

THE UNIVERSITY of EDINBURGH

Text Technologies for Data Science

INFR11145

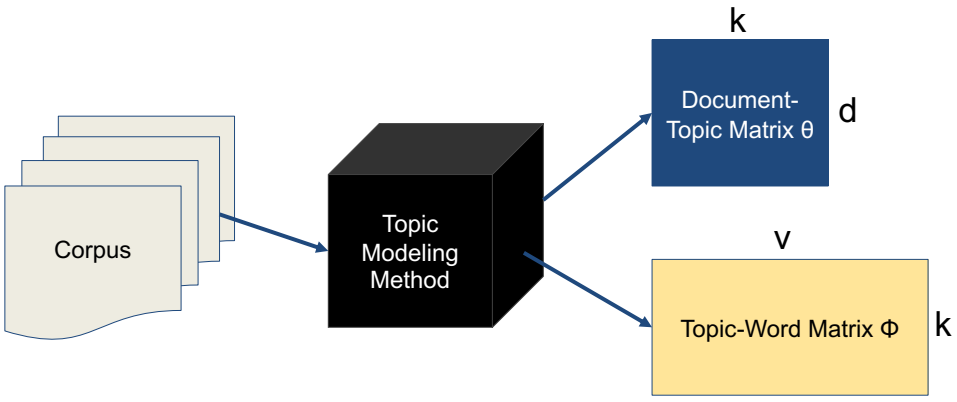
Comparing Text Corpora (2): Topic Models

