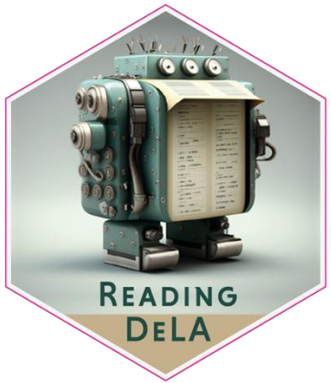
Bachelor's and Master's Thesis Topics in Educational Technologies

Provided by

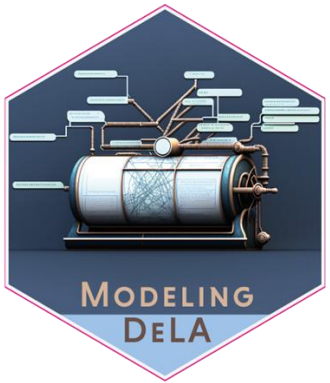
EduTec Educational Technologies

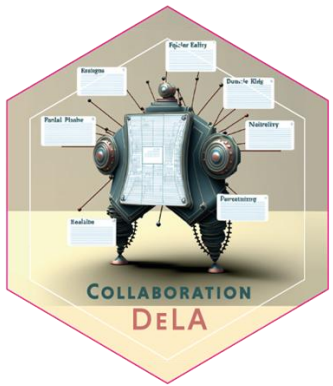
Category 1: Highly-Informative Learning Analytics



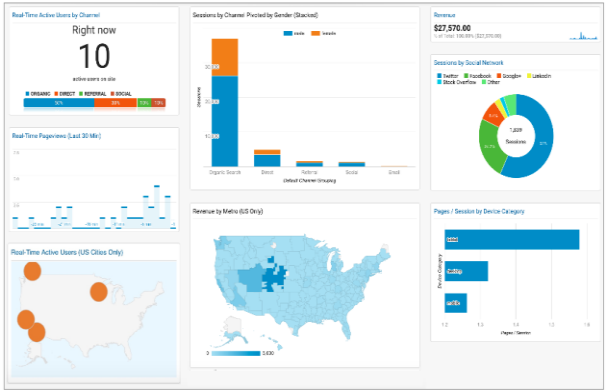








[1.1] [DEV] Learning Analytics Dashboard



Suitable for:

Bachelors & Masters

First examiner (Betreuer):

Prof. Dr. Hendrik Drachsler

Mentor:

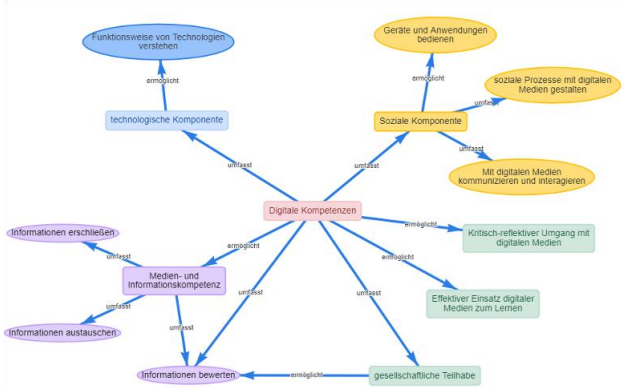
Atezaz Ahmad
ahmad@studiumdigitale.uni-frankfurt.de

Background. Learning Analytics Dashboards (LADs) provide users with insights about their study progress and an overview of the system through visualizations. LADs consist of metrics and indicators. Metrics are the unit of measurement (e.g., number of views/courses, total keywords/resources, keywords frequency, etc.). We refer to this unit of measurement as metrics. Metrics are used to create indicators; an indicator results from analyzing one or multiple metrics and gives a more comprehensive picture of a particular abstract, e.g., course analytics, self-reflection, engagement, etc.

Task. The task is to develop a learning analytics indicators dashboard (e.g., link) that provides an overview of the platform. An extensive JSON dataset containing over a million courses will be provided. The dashboard should be able to give a complete overview of the system. The visualization should include course analytics (numbers), world cloud (keywords and course description), and course distribution by location (country and cities), for example link.

