Introduction to Text Modeling

Ayush Tewari

November 18th, 2025

Adapted from Carl Edward Rasmussen

Key concepts

- modeling document collections
- probabilistic models of text
- Zipf's law
- bag of words representations

Modelling text documents

Here is an article from the Daily Kos (a US political blog) from Feb 16 2014:

```
GOP abortion foes are criminalizing the doctor-patient relationship

"The doctor-patient relationship." For more than 20 years, conservative propagandists
and their Republican allies have used that four-word bludgeon to beat back universal
health care reform.  In 1994, GOP strategist Bill Kristol warned that "the Clinton
Plan is damaging to the quality of American medicine and to the relationship between
the patient and the doctor." Kristol's successful crusade to derail Bill Clinton's
reform effort was greatly aided by future "death panels" fabulist Betsy McCaughey,
who wrongly warned that Americans would even lose the right to see the doctor of
their choice.  Twelve years later, President George W. Bush proclaimed, "Ours is
a party that understands the best health care system is when the doctor-patient
relationship is central to decision-making."

With the victory of Barack Obama in 2008, GOP spinmeister Frank Luntz told
Republicans obstructing the Affordable Care Act in Congress to once again "call
for the 'protection of the personalized doctor-patient relationship.'" And during the
2012 campaign, the GOP platform declared the party would "ensure the doctor-patient
relationship."
...
```

Why would we model text documents?

How could we model this document?

