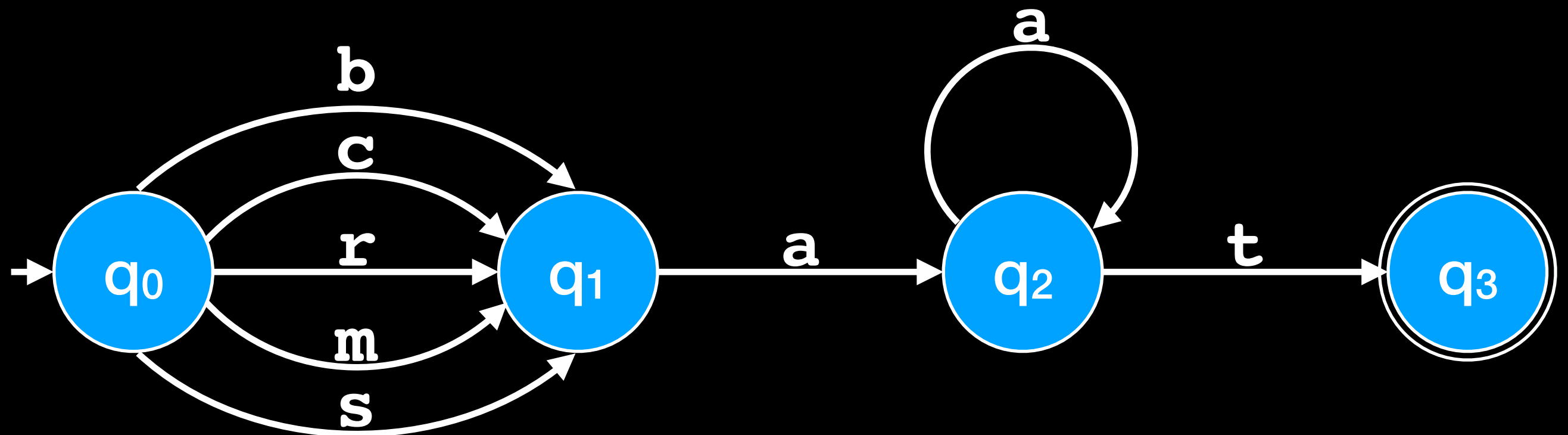


`[bcrms]a+t`

# Finding Important Words: TF-IDF

# Some Words are Just More Interesting…
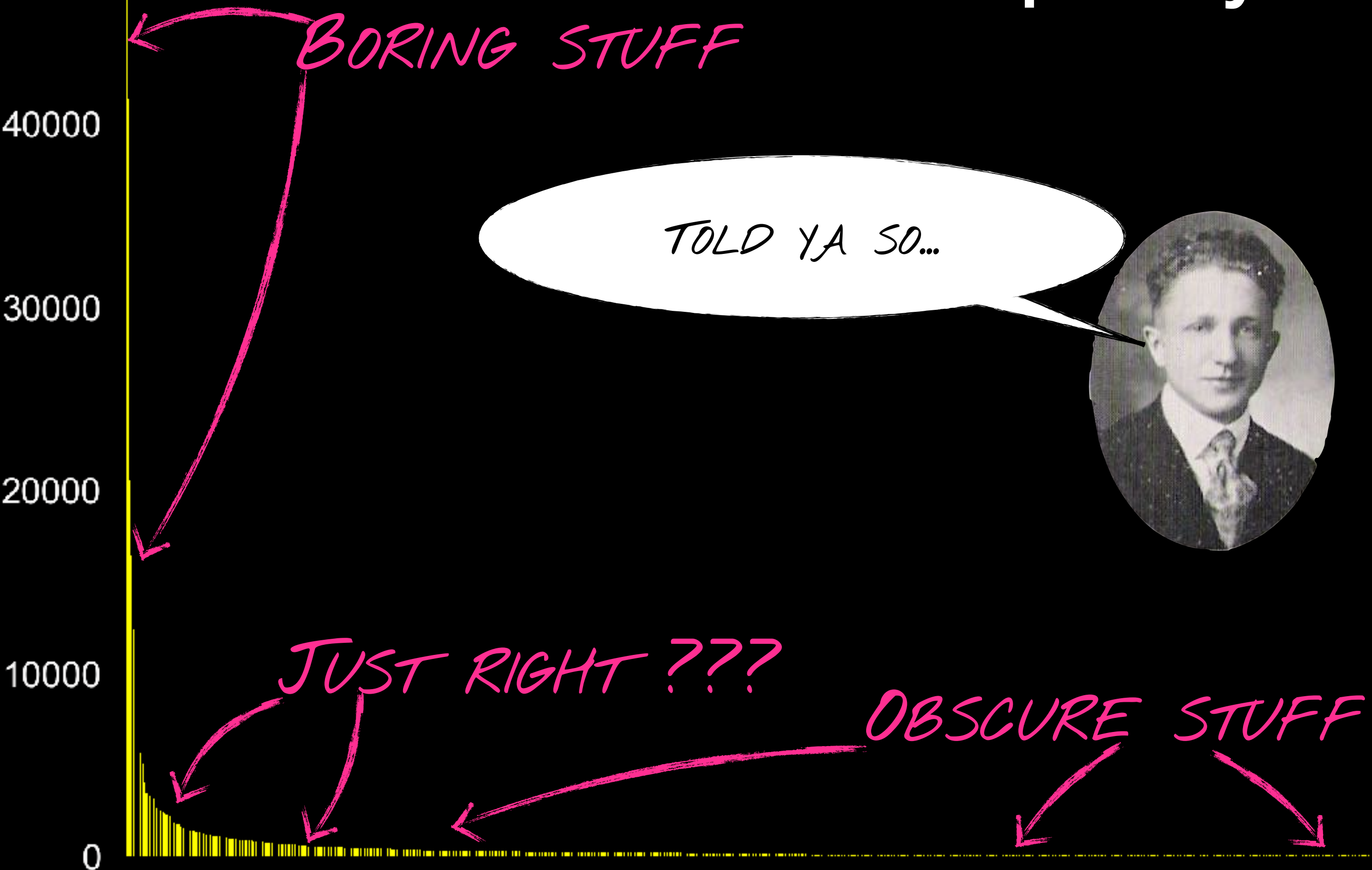
**Bocconi**

# Karen Spärck Jones



1935–2007

