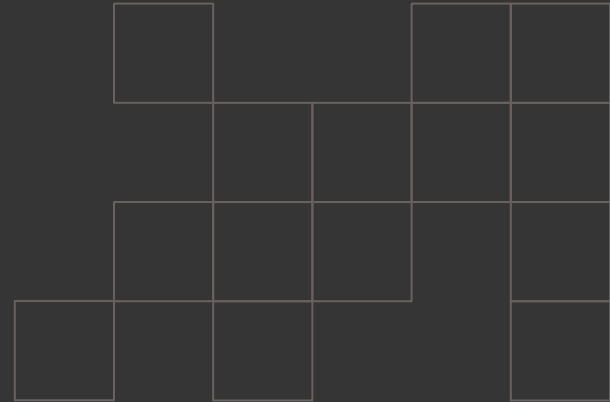25th August 2025

Effective use of AI tools
Software Engineering
Vs
Business Administration
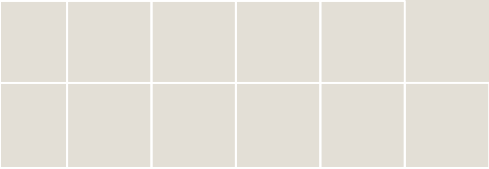
By Matei Avram

Introduction

Research Question

- ChatGPT introduced in Nov. 30, 2022

- Increasing speculation about future of Software Engineering field

Can software engineering skills translate into effective AI usage performance compared to business students in a real life engineering scenario?

Sub-Questions

- Does prior SE domain experience measurably improve prompt engineering skills?

- How do SE students compare to BA students when using AI tools to solve task in a real life work setting?
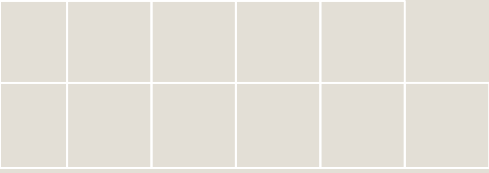
Importance

- Helps understanding where the job market might be headed

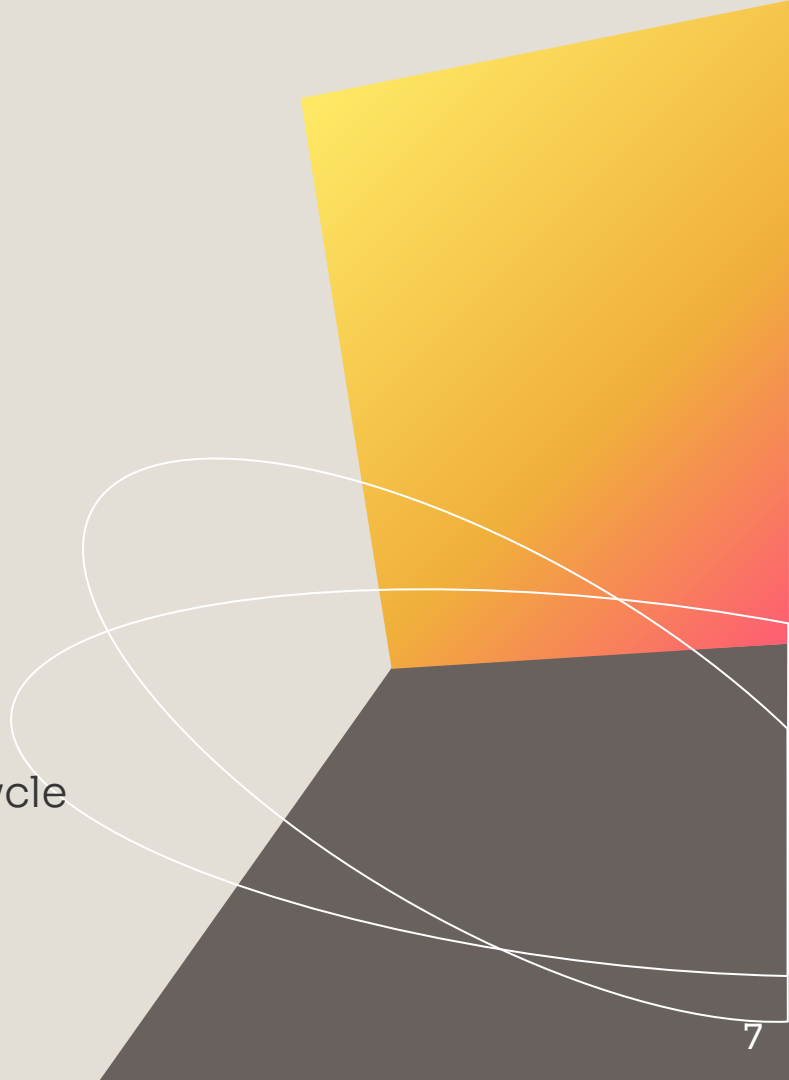- Helps universities better prepare students for real life working environments

