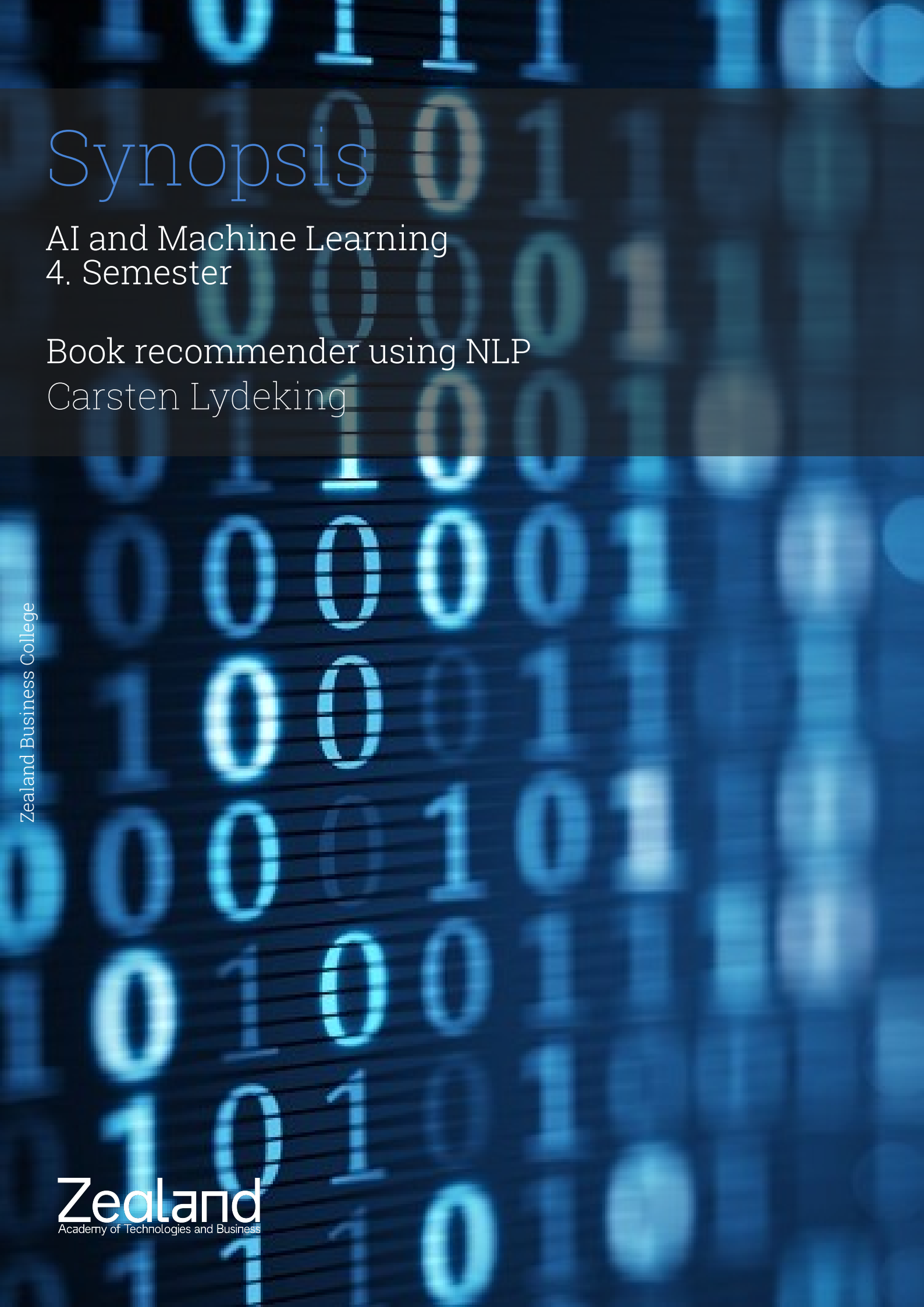

Synopsis

AI and Machine Learning
4. Semester

Book recommender using NLP

Carsten Lydeking

Zealand Business College

Zealand
Academy of Technologies and Business

Synopsis

AI and Machine Learning
4. Semester

Carsten Lydeking

cal002@edu.zealand.dk

Lecturer, AI and ML:	Jens Peter Andersen
Project Deadline:	30/05/2025
Handed-in:	08/05/25
Semester:	4. Semester
Word Count:	xxx Characters (including spaces)

