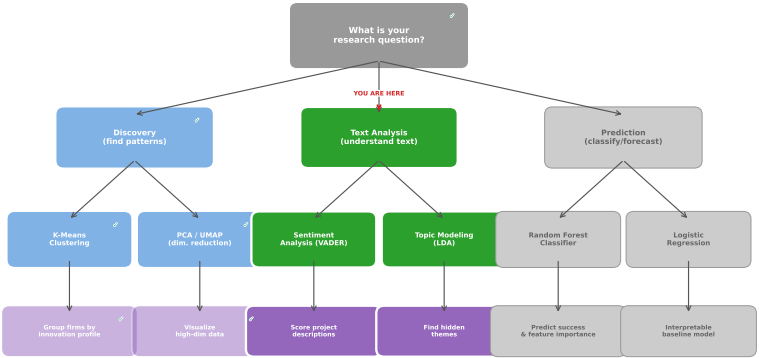
ML Technique Decision Tree



Your research question determines your method. We are exploring the Text Analysis branch.

Words Are Data

When 500 innovators describe their projects,

their word choices contain signals
about strategy, confidence, and focus.

Optimism, uncertainty, ambition, caution —
all hiding in the text.

Language is not just communication — it is measurable data. Today we learn to read it systematically.

Text as Research Data

Innovation text sources:

- Patent abstracts (millions available)
- Annual reports and filings
- Innovation survey descriptions
- Startup pitch decks
- Academic paper abstracts
- News articles about innovation

What NLP can extract:

- **Sentiment:** Positive/negative/neutral tone
- **Topics:** What themes emerge?
- **Similarity:** Which projects resemble each other?
- **Keywords:** What terms define each group?
- **Classification:** Auto-categorize by type

Innovation descriptions are not just words — they are data. Every sentence contains signals about strategy, confidence, and focus.

The NLP Pipeline

Raw Text → Tokens → Numbers → Analysis

Step 1: Tokenization
Split text into words ("tokens").
"AI-driven fraud detection" → [AI, driven, fraud, detection]

Step 2: Representation
Turn words into numbers machines can process.

- **TF-IDF:** Word importance scores
- **Embeddings:** Dense vectors capturing meaning

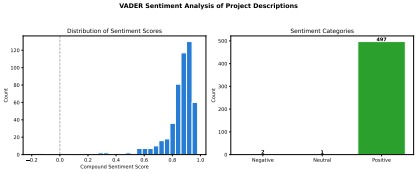
Step 3: Analysis
Apply ML to the numerical representation.

- Sentiment scoring
- Clustering (on text vectors)
- Classification
- Topic modeling

Key insight:
Like cleaning raw survey data — process it before you analyze it.

TF-IDF: Term Frequency–Inverse Document Frequency. Higher score = more distinctive in that document.

Sentiment Analysis



VADER approach:

- Lexicon-based (word lists)
- Handles negation: "not good" $\rightarrow$ negative
- Compound score: -1.0 to $+1.0$
- Fast, no training needed

Research questions:

- Are the successful innovators more confident in how they describe their work?
- Does sentiment vary by industry?

VADER: Valence Aware Dictionary and sEntiment Reasoner. Works well for short texts without training.

Word Embeddings: Meaning as Geometry

The breakthrough:

Words that appear in similar contexts have similar meanings.

Result: Each word (or sentence) becomes a point in high-dimensional space where:

- Similar meanings = nearby points
- "fintech" is close to "banking"
- "AI" is close to "machine learning"

For research:

- Measure similarity between projects
- Find projects that discuss similar innovations
- Visualize the "innovation landscape" in 2D
- No manual keyword lists needed

We use:
Sentence-transformers (local, no API).
Converts each description to a 384-dimensional vector.

Technically: embeddings are learned functions $f : \text{text} \to \mathbb{R}^d$ preserving semantic similarity.

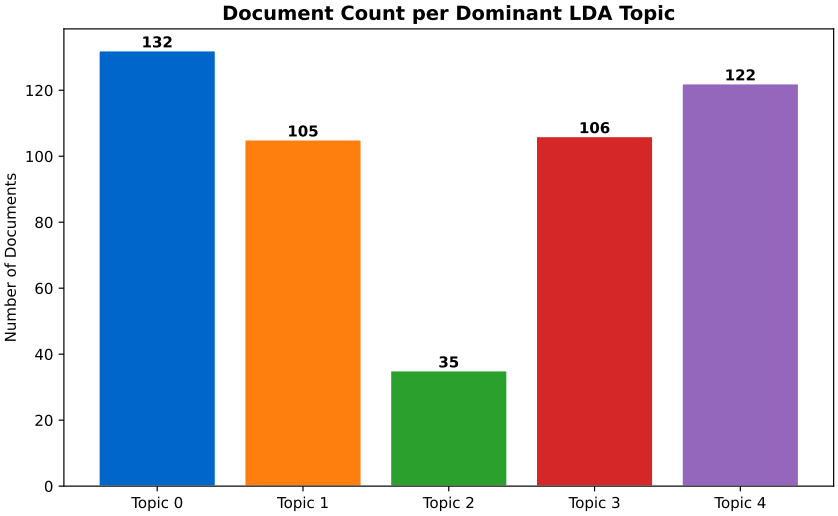
Live Demo: Analyzing Innovation Descriptions

Jupyter Notebook: 03_nlp_sentiment.ipynb

- TF-IDF: most distinctive words per industry
- VADER sentiment analysis
- Sentence embeddings + UMAP visualization
- Semantic similarity between projects

Not which words appear, but which words are distinctive. If everyone says "innovation," it's meaningless.

Topic Modeling: Hidden Themes



LDA reveals topics no one explicitly stated — patterns across 500 descriptions that no single reader would notice.

LDA: Latent Dirichlet Allocation

The intuition:

Every document is a *mixture of topics*. Every topic is a *mixture of words*.

Example:
A project description might be:
40% "digital technology" +
35% "healthcare" +
25% "business model"

LDA discovers:

- What topics exist
- Which words define each topic
- How much of each topic per document

For innovation research:

- Discover innovation themes in patent corpora
- Track topic evolution over time
- Compare topic distributions across industries
- Identify emerging research areas

Research question

"What innovation themes characterize the most successful projects?"

LDA assumes a generative process: $p(\mathbf{word}|\mathbf{doc}) = \sum_k p(\mathbf{word}|\mathbf{topic}_k) \cdot p(\mathbf{topic}_k|\mathbf{doc})$

Live Demo: Topic Modeling

Jupyter Notebook: 04_topic_modeling.ipynb

- Prepare text (tokenize, remove stopwords)
- Fit LDA with 5 topics
- Inspect top words per topic
- Assign topics to innovation projects
- Cross-tabulate topics $\times$ industries

If time allows: compare LDA with different numbers of topics. Coherence score for evaluation.

Your Turn: Where Does This Fit?

"When does analyzing language reveal truth versus impose meaning?"

Take 60 seconds. Discuss with your neighbor.

Map Check: Where Are We?

Branches explored:

- ✓ Discovery: 4 innovator archetypes identified
- ✓ Text: sentiment patterns + 5 hidden topics extracted

Next on the map:

- Synthesis: crossing Discovery × Text
- Do the branches tell a consistent story?

Two branches explored. Now we cross-reference them — do the findings converge?