

Topic Modeling & Ideation

Discovering What You Didn't Know You Were Looking For

Week 5: Machine Learning for Smarter Innovation

Transform 1 Million Comments into 10 Innovation Opportunities

Four Stages of Discovery

1. **The Hidden Pattern Problem** - Why we miss what matters most
2. **Understanding Hidden Structure** - How documents mix topics
3. **The Algorithm Arsenal** - Four ways to unmix topics
4. **Innovation Through Discovery** - From patterns to products

Core Question: How do you find themes you didn't know existed in data too large to read?

Topic modeling reveals latent structure - probabilistic decomposition exposes thematic patterns invisible through direct observation

What Are People Really Saying?

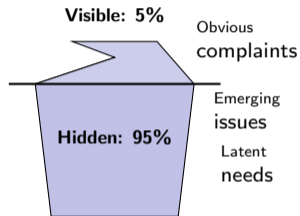
The Scenario:

- Online retailer with 1M reviews
- Need insights by tomorrow
- Competitors analyzing manually
- Missing patterns = lost opportunities

Manual Approach:

- Read 100 reviews/day
- 10,000 days to finish (27 years!)
- Cost: 50 analysts \times \$50K = \$2.5M/year
- Still miss cross-cutting themes

What You're Missing:



Result: You see complaints,
miss opportunities

Volume necessitates automation - pattern discovery scales beyond manual capacity when data growth exceeds analyst availability

When You Can't See the Forest for the Trees

Blockbuster (2000-2010):

- Had millions of rental records
- Categorized by genre (Action, Drama)
- Missed micro-preferences
- Couldn't see "Films with strong female leads from the 80s"
- Result: Bankruptcy in 2010

Netflix (Same Period):

- Applied topic modeling to viewing data
- Discovered 76,897 micro-genres
- "Critically-acclaimed emotional dramas"
- "Witty foreign thrillers"
- Result: \$240B market cap

The Pattern Discovery Gap:

[Chart: Pattern Discovery Comparison]

Netflix found:

- Micro-genres humans never named
- Cross-category preferences
- Time-based viewing patterns
- Mood-driven selections

Algorithmic pattern detection reveals latent structure - computational approaches expose relationships human intuition overlooks

Our Brains Aren't Built for Big Data

Human Limits:

1. Cognitive Capacity

- Can track 7 categories at once
- After 50 items: accuracy drops 40%
- After 500 items: random guessing

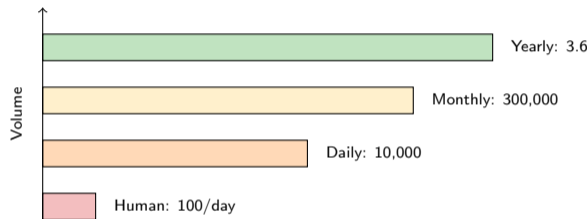
2. Consistency Problem

- Same text, different day = different category
- Two analysts = 60% agreement max
- Fatigue changes decisions

3. Bias Blindness

- See what we expect to see
- Miss emerging trends
- Overlook weak signals

Scale Comparison:

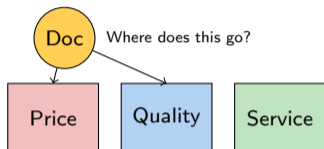


The Gap: Human capacity is linear, data growth is exponential

Real-time analysis demands computational methods - latency requirements eliminate manual processing as viable option

When Topics Don't Fit in Boxes

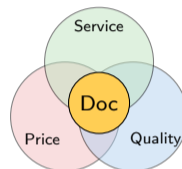
Traditional Categories:



Real Review: "Great value for money, though shipping was slow. Product quality exceeded expectations given the price point."

Problem: Mentions price, quality, AND service - which box?