Instructor:
Björn Ross

11-Nov-2026

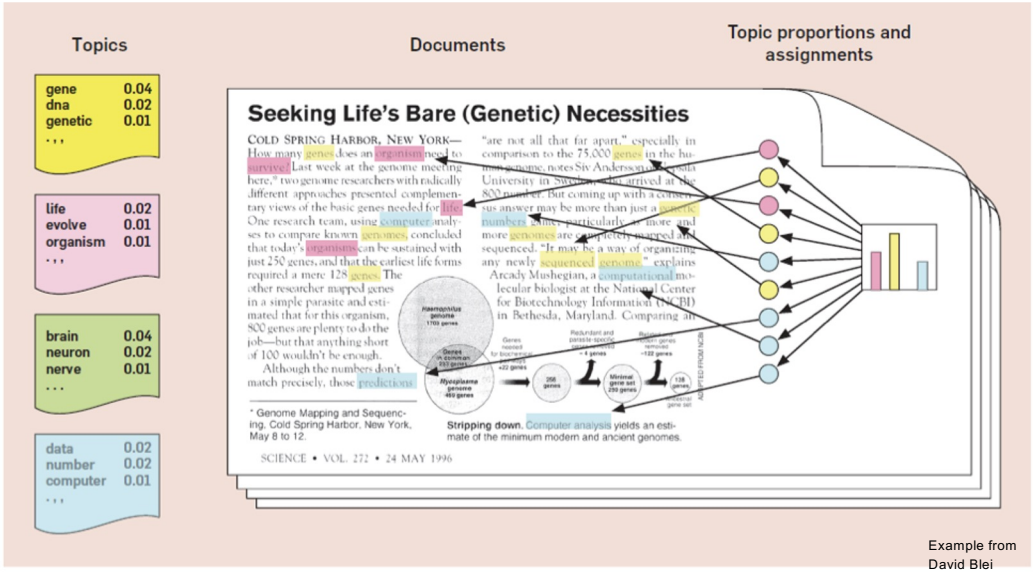
1

Topic Modeling





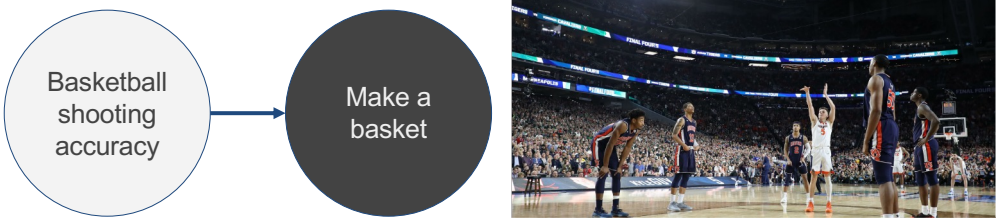
2



Example from David Blei

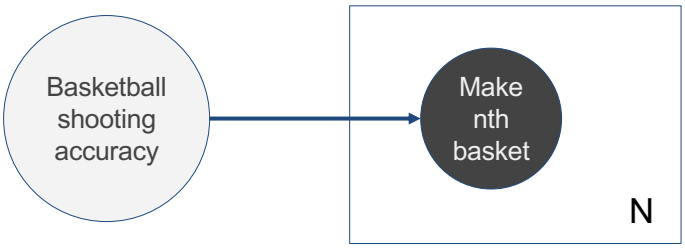
LDA Overview

Background: Plate Notation



5

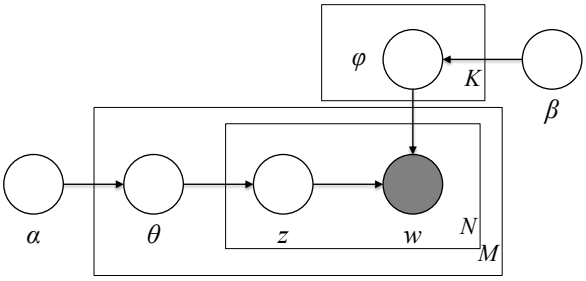
Background: Plate Notation



6

Latent Dirichlet Allocation

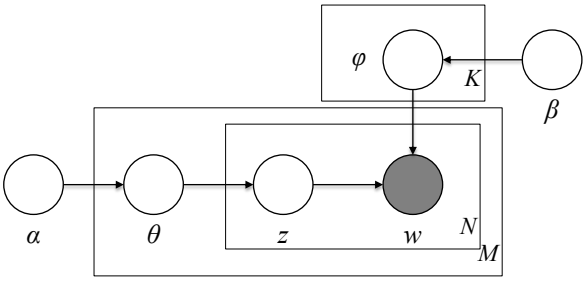




Smoothed LDA (Blei et al. 2003; Griffiths & Steyvers 2004)

θ: a document's distribution over topics (one per document)
α: parameter of the Dirichlet distribution each θ is drawn from
φ: a topic's distribution over words (one per topic; all of them together: Φ)
β: parameter of the Dirichlet distribution each φ is drawn from

Latent Dirichlet Allocation

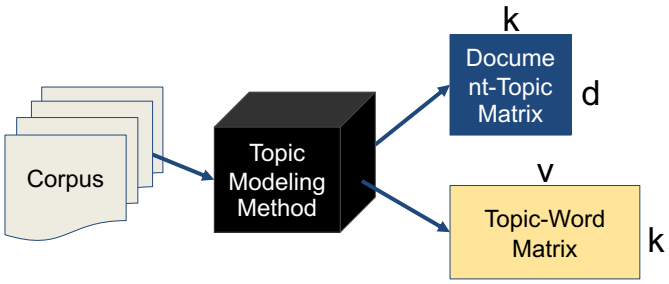


Smoothed LDA (Blei et al. 2003; Griffiths & Steyvers 2004)

$$p(\theta, \mathbf{z}, \mathbf{w} \mid \alpha, \Phi) = p(\theta \mid \alpha) \prod_{n=1}^{N} p(z_n \mid \theta)\, p(w_n \mid z_n, \Phi)$$

Model Inference

- Want to learn the model parameters
- Exact inference becomes intractable



Model Inference

- Instead, use an approximate method such as:
  - Gibbs sampling
  - Variational Inference

