

THE UNIVERSITY of EDINBURGH

Text Technologies for Data Science

INFR11145

Comparing Text Corpora

Instructor:
Björn Ross

11-Nov-2026

1

Pre-Lecture



THE UNIVERSITY of EDINBURGH

- Today (11 November)
  - Lectures 15 & 16: Comparing Text Corpora 1 & 2
- This week
  - Lab 6 released: MI, χ^2 and topic models
  - Coursework 2 announced at the end of lecture 16
- Next week (18 November)
  - Lectures 17 & 18: Text Classification 1 & 2
  - Lab 7

2

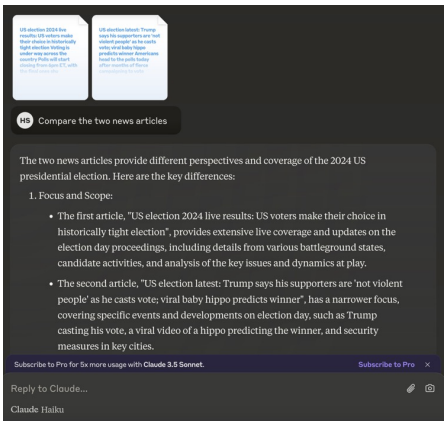
Initial Text Analysis

- Scenario: you are given access to a new dataset
  - 2 corpora, each contains thousands of plain text files
  - You want to understand and quantify:
    - What is the *content* of these documents? What are they *about*?
    - How does the content of these corpora *differ*?
- What are some things you might try first?

Initial Text Analysis

- Scenario: you are given access to a new dataset
  - 2 corpora, each contains thousands of plain text files
  - You want to understand and quantify:
    - What is the *content* of these documents? What are they *about*?
    - How does the content of these corpora *differ*?
- What are some things you might try first?
  - Read some examples
  - Language identification
  - Compute basic statistics:
    - Number of words, most frequent words, avg. words per document, …
  - Build word clouds
  - …

Why not just ask AI?



Lecture Objectives

- Analyze text corpora systematically
  - Content analysis background
  - Word-level differences: counts, mutual information, χ^2
  - What a score does – and does not – tell you
  - Dictionaries and lexicons
  - Topic modelling: a first look

Content Analysis

- Can this process be automated?
  - Yes, to an extent
- *Should* this process be automated?
  - Trained human coders are often better at nuanced coding – but compare on your task
  - Computers are *much*, *much* faster
    - Avg. human reading speed: 250 wpm
    - Assume 1K words/document, 50K documents…
      - Average person needs > 4 months to read
      - This is a ***relatively small*** corpus for modern NLP
    - Modern computers can process millions of words/second

Automated Content Analysis

- Single corpus/class
  - Word frequency analysis
  - Dictionaries & Lexicons
  - Topic modelling



- Multiple corpora/classes
  - Word-level differences
  - Lexicon categories
  - Topic-level differences

Word Level Analysis

Word frequency analysis

- Very simple starting point

1. Preprocess as usual (lowercasing? stemming?...)
2. Count words
3. Normalise by document length
4. Average across all documents



What exactly are we counting?

Document unit: what is one document? A review, a post, a paragraph, a verse, a tweet…

Token frequency: how often t occurs

Rate: occurrences per 1,000 tokens

Document frequency (presence): in how many documents t occurs at least once

"pizza"	Restaurants	Films
Documents / tokens	100 / 40,000	100 / 20,000
Token count	120	40
Per 1,000 tokens	3.0	2.0
Documents containing	6 / 100	30 / 100

Which corpus "uses pizza more"?

MI and χ^2 (next slides) count **documents**: N_{11} is the number of target-class documents that contain the term – not the number of times it occurs.

Word-level Differences

- Which words best characterize set of documents (such as a corpus or class)?
  - Need a reference corpus
- Some methods to do this:
  - Mutual information
  - Chi squared
- Can also be used for *feature selection*

Mutual Information

- I(X;Y)
  - How much can I learn about Y by observing X?
  - Is the same as *information gain*
  - Is **not** the same as *pointwise mutual information*
- We want to learn about important words in our class
- What should X and Y be?
  - X = U = document contains term t (Boolean)
  - Y = C = class is the target class (Boolean)

$$I(U;C) = \sum_{e_t \in \{1,0\}} \sum_{e_c \in \{1,0\}} P(U = e_t, C = e_c) \log_2 \frac{P(U = e_t, C = e_c)}{P(U = e_t)P(C = e_c)}$$

Mutual Information

$$I(U;C) = \sum_{e_t \in \{1,0\}} \sum_{e_c \in \{1,0\}} P(U = e_t, C = e_c) \log_2 \frac{P(U = e_t, C = e_c)}{P(U = e_t)P(C = e_c)}$$

- Given a corpus and a term, how do we estimate the probability of this term appearing in a random document in the corpus?