Topic Modeling Solution:



Document Mixture:

- 40% about price/value
- 35% about quality
- 25% about service

Benefit: Captures full meaning, not forced choice

Every document is a unique mixture of topics - forcing single categories loses information

From Human Limits to Machine Intelligence

What Topic Modeling Does:

1. Discovers Hidden Themes

No predefined categories; themes emerge from data; finds unexpected connections

2. Handles Scale

1M documents in hours; consistent analysis; never gets tired

3. Captures Nuance

Documents as topic mixtures; probabilistic understanding; cross-cutting themes

4. Evolves with Data

Detects emerging trends; tracks topic evolution; adapts to new patterns

The Transformation:



`charts/topic_discovery_landscape.pdf`

A Simple Way to Think About Topics

Think of Cooking:

Ingredients = Words

- Tomato, cheese, basil, pasta...
- Each has different uses
- Can appear in many dishes

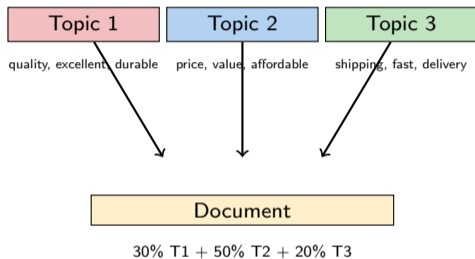
Recipe Types = Topics

- Italian: pasta, tomato, basil, olive oil
- Mexican: beans, corn, chili, lime
- Asian: rice, soy, ginger, sesame

Actual Dish = Document

- Fusion pasta: 60% Italian, 40% Asian
- Uses ingredients from both
- Mixed in specific proportions

The Document Recipe:



Key Insight: Every document mixes multiple topics, just like fusion cuisine mixes cooking styles

Proportional mixture representations preserve information - hard category assignment discards distributional structure present in multithematic content

Which Words Define Each Theme?

What Is a Topic?

A list of words with probabilities (sum to 100%); high prob = core to topic;
low prob = rarely appears

Example: "Customer Service" Topic

Word	Probability
service	15%
support	12%
helpful	10%
response	8%
quick	7%
team	6%
...	...

Visual Distribution:

charts/topic_word_distribution.pdf

Real Documents Are Never Pure

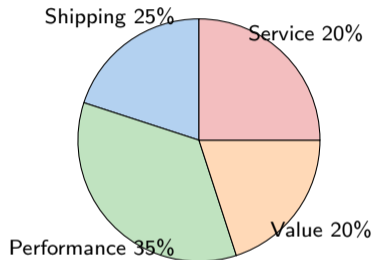
A Real Product Review: "The laptop arrived quickly and was well packaged. Performance is excellent for the price, though battery life could be better. Customer service was helpful when I had questions about setup."

Topic Breakdown:

- **Shipping (25%):** arrived, quickly, packaged
- **Performance (35%):** excellent, battery, performance
- **Value (20%):** price, worth
- **Service (20%):** customer, helpful, questions

The Math: $P(\text{word—doc}) = \sum P(\text{word—topic}) \times P(\text{topic—doc})$

Topic Mixture Visualization:

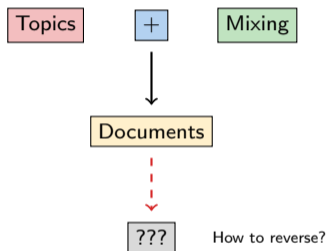


No document is 100% one topic - real communication always blends themes

Proportional topic mixing reflects natural communication patterns - written expression typically combines multiple thematic elements rather than maintaining topical purity

Reverse Engineering the Recipe

The Problem:



Given: Mixed documents

Find: Original topics

Challenge: Many valid solutions!

Like Having a Smoothie:

Taste the final blend, identify ingredients, determine proportions - without the recipe!

How Algorithms Solve It:

1. Pattern Recognition

Words that appear together; consistent co-occurrences; statistical regularities

2. Iterative Refinement

Start with random guess, improve topic definitions, adjust mixtures, repeat until stable

3. Optimization

Maximize coherence; minimize reconstruction error; balance specificity/coverage

The Magic: Algorithms find patterns humans can't see in millions of documents

Scale enables precision - larger corpus sizes reveal subtler thematic distinctions invisible in smaller samples

Organizing Text as Numbers

Step 1: Count Words

	quality	price	service
Review 1	3	1	0
Review 2	0	2	4
Review 3	2	3	1
Review 4	1	0	5