Gibbs Sampling for LDA

Goal: Learn Φ, θ given a set of documents D

Φ = topic-word probabilities
θ = document-topic probabilities

Known:
corpus, α, β and the probability that a word is from a topic conditional on the assignments of all other words to topics

$$P(z_i = j \mid \mathbf{z}_{-i}, w_i, d_i, \cdot) \propto \frac{C^{WT}_{w_i,j} + \beta}{\sum_{w=1}^{W} C^{WT}_{wj} + W\beta} \frac{C^{DT}_{d_i,j} + \alpha}{\sum_{t=1}^{T} C^{DT}_{d_i t} + T\alpha}$$

T: number of topics (k)
W: vocabulary size

Note: the ∝ symbol means "proportional to"

Gibbs Sampling for LDA

Want to learn Φ, θ given a set of documents D

1. Assign each word a topic randomly
2. Calculate count matrices
3. Repeat until convergence:
   - For every document d
     - For every word i
       - Decrement count matrices C^{WT} and C^{DT} for current topic assignment
       - Sample a new topic assignment
       - Increment count matrices C^{WT} and C^{DT} for new topic assignment
4. Calculate Φ and θ

Gibbs Sampling for LDA

d1 Green eggs and ham.
d2 Ham and green peppers.
d3 Ham and cheese.

d1 Green eggs and ham.
d2 Ham and green peppers.
d3 Ham and cheese.

Random initialization.



Gibbs Sampling for LDA

C^{WT}	green	eggs	and	ham	peppers	cheese
t1	1	1	1	1	1	1
t2	1	0	2	2	0	0

Green eggs and ham.
Ham and green peppers.
Ham and cheese.

C^{DT}	d1	d2	d3
t1	2	2	2
t2	2	2	1



Gibbs Sampling for LDA

Assume (for the moment) $\alpha = \beta = 0$

Φ	green	eggs	and	ham	peppers	cheese
t1	0.17	0.17	0.17	0.17	0.17	0.17
t2	0.20	0.00	0.40	0.40	0.00	0.00

Green eggs and ham.
Ham and green peppers.
Ham and cheese.

θ	d1	d2	d3
t1	0.50	0.50	0.66
t2	0.50	0.50	0.33

Gibbs Sampling for LDA

C^{WT}	green	eggs	and	ham	peppers	cheese
t1	1	1	1	1	1	1
t2	1	0	2	2	0	0

Green eggs and ham.
Ham and green peppers.
Ham and cheese.

C^{DT}	d1	d2	d3
t1	2	2	2
t2	2	2	1

Gibbs Sampling for LDA

C^{WT}	green	eggs	and	ham	peppers	cheese
t1	1	1	1	1	1	1
t2	1	0	2	2	0	0

Green eggs and ham.
Ham and green peppers.
Ham and cheese.

C^{DT}	d1	d2	d3
t1	2	2	2
t2	2	2	1

Gibbs Sampling for LDA

$$\frac{C^{WT}_{w_i j} + \beta}{\sum_{w=1}^{W} C^{WT}_{wj} + W\beta} \frac{C^{DT}_{d_i j} + \alpha}{\sum_{t=1}^{T} C^{DT}_{d_i t} + T\alpha}$$

Assume (for the moment) $\alpha = \beta = 0$

C^{WT}	green	eggs	and	ham	peppers	cheese
t1	0	1	1	1	1	1
t2	1	0	2	2	0	0

Green eggs and ham.
Ham and green peppers.
Ham and cheese.

C^{DT}	d1	d2	d3
t1	1	2	2
t2	2	2	1

Gibbs Sampling for LDA

C^{WT}	green	eggs	and	ham	peppers	cheese
t1	0	1	1	1	1	1
t2	2	0	2	2	0	0

Green eggs and ham.
Ham and green peppers.
Ham and cheese.

C^{DT}	d1	d2	d3
t1	1	2	2
t2	3	2	1

Gibbs Sampling for LDA

C^{WT}	green	eggs	and	ham	peppers	cheese
t1	0	1	1	1	1	1
t2	2	0	2	2	0	0

Green eggs and ham.
Ham and green peppers.
Ham and cheese.