Skills. JavaScript, HTML, and visualization libraries like Highcharts, Echarts, D3, or C3.js can be used. The mentioned libraries and languages are not mandatory to be used. You can use any other programming language or technology.

[1.2] Learning Analytics for MindMap-Tools



Suitable for:

Masters

First examiner (Betreuer):

Prof. Dr. Hendrik Drachsler

Mentor:

Tornike Giorgashvili

giorgashvili@studiumdigitale.uni-frankfurt.de

Background. The integration of Learning Analytics into Learning Management Systems, especially Moodle, offers significant potential for improving learning success through formative feedback. Despite the increasing use of MindMap tools as didactic resources in Moodle, efficient strategies for analyzing the generated process data are lacking. This gap hinders the provision of precise, implementation- and content-oriented feedback, which is essential for individual learner support.

Task. The aim is to address this challenge by developing specific data identification strategies and thus contribute to optimizing the use of Learning Analytics in educational technology.

Skills. Basic knowledge in the field of e-learning and a fundamental familiarity with Learning Management Systems, especially Moodle, are required. Also necessary are basic skills in data analysis and processing, preferably with experience in using database management systems such as MongoDB. Knowledge in the development and application of Learning Analytics or similar data-driven analytical methods in the education sector is advantageous but not mandatory.

[1.3] [DEV] Ask the paper: AI-based Extraction System for Understanding Research

Background: In many research scenarios, users want to ask a specific questions about scientific papers, e.g., “Which methods are used for task X?” or “What materials are related to this method?” A QA system can automatically answer these questions by extracting entities and relations from papers and presenting structured, understandable answers.
Tasks: 1. ***Entity & Relation Extraction***, identify key entities and their relationships from scientific papers. 2. ***Information Retrieval*****,** fetch relevant entities and relations based on user questions. 3. ***Answer Generation***, use LLMs to convert structured data into clear, natural language answers, optionally visualizing relationships.
Skills: Familiarity with models like BERT, GPT, zero-shot, few-shot or fine-tuning approaches for scientific text. Software Development, such as Backend (Python, databases, APIs), frontend (web frameworks, visualization), and system integration.

Suitable for: Bachelors & Masters

First examiner (Betreuer):

Prof. Dr. Hendrik Drachsler

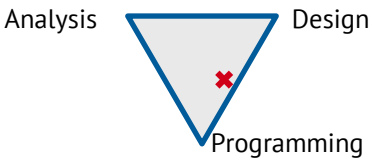
Supervisor:

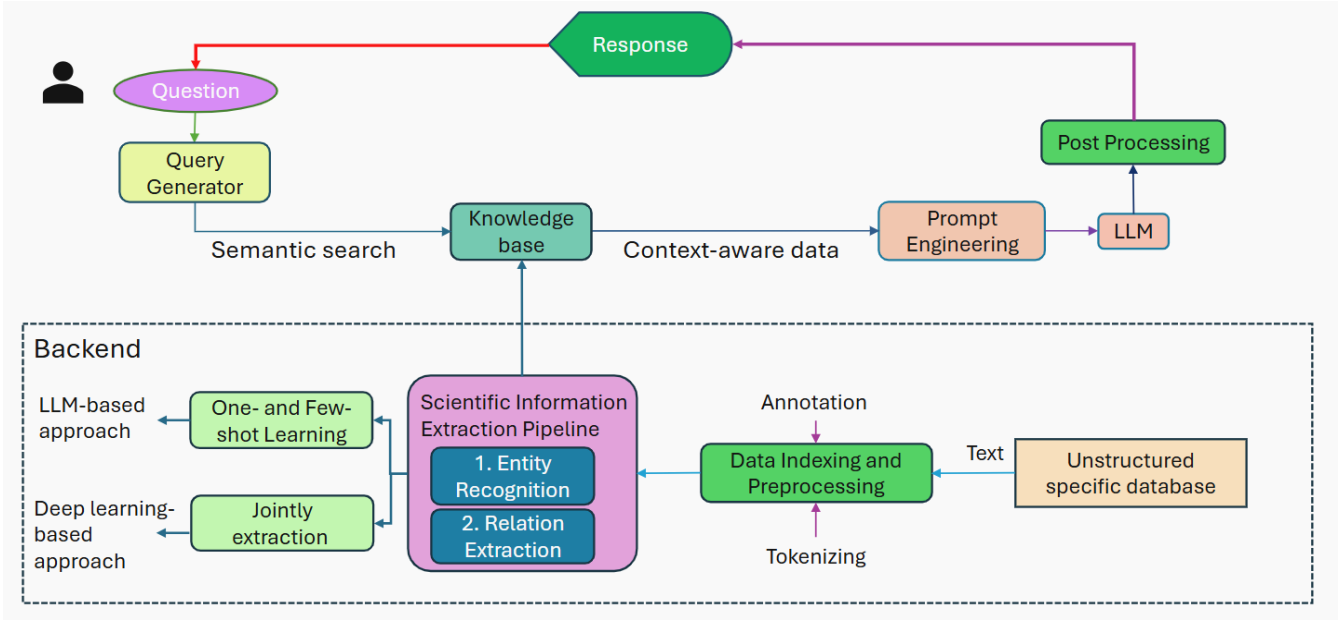
Atezaz Ahmad / Amir Rajabi

ahmad@sd.uni-frankfurt.de

a.rajabi@dipf.de

Evaluation: Design-driven evaluation





[1.4] Agentic Learning Analytics for Programming Process Feedback

Background: Learning programming is a process of repeated experimentation, error diagnosis, code revision, testing, and reflection. Conventional assessment usually considers only the final submission and therefore provides limited information about how students approached a programming task. Development environments such as Jupyter, GitHub, GitLab, and browser-based IDEs can generate fine-grained learning-process data, including code revisions, execution events, errors, test results, commits, and help-seeking activities. However, these data must be transformed into pedagogically meaningful indicators before they can support students and educators. The proposed thesis investigates how learning analytics and an LLM-based agent can provide transparent, contextualized, and actionable feedback during programming activities.

Suitable for: BA & MA

First examiner (Betreuer):

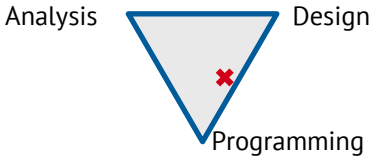
Prof. Dr. Hendrik Drachsler

Supervisors: Atezaz Ahmad, Prof. Basit Shahzad
ahmad@sd.uni-frankfurt.de

Task. Develop a learning-analytics platform that collects and analyses programming-process data from a selected environment, such as JupyterLab, GitHub, GitLab, or a web-based coding platform.

Expected Result. A functional learning-analytics web application consisting of: A data-collection component integrated with a programming environment. A reproducible analytics pipeline for computing programming-process indicators. Student and educator dashboards. An agentic feedback component grounded in event data, source code, test results, and approved learning materials. An evaluation of indicator validity, intervention accuracy, usability, feedback quality, and preliminary learning effects.

Skills. Python or JavaScript/TypeScript, data analysis, learning analytics, LLM APIs, software development APIs, dashboard development, databases, and basic experimental design.





[1.5] Data-Enriched Learning Activities in Jupyter Notebooks

Background: Jupyter Notebooks can capture detailed learning-process data, including code edits, cell executions, errors, help requests, and interactions with an AI tutor. The JELAI research demonstrates an architecture for collecting these data and enriching LLM prompts with the learner's current code, errors, objectives, and interaction history. However, its current contribution is primarily a technical framework; further work is required to transform the collected data into configurable, pedagogically grounded learning activities and proactive interventions.

Suitable for: BA & MA

First examiner (Betreuer):

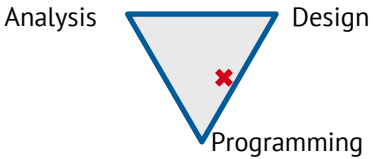
Prof. Dr