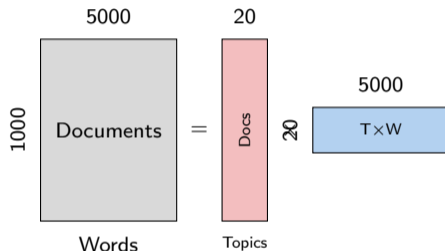
Step 2: Find Patterns

- Reviews 1,3: quality + price
- Reviews 2,4: service-focused
- Hidden structure emerges

Step 3: Decompose

- Original = Topics \times Mixtures
- $1000 \times 5000 = (1000 \times 20) \times (20 \times 5000)$
- Huge matrix \rightarrow Two smaller ones

Visual Decomposition:



Benefit: Compress millions of words into 20 meaningful topics


Dimensionality reduction preserves signal while eliminating noise - low-rank approximations capture dominant patterns efficiently

Measuring Quality Without Ground Truth

Good Topics Are:

1. **Coherent** - Words belong together, make semantic sense
[GOOD] {pizza, pasta, Italian, restaurant}
2. **Distinctive** - Different from others, clear boundaries
[BAD] Topic 1 and 2 both about "food"
3. **Interpretable** - Humans understand, can label easily, actionable

Quality Metrics:



charts/topic_coherence_plot.pdf

What You Now Understand

Core Concepts:

- Documents mix multiple topics
- Topics are word probabilities
- Goal: unmix the smoothie
- Matrix decomposition helps
- Quality matters more than quantity

The Challenge:

- Given: Mixed documents
- Find: Hidden topics
- Make: Useful for innovation

Next: Four Approaches

LDA: Probabilistic

NMF: Parts-based

LSA: Semantic

BERT: Context

Each algorithm unmixes topics differently - like different chefs approaching the same ingredients

Next: Deep dive into each algorithm

Problem formulation enables algorithmic solutions - understanding mixture decomposition requirements guides method selection and evaluation

Different Ways to Find Hidden Themes

charts/algorithm_comparison_matrix.pdf

Our Toolkit:

1. **LDA**
The probabilistic chef
"What's the recipe probability?"
2. **NMF**
The LEGO builder
"What parts combine?"
3. **LSA**
The meaning compressor
"What's the essence?"
4. **BERTopic**
The context reader
"What's the full meaning?"

Trade-offs:

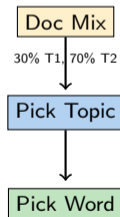
- Speed vs Quality
- Interpretability vs Accuracy
- Simple vs Complex

The Probabilistic Recipe Finder

How LDA Thinks:

- Documents are recipe cards
- Topics are ingredient lists
- Each word is randomly picked:
 1. Pick a topic (from document's mix)
 2. Pick a word (from that topic)
- Work backwards from words to topics

The Process:



Real Example: Input: 1000 restaurant reviews

Output: 5 topics discovered

Topic	Top Words
Food	pizza, pasta, taste
Service	waiter, friendly, quick
Ambiance	cozy, music, romantic
Price	expensive, value, worth
Location	parking, convenient

Performance:

- Speed: **Medium** (5 min/1000 docs)
- Quality: **High**
- Interpretability: **Excellent**

Use LDA when: You need interpretable topics with probability estimates

Probability All the Way Down

The Generative Story:


- 1. For each document:**
Draw topic proportions
e.g., [0.3, 0.5, 0.2] for 3 topics
- 2. For each word position:**
Pick a topic from proportions
Pick a word from that topic

The Math (Simplified):

$$P(\text{word—doc}) = \sum P(\text{word—topic}) \\ \times P(\text{topic—doc})$$

"Word probability = Sum of (word in topic × topic in document)"

Visual Process:



charts/lda_document_topics.pdf

The LEGO Block Builder

How NMF Thinks:

Topics = LEGO sets; documents built from blocks; only adding, never subtracting; each part contributes positively

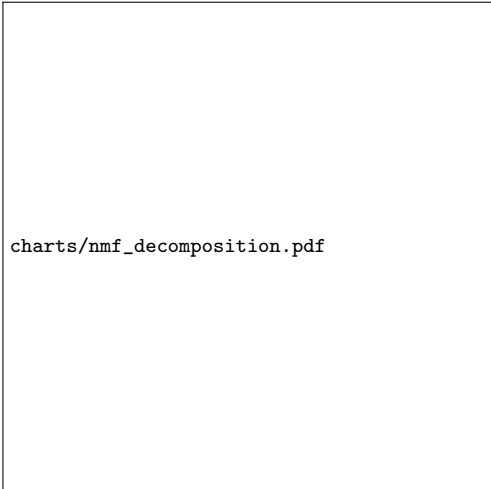
The Decomposition: $V = W \times H$

V: documents (1000×5000); W: doc-topics (1000×20); H: topic-words (20×5000); all ≥ 0

Why "Parts-Based"?