5

Related work

LLMs fundamentals

- Generates words
- Susceptible to hallucinations

Usage

- Throughout the Software Development life cycle
- Concerns about project quality
- Some concerns about performance

Prompt engineering techniques

Chain-of-thought prompting

Boost model reasoning by breaking problems into steps.

Tree-of-thought prompting

Request model to explore multiple next steps to reach the goal

Maieutic prompting

Deepen model understanding through iterative explanations

Prompt engineering techniques

Least-to-most prompting

Ask model to break problem into sub-problems and solve sequentially
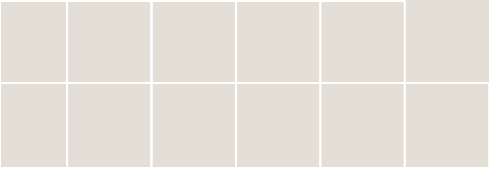
Self-refine prompting

Request model to improve its solution through self-critique and revision

Directional-stimulus prompting

Guide models using key-words to describe output generation.
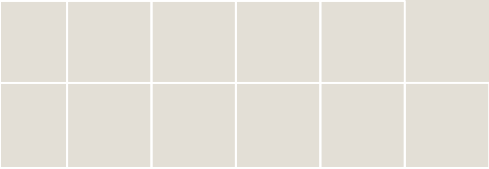
Study Design

How?

Conduct an experiment:

- 2 groups (SE vs BA)

