Leveraging LLMs for Automated Feedback Generation on Exercises

Master‘s Thesis Intermediate Presentation

Author: Felix Timotheus Johannes Dietrich

Supervisor: Prof. Dr. Stephan Krusche

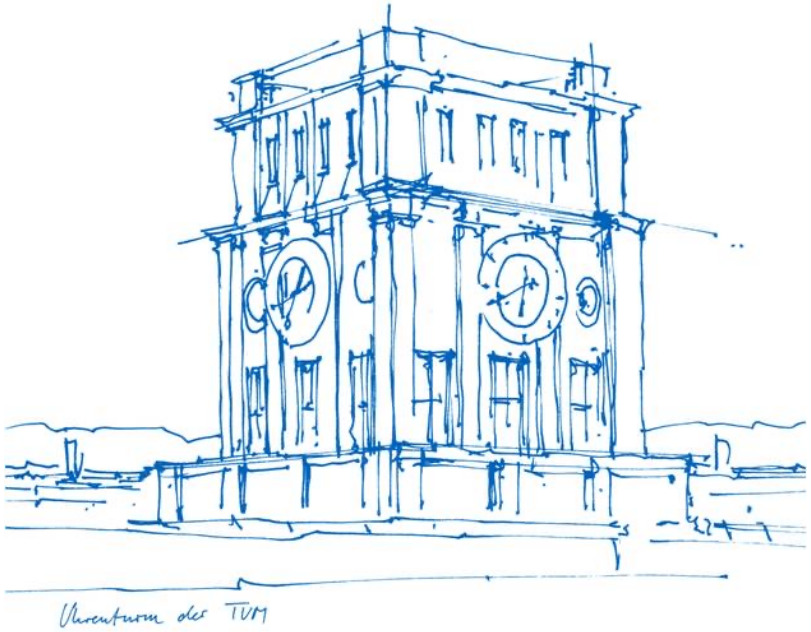
Advisor: Maximilian Sölch, M.Sc.

Technical University of Munich

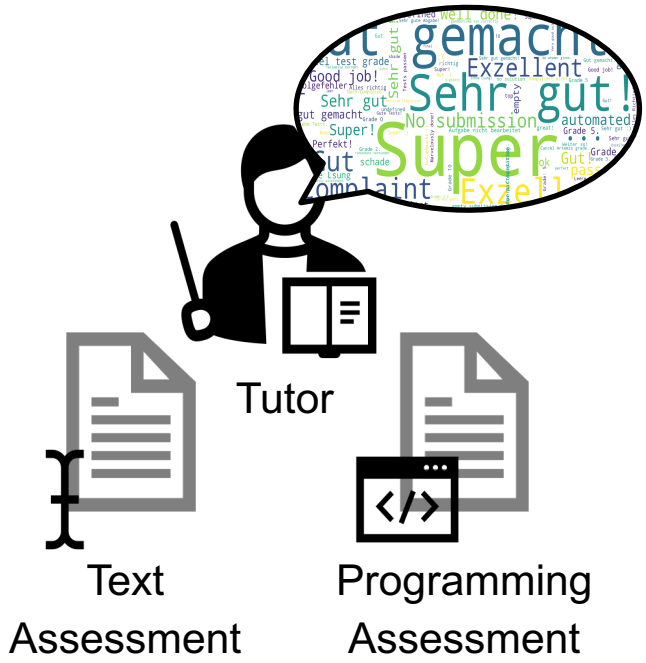
TUM School of Computation, Information and Technology

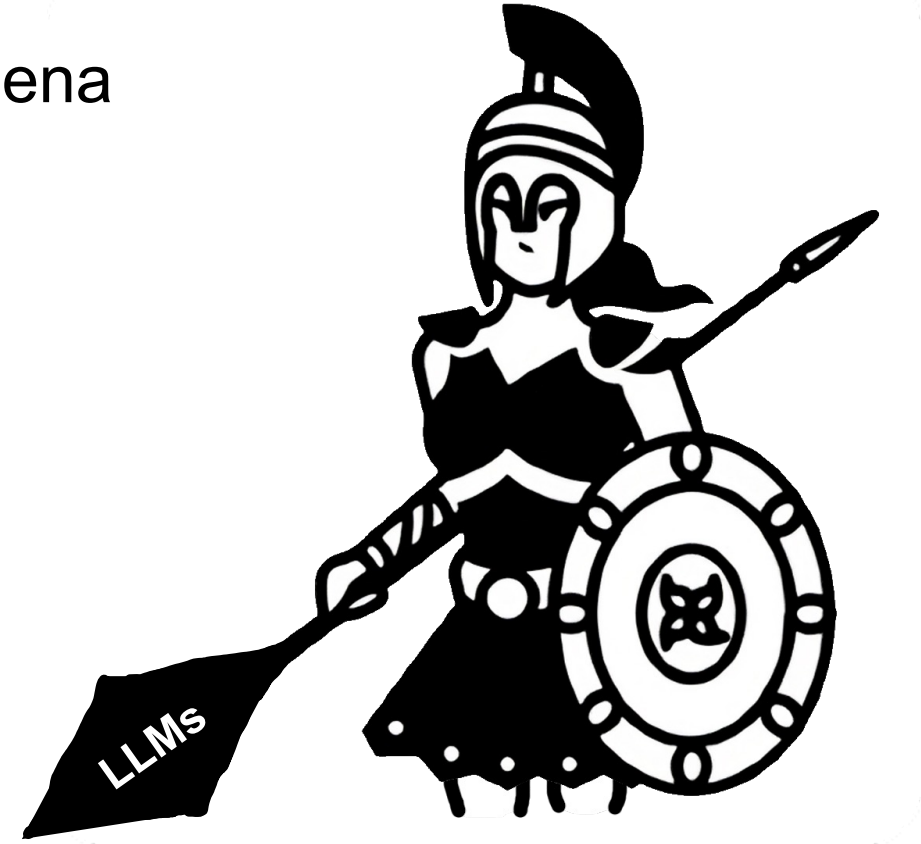
Chair for Applied Software Engineering

Munich, 11. September 2023



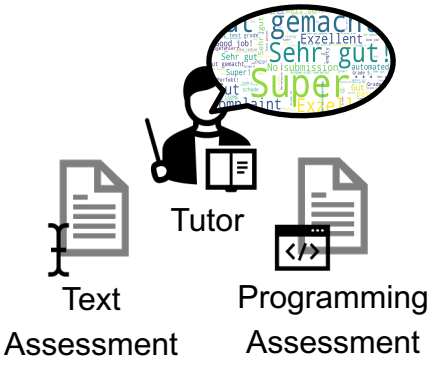
LLMs – A New Weapon for Athena



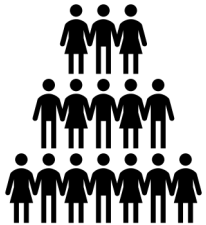


Outline

Problem and Motivation → Requirements Analysis → System Design → Text Exercises → Programming Exercises → Status → Future Work