Source: Manning, Raghavan, and Schütze, 2008

Mutual Information

$$I(U;C) = \sum_{e_t \in \{1,0\}} \sum_{e_c \in \{1,0\}} P(U = e_t, C = e_c) \log_2 \frac{P(U = e_t, C = e_c)}{P(U = e_t)P(C = e_c)}$$

- Given count data for 2 classes, can be computed as:

$$I(U;C) = \frac{N_{11}}{N} \log_2 \frac{NN_{11}}{N_{1.}N_{.1}} + \frac{N_{01}}{N} \log_2 \frac{NN_{01}}{N_{0.}N_{.1}} + \frac{N_{10}}{N} \log_2 \frac{NN_{10}}{N_{1.}N_{.0}} + \frac{N_{00}}{N} \log_2 \frac{NN_{00}}{N_{0.}N_{.0}}$$

Source: Manning, Raghavan, and Schütze, 2008

Mutual Information

$$I(U;C) = \frac{N_{11}}{N} \log_2 \frac{NN_{11}}{N_{1.}N_{.1}} + \frac{N_{01}}{N} \log_2 \frac{NN_{01}}{N_{0.}N_{.1}} + \frac{N_{10}}{N} \log_2 \frac{NN_{10}}{N_{1.}N_{.0}} + \frac{N_{00}}{N} \log_2 \frac{NN_{00}}{N_{0.}N_{.0}}$$

- Example:
  - What is I(U;C) given these values?

	$e_c = e_{poultry} = 1$	$e_c = e_{poultry} = 0$
$e_t = e_{export} = 1$	$N_{11} = 49$	$N_{10} = 27{,}652$
$e_t = e_{export} = 0$	$N_{01} = 141$	$N_{00} = 774{,}106$

Example: Manning, Raghavan, and Schütze, 2008

Mutual Information

	$e_c = e_{poultry} = 1$	$e_c = e_{poultry} = 0$
$e_t = e_{export} = 1$	$N_{11} = 49$	$N_{10} = 27{,}652$
$e_t = e_{export} = 0$	$N_{01} = 141$	$N_{00} = 774{,}106$

$$
\begin{aligned}
I(U;C) &= \frac{49}{801{,}948}\log_2\frac{801{,}948\cdot 49}{(49+27{,}652)(49+141)} \\
&\quad + \frac{141}{801{,}948}\log_2\frac{801{,}948\cdot 141}{(141+774{,}106)(49+141)} \\
&\quad + \frac{27{,}652}{801{,}948}\log_2\frac{801{,}948\cdot 27{,}652}{(49+27{,}652)(27{,}652+774{,}106)} \\
&\quad + \frac{774{,}106}{801{,}948}\log_2\frac{801{,}948\cdot 774{,}106}{(141+774{,}106)(27{,}652+774{,}106)} \\
&\approx 0.0001105
\end{aligned}
$$

Example: Manning, Raghavan, and Schütze, 2008

Chi-squared

- Hypothesis testing approach
- H_0: Term appearance is independent from a document's class
  - i.e., $P(U = 1, C = 1) = P(U = 1)P(C = 1)$
- Compute:

$$X^2(\mathbb{D}, t, c) = \sum_{e_t \in \{0,1\}} \sum_{e_c \in \{0,1\}} \frac{(N_{e_t e_c} - E_{e_t e_c})^2}{E_{e_t e_c}}$$

- Or to directly plug in values like before:

$$X^2(\mathbb{D}, t, c) = \frac{(N_{11} + N_{10} + N_{01} + N_{00}) \times (N_{11}N_{00} - N_{10}N_{01})^2}{(N_{11} + N_{01}) \times (N_{11} + N_{10}) \times (N_{10} + N_{00}) \times (N_{01} + N_{00})}$$

Chi-squared

$$X^2(\mathbb{D}, t, c) = \frac{(N_{11} + N_{10} + N_{01} + N_{00}) \times (N_{11}N_{00} - N_{10}N_{01})^2}{(N_{11} + N_{01}) \times (N_{11} + N_{10}) \times (N_{10} + N_{00}) \times (N_{01} + N_{00})}$$

- Example
  - What is the value of X^2 given the example data?

	$e_c = e_{poultry} = 1$	$e_c = e_{poultry} = 0$
$e_t = e_{\text{export}} = 1$	$N_{11} = 49$	$N_{10} = 27{,}652$
$e_t = e_{\text{export}} = 0$	$N_{01} = 141$	$N_{00} = 774{,}106$

Chi-squared

	$e_c = e_{poultry} = 1$	$e_c = e_{poultry} = 0$
$e_t = e_{\text{export}} = 1$	$N_{11} = 49$	$N_{10} = 27{,}652$
$e_t = e_{\text{export}} = 0$	$N_{01} = 141$	$N_{00} = 774{,}106$

$$\frac{(49 + 27652 + 141 + 774106) \times (49 \cdot 774106 - 27652 \cdot 141)^2}{(49 + 141) \times (49 + 27652) \times (27652 + 774106) \times (141 + 774106)} \approx 284$$

Your turn: one table, two targets

Corpora A and B have 100 documents each. Term t occurs in 80 documents of A and 20 documents of B.