- Became a teacher before starting CS career at Cambridge

- Laid the foundation for modern NLP, Google Search, text classification

- Campaigned for more women in CS

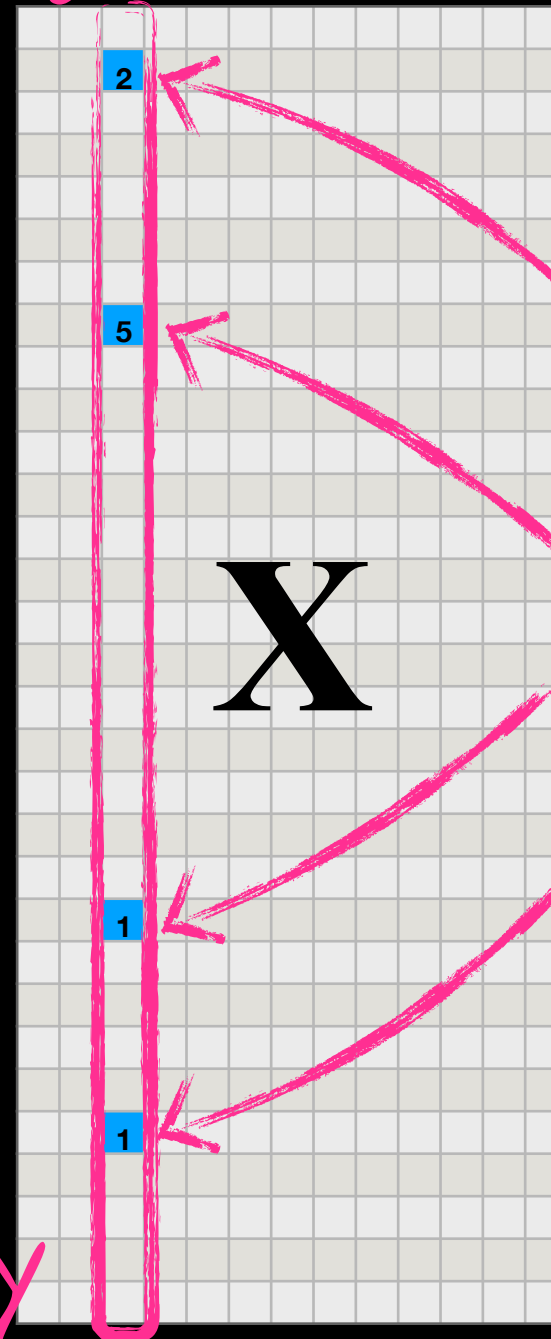- Namesake of prestigious CS prize