- Solve tasks within a simulated platform with access to ChatGPT o4-mini

- Analyze data

Platform

Mock-up E-Ticket web application where users can insert or delete tickets

Tasks

1. Add html form to insert ticket
2. Implement the DELETE ticket functionality
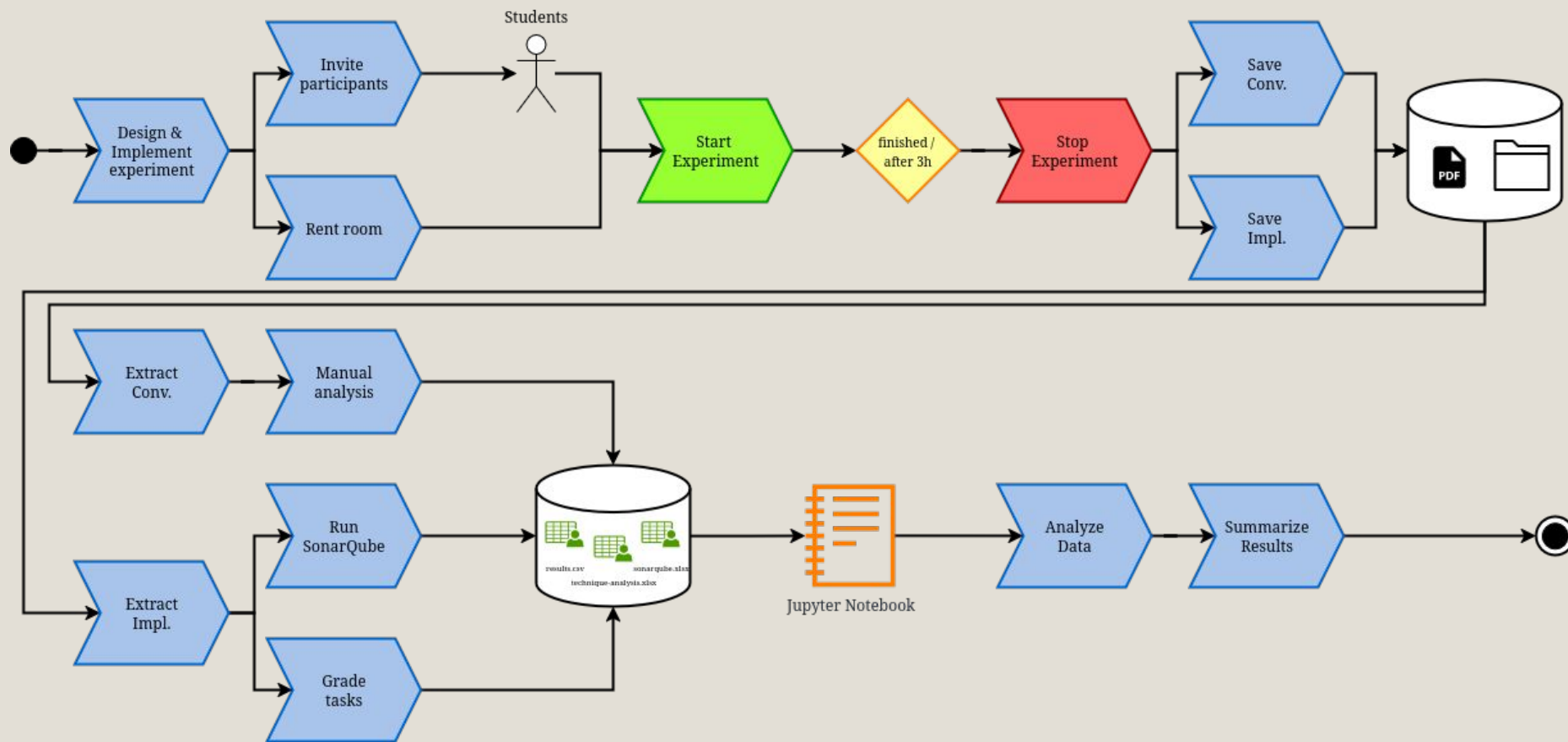3. Update ticket prices according to dynamic pricing formula
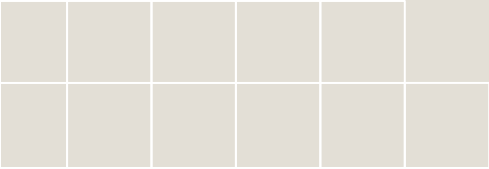
What are we analyzing?

- Task success rate

- Final student project implementation quality using SonarQube

- Manual analysis of students' conversations with the AI model

Students

Design & Implement experiment → Invite participants → Rent room

Start Experiment → finished / after 3h → Stop Experiment → Save Conv. / Save Impl.