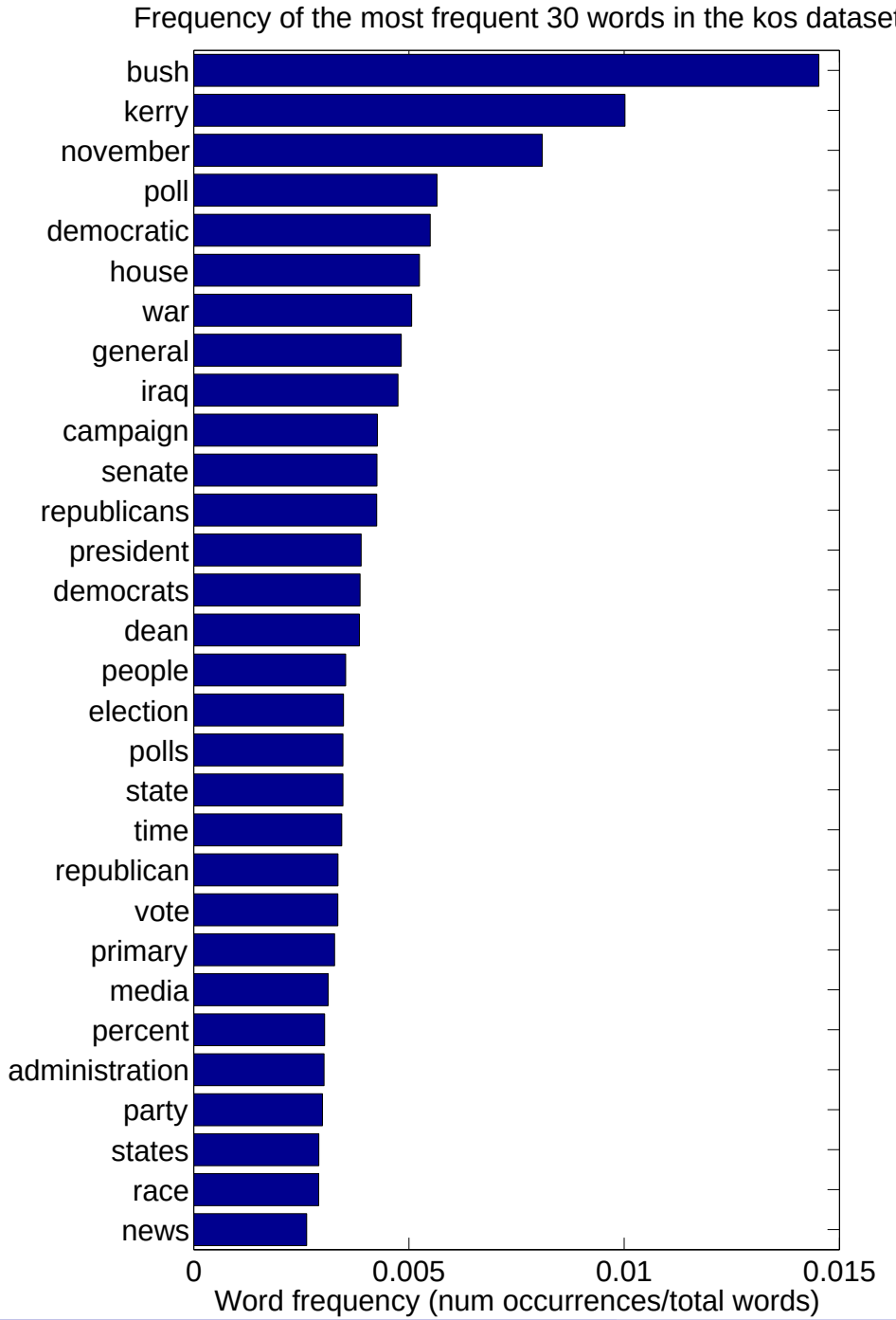
Example: word counts in text

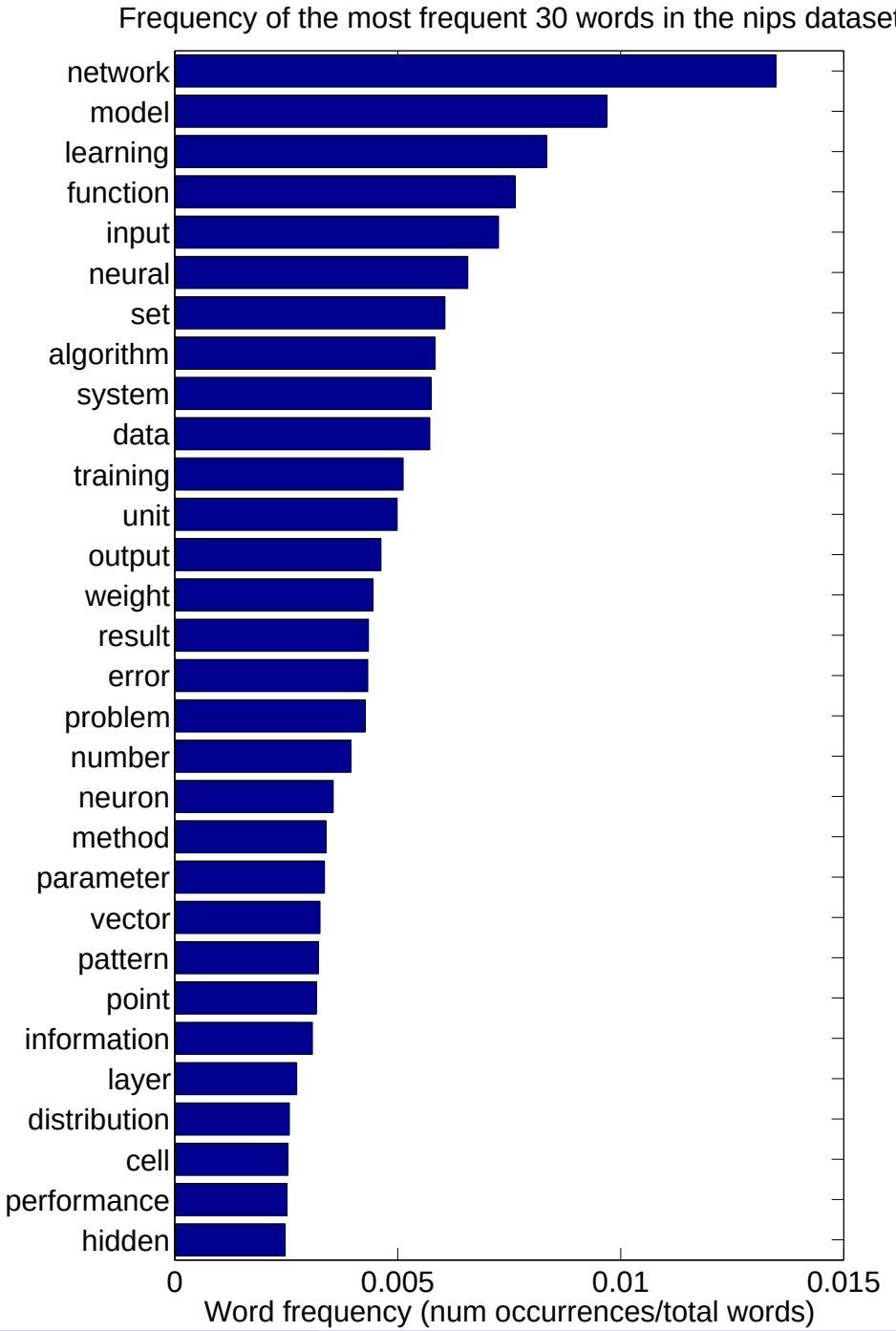
Consider describing a text document by the frequency of occurrence of every distinct word.