Problems



Scalability of
Manual Feedback

Assessment is
Time Consuming

Feedback
Quality/Quantity Suffers

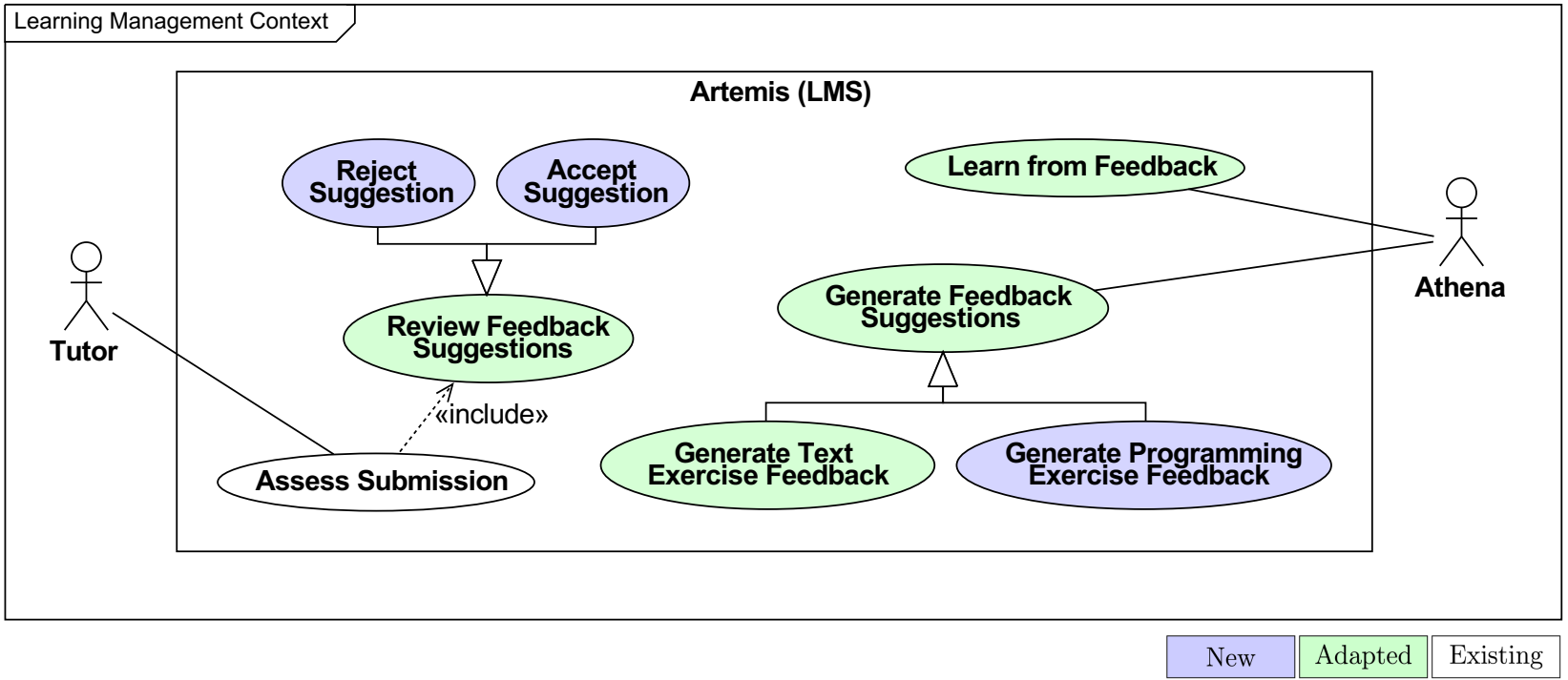
Motivation

Improve Learning Experience

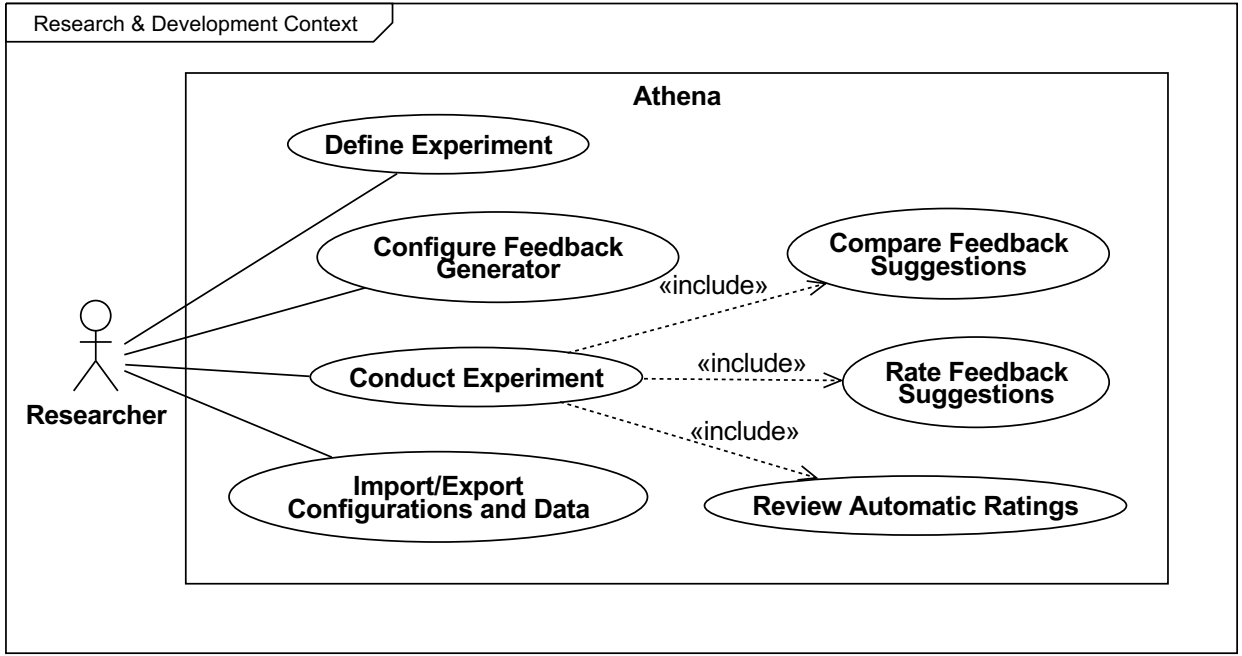
Reduced Workload for Tutors

Leverage LLMs

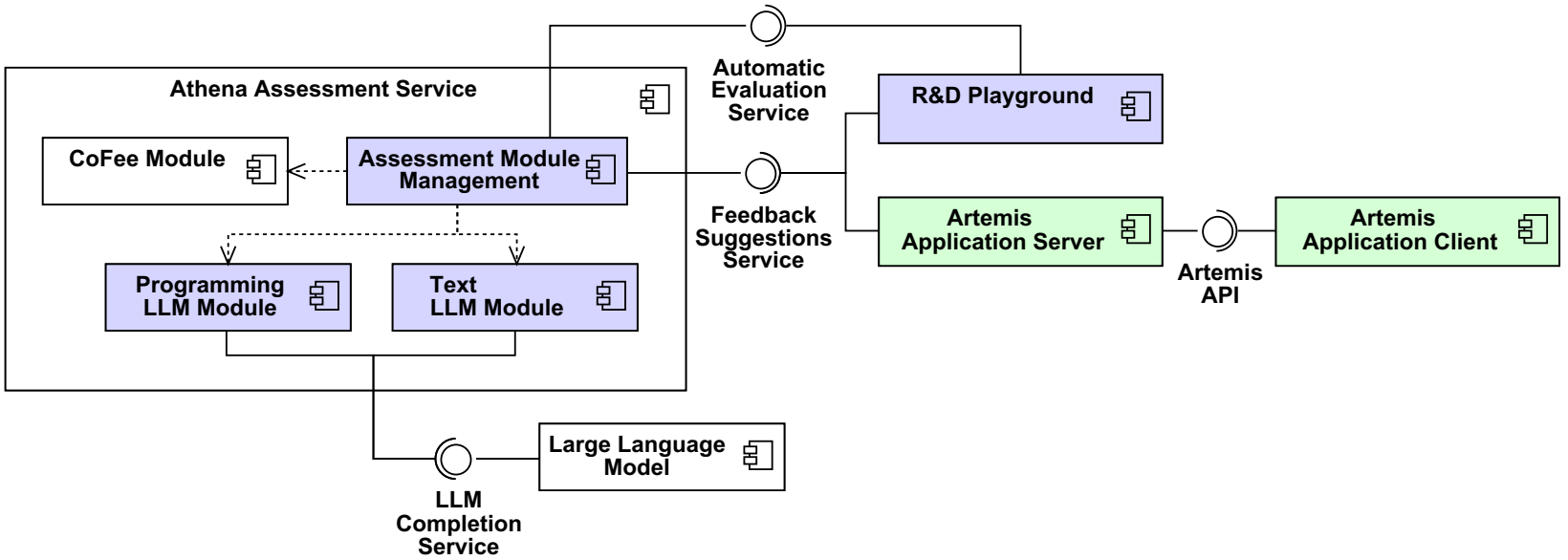
Requirements Analysis



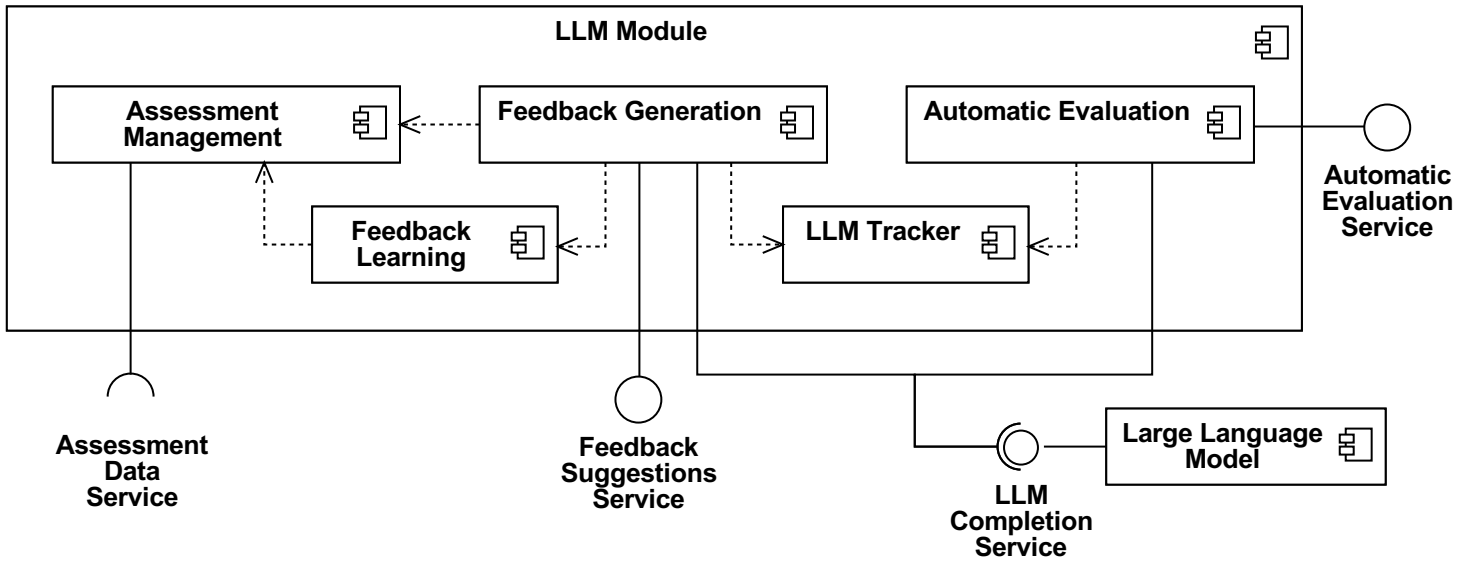
Requirements Analysis



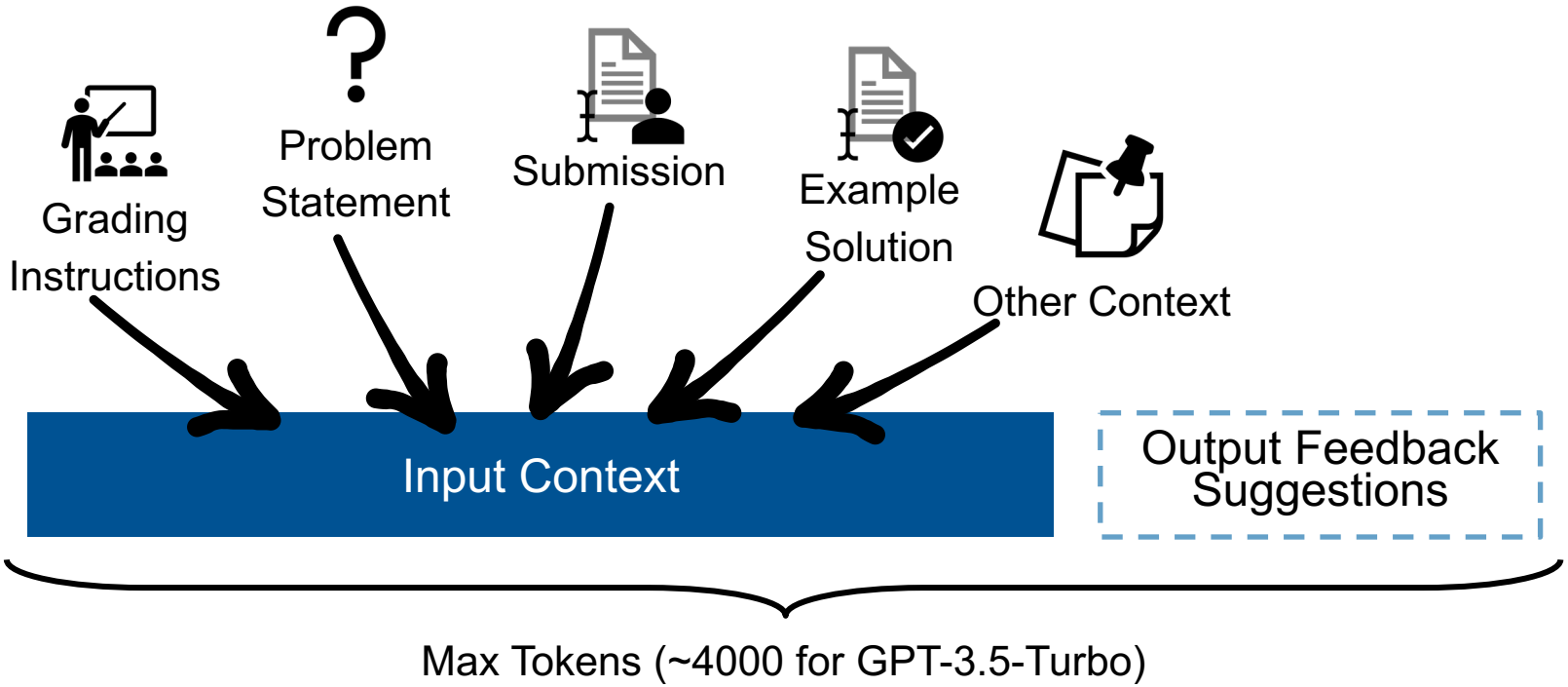
System Design



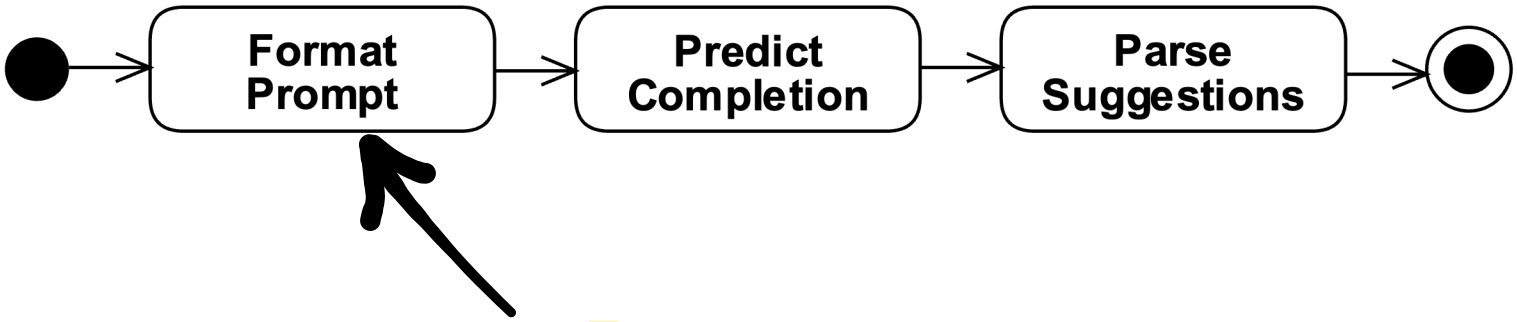
System Design



How to Generate Feedback for Text Exercises?



How to Generate Feedback for Text Exercises?



*Simplified for presentation

Tutor

Text Assessment

DEMO

Text Exercises

LLM-as-a-Judge

GPT-4

System Message

```
1  You are now an evaluator for feedback accuracy generated by
    a machine-learning system.
2
3  # Task
4  Your task is to estimate if a human tutor would accept or
   reject the feedback suggestion and how much modification is
    needed to make the feedback useful.
5
6  # Score Criteria
7  Accept feedback that is useful to the tutor, meaning that
   it can be applied to the submission with minor or no
   modification. Our goal is to reduce the workload of tutors
   and reduce their cognitive load. Reject feedback that is
   not useful and would burden the tutor.
8
9  Put the focus on the description of the feedback, the title
    is optional. The `line_start` and `line_end` should make
   sense with respect to the submission but do not need to be
   exact. Credits should make sense with respect to the
   feedback and the submission but also do not need to be
   exact.
10