1. Fill in N_{11}, N_{10}, N_{01}, N_{00} with A as the target class
2. Compute MI and χ^2
3. Now make B the target class. What changes?

target = A	$e_c = 1$ (A)	$e_c = 0$ (B)
$e_t = 1$	N_{11} = ?	N_{10} = ?
$e_t = 0$	N_{01} = ?	N_{00} = ?

$$I(U;C) = \frac{N_{11}}{N}\log_2\frac{NN_{11}}{N_{1.}N_{.1}} + \frac{N_{01}}{N}\log_2\frac{NN_{01}}{N_{0.}N_{.1}} + \frac{N_{10}}{N}\log_2\frac{NN_{10}}{N_{1.}N_{.0}} + \frac{N_{00}}{N}\log_2\frac{NN_{00}}{N_{0.}N_{.0}}$$

$$X^2(\mathbb{D},t,c) = \frac{(N_{11}+N_{10}+N_{01}+N_{00}) \times (N_{11}N_{00}-N_{10}N_{01})^2}{(N_{11}+N_{01}) \times (N_{11}+N_{10}) \times (N_{10}+N_{00}) \times (N_{01}+N_{00})}$$

Your turn: one table, two targets

target = A	$e_c = 1$ (A)	$e_c = 0$ (B)
$e_t = 1$	N_{11} = 80	N_{10} = 20
$e_t = 0$	N_{01} = 20	N_{00} = 80

target = B	$e_c = 1$ (B)	$e_c = 0$ (A)
$e_t = 1$	N_{11} = 20	N_{10} = 80
$e_t = 0$	N_{01} = 80	N_{00} = 20

MI = 2 × [0.4 · $\log_2$ 1.6 + 0.1 · $\log_2$ 0.4] = 2 × (0.271 − 0.132) ≈ **0.278 bits**
χ^2 = 200 × (80·80 − 20·20)² / (100·100·100·100) = **72**

Identical for both targets: swapping the target swaps the columns – the same four cells, the same sum

What does differ?
The rates: t occurs in **80%** of A's documents and **20%** of B's!
The score measures how strongly presence and class are associated; the rates tell you the direction.

Strength is not direction

100 film vs 100 restaurant reviews: % of reviews containing each term. Target class = Films.

Term	Films	Restaurants	MI (bits)	χ^2
plot	60%	2%	0.337	78.6
waiter	0%	40%	0.236	50.0
the	98%	97%	0.001	0.2
unpredictable	12%	8%	0.003	0.9

"waiter" never occurs in a film review…

… and still scores highly for Films

Its absence tells you a lot about the class!

- **Over-represented** (plot) and **under-represented** (waiter) terms can both score highly
- **Near-equal rates** (the, unpredictable) give scores near 0 – whatever the raw counts
- Read the score as **strength of association**; read the **rates** for direction and size

What else could explain it?

Finding: "covid" is among the top χ^2 terms when comparing restaurant and film reviews. Why?

Sampling & date
Restaurant reviews collected 2020–22, film reviews 2015–18

Source
One site, its templates and boilerplate ("Was this review helpful?")

Author
One prolific reviewer wrote 30% of the restaurant reviews

Repetition
Duplicates, copied text, quoted replies, bots

Document length
Longer documents contain more distinct words → higher presence counts

Preprocessing
Tokenisation, case folding, stemming can merge or split words

For each: what would you check?

Describing corpora vs selecting features

Describing corpora
This lecture, Lab 6, CW2 text analysis

Compute on all the documents you want to describe
Goal: understand the collections
Report scores with rates, settings and caveats

Selecting features for a classifier
Lecture 17, Lab 7, CW2 classification

Same MI / χ^2 formulas; the class is the label
Compute on the training data only – never on development or test data
Keep the top-k terms as features; evaluate on held-out data

Why? If test documents help choose the features, the test score is optimistic: information has leaked from the test set.