C^{DT}	d1	d2	d3
t1	1	2	2
t2	3	2	1

Gibbs Sampling for LDA

$$\frac{C^{WT}_{w_i j} + \beta}{\sum_{w=1}^{W} C^{WT}_{wj} + W\beta} \frac{C^{DT}_{d_i j} + \alpha}{\sum_{t=1}^{T} C^{DT}_{d_i t} + T\alpha}$$

Assume (for the moment) $\alpha = \beta = 0$

C^{WT}	green	eggs	and	ham	peppers	cheese
t1	0	0	1	1	1	1
t2	2	0	2	2	0	0

Green eggs and ham.
Ham and green peppers.
Ham and cheese.

C^{DT}	d1	d2	d3
t1	0	2	2
t2	3	2	1

Gibbs Sampling for LDA

$$\frac{C^{WT}_{w_i j} + \beta}{\sum_{w=1}^{W} C^{WT}_{wj} + W\beta} \frac{C^{DT}_{d_i j} + \alpha}{\sum_{t=1}^{T} C^{DT}_{d_i t} + T\alpha}$$

$C^{WT} + \beta$	green	eggs	and	ham	peppers	cheese
t1	0.01	0.01	1.01	1.01	1.01	1.01
t2	2.01	0.01	2.01	2.01	0.01	0.01

Green eggs and ham.
Ham and green peppers.
Ham and cheese.

$C^{DT} + \alpha$	d1	d2	d3
t1	0.01	2.01	2.01
t2	3.01	2.01	1.01

Gibbs Sampling for LDA

- Repeat until convergence
- Probabilistic algorithm – results depend on random initialisation and random samples!

Comparing corpora with one LDA model

1. Fit **one** model on all documents from all corpora; keep a separate record of each document's corpus

2. Get the **full** topic distribution θ_d for every document – all k values (gensim: minimum_probability=0)

3. Average θ_d over the documents of each corpus

4. Compare topic by topic – and read representative documents before naming a topic

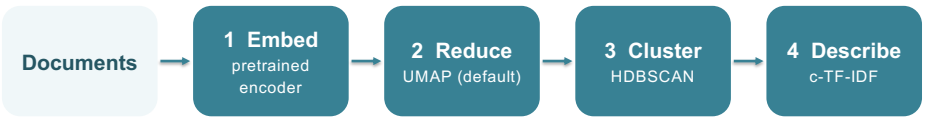
doc	topic 1	topic 2	topic 3
a1	0.70	0.20	0.10
a2	0.60	0.30	0.10
a3	0.10	0.10	0.80
b1	0.10	0.80	0.10
b2	0.20	0.70	0.10
b3	0.10	0.30	0.60
avg A	0.47	0.20	0.33
avg B	0.13	0.60	0.27

- Topic 3 is the top topic of neither corpus – but a third of A and over a quarter of B
- A missing value means 0, not "skip this document"

BERTopic

Why another topic model?

- **LDA only sees word counts:** "plot" and "story" are as unrelated as "plot" and "pizza"
- **Short documents** (reviews, tweets, verses) give little co-occurrence evidence
- **Idea:** start from pretrained document embeddings (lecture 12), then group and describe



Optional 5th step: fine-tune the representation or labels (e.g. keywords, an LLM-written label)

Each box is a separate, swappable model. A pretrained encoder followed by simple, inspectable steps: a common industry pattern (see also RAG, lecture 20).

Running example: 13 short reviews

id	corpus	review
d1	Films	The plot was clever and the ending surprised me.
d2	Films	A gripping story with a twist I never saw coming.
d3	Films	The script was weak and the plot made no sense.
d4	Films	Great acting, but the story dragged in the middle.
d5	Restaurants	The pasta was delicious and the sauce was rich.
d6	Restaurants	Bland soup, cold bread and an overcooked steak.
d7	Restaurants	Fresh fish, tasty sauce and a lovely dessert.
d8	Restaurants	The pizza was soggy and the cheese tasted cheap.
d9	Restaurants	Our waiter was friendly and the service was quick.
d10	Restaurants	We waited an hour and the staff ignored us.
d11	Restaurants	Rude staff and slow service ruined the evening.
d12	Films	The usher was friendly and the staff were helpful.
d13	Films	Parking near the cinema is impossible at weekends.

