Decoding the Epistemic Climate

An Analysis of Computer Science Course Materials Using NLP

Mitch Gerhardt — December 5, 2024

Big Picture!

How do engineering course materials shape students' understanding of engineering knowledge?

Why this matters:

- Engineers tackle complex problems requiring a sophisticated understanding of knowledge (Litzinger et al., 2013)
- Educational materials shape how students view and use knowledge (Freuht, 2010; Muis et al., 2016)
- Understanding these influences can improve engineering education (Hofer, 2001)

Our Approach:

- Analyze engineering textbooks using NLP
- Examine how knowledge is presented and described
- Identify patterns in knowledge representation

What is a knowledge type?

Defining textbook “knowledge”

Purpose:

- Serves specific educational & professional goals
- Has distinct learning outcomes
- Contributes uniquely to engineering understanding
- Aligns with institutional frameworks (e.g., NPSE’s Engineering BOK)

Linguistic Characteristics:

- Uses distinct vocabulary and terminology
- Has characteristic sentence structures
- Employs specific rhetorical patterns
- Can be recognized using NLP

Epistemic Implications:

- Shapes what “counts” as knowledge
- Influences how knowledge is justified
- Affects the perceived certainty of knowledge
- Determines authority sources
- Impacts students’ understanding of the profession

Purpose: Theoretical understanding of algorithms

"Quicksort has an average time complexity of O(n log n). It uses a divide-and-conquer strategy by selecting a pivot element and partitioning the array around it."

Linguistic: Uses formal notation, definitions, theoretical terms

Epistemological: Based on mathematical proofs, presented as certain

“Quicksort has an average time complexity of O(n log n). It uses a divide-and-conquer strategy by selecting a pivot element and partitioning the array around it.”

“When implementing QuickSort, choose the median of three elements as the pivot to avoid worst-case performance on nearly-sorted arrays.”

“In professional systems, use built-in sorting functions for arrays under 50 elements. Custom QuickSort implementations rarely outperform optimized system libraries.”

Purpose: Practical implementation guidance
Linguistic: Uses imperative statements, conditional logic
Epistemological: Knowledge validated through performance testing

Purpose: Real-world best practices
Linguistic: References industry context, trade-offs
Epistemological: Authority from professional experience, multiple valid approaches

Determining Knowledge Types in CS Textbooks

Using our definition of “knowledge type,” what types exist?

Process:

- Manual textbook analysis
- AI-assisted pattern identification
- Framework refinement
- Final validation

Key types identified:

- Conceptual
- Procedural
- Epistemic
- Ethical
- Uncertainty
- Historical
- Interdisciplinary
- Metacognitive
- Mathematical
- Professional

Example: A sorting algorithm can be presented as conceptual (time complexity theory), procedural (implementation steps), professional (production best practices), or a combination!

Current Approaches to Textbook Analysis

Close Reading & Coding:

- Time-intensive manual analysis of content (Bock, 2018; Weinbrenner, 1992)
- Limited to small sample sizes
- Challenges in maintaining coding consistency
- Requires domain expertise

Frequency Analysis:

- Counts specific terms or concepts
- May miss semantically similar language (Katz et al., 2021)
- Reduces complex ideas to simple units
- May miss deeper meaning and connections

Common Challenges:

- Difficult to compare across contexts and time periods
- Resource-intensive methods limit sample size (Son & Diletti, 2017)
- Coding schemes may not transfer between textbooks (Rezat & Strässer, 2015)

Textbook Knowledge Content:

- Son & Diletti (2017) - 17 US and Chinese textbooks; Compared the content and problems presented
- Okeefe (2012) - Analyzed textbook Content, Structure, Expectation, and Language
- Latour (2007) - Notes the move from epistemological to ontological foundations in science textbooks

Data Generation Approach

Currently, no training data for knowledge type classification

Approach:

- Use LLMs to generate synthetic textbook passages
- Control for topic, context, location
- Create a diverse, balanced dataset

Content Controls:

- Theoretical discussion
- Mathematical derivation
- Case studies
- Worked examples

Location Controls:

- Section beginning
- After equation
- Before exercise
- Sidebar

Courses:

- Taken from the VT undergraduate curriculum (textbook from each required CS course)
- *Examples*: Software Design, Data Structures, Computer Systems, Programming Languages

Next Steps

From data to knowledge classification

Current Progress:

- Generated 10k labeled training samples
- Developed BERT-based classifier
- Achieved ~79% micro F1-score

Architecture:

- Includes attention pooling
- Multi-layer classifier head with dropout
- Focal loss for imbalanced classes
- Dynamic threshold optimization per class

Performance:

- Strongest classes: Ethics (0.97), Historical (0.86)
- Weakest classes: Math (0.68), Procedural (0.68)
- Class-specific thresholds: 0.55 - 0.8

Upcoming Work:

- Evaluate classifier on real textbook passages
- Analyze the distribution of knowledge types
- Compare patterns across CS courses

Thanks!

Questions?

Supplemental Slides

Curriculum: This encompasses the instructional content, materials, and resources used
- Textbooks, lecture notes, assignments
- Types of content, e.g., open-ended, patterns, longitudinal

Pedagogy: The pedagogical approach involves how knowledge is taught and constructed in the classroom
- Pair programming
- Lecture-based instruction
- Encouraging discussions

Support: This involves scaffolding students through changes in their understanding of knowledge
- Supporting students during uncertainty
- Acknowledge emotional aspects of learning
- Helping students develop strategies

PACES

Muis et al., 2016

Authority: This relates to how knowledge sources and expertise are positioned in the learning environment
- Teachers always “correct”?
- Peer discussions
- Having students justify decisions vs implementing

Evaluation: This involves how learning is assessed and what is valued
- Exams, assignments, grading
- Types of evaluation, e.g., real-world projects, process & justification, reflection

Aims and Values: The epistemic goals someone seeks to achieve and what they consider important in developing knowledge - like pursuing understanding, creating accurate models, or developing useful theories

Reliable Processes: The methods and practices someone trusts and uses to create or validate knowledge - for example, conducting experiments, engaging in systematic observation, or verifying through peer review

AIR

Chinn et al., 2010, 2014

Ideals: The standards or criteria that someone uses to evaluate knowledge claims and justify their beliefs - such as requiring evidence, valuing logical consistency, or demanding replicability

Aims and Values: The epistemic goals someone seeks to achieve and what they consider important in developing knowledge - like pursuing understanding, creating accurate models, or developing useful theories

Reliable Processes: The methods and practices someone trusts and uses to create or validate knowledge - for example, conducting experiments, engaging in systematic observation, or verifying through peer review

AIR

Ideals: The standards or criteria that someone uses to evaluate knowledge claims and justify their beliefs - such as requiring evidence, valuing logical consistency, or demanding replicability

Example: When Sarah encounters a performance issue in her web application (aim: to understand and optimize system performance), she evaluates different solutions by examining their Big O complexity and running benchmark tests (epistemic ideals: efficiency and empirical verification), ultimately choosing to validate her understanding by implementing the solution, running comprehensive tests, and having her code reviewed by senior developers on her team (reliable epistemic processes).