11 # Submission (with sentence numbers <number>: <sentence>):
12 {submission}
13
14 # Example (Human) Feedback:
15 {true_feedbacks}
```

Evaluation Prompt

Human Message

```
1  ### Model Output:
2  {predicted_feedbacks}
```

Approximate Human Preference

👍 Accept | 👎 Reject

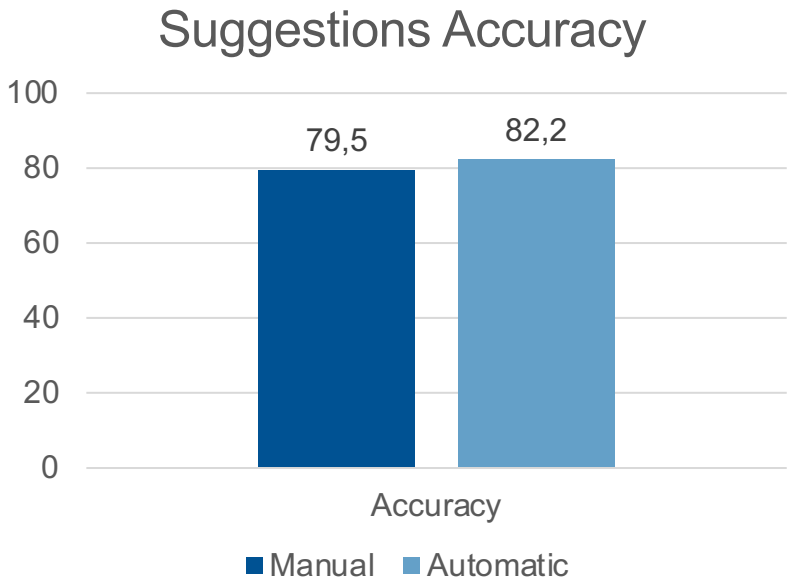
GPT-4 estimates **accepted** with **minor** modification

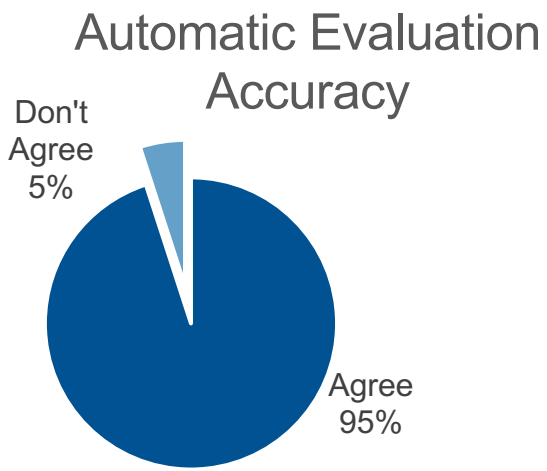
Evaluation – Text Exercises (Multiple LLMs)

Model	Tokens	Time (s)	Feedbacks	Est. Acc. (%)
GPT-3.5-Turbo	1522	5.58	3.03	82.2%
GPT-4	1406	10.09	3.32	80.1%
LLaMA-2-13b-Chat	2008	22.21	1.91	44.5%

Benchmarked on 100 submissions of H04E01 Coupling and Cohesion

Evaluation – Text Exercises (LLM-as-a-judge)





Benchmarked on 100 submissions of H04E01 Coupling and Cohesion (303 pieces of feedback)

Evaluation – Text Exercises (Multiple Exercises)

ID	Tokens	Time (s)	Feedbacks	Est. Acc. (%)
4160	1522	5.58	3.03	82.2
4101	1813	3.43	1.22	64.8
4238	2145	9.41	6.75	78.3
4082	2196	10.24	7.04	73.6
4162	1961	8.30	3.01	98.0

Benchmarked on 100 submissions each

Evaluation – Text Exercises (Multiple Exercises)

ID	Tokens	Time (s)	Feedbacks	Est. Acc. (%)
4160	1522	5.58	3.03	82.2
4101	1813	3.43	1.22	64.8
4238	2145	9.41	6.75	78.3
4082	2196	10.24	7.04	73.6
4162	1961	8.30	3.01	98.0

~$0.004 per submission with GPT-3.5-Turbo

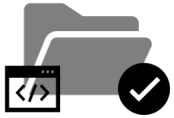
Performs well and costs are low 🎉

Next Step: Live evaluation on Artemis

Programming Exercises

How to Generate Feedback for Programming Exercises?

Long Problem Statement



Solution Repository

Repository Differences

Tests Results

Too Much Context!!