The UCI *Bag of Words* dataset from the University of California, Irvine. [1]
For illustration consider two collections of documents from this dataset:

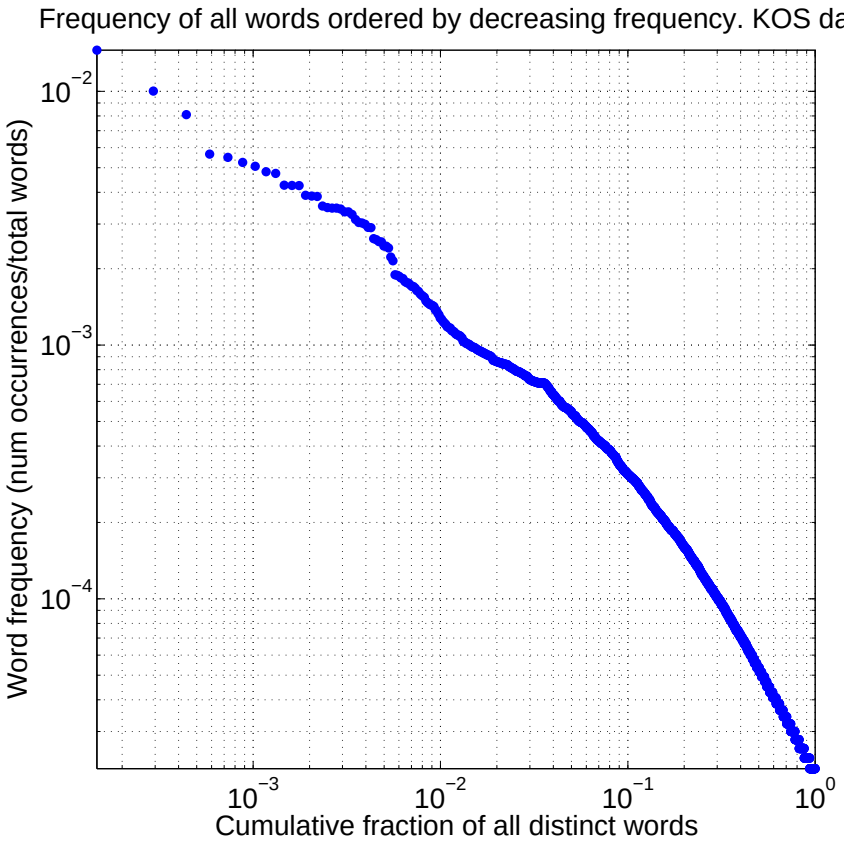
- KOS (political blog — `http://dailykos.com`):
  - $D = 3,430$ documents (blog posts)
  - $n = 353,160$ words
  - $m = 6,906$ *distinct* words
- NIPS (machine learning conference — `http://nips.cc`):
  - $D = 1,500$ documents (conference papers)
  - $n = 746,316$ words
  - $m = 12,375$ *distinct* words