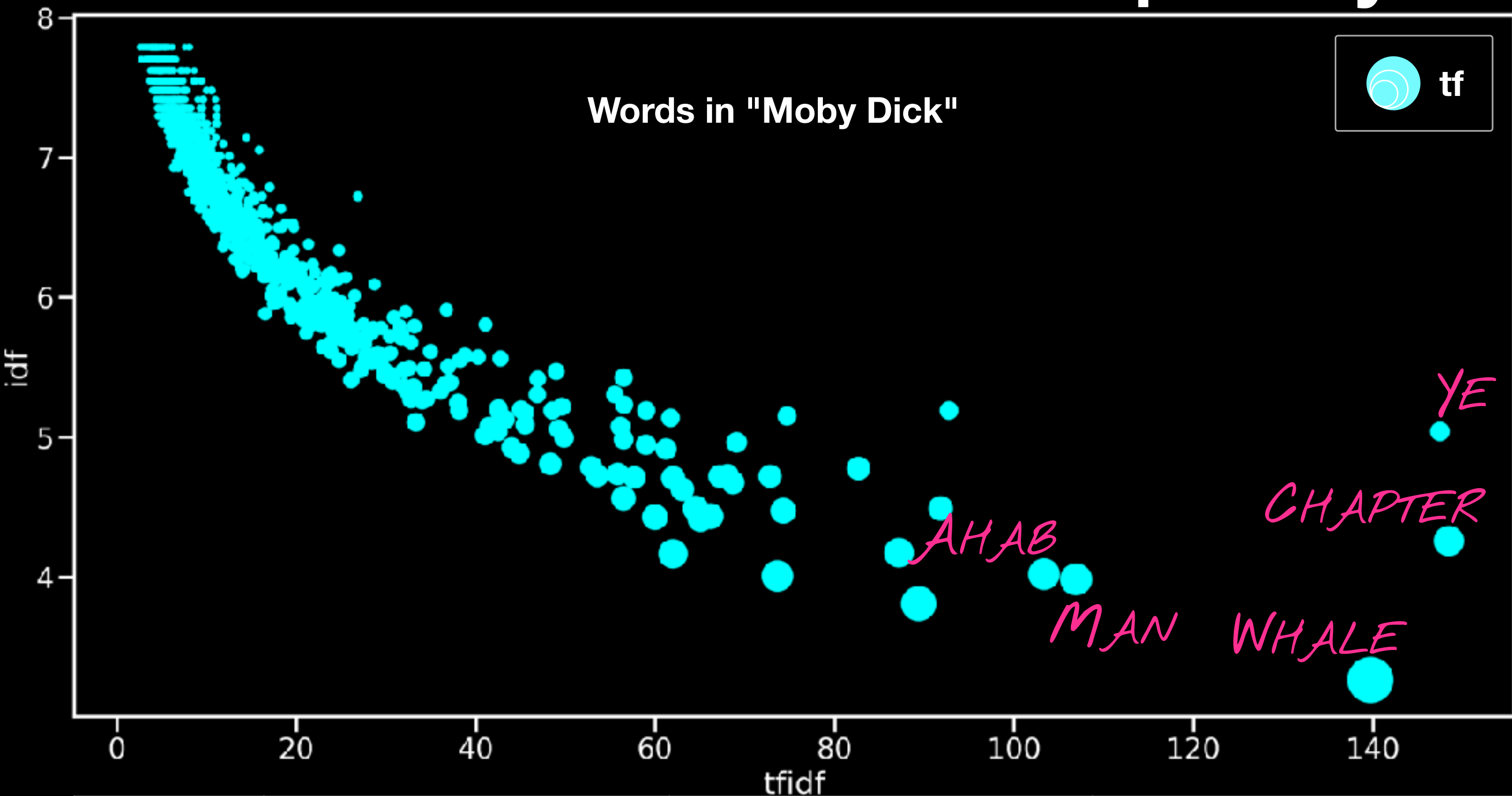# Document and Term Frequency

# Putting it Together

How often we saw the word

$$TFIDF(w) = TF(w) \cdot \log \frac{N}{df(w)}$$

Adjusted By How many Documents

BOCCONI

# Document and Term Frequency



Words in "Moby Dick"