Long Grading Instructions

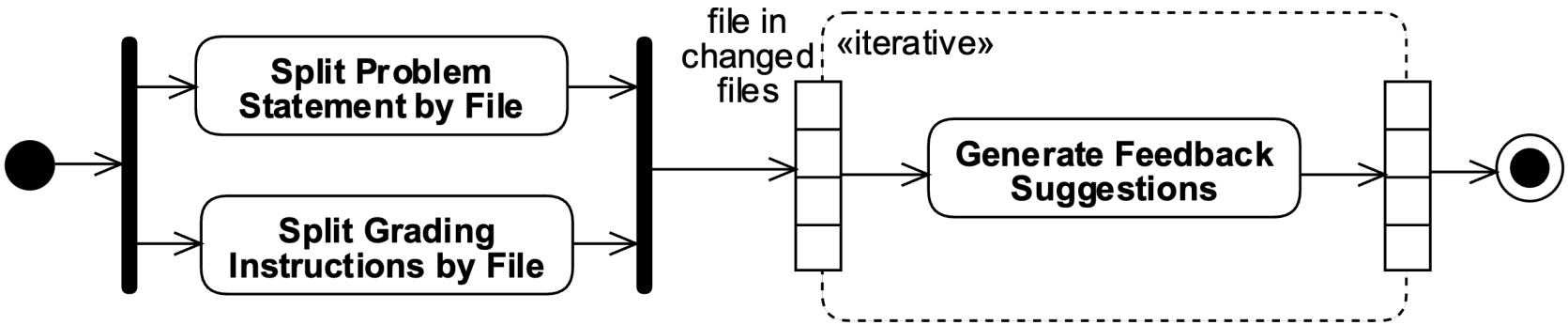
Submission Repository

Build Outputs

Template Repository

Much more Context

Generate Feedback Suggestions per File



*Simplified for presentation

Tutor

Programming Assessment

DEMO

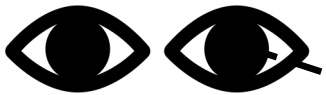
Programming Exercises

Evaluation – Programming Exercises

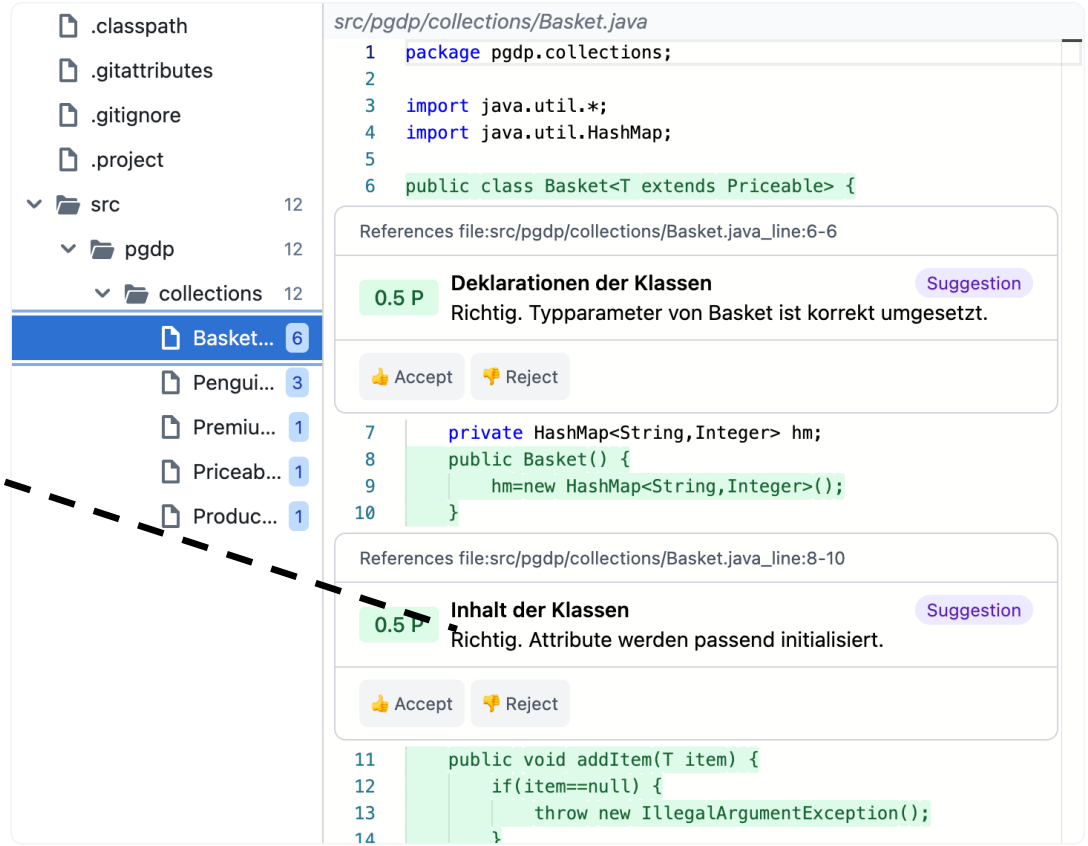
Model	Tokens	Time (s)	Feedbacks
GPT-3.5-Turbo	12953	57.08	10.81
GPT-4	11409	160.85	18.70

~$0,026 per submission with GPT-3.5-Turbo
~$0,35 per submission with GPT-4

Evaluated on 48 submissions each (Shoppende Pinguine)



Eyeballing the Suggestions



There is some usable and correct feedback

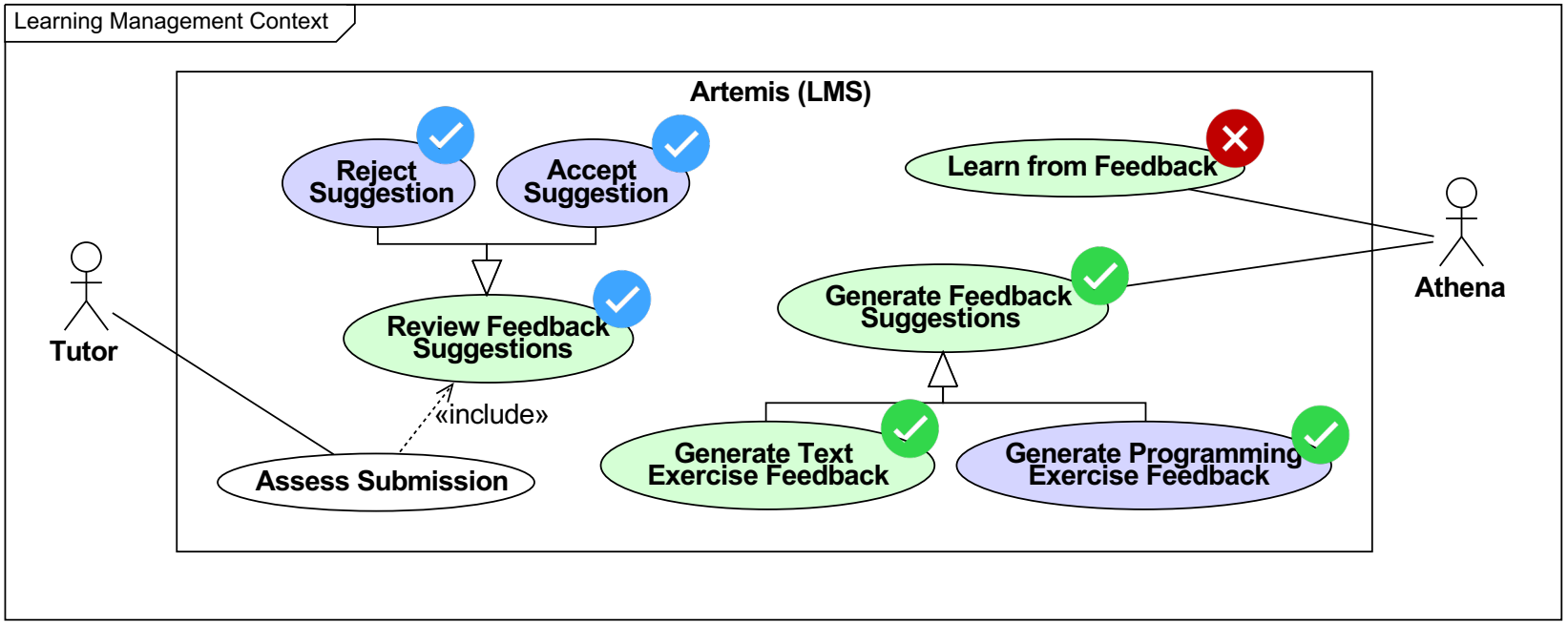
Evaluation – Takeaways

Not usable for tutors, ***yet***
But overall very promising with a lot of potential!