Extract Conv. → Manual analysis

Extract Impl. → Run SonarQube / Grade tasks

results.csv   sonarqube.xlsx
technique-analysis.xlsx

Jupyter Notebook → Analyze Data → Summarize Results

Results
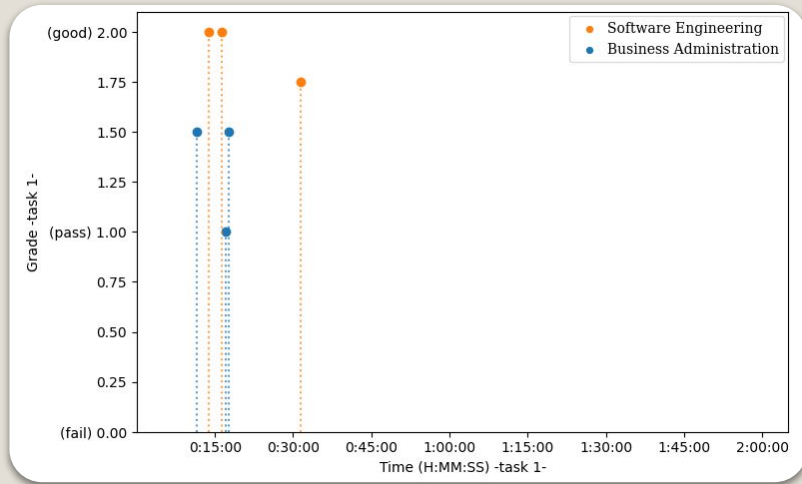
Task success rate

Measured **Time & Grade**

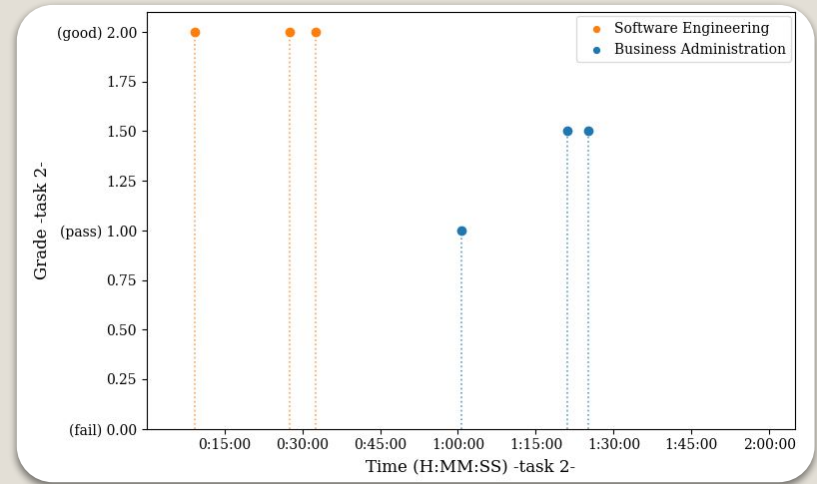Individual performance   ⇒ Scatter plots

Avg. Group performance   ⇒ Bar plots

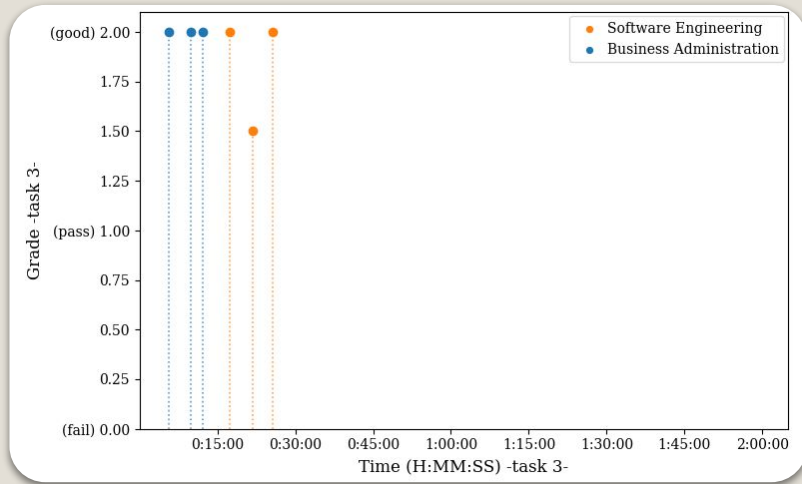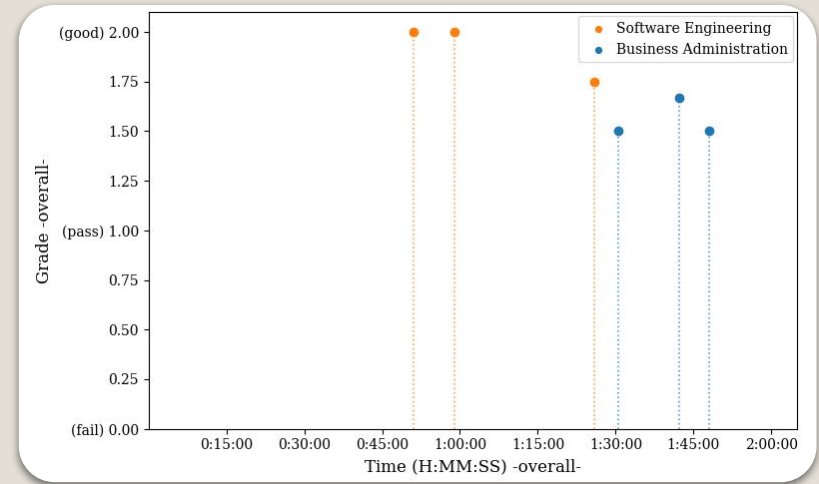# Student completion time & grade distribution