Face = eyes + nose + mouth; Review = quality + price + service

Visual Decomposition:



charts/nmf_decomposition.pdf

The Meaning Compressor

How LSA Thinks:

Words have hidden meanings ("Car" \approx "Vehicle"); compress to essential concepts; like MP3 for text

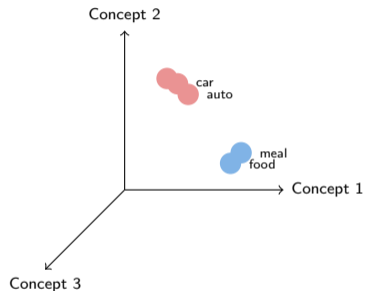
$$\text{SVD: } A = U \times \Sigma \times V^T$$

A: doc-term; U: doc concepts; Σ : importance; V: term concepts

Dimension Reduction:

5000 words \rightarrow 100 concepts; keep important patterns; lose noise

Semantic Space:



What It Finds:

- Synonyms automatically grouped
- Related concepts connected
- Hidden relationships revealed

Performance:

- Speed: **Very Fast** (30 sec/1000)

The Modern Context Master

How BERTopic Thinks:

- Uses BERT's language understanding
- "Bank" (money) "Bank" (river)
- Context determines meaning
- Clusters similar meanings

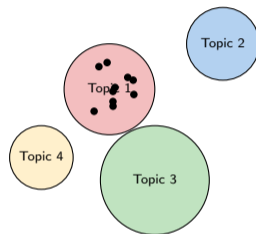
The Process:

1. Embed documents with BERT
2. Reduce dimensions (UMAP)
3. Cluster embeddings (HDBSCAN)
4. Extract topics with TF-IDF

Why It's Better:

- Understands context
- Handles short texts well
- Finds nuanced topics
- Dynamic number of topics

Visual Clustering:



Example Topics (More Nuanced):

Topic	Description
1	Frustrated with slow shipping
2	Delighted by surprise quality
3	Confused about setup process

Performance:

- Speed: **Slow** (10 min/1000)
- Quality: **Excellent**
- Interpretability: **High**

Which Tool for Which Job?

charts/algorithm_speed_quality_tradeoff.pdf

Decision Guide:

Use LDA when:

- Need probability estimates
- Want interpretable topics
- Have medium-length texts

Use NMF when:

- Finding product features
- Need fast results
- Want additive parts

Use LSA when:

- Finding similar documents
- Need very fast processing
- Dimension reduction

Use BERTopic when:

- Quality is critical
- Have short texts (tweets)
- Need nuanced topics

What to Expect in Practice

On 10,000 Reviews:

Algorithm	Time	Topics	Quality
LDA	5 min	20	85%
NMF	2 min	20	78%
LSA	30 sec	20	72%
BERTopic	15 min	23	92%

Quality Metrics:

- Coherence score (0-100)
- Human evaluation
- Actionability of insights

Scalability:

Dataset Size	Best Choice	Time
≤1K docs	BERTopic	5 min
1K-10K	LDA	10 min
10K-100K	NMF	30 min
≥100K	LSA→LDA	1 hour

Industry Usage:

- Netflix: LDA (content)
- Amazon: NMF (reviews)
- Google: LSA + modern variants
- Startups: BERTopic

Reality check: All algorithms find useful patterns - perfect is enemy of good

Benchmark comparisons quantify trade-offs - controlled evaluation reveals relative strengths across speed, quality, and scalability dimensions

How Themes Change Over Time

Dynamic Topic Modeling:

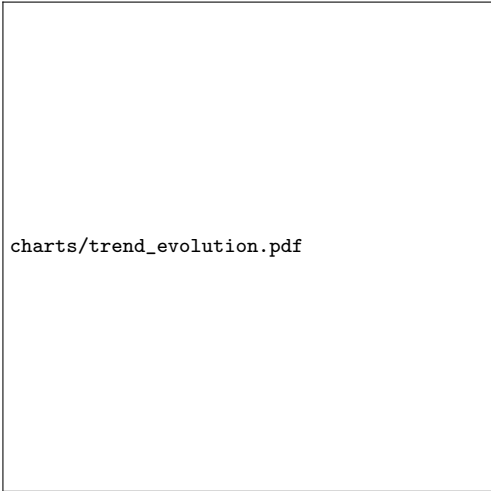
- Topics aren't static
- Language evolves
- New themes emerge
- Old themes fade

Example: Smartphone Reviews

- 2010: "Battery life, small screen"
- 2015: "Camera quality, apps"
- 2020: "5G, privacy, ecosystem"
- 2024: "AI features, sustainability"

How to Track:

- Run topic modeling by time period
- Align topics across periods
- Track word probability changes
- Identify emerging themes early



charts/trend_evolution.pdf

Making Topic Modeling Work

Data Preparation:

Remove stop words, keep domain terms, min 50 words/doc, 1000+ docs

Parameter Tuning:

Start with 20 topics, try 10/30/50, check coherence, get human feedback

Quality Checks:

Do topics make sense? Actionable? Reveal insights? Can you name them?

Common Mistakes:

Too few docs ($i < 100$), too many topics ($i > 100$), not removing boilerplate, ignoring domain knowledge

Success Factors:

Clean data, iterative refinement, human validation, clear use case, action plan

Remember: Topic modeling is exploratory - embrace unexpected discoveries

Domain expertise guides interpretation - algorithmic output requires contextual knowledge to transform statistical patterns into actionable insights

How Topic Modeling Changed Entertainment

The Challenge (2006):

- 100,000 DVDs in catalog
- Basic genres: Action, Comedy, Drama
- Users couldn't find what they wanted
- 60% of catalog never rented

Topic Modeling Applied:

- Analyzed viewing patterns
- User reviews and ratings
- Plot summaries and scripts
- Actor/director combinations

Discovered Patterns:

- "Quirky Independent Movies"
- "Dark Comedies from the 1980s"
- "Emotional Fight-the-System Documentaries"

The Innovation:

Before: 20 genres



After: 76,897 micro-genres

Business Impact:

- 75% of views from recommendations
- 18% increase in engagement
- 80% catalog utilization (vs 40%)
- \$1B saved in content acquisition

Key Insight: People don't want "action movies" - they want "Visually-striking nostalgic action dramas"

Granular categorization reveals preferences - hierarchical topic decomposition exposes latent taste structures beyond conscious user awareness

Music Discovery Through Emotional Topics

Traditional Categories:

- Rock, Pop, Jazz, Classical
- Happy, Sad, Energetic
- Missing nuanced emotions

Topic Modeling on:

- 4 billion playlists
- Listening patterns by time
- Skip rates and repeats
- Playlist names and descriptions

Discovered Moods:

- "Monday motivation"
- "Rainy day contemplation"
- "Late night coding"
- "Sunday morning coffee"
- "Post-breakup empowerment"

The Innovation Map:



47 New Products from Hidden Connections

The Challenge:

- 100,000+ patents in portfolio
- Siloed R&D departments
- Missing cross-applications
- Duplicate research efforts

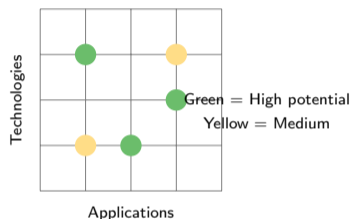
Topic Modeling Applied:

- All patent descriptions
- Research papers
- Lab notebooks
- Customer feedback

Unexpected Discoveries:

- Adhesive + Medical = Surgical tape
- Abrasive + Dental = Tooth whitening
- Reflective + Fashion = Safety clothing

Cross-Pollination Matrix:



Innovation Results:

- 47 new product ideas identified
- 12 launched within 18 months
- \$120M revenue in year 1
- 30% reduction in R&D redundancy