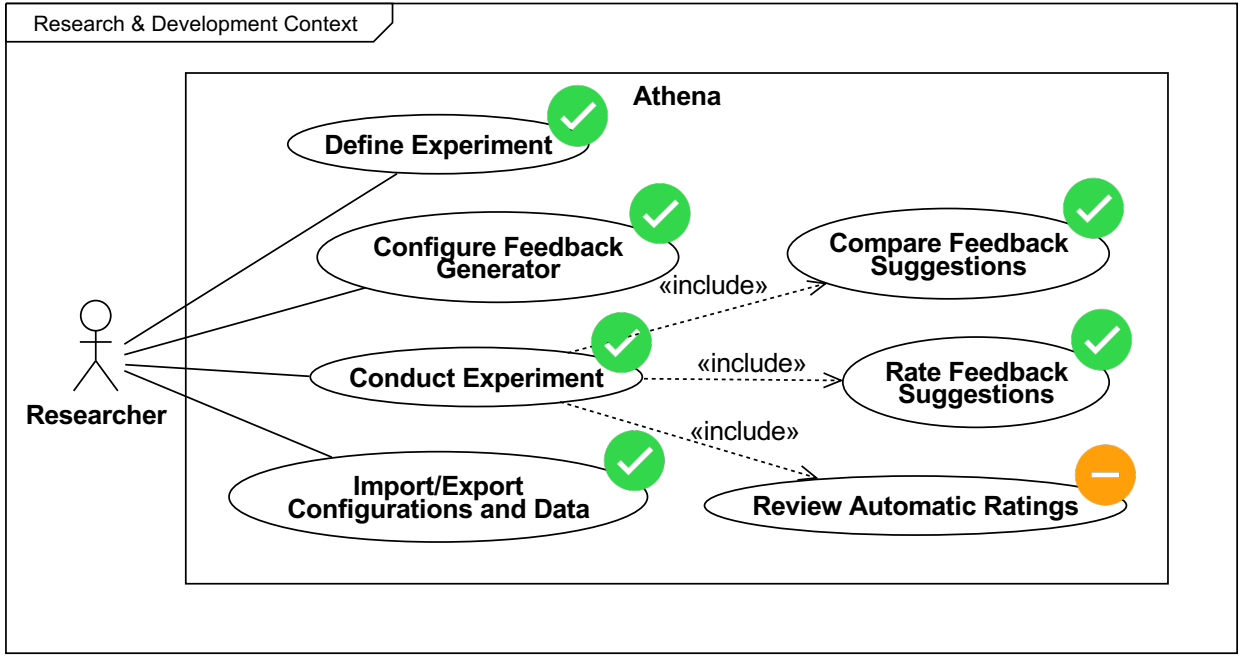
Next Step: More sophisticated approach

Status

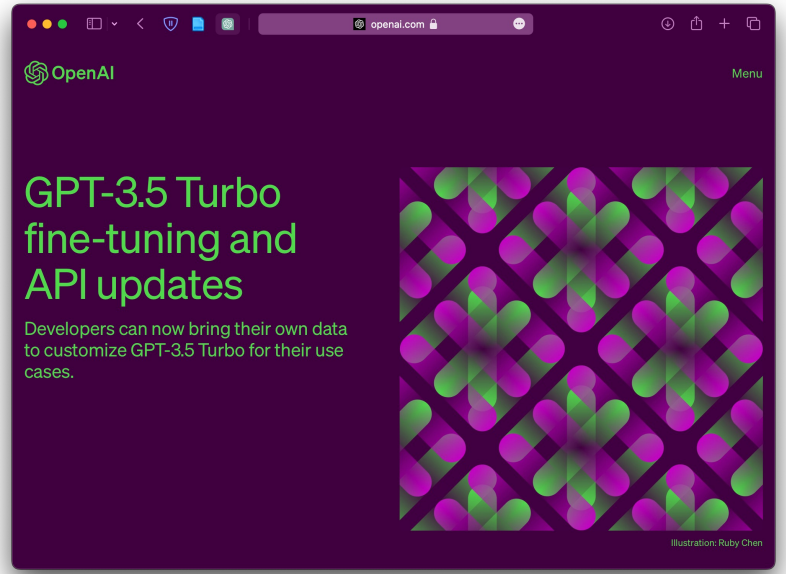
Implemented by Collaborator



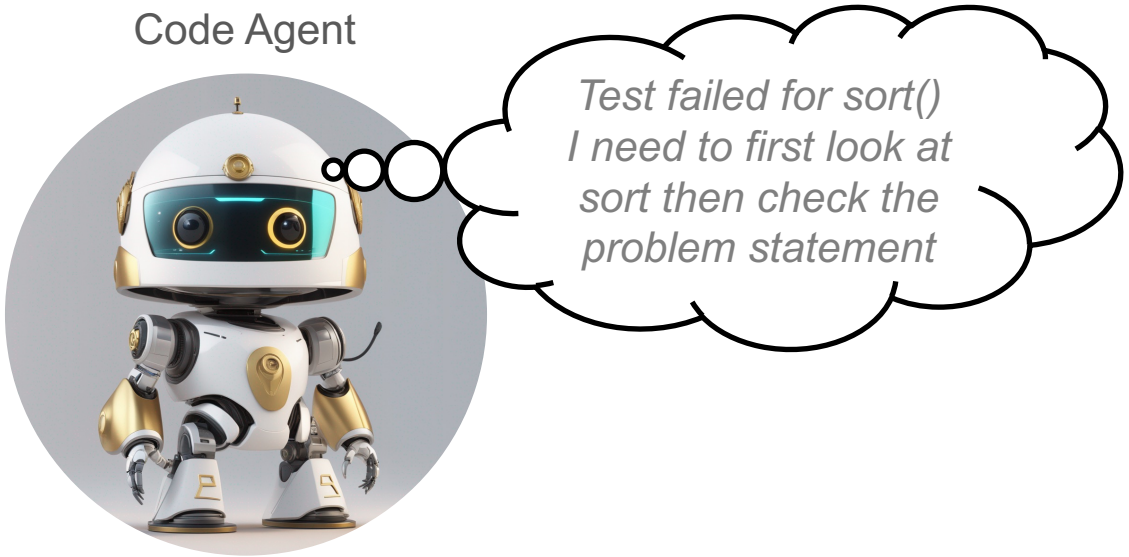
Status


Research & Development Context
Athena
Define Experiment
Configure Feedback Generator
Compare Feedback Suggestions
«include»
Researcher
Conduct Experiment
«include»
Rate Feedback Suggestions
«include»
Import/Export Configurations and Data
Review Automatic Ratings

Future Work – Fine-Tuning

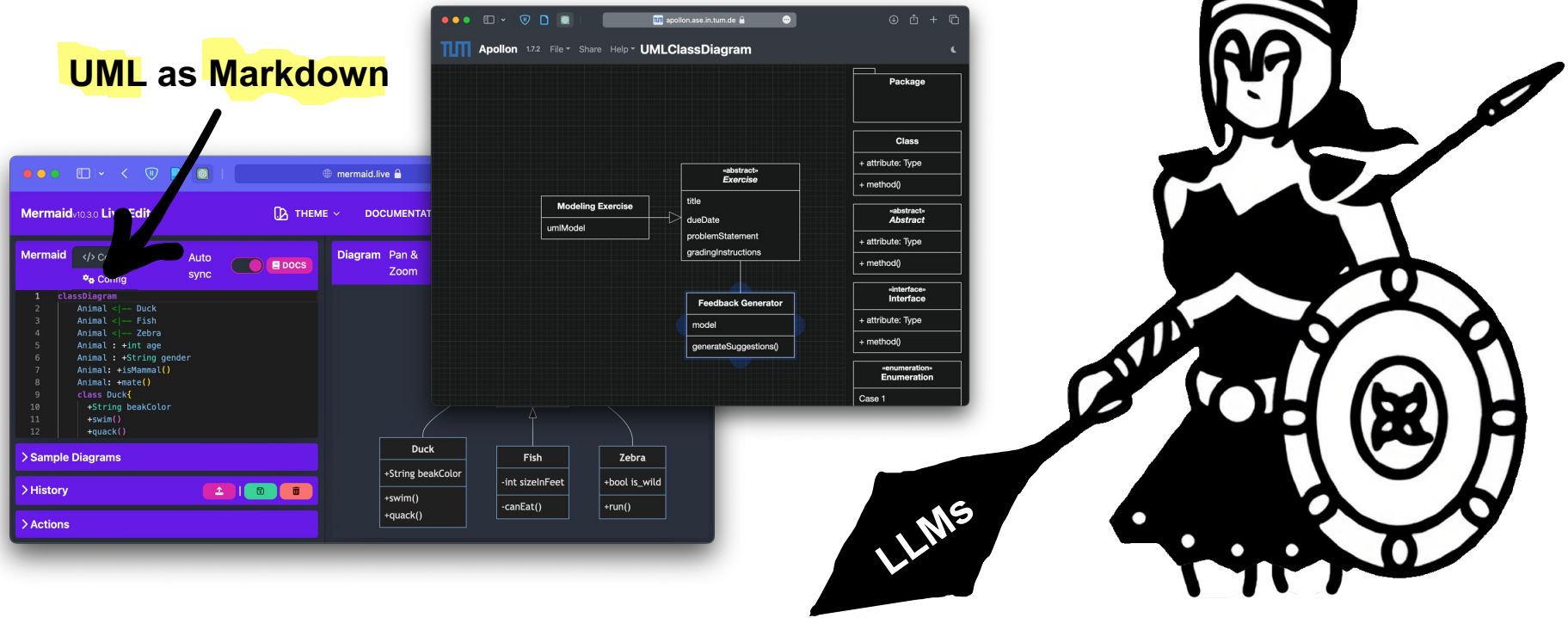


Future Work – Agentic Approach for Programming Exercises



Emulate the tutor's actions!

Future Work – Modeling Exercises

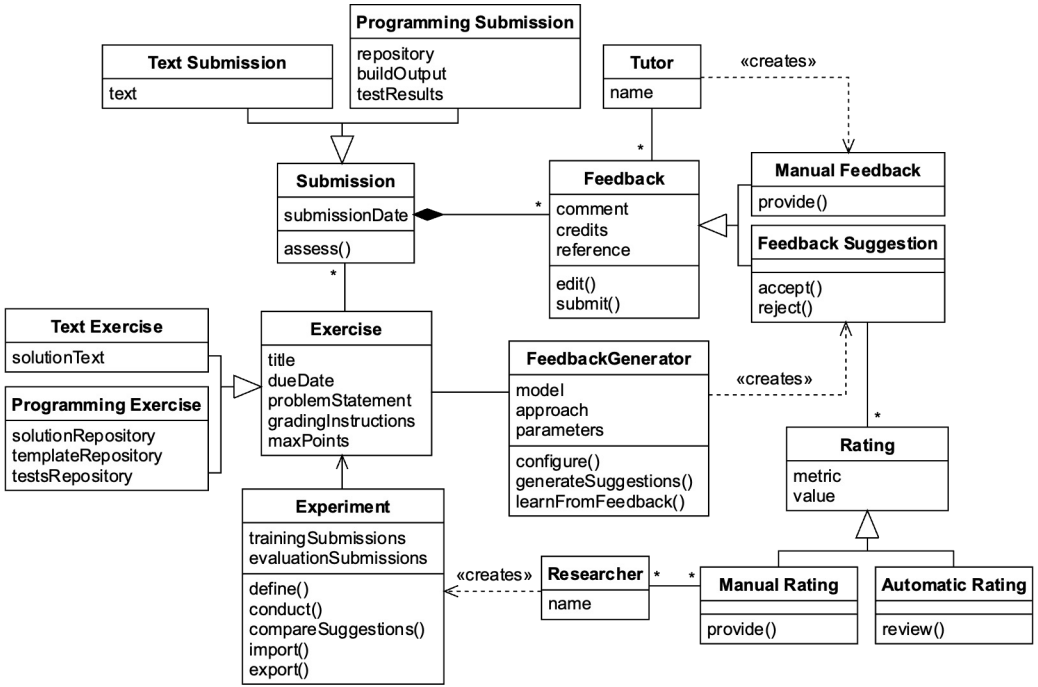


Thanks!