Task 1
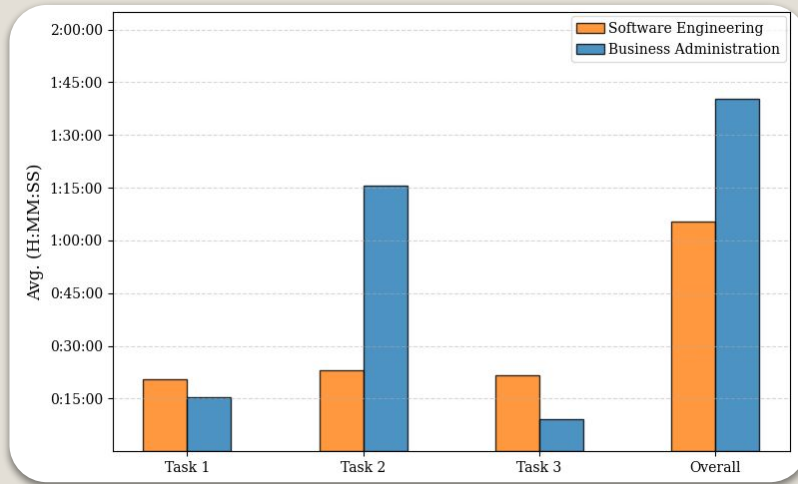

Task 2

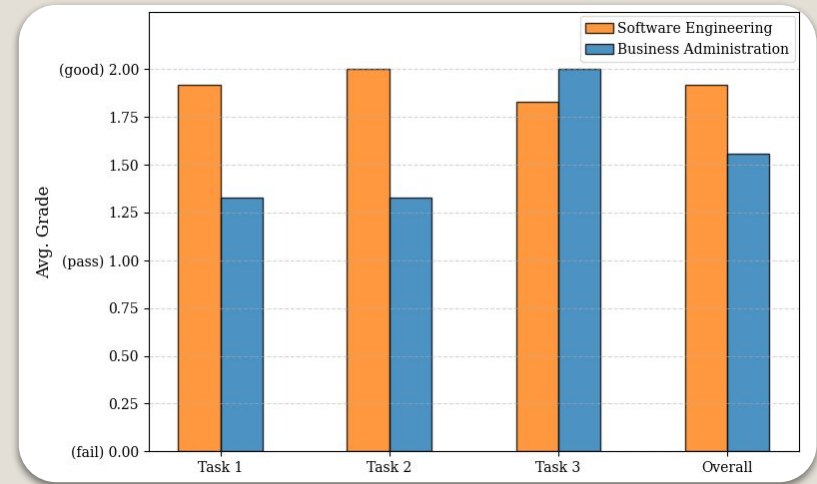# Student completion time & grade distribution