Step 1 – Embed

Each review → a 300-dimensional vector

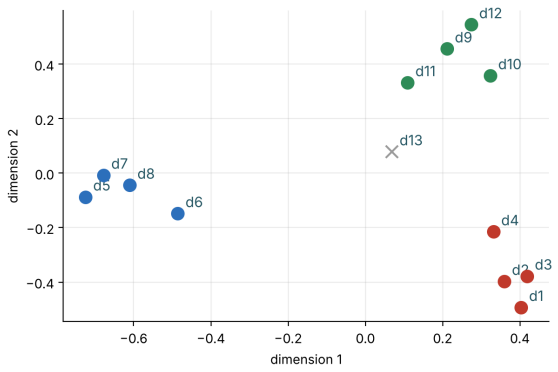
	dim 1	dim 2	dim 3	dim 4	…
d1	−0.136	−0.043	−0.053	−0.042	…
d2	−0.161	−0.011	−0.032	−0.022	…
d5	−0.105	−0.041	0.061	−0.031	…

pair	cosine	shared content words
d1 – d2 (story, story)	0.86	none
d1 – d3 (story, story)	0.83	plot
d5 – d7 (food, food)	0.86	sauce
d1 – d5 (story, food)	0.20	none

- **Similar meaning → nearby vectors, even with no words in common** (d1 and d2)
- The encoder is pretrained and **frozen**: nothing is learned from our 13 reviews

Encoder here: average of pretrained word vectors (spaCy en_core_web_md, content words only). **BERTopic's default:** sentence-transformer all-MiniLM-L6-v2, 384 dimensions.

Steps 2–3 – Reduce and cluster



Reduce: 300 → 3 dims (PCA here; default UMAP → 5). Density is hard to find in hundreds of dimensions

Cluster: HDBSCAN, min_cluster_size = 3. Dense regions; no k in advance

- topic 0: d1–d4
- topic 1: d5–d8
- topic 2: d9–d12
- outlier (−1): d13

One cluster per document (or none) – not a mixture as in LDA

Step 4 – Describe: c-TF-IDF

$$W_{t,c} = tf_{t,c} / (\text{words in c}) \times \ln(1 + A / f_t)$$

- **Join** all documents of a cluster into one "class document"
- $\mathbf{tf_{t,c}}$: count of t in cluster c (stop words removed)
- $\mathbf{f_t}$: count of t over all clusters, outliers included
- **A**: average words per cluster, truncated to an integer
- **ln**: natural log; the +1 keeps weights positive

Cluster	docs	words
−1 (outlier)	d13	5
0	d1–d4	18
1	d5–d8	21
2	d9–d12	18
	A = int(62 / 4)	15

Cluster 2 counts: staff 3, friendly 2, service 2, waiter 1, quick 1, waited 1, hour 1, … (18 words)

Your turn: the weight of "staff"

	staff	friendly
$tf_{t,2}$ (count in cluster 2)	3	2
words in cluster 2	18	18
f_t (count in all clusters)	3	2
A	15	15

Compute $W_{staff,2}$ and $W_{friendly,2}$

ln 6 ≈ 1.792
ln 8.5 ≈ 2.140

$$W_{t,c} = tf_{t,c} / (\text{words in c}) \times \ln(1 + A / f_t)$$

Then: what if two of the food reviews had also mentioned the staff?

Your turn: the weight of "staff"

$W_{staff,2} = 3/18 \times \ln(1 + 15/3) = 0.167 \times 1.792 = \mathbf{0.299}$

$W_{friendly,2} = 2/18 \times \ln(1 + 15/2) = 0.111 \times 2.140 = \mathbf{0.238}$

What if two food reviews also said "staff"?