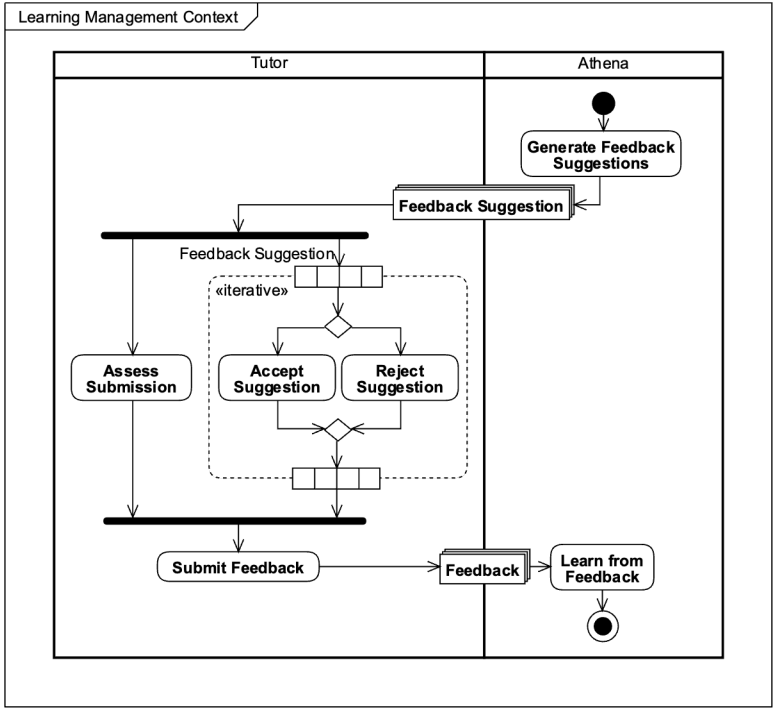
Further reading:

- My thesis
- Paul Schwind's thesis

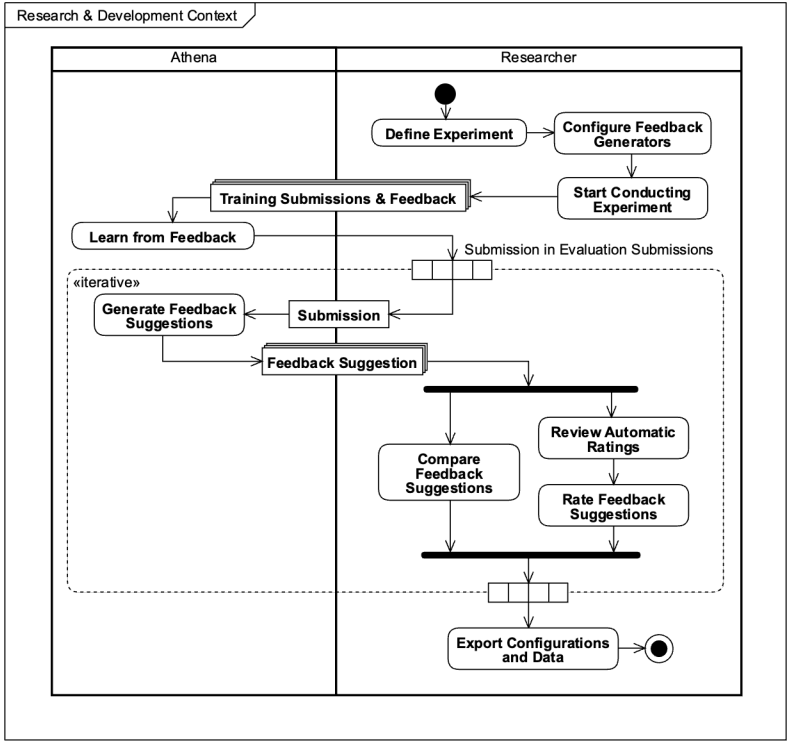
Backup Slides – Analysis Object Model



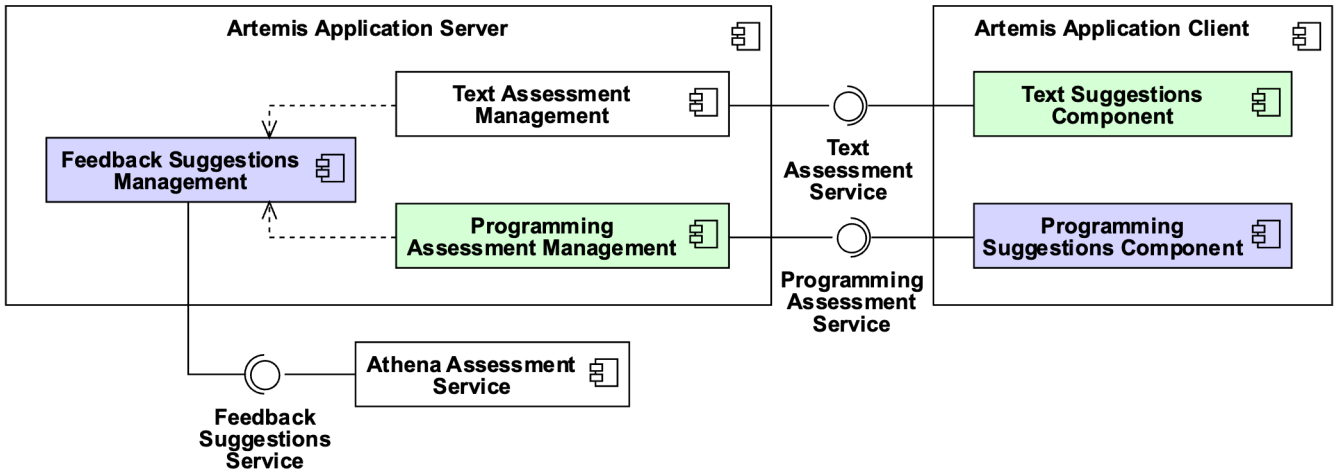
Backup Slides – Dynamic Model



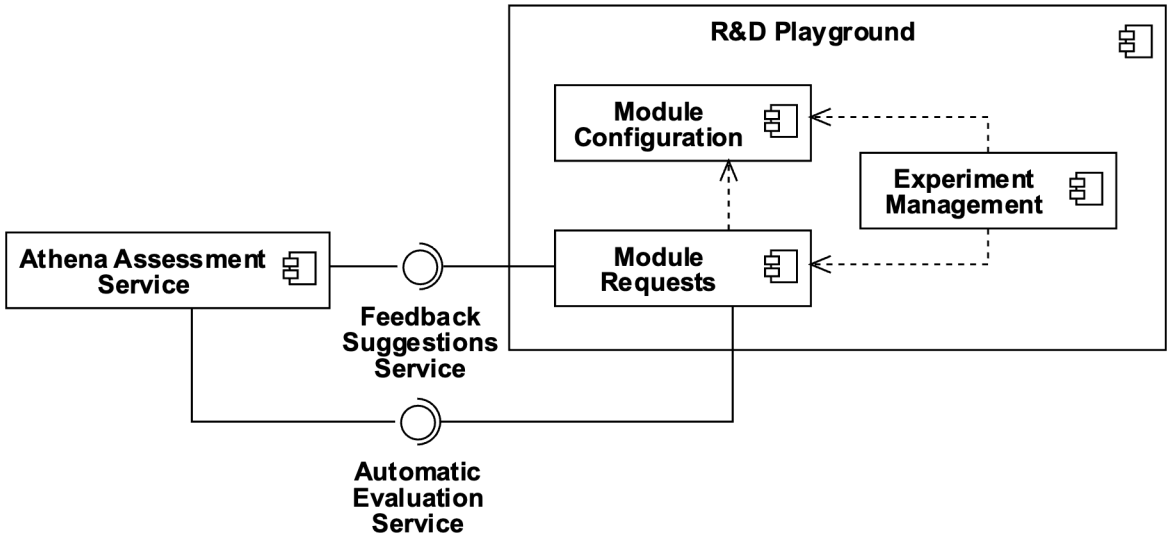
Backup Slides – Dynamic Model



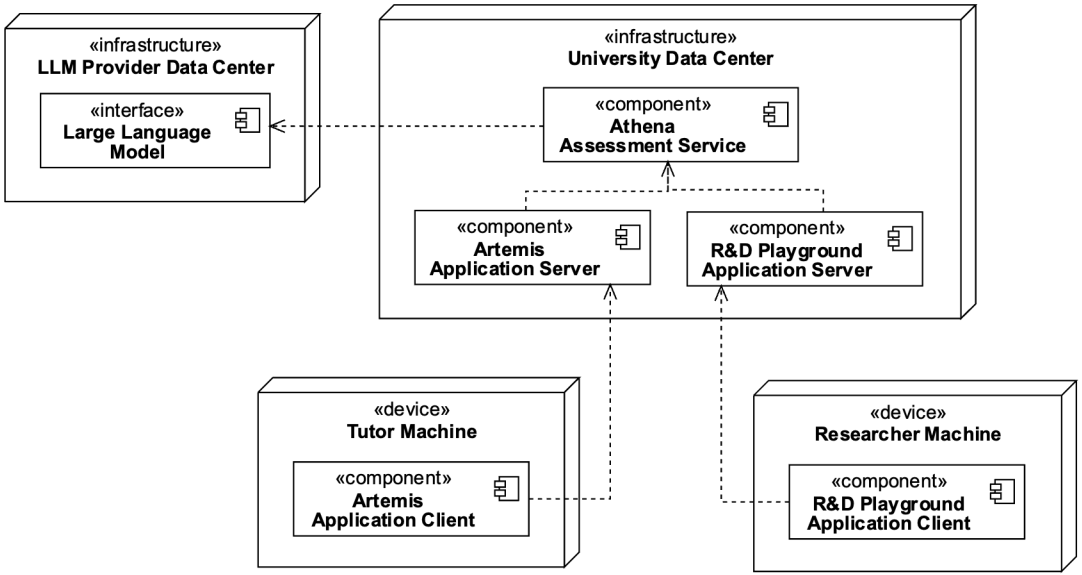
Backup Slides – Artemis



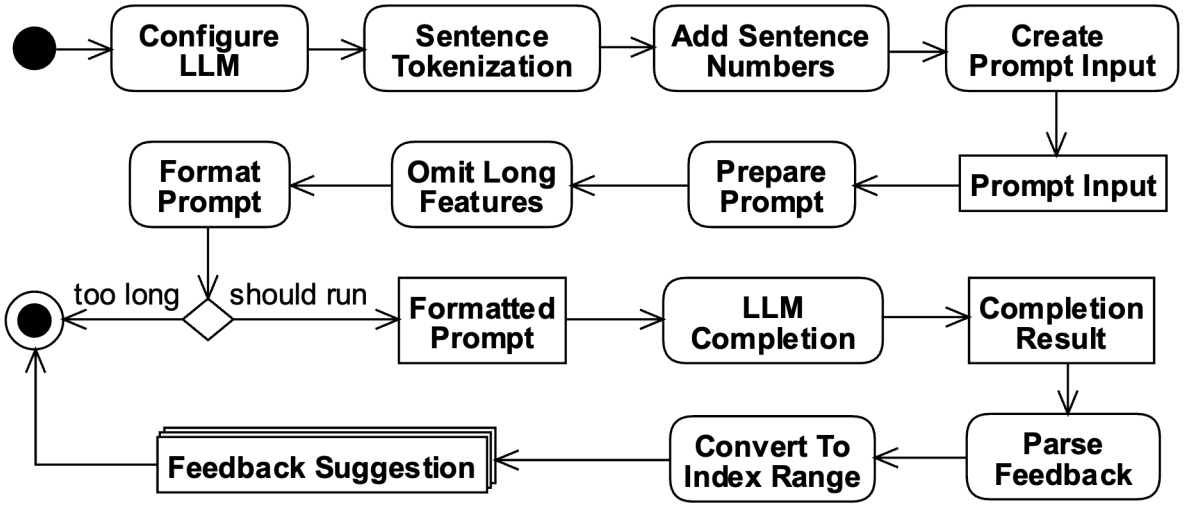
Backup Slides – Playground



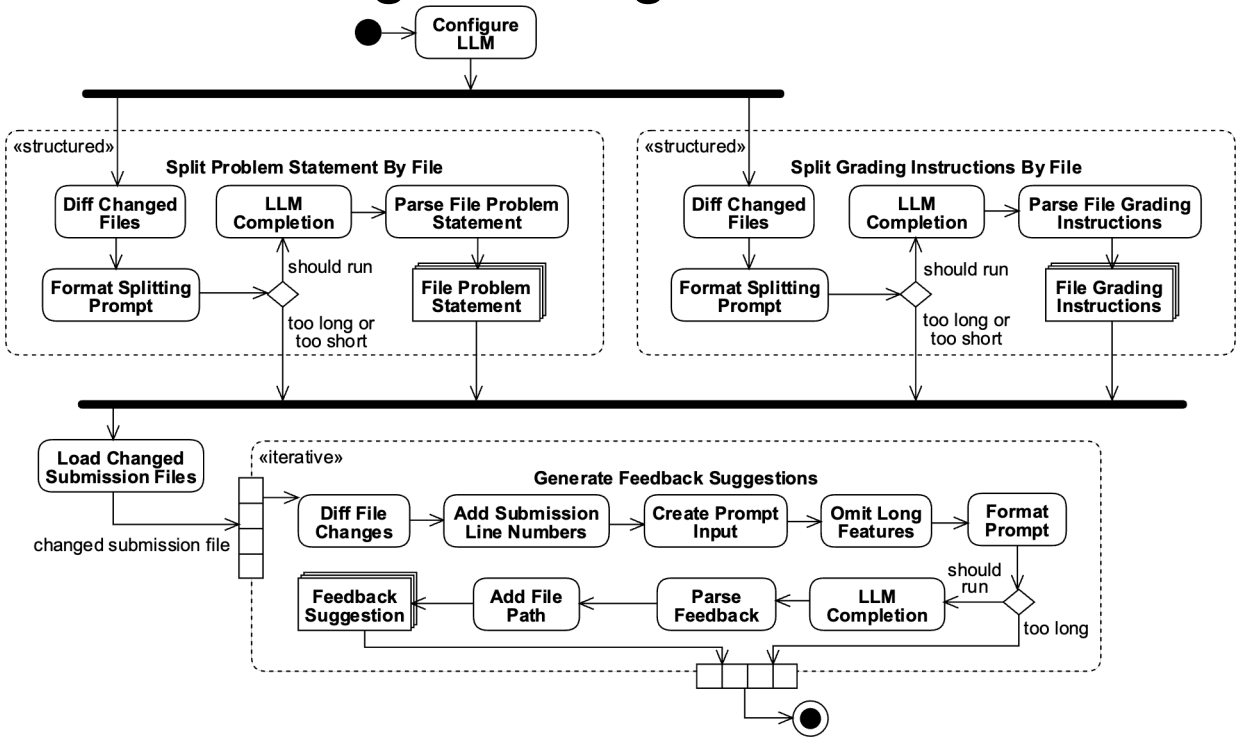
Backup Slides – Hardware-Software Mapping



Backup Slides – Text Exercises



Backup Slides – Programming Exercises



Backup Slides

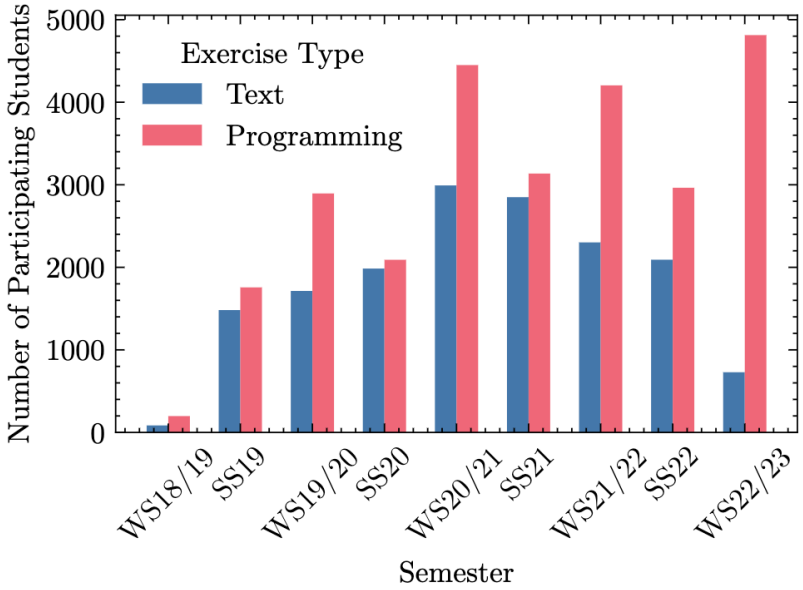


Figure A.2: Student Participation Trends in Text and Programming Exercises Across Semesters.

Backup Slides

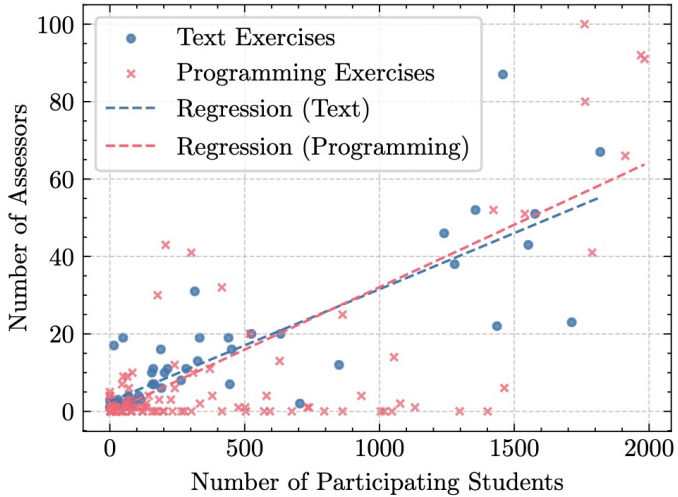


Figure A.3: Student-to-Assessor Ratios in Artemis Courses. Data shows the correlation between the number of participating students and assessors, broken down by text and programming exercises per course. The regression lines have coefficients of 0.029 for text and 0.032 for programming exercises and exclude courses solely reliant on automated assessments, *i.e.* zero assessors.

Backup Slides – Text Exercises

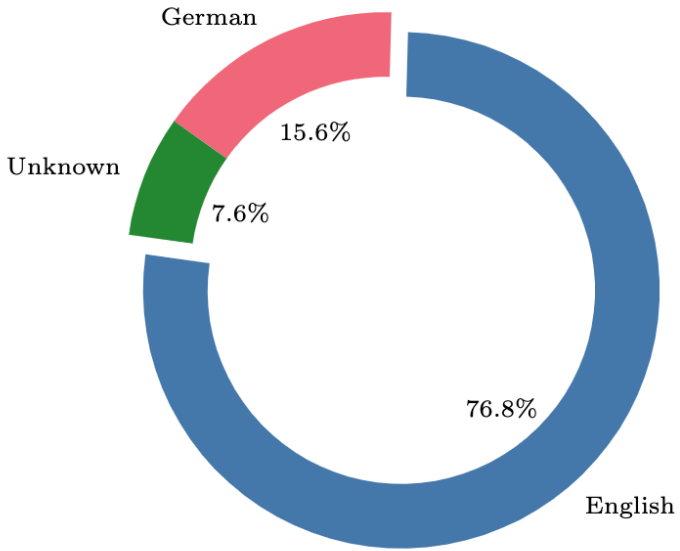


Figure A.4: Distribution of Languages in Text Exercises.

Backup Slides – Text Exercises

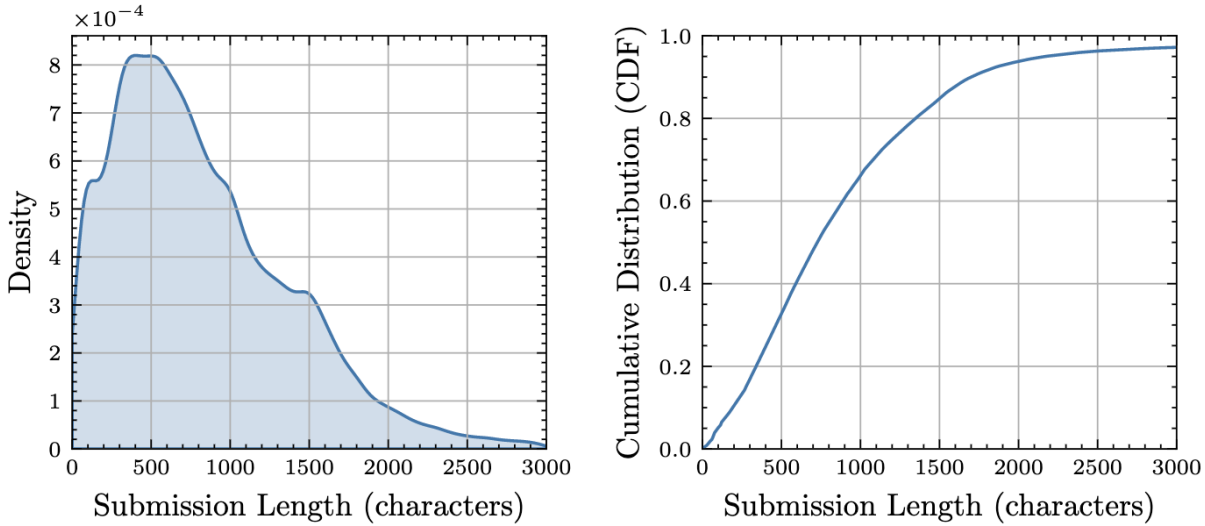


Figure A.5: Distribution of Text Submissions Length in Characters.

