

Book Recommender using NLP

Carsten Lydeking

Zealand Business College

Oral Exam – AI and ML, 4th Semester

2025-06-11

Book Recommender using NLP

Book Recommender using NLP

Carsten Lydeking
Zealand Business College
Oral Exam – AI and ML, 4th Semester



- **Growing privacy and cost concerns** with cloud-based systems
- Traditional systems require **user profiles and data collection**
- **Goal:** Build a fully offline, content-based book recommender
- **Research Question:**

How can a local ML model recommend books based on natural language descriptions?

- **Key Requirements:** No external APIs, no user tracking, semantic understanding

2025-06-11

Book Recommender using NLP

└ Motivation and Problem

Key Points (45 seconds):

- Start with privacy motivation - very current topic
- Emphasize "fully offline" - no cloud APIs
- Research question is central - pause after reading it
- If asked about motivation: mention GDPR, data breaches, user control

Potential Questions:

- "Why privacy focus?" → GDPR compliance, user trust, data breaches
- "What's wrong with cloud?" → Data collection, profiling, vendor lock-in

Motivation and Problem

- Growing privacy and cost concerns with cloud-based systems
- Traditional systems require user profiles and data collection
- **Goal:** Build a fully offline, content-based book recommender
- **Research Question:**

How can a local ML model recommend books based on natural language descriptions?

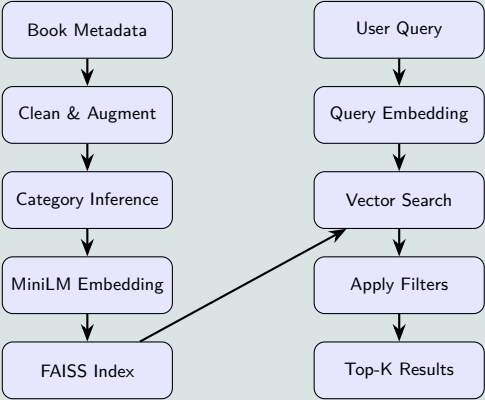
- **Key Requirements:** No external APIs, no user tracking, semantic understanding



Core Components:

- **Data Pipeline:** Clean & augment book metadata
- **Category Inference:** Zero-shot classification + fallback rules
- **Semantic Embedding:** MiniLM sentence transformers
- **Vector Search:** FAISS similarity matching
- **Local UI:** Streamlit interface

Key Innovation: Fully local semantic search without cloud dependencies



2025-06-11

Book Recommender using NLP

Architecture Overview

- Core Components:**
- Data Pipeline: Clean & augment book metadata
 - Category Inference: Zero-shot classification + fallback rules
 - Semantic Embedding: MiniLM sentence transformers
 - Vector Search: FAISS similarity matching
 - Local UI: Streamlit interface
- Key Innovation:** Fully local semantic search without cloud dependencies



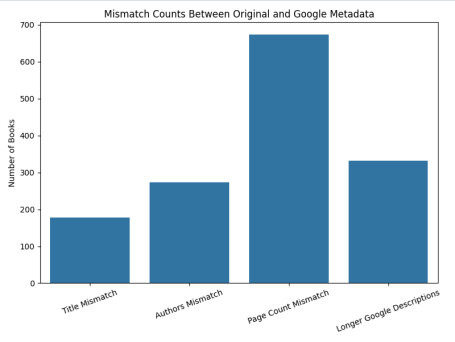


Data Quality Challenges:

- ▶ Started with **6,800 books** from multiple sources
- ▶ **Issues found:**
 - ▷ Missing author/category information
 - ▷ Very short descriptions (< 9 words)
 - ▷ Inconsistent categorization across sources

Data Engineering Solutions:

- ▶ **API enrichment:** OpenLibrary & Google Books
- ▶ **Quality filtering:** Remove inadequate descriptions
- ▶ **Final dataset:** 5,160 high-quality books

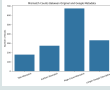


Data inconsistencies across sources required systematic cleaning and validation

2025-06-11

Dataset Exploration & Processing

- Data Quality Challenges:**
- ▶ Started with 6,800 books from multiple sources
 - ▶ **Issues found:**
 - ▷ Missing author/category information
 - ▷ Very short descriptions (< 9 words)
 - ▷ Inconsistent categorization across sources
- Data Engineering Solutions:**
- ▶ API enrichment: OpenLibrary & Google Books
 - ▶ Quality filtering: Remove inadequate descriptions
 - ▶ Final dataset: 5,160 high-quality books



Data inconsistencies across sources required systematic cleaning and validation

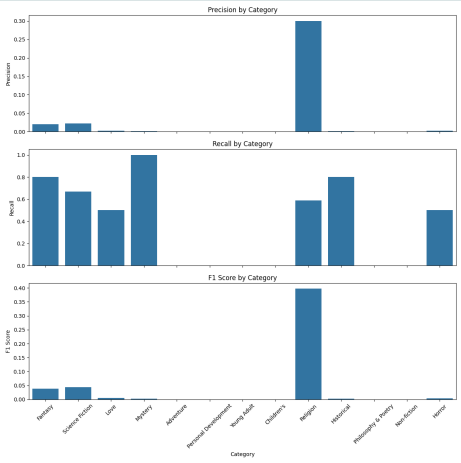


Two-Tier Classification Approach:

- **Primary:** Zero-shot classification with BART-MNLI
 - ▷ No training data required
 - ▷ 13 predefined book categories
 - ▷ Confidence scoring for predictions
- **Fallback:** Rule-based keyword matching
 - ▷ When confidence < threshold
 - ▷ Genre-specific keyword patterns

Quality Control:

- Description length \geq 200 chars
- Average confidence \geq 0.2
- Maximum confidence \geq 0.4



Results focus on high-confidence predictions rather than perfect recall across all categories

2025-06-11

Book Recommender using NLP

Category Inference Strategy

Category Inference Strategy

Two-Tier Classification Approach:

- **Primary:** Zero-shot classification with BART-MNLI
 - ▷ No training data required
 - ▷ 13 predefined book categories
 - ▷ Confidence scoring for predictions
- **Fallback:** Rule-based keyword matching
 - ▷ When confidence < threshold
 - ▷ Genre-specific keyword patterns

Quality Control:

- Description length \geq 200 chars
- Average confidence \geq 0.2
- Maximum confidence \geq 0.4

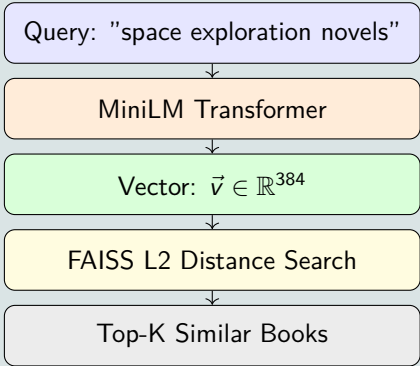


Sentence Embedding with MiniLM:

- **Model:** all-MiniLM-L6-v2
- **Input format:**
"Title: ... Author: ... Description: ..."
- **Output:** 384-dimensional vectors
- **Advantage:** Semantic similarity beyond keywords

Vector Search with FAISS:

- **Index:** 5,160 book embeddings
- **Search:** L2 distance (exact search)
- **Performance:** < 10ms query time
- **Local:** No external dependencies



End-to-end semantic search in < 200ms

2025-06-11

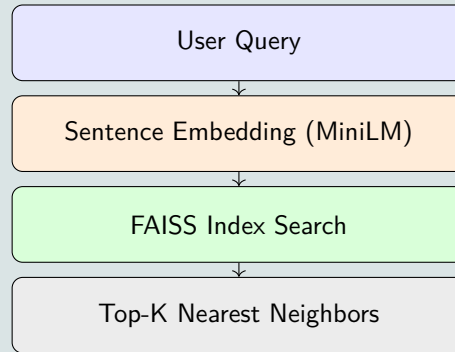
Book Recommender using NLP

Semantic Embedding & Vector Search





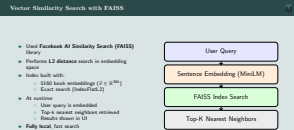
- Used **Facebook AI Similarity Search (FAISS)** library
- Performs **L2 distance** search in embedding space
- Index built with:
 - ▷ 5160 book embeddings ($\vec{v} \in \mathbb{R}^{384}$)
 - ▷ Exact search (IndexFlatL2)
- At runtime:
 - ▷ User query is embedded
 - ▷ Top-k nearest neighbors retrieved
 - ▷ Results shown in UI
- **Fully local**, fast search



2025-06-11

Book Recommender using NLP

Vector Similarity Search with FAISS





Implementation Strengths:

- **Privacy-preserving** by design
- **Semantic understanding** beyond keyword matching
- **Lightweight** - runs on consumer hardware
- **Modular architecture** for easy extension

Current Limitations:

- **No personalization** - stateless by design
- **Dataset scope** - 5,160 books vs. commercial scale
- **Cold start problem** for new books
- **No feedback learning** - static recommendations

Future Research Directions:

- **Hybrid approach:** Combine content-based with collaborative filtering
- **Better embeddings:** Experiment with domain-specific models
- **Privacy-preserving personalization:** Local user preference learning
- **Multi-modal features:** Include cover images, genre embeddings

"Demonstrates that local-first ML - or using a more popular term - edge AI, can provide meaningful semantic recommendations without compromising user privacy or requiring cloud infrastructure."

2025-06-11

Book Recommender using NLP

└ Critical Analysis & Future Directions

Implementation Strengths:

- Privacy-preserving by design
- Semantic understanding beyond keyword matching
- Lightweight - runs on consumer hardware
- Modular architecture for easy extension

Current Limitations:

- No personalization - stateless by design
- Dataset scope - 5,160 books vs. commercial scale
- Cold start problem for new books
- No feedback learning - static recommendations

Future Research Directions:

- Hybrid approach: Combine content-based with collaborative filtering
- Better embeddings: Experiment with domain-specific models
- Privacy-preserving personalization: Local user preference learning
- Multi-modal features: Include cover images, genre embeddings

"Demonstrates that local-first ML - or using a more popular term - edge AI, can provide meaningful semantic recommendations without compromising user privacy or requiring cloud infrastructure."



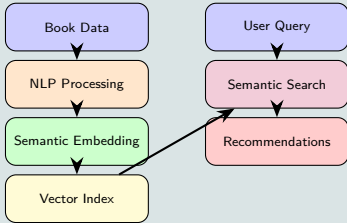
Research Question Answered:

How can a local ML model recommend books based on natural language descriptions?

Solution Implemented:

- **Semantic embeddings** with MiniLM transformers
- **Vector similarity search** using FAISS
- **Zero-shot classification** for categorization
- **Privacy-first design** - fully local processing
- **Key Contributions:** *Proof-of-concept that modern NLP enables practical, privacy-preserving recommendation systems*

System Architecture:



Impact & Applications:

- Educational tool for privacy-aware ML
- Foundation for local-first recommendation systems
- Demonstrates transformer accessibility on consumer hardware

2025-06-11

Book Recommender using NLP

Conclusion

Conclusion

Research Question Answered:

How can a local ML model recommend books based on natural language descriptions?

Solution Implemented:

- Semantic embeddings with MiniLM transformers
- Vector similarity search using FAISS
- Zero-shot classification for categorization
- Privacy-first design - fully local processing
- Key Contributions: *Proof-of-concept that modern NLP enables practical, privacy-preserving recommendation systems*

System Architecture:

Impact & Applications:

- Educational tool for privacy-aware ML
- Foundation for local-first recommendation systems
- Demonstrates transformer accessibility on consumer hardware



Thank you for your attention!



Questions?

2025-06-11

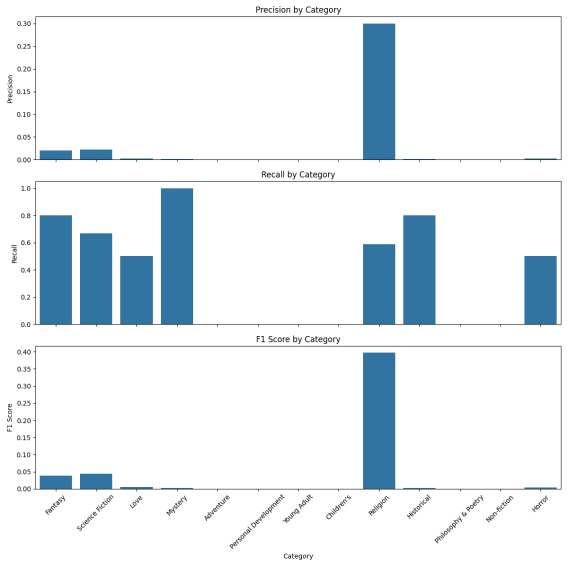
└ End of Presentation

Thank you for your attention!



Questions?

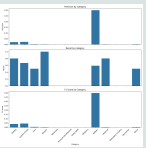
Backup: Per-Category Metrics



2025-06-11

Book Recommender using NLP

Backup: Per-Category Metrics





- ▶ Fallback used when zero-shot model confidence was low
- ▶ Example keywords:
 - ▷ **Fantasy:** magic, wizard, dragon
 - ▷ **Science Fiction:** space, AI, dystopia
 - ▷ **Love:** romance, passion, relationship
 - ▷ **Mystery:** detective, clue, crime

2025-06-11

└ Backup: Fallback Keywords

- ▶ Fallback used when zero-shot model confidence was low
- ▶ Example keywords:
 - ▷ **Fantasy:** magic, wizard, dragon
 - ▷ **Science Fiction:** space, AI, dystopia
 - ▷ **Love:** romance, passion, relationship
 - ▷ **Mystery:** detective, clue, crime



- Embedding time per book: ≈ 2 ms (batch embedding)
- Query embedding: ≈ 50 -200 ms
- FAISS search: < 10 ms
- UI render time: ≈ 1 -2 seconds (including image loading)
- All processing fully local on consumer-grade laptop

2025-06-11

Book Recommender using NLP

Backup: Performance

- Embedding time per book: ≈ 2 ms (batch embedding)
- Query embedding: ≈ 50 -200 ms
- FAISS search: < 10 ms
- UI render time: ≈ 1 -2 seconds (including image loading)
- All processing fully local on consumer-grade laptop



Example Query 1:

"Books about artificial intelligence and ethics"

Top Results:

- ▶ "The Alignment Problem" - Brian Christian
- ▶ "Life 3.0" - Max Tegmark
- ▶ "Weapons of Math Destruction" - Cathy O’Neil

Example Query 2:

"mystery novels with unreliable narrators"

Top Results:

- ▶ "Gone Girl" - Gillian Flynn
- ▶ "The Girl on the Train" - Paula Hawkins
- ▶ "In the Woods" - Tana French

What This Demonstrates:

- ▶ **Semantic understanding** beyond keywords
- ▶ **Abstract concept matching** (ethics, unreliable narrators)
- ▶ **Cross-genre discovery** potential

Evaluation Challenges:

- ▶ No ground truth for "perfect" recommendations
- ▶ Subjective nature of book preferences
- ▶ **Solution:** Focus on semantic relevance rather than prediction accuracy

Discussion Starters:

- ▶ How could one evaluate recommendation quality
- ▶ Books with sparse descriptions
- ▶ Could this approach work for other domains?

2025-06-11

Book Recommender using NLP

Backup: Concrete Examples

Example Query 1:
"Books about artificial intelligence and ethics"
Top Results:

- ▶ "The Alignment Problem" - Brian Christian
- ▶ "Life 3.0" - Max Tegmark
- ▶ "Weapons of Math Destruction" - Cathy O’Neil

Example Query 2:
"mystery novels with unreliable narrators"
Top Results:

- ▶ "Gone Girl" - Gillian Flynn
- ▶ "The Girl on the Train" - Paula Hawkins
- ▶ "In the Woods" - Tana French

What This Demonstrates:

- ▶ Semantic understanding beyond keywords
- ▶ Abstract concept matching (ethics, unreliable narrators)
- ▶ Cross-genre discovery potential

Evaluation Challenges:

- ▶ No ground truth for "perfect" recommendations
- ▶ Subjective nature of book preferences
- ▶ **Solution:** Focus on semantic relevance rather than prediction accuracy

Discussion Starters:

- ▶ How could one evaluate recommendation quality
- ▶ Books with sparse descriptions
- ▶ Could this approach work for other domains?