| word | tf | idf | tfidf |
|---|---|---|---|
| ye | 467 | 4.257380 | 148.497079 |
| chapter | 171 | 5.039475 | 147.504638 |
| whale | 1150 | 3.262357 | 139.755743 |
| man | 525 | 3.982412 | 106.932953 |
| ahab | 511 | 4.019453 | 103.357774 |

Bocconi

# Variants

| | TF |
|---|---|
| **binary** | *1 if word in D, else 0* |
| **raw** | *c(word, D)* |
| **relative** | *c(word, D) / len(D)* |
| **smooth** | *log(c(word, D) + 1)* |

| | IDF |
|---|---|
| **regular** | $\log \dfrac{N}{df(word)}$ |
| **smooth** | $\log \dfrac{N}{df(word) + 1} + 1$ |

**Bocconi**

# The Probability of Words: Entropy

# What's More Likely?

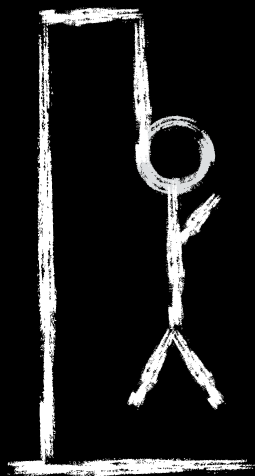We finish each others' ___*Sentences*___

*Leftovers*

*Movies*

Bocconi

# Claude Shannon

1916-2001

- His master's thesis founded a new field: digital circuits

- Invented entropy (to quantify language)
  …and a flame-throwing trumpet!

- Enabled NLP, cryptography, modern computers…

- Died of Alzheimer's, oblivious to his own inventions' impact
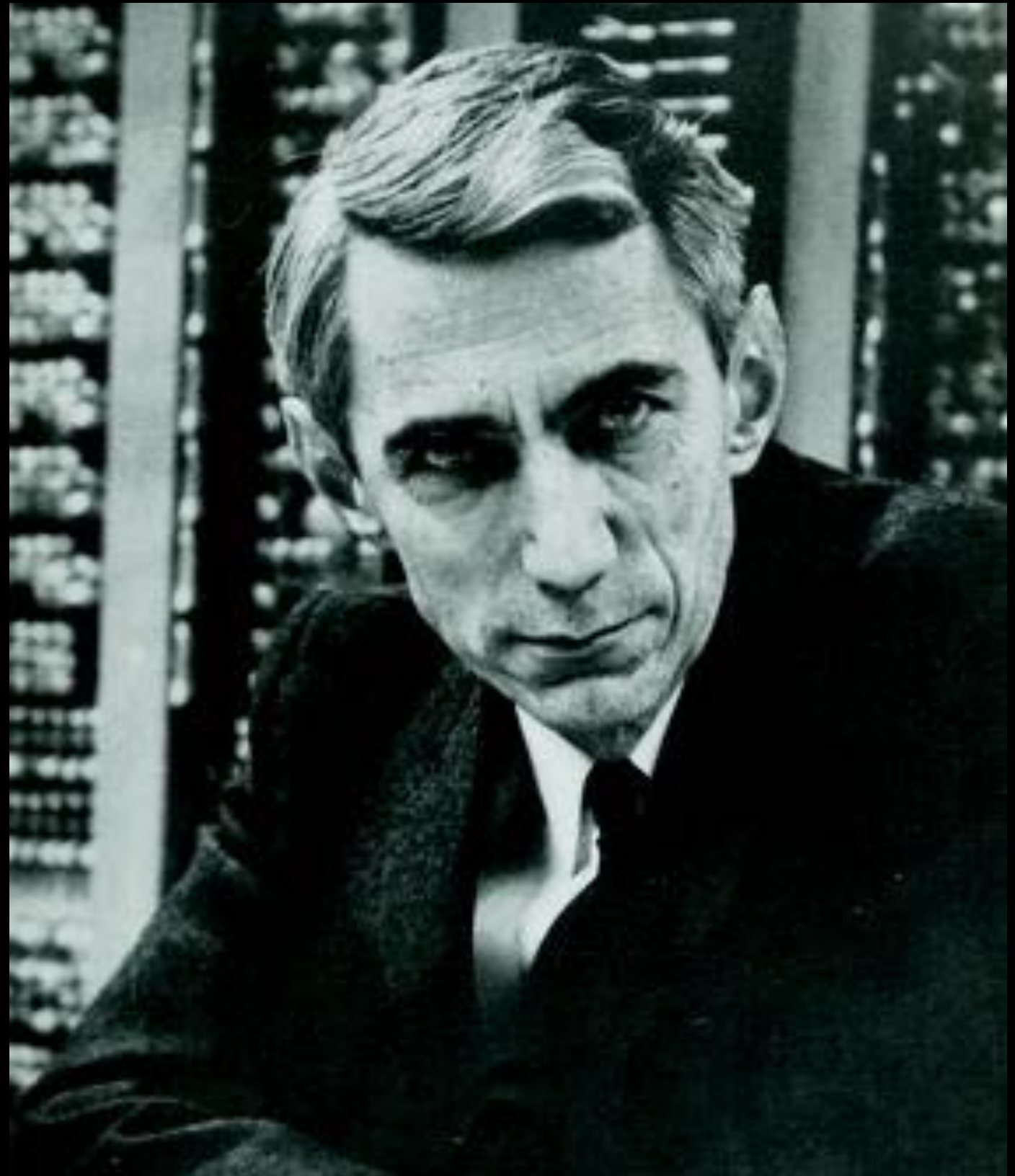
# Entropy