Hidden connections in existing knowledge = breakthrough innovations

Cross-domain topic analysis reveals innovation opportunities - decomposing proprietary knowledge bases exposes non-obvious technology transfer pathways

Turning Complaints into Features

charts/topics_to_opportunities.pdf

The Process:

1. Collect all feedback channels
2. Run topic modeling (LDA)
3. Identify pain point themes
4. Quantify impact
5. Prioritize solutions

Example Topics → Features:

Topic Found	Feature Built
"Confusing setup"	Onboarding wizard
"Battery anxiety"	Power-saving mode
"Lost features"	Search function
"Slow loading"	Cache system

Impact:

- 40% reduction in complaints
- 28% increase in retention
- 50% faster feature validation

60% Faster Insights, 3x More Patterns

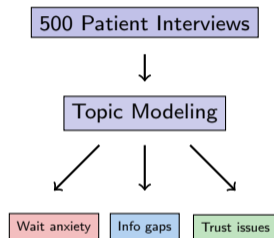
Traditional Research:

- 100s of interviews
- 1000s of sticky notes
- Manual affinity mapping
- 2-3 weeks synthesis
- 5-10 insights found

With Topic Modeling:

- Same interviews transcribed
- LDA + NMF combination
- Automatic theme discovery
- 3 days to insights
- 15-30 patterns found

Healthcare Project Example:



Insights Discovered:

- "Waiting room anxiety" → Redesigned space
- "Information blackout" → Status system
- "Provider trust" → Communication training

Design Impact:


- Patient satisfaction +34%
- Staff efficiency +22%
- Unexpected insights: 12

From Raw Data to Product Launch

The 5-Step Process:

- 1. Data Collection:** Customer feedback, market research, competitor analysis, patent databases
- 2. Topic Discovery:** Run multiple algorithms, validate with experts, name themes
- 3. Opportunity Mapping:** Size each opportunity, assess feasibility, check market fit
- 4. Prioritization:** Impact vs effort matrix, resource requirements, strategic alignment

The Innovation Funnel:



`charts/innovation_funnel_topics.pdf`

When to Use Which Topic Modeling Method: Judgment Criteria

`charts/topic_modeling_decision.pdf`

45 Minutes to Find Hidden Gold

Basic (15 min): Manual Theme Finding

- Read 20 reviews
- Identify 3 themes
- Count theme frequency
- No coding required

Deliverable: Theme list with examples

Success Criteria:

- 3 distinct themes
- 5 examples each
- Clear naming

Intermediate (30 min): Run Topic Modeling

- Use provided code
- Load 1000 reviews
- Run LDA with $k=10$
- Interpret topics

Deliverable: Topic visualization + labels

Tools Provided:

- Jupyter notebook
- Pre-processed data
- LDA template

Advanced (45 min): Innovation Pipeline

- Compare 3 algorithms
- Optimize topic count
- Map to opportunities
- Prioritize top 3

Deliverable: Innovation opportunity report

Bonus Challenge:

- Dynamic topics over time
- Competitor comparison
- ROI estimation

Dataset: 5,000 product reviews from emerging startup

Comparative analysis validates interpretation - multiple analysts extracting independent themes from identical corpora reveals both algorithmic consistency and subjective labeling variance

From Text Chaos to Innovation Strategy

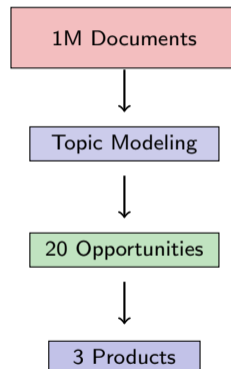
Technical Skills Acquired:

Run topic modeling on any text; Choose LDA/NMF/LSA/BERTopic;
Evaluate quality with coherence; Visualize distributions; Track evolution

Business Applications:

Customer feedback synthesis; Patent mapping; Research organization; Social media trends; Content recommendation

Innovation Capabilities:



ROI Example:

Investment: 1 week — Discovery: 15 hidden needs — Result: 3 features — Revenue: +22% in 6 months

Topic Modeling Mastered

You Can Now:

- Find hidden themes in massive text collections
- Choose the right algorithm for your data
- Transform unstructured feedback into structured insights
- Discover innovation opportunities others miss

Next Week: Generative AI for Rapid Prototyping