Backup Slides – Text Exercises

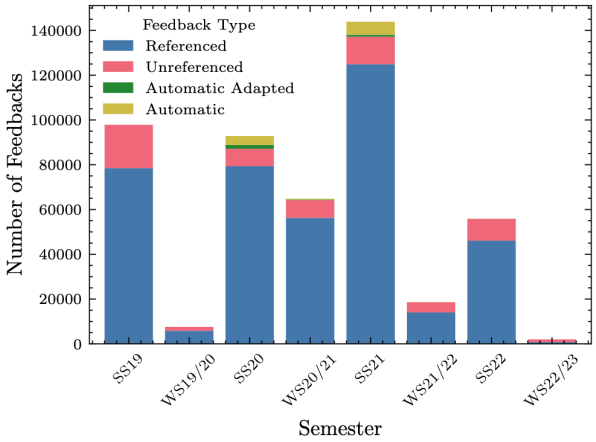


Figure A.6: Distribution of Feedback Types for Text Exercises Across Semesters. "Automatic" and "Automatic Adapted" refer to feedback provided by CoFee [BB19]. "Referenced" feedback is linked to a specific text passage, whereas "Unreferenced" feedback is not.

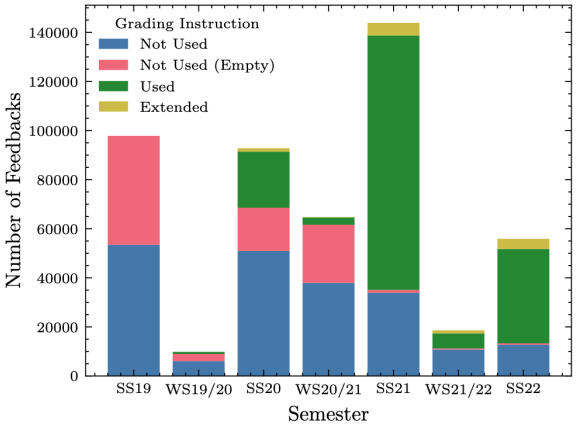
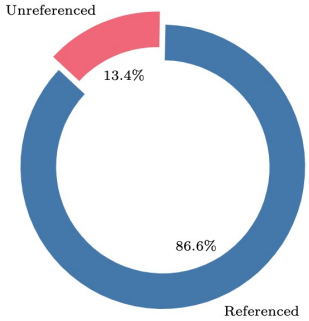
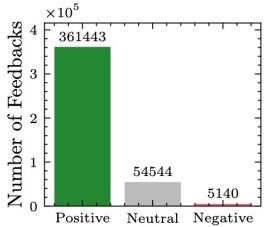


Figure A.7: Usage of Structured Grading Instructions in Text Exercises. "Not Used (Empty)" refers to feedback that has no content and liked grading instruction, probably due to deletion of the grading instruction. "Extended" denotes feedback that is linked to a grading instruction, but also additional comments.

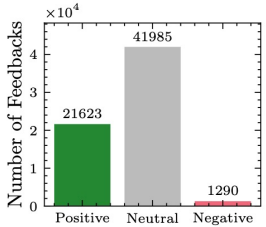
Backup Slides – Programming Exercises



(a) Distribution of Referenced and Unreferenced Feedback.



(b) Types of Referenced Feedback.



(c) Types of Unreferenced Feedback.

Figure A.8: Analysis of Referenced and Unreferenced Feedback in Text Exercises.

Backup Slides – Programming Exercises

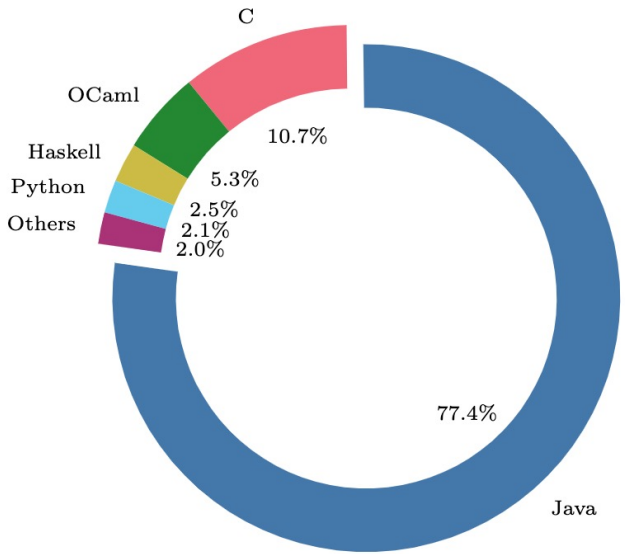
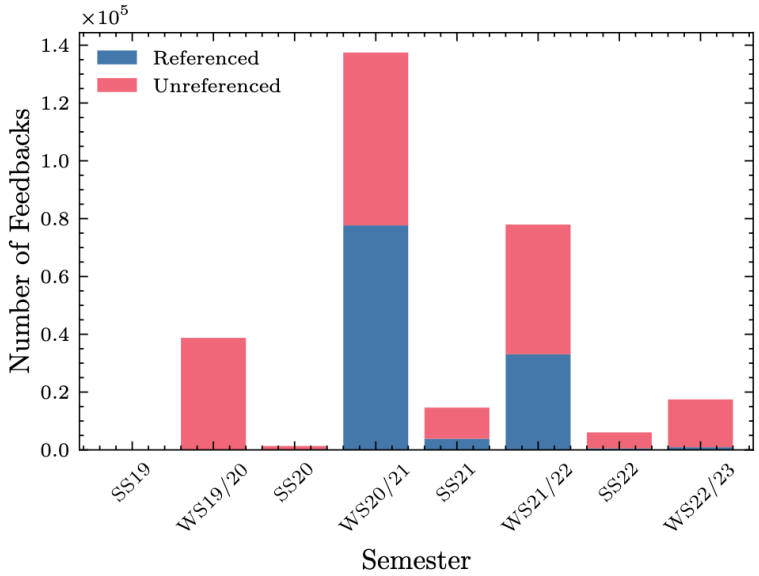
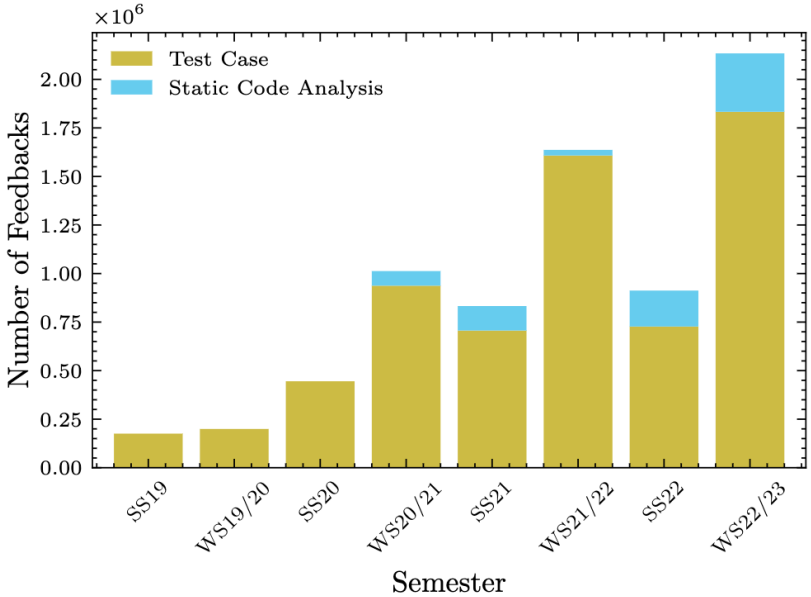


Figure A.9: Distribution of Programming Languages in Programming Exercises.

Backup Slides – Programming Exercises



(a) Manual Feedback Types.



(b) Automatic Feedback Types.

Figure A.10: Distribution of Feedback Types for Programming Exercises Across Semesters.

Backup Slides – Programming Exercises

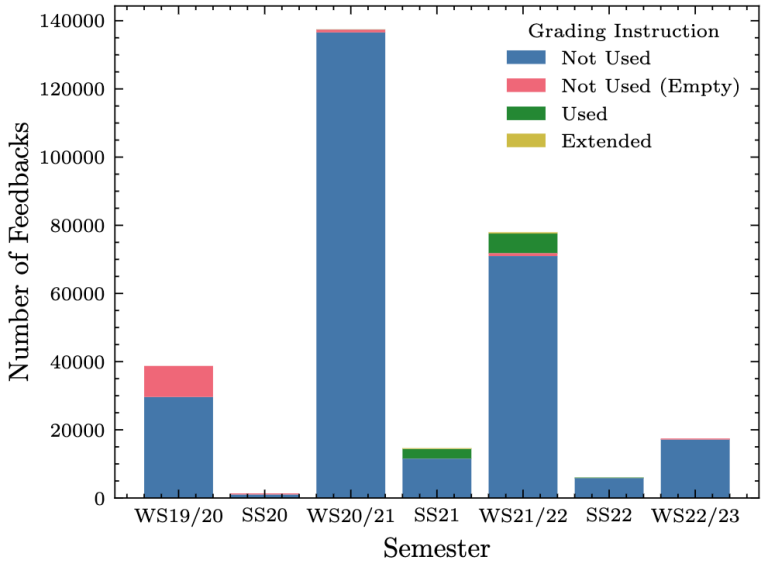
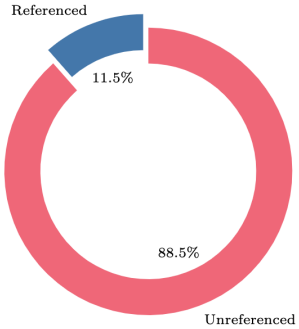
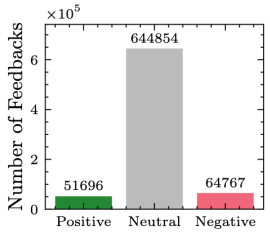


Figure A.11: Usage of Structured Grading Instructions in Programming Exercises for Manual Feedback.

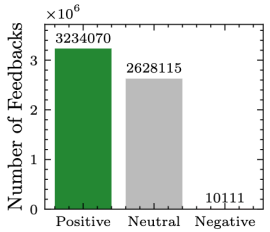
Backup Slides – Programming Exercises



(a) Distribution of Referenced and Unreferenced Feedback.



(b) Types of Referenced Feedback.



(c) Types of Unreferenced Feedback.

Figure A.12: Analysis of Referenced and Unreferenced Feedback in Programming Exercises.