entropy

$$H(X) = -\sum_x p(x) \log p(x)$$

**Information**

0

$p(x)$

1

26

Bocconi

# Probability of a Word

*"It must be recognized that the notion 'probability of a sentence' is an entirely useless one, under any known interpretation of this term."*

**Noam Chomsky**

- Choose a word $w$

- Open a page at random and point at a word:
Is it $w$?

HOW OFTEN WE HAVE SEEN W

$$P(w) = \frac{c(w)}{\sum_{v \in V} c(v)}$$

...ALL WORDS

Bocconi

# Entropy in Use

*WHAT'S THE NEXT LETTER?*



$H(X) \approx 4.38$

*FAIR CHANCE...*

$H(X) \approx 5.73$

*UNPREDICTABLE...*

e t a o n i s h r l d u m c w g f , p y b v k , - ;   ' ! q j x ? z ( ) : 0 1 2 5 7 8 3 * 4 6 9 _ $ [ ] &
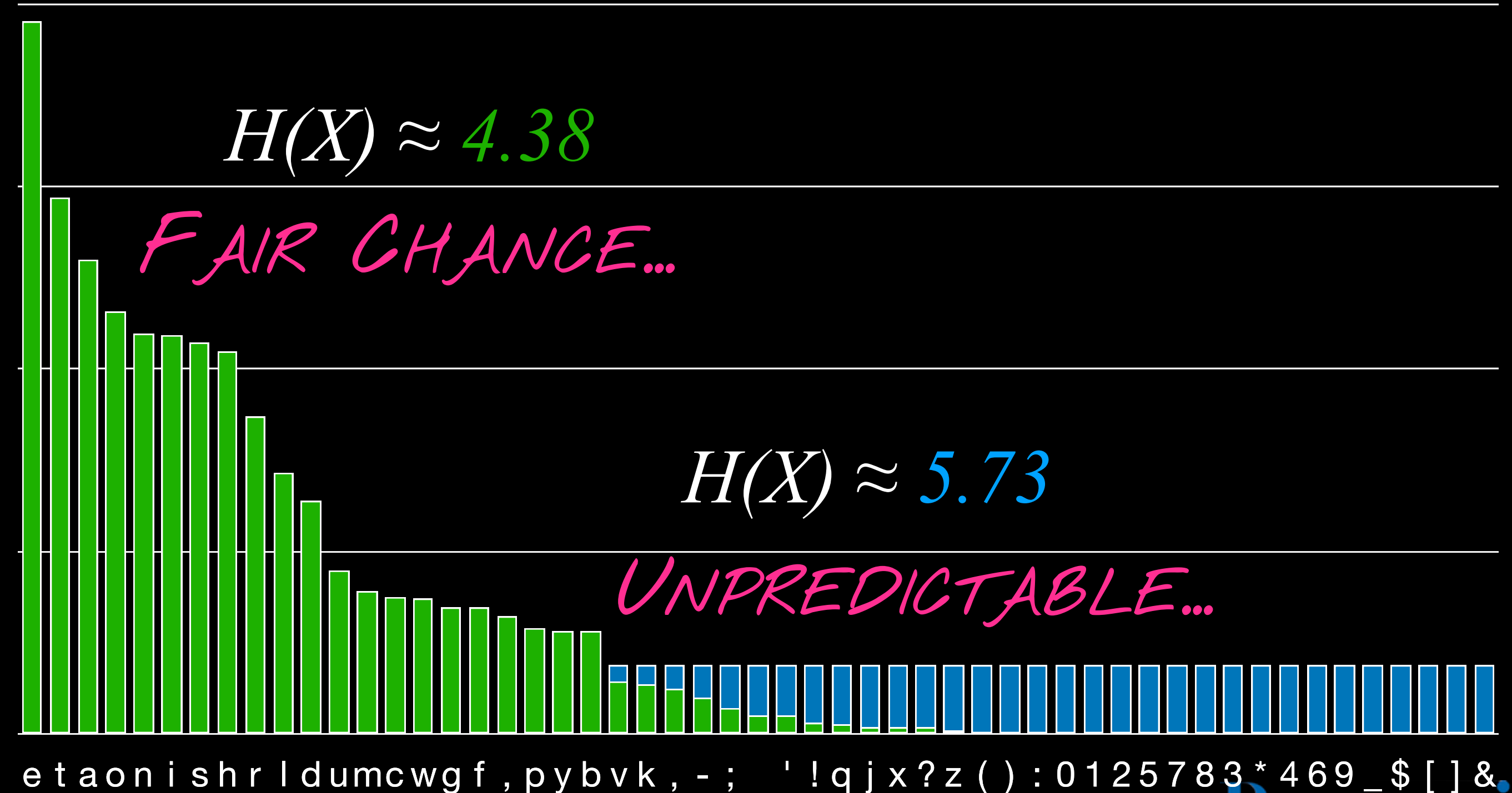
# Telling Neighbors: Pointwise Mutual Information

# Some are not like the Others

frying pan

social media

*FREQUENT BIGRAMS*

and thought

*COLLOCATIONS*

New York

of the

Bocconi

# Mutual Informativity

How well can we guess the Blank?

social _____        and _____

_____ media         _____ the

**Bocconi**

# Pointwise Mutual Information

*CHANCE OF SEEING THEM TOGETHER*

$$PMI(x, y) = log \frac{P(x, y)}{P(x) P(y)}$$

*...SEEING EITHER*

| x | y | c(x) | c(y) | c(xy) | P(x) | P(y) | P(x, y) | PMI(x; y) |
|---|---|------|------|-------|------|------|---------|-----------|
| moby | dick | 83 | 83 | 82 | 0.0003 | 0.0003 | 0.0003 | 3.48 |
| captain | ahab | 327 | 511 | 61 | 0.0013 | 0.0020 | 0.0002 | 1.97 |
| white | whale | 280 | 1150 | 106 | 0.0011 | 0.0045 | 0.0004 | 1.93 |
| under | the | 119 | 14175 | 45 | 0.0005 | 0.0553 | 0.0002 | 0.83 |
| is | a | 1690 | 4636 | 110 | 0.0066 | 0.0181 | 0.0004 | 0.56 |

c(X) = 256,149
c(XY) = 256,148

Bocconi

# Wrapping up

# Take home points

- **Regular expressions** allow us to search for flexible patterns

- **Entropy** allows us to quantify how surprising/predictable something is

- **TF-IDF** finds "bursty" words: medium frequency overall, but concentrated in few documents

- **PMI** tells us how likely one word is to occur with/without another to find **collocations**

Bocconi