Task 3

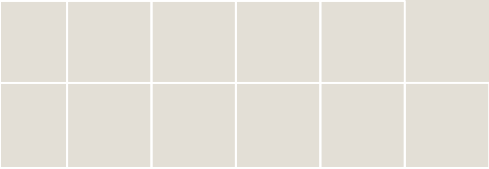
Overall

Avg. Group completion time & grade

Time

Grade

SE students more consistent faster average times
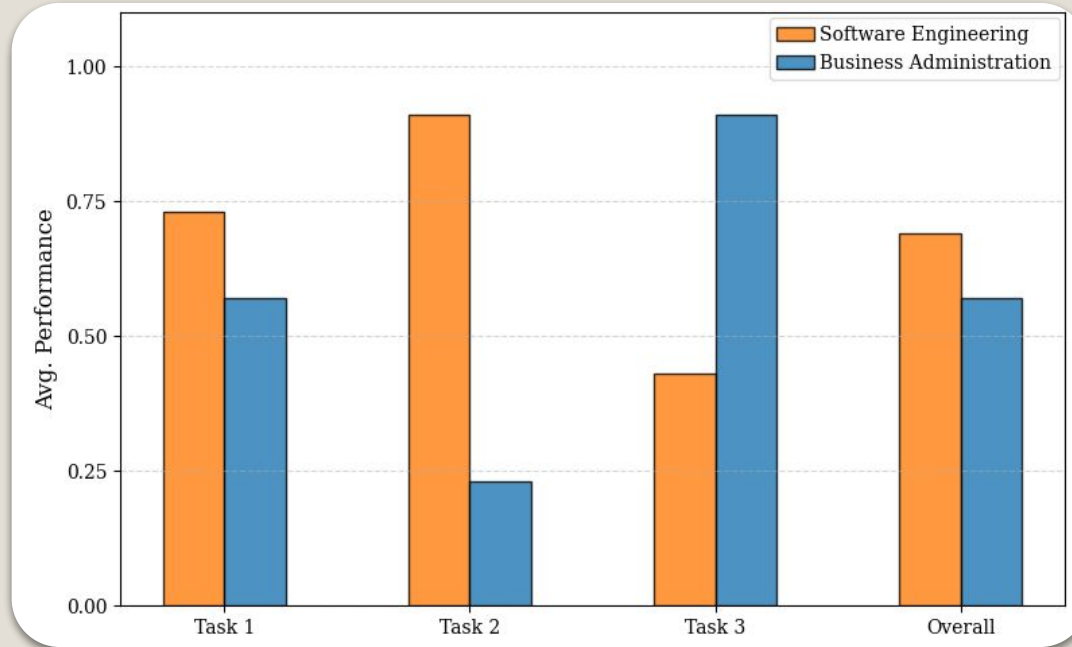
SE students higher average grade

Avg. Group performance

1. Normalized **Time & Grade** metrics

2. Invert **Time** as more time spent = decreased performance

3. Avg. the two metrics

Avg. Group performance

On average SE students outperformed BA students

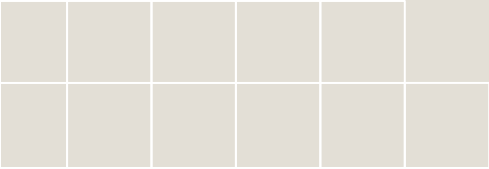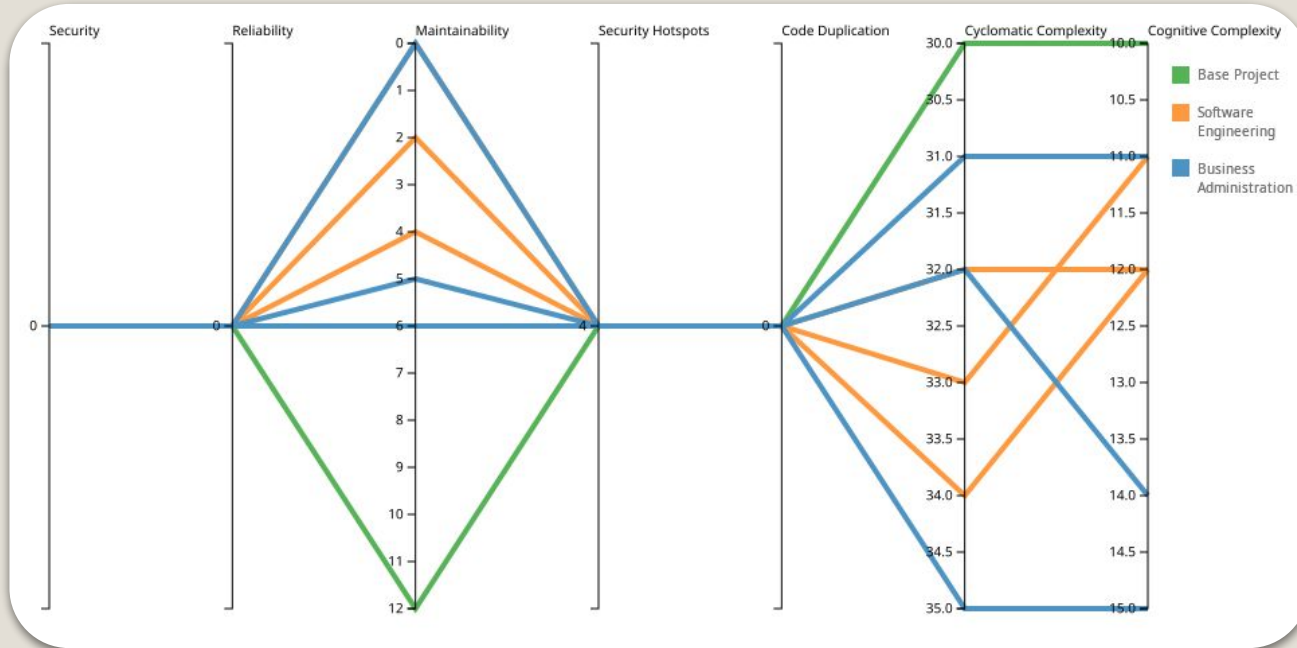# SonarQube analysis