$f_{staff} = 5$, cluster 1 has 23 words, A = int(64/4) = 16: $W_{staff,2} = 3/18 \times \ln(1 + 16/5) = 0.167 \times 1.435 = \mathbf{0.239}$

- Same count in cluster 2, lower weight: a word spread across clusters is less characteristic of any one
- Matches BERTopic's own output for this example (top words of topic 2: staff 0.299, friendly 0.238, service 0.238, …)

Topic words – then read the documents

Topic	Top c-TF-IDF words	Documents (corpus)	Label?
0	plot, story, saw, script, surprised	d1–d4 (all Films)	story & plot
1	sauce, bread, cheap, bland, pizza	d5–d8 (all Restaurants)	food quality
2	staff, friendly, service, rude, waiter	d9–d11 Restaurants, d12 Films	service & staff
−1	near, cinema, weekends, parking, impossible	d13 (Films)	not a topic

- **Ties:** in small clusters many words share a weight (saw, script, surprised: 0.154) – don't over-read the order
- **A label is a hypothesis –** yours or an LLM's. Read the representative documents: do they support it?
- **The outlier "topic" has the most distinctive words of all** (0.555) – because it is a single document

Comparing corpora: the denominator

Topic	Films: all 6 reviews	Films: excluding outliers (5)	Restaurants: all 7
0 story & plot	4/6 = 67%	4/5 = 80%	0/7 = 0%
1 food quality	0/6 = 0%	0/5 = 0%	4/7 = 57%
2 service & staff	1/6 = 17%	1/5 = 20%	3/7 = 43%
−1 outliers	1/6 = 17%	–	0/7 = 0%

- **Proportion = documents in topic ÷ documents in corpus:** say whether outliers are in the denominator
- **Report the outlier rate per corpus** – it can differ, and it changes every other number
- **LDA instead averages mixtures:** every document contributes a little to every topic

In practice: politicians on Truth Social

Data and pipeline

107,393 posts by 129 US political figures (99 men, 30 women)

Political or not: zero-shot Llama 3.1 8B, checked against two human coders (κ = 0.75; macro-F1 0.90)

BERTopic fitted separately for each subset (men / women × political / not), tuned for topic coherence

What we found

Shared: Israel/Palestine, Russia/Ukraine, inflation, Kamala Harris

More often men: immigration and the border, oil and gas, schools

More often women: the Chinese Communist Party, the courts, campus antisemitism, gun rights

- **Separate models per subset** → topics compared by reading them, not by numbers
- **One author dominates** (Trump) → classifier retrained without him: F1 0.90 → 0.88
- *The authors: themes are "interpretive rather than definitive"*

Bidewell, Deligianni, Elmas, Llewellyn & Ross (2026). Gendered communication patterns of political elites on Truth Social. WebSci '26

Wrap-up

- LDA: each document a mixture of k topics; Gibbs sampling = decrement → sample → increment
- Comparing corpora with LDA: one model, full distributions, averaged per corpus
- BERTopic: embed → reduce → cluster → c-TF-IDF; one cluster or outlier per document
- Either way: read documents, rerun, question labels, report settings and denominators

- *Can you trace an LDA update, explain BERTopic's stages, and justify a corpus comparison using documents – not just word lists?*

Readings

- Manning: IR book section 13.5
- “Probabilistic Topic Models” by David Blei
- “Latent Dirichlet Allocation” by David Blei, Andrew Y. Ng, and Michael I. Jordan
- “Probabilistic Topic Models” by Mark Steyvers and Tom Griffiths (Gibbs sampling)
- “BERTopic: Neural topic modeling with a class-based TF-IDF procedure” by Maarten Grootendorst (2022), arXiv:2203.05794
- BERTopic documentation: maartengr.github.io/BERTopic (algorithm page)
- Optional: UMAP (McInnes, Healy & Melville, 2018); HDBSCAN (Campello, Moulavi & Sander, 2013)

- To watch:
  - Guest lecture (2017) by David Blei at University of Edinburgh School of Informatics