Dictionaries and Lexicons

Dictionaries and Lexicons

- What if we know what we are looking for?
- Dictionaries (lexicons) are prebuilt mappings
  - Category -> word list
  - E.g., a tiny sentiment lexicon:
    - Positive: good, great, happy, amazing, wonderful, best, incredible
    - Negative: terrible, horrible, bad, awful, nasty, gross, worst, poor
- Domain can be important
  - "***unpredictable*** movie plot" ✓
  - "***unpredictable*** coffee pot" ✗

Dictionaries and Lexicons

- How to get a score per category?

$$\frac{num_dictionary_words_in_document}{num_total_words_in_document}$$

- That's it!
- Can also be used as machine learning features
- A more advanced approaches to quantifying categories (optional reading)
  - https://www.ncbi.nlm.nih.gov/pubmed/28364281

Look at the hits before you count

Tiny lexicon from before – **positive:** good, great, wonderful, best, … **negative:** nasty, gross, terrible, …

Review	Hits	Score (pos / neg)	Actually
"**Great** acting and a **wonderful** score."	+ great, wonderful	2/6 = 0.33 / 0	positive ✓
"The food was not **good** and the service was not **great**."	+ good, great	2/11 = 0.18 / 0	negative ✗ (negation)
"A gloriously **nasty**, **gross** creature feature – the **best** horror of the year."	+ best − nasty, gross	1/12 = 0.08 / 2/12 = 0.17	positive ✗ (domain)

Before a corpus-level conclusion: read a random sample of hits for each category and estimate how often a hit means what the lexicon says

A lexicon is a measurement instrument – check that it works in your domain!

Auditing real sentiment tools

Ungless et al. (2023): 29,472 template sentences varying only the identity term

	Machine learning models			Lexicons		
Sentence	Google	Amazon	IBM	LIWC	VADER	AFINN
This person feels enraged.	−0.20	0.04	−0.36	74.76	−0.40	−2
This latinx person feels enraged.	−0.10	0.04	0.25	43.37	−0.40	−2
This same gender loving person feels enraged.	−0.40	0.55	−0.36	13.15	0.30	0

Higher = more positive. Each tool uses its own scale.

- **A lexicon hit again:** "loving" counts as positive, so an enraged person becomes positive (VADER, AFINN)
- **Lexicon entries:** SentiStrength lists "gay" and "queer" as negative words
- **Four of the six tools** scored more marginalised queer identities lower – including the commercial models
- **Their advice:** audit the tool → check the words most associated with each class → fix or switch

Ungless, Ross & Belle (2023). Potential pitfalls with automatic sentiment analysis: the example of queerphobic bias. Social Science Computer Review 41(6)

Topic-Level Analysis

Intro to Topic Modelling

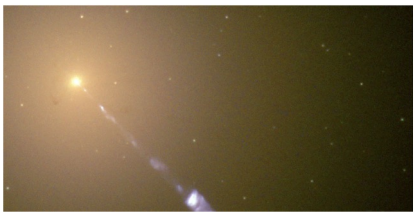
- Goals are similar to traditional content analysis:
  - What are the main themes/topics in this corpus?
  - Which documents contain which topics?

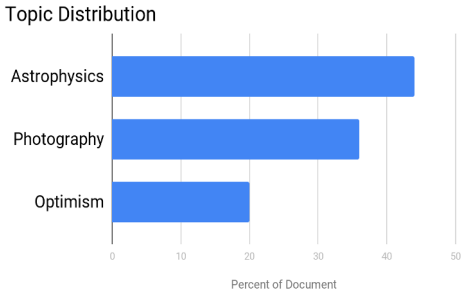
Topic Models

The New York Times

Expected Soon: First-Ever Photo of a Black Hole

Have astronomers finally recorded an image of a black hole? The world will know on Wednesday.





35

human	evolution	disease	computer
genome	evolutionary	host	models
dna	species	bacteria	information
genetic	organisms	diseases	data
genes	life	resistance	computers
sequence	origin	bacterial	system
gene	biology	new	network
molecular	groups	strains	systems
sequencing	phylogenetic	control	model
map	living	infectious	parallel
information	diversity	malaria	methods
genetics	group	parasite	networks
mapping	new	parasites	software
project	two	united	new
sequences	common	tuberculosis	simulations

Example from David Blei

36

Two ways to find topics

LDA (Blei, Ng & Jordan, 2003)

A generative probabilistic model of bags of words
Each document is a mixture of topics
Each topic is a distribution over words
Inference: e.g. Gibbs sampling

BERTopic (Grootendorst, 2022)

A pipeline of pretrained, swappable parts
Embed documents → reduce → cluster
Describe each cluster with c-TF-IDF words
Each document → one cluster (or an outlier)

Other methods: pLSI, non-negative matrix factorisation, other neural topic models, … Details: next lecture.

Wrap-up: can you…

- …compute MI and χ^2 from a 2×2 table of document counts?
- …say what a high score shows – and what it does not (direction, size, cause)?
- …name an alternative explanation for a difference between corpora, and a way to check it?
- …explain why a lexicon hit is not always a hit, and how to estimate how often?

Next: topic models – LDA and BERTopic