Project quality ⇒ Parallel coordinates plots

- Security
- Reliability
- Maintainability ⇒ Modified
- Security hotspots
- Code Duplication %
- Cyclomatic Complexity
- Cognitive Complexity

Comparison of software quality metrics

AI conversation analysis

- Searched for prompt engineering techniques

- Reviewed workflow used by students and its impact.
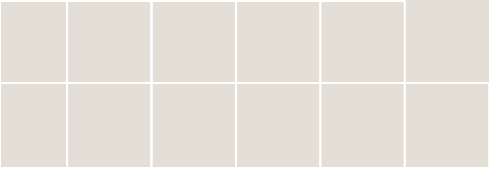
Summary of findings

Software Engineering

- Strongly used "Directional-stimulus prompting"

- Employed some use of "Chain-of-thought", "Least-to-most" and sometimes "Self-refine" prompting techniques

- Used some "Maieutic prompting" for debugging

Business Administration

- Used some "Directional-stimulus prompting"

- Employed more "Least-to-most" rather then "Chain-of-thought" prompting styles, however to little extent

- Used "Maieutic prompting" to understand core Software Engineering concepts

Discussion

Observations

- SE students had a more accuracy oriented approach (seemingly aware of the model's potential limitations)

- BA students used the model as a guiding tutor (rather than a collaborative tool)

- Both groups' prompts were successful more due to the model's ability to understand their intentions.
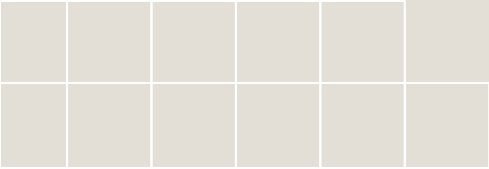
Study limitations

- Task description copy-pasting

- Simplified business task

- Standardized browser

- Mock-up application size

- Number of participants

Conclusion

Conclusion

- On average SE students outperformed BA students in this experiment

- Domain knowledge had a measurable impact in navigating the environment and effectively validating the model's outputs
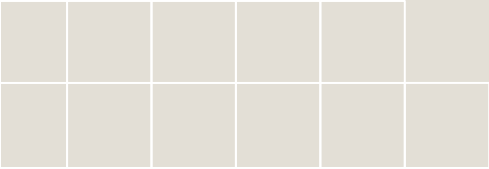
Implications

- Software engineering expertise is still required to utilize LLMs effectively. Otherwise companies risk confidentiality issues and significant time increase in solving platform issues.

- Students would benefit from prompt engineering courses to learn how to communicate their requests more accurately

Future work

New experiment:

- Compare 2 teams:
    - Senior software engineers + AI
    - Senior software engineers + juniors
- Analyze team performance

Effective use of AI tools
Software Engineering
Vs
Business Administration

By Matei Avram