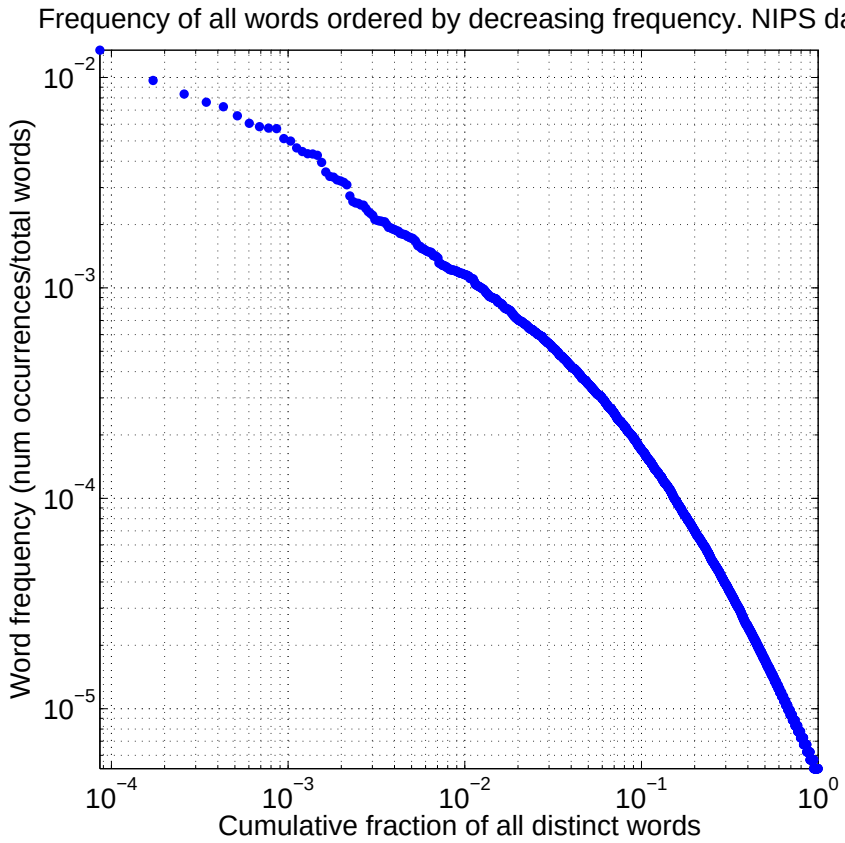
[1] `http://archive.ics.uci.edu/ml/machine-learning-databases/bag-of-words/`


Frequency of the most frequent 30 words in the kos dataset
bush
kerry
november
poll
democratic
house
war
general
iraq
campaign
senate
republicans
president
democrats
dean
people
election
polls
state
time
republican
vote
primary
media
percent
administration
party
states
race
news
0
0.005
0.01
0.015
Word frequency (num occurrences/total words)


Frequency of the most frequent 30 words in the nips dataset
network
model
learning
function
input
neural
set
algorithm
system
data
training
unit
output
weight
result
error
problem
number
neuron
method
parameter
vector
pattern
point
information
layer
distribution
cell
performance
hidden
0
0.005
0.01
0.015
Word frequency (num occurrences/total words)

Different text collections, similar behaviour





Zipf's law states that *the frequency of any word is inversely proportional to its rank in the frequency table*.

Automatic Categorisation of Documents

Can we make use of the *statistical distribution* of words, to build an automatic document categorisation system?

- The learning system would have to be *unsupervised*
- We don't *a priori* know what categories of documents exist
- It must *automatically discover* the structure of the document collection
- What should it even mean, that a document belongs to a category, or has certain properties?

How can we design such a system?