Cover: Generated image of binary using DALL-E
Style: ZBC template – created by Carsten Lydeking (Cally)

Zealand
Academy of Technologies and Business

Contents

1 Introduction

1.1. Motivation

The purpose of this project is to explore the feasibility of running local language models in privacy-preserving environments. To this end, a book recommendation system has been selected as the practical foundation. The recommender leverages Natural Language Processing (NLP) techniques to analyze both book descriptions and free-form user queries, enabling content-based recommendations.

In contrast to most cloud-based recommendation systems, this project demonstrates a fully local setup where all data processing and inference occurs on the user's own machine. This ensures that no user data is transmitted or stored externally, providing strong privacy guarantees.

The goal is to evaluate whether smaller transformer-based models, specifically sentence transformers like MiniLM, are capable of delivering high-quality recommendations in an offline context. Techniques such as semantic embedding, vector similarity search, and metadata filtering are investigated to improve recommendation quality. In doing so, the project explores a broader question of how accessible and effective local AI systems can be for individual users.

1.2. Problem Definition

The project is centered around the following research question:

> *How can a local ML model be used to recommend books based on natural language descriptions, relying only on locally running models?*

This leads to three guiding sub-questions:

1. *What techniques exist for embedding text into meaningful vectors?*
2. *How can vector similarity be used for finding similar books?*
3. *What are the limitations of a local, content-only recommender?*

These sub-questions form the conceptual framework for the research and analysis presented in the following chapters.

1.3. Planning

The project was conducted over five weeks and divided into the following phases:

- **Week 1 –** Literature review and background research on semantic similarity, embedding models, and recommender systems.

- **Week 2 –** Data exploration and cleaning, including handling of missing values and metadata normalization.
- **Week 3 –** Embedding and indexing: generating dense sentence embeddings and setting up FAISS for similarity search.
- **Week 4 –** Development of a prototype user interface and live experimentation with queries and filters.
- **Week 5 –** Final testing, evaluation, and writing of the synopsis.

2

Methodology and Structure

This project follows a research-based prototype methodology. Rather than developing a commercial software product, the goal is to explore the scientific and technical feasibility of running semantic book recommendations using local machine learning tools.

The findings—both theoretical and experimental—constitute the core deliverables of this project.

2.1. Research Approach

The project is based on a combination of literature review and hands-on implementation. Each aspect of the methodology was selected to support answering the sub-questions defined in Section 1.2.

- **Literature Review:** Relevant topics included natural language processing (NLP), sentence embedding techniques, recommender systems, and similarity search. Key technologies such as `MiniLM`, `FAISS`, and `Streamlit` were studied through documentation and academic sources.
- **Implementation:** The following tools and libraries were used throughout the project:
  - `Python` for scripting, preprocessing, and experimentation.
  - `pandas`, `seaborn`, and `matplotlib` for data analysis and visualization.
  - `sentence-transformers` (MiniLM-L6-v2) to generate semantic vector representations.
  - `FAISS` to index and search high-dimensional embeddings efficiently.
  - `Streamlit` to build an interactive, privacy-preserving user interface.
- **Testing:** Practical testing included a range of user queries and filtering conditions to observe whether recommendations matched the query intent semantically.

2.2. Evaluation Criteria

As no supervised training or labeled ground-truth data was involved, the system is evaluated using non-traditional metrics. The following criteria were applied:

- **Qualitative Relevance:** Whether the recommendations appear semantically relevant to a human evaluator.
- **Responsiveness:** How quickly the system responds to user queries on consumer hardware.

- **Offline Capability:** Verification that all processing occurs locally, without internet access.
- **Scalability:** Exploration of performance with larger datasets and indexing sizes.

This methodology supports the central research question posed in section 1.2, and particularly sub-questions 1 and 2, by grounding each system component in both theoretical research and empirical testing.

3

conclusion

This is the conclusion of the document. It summarizes the main points and findings discussed in the previous chapters. The conclusion also reflects on the implications of the research and suggests areas for future work or study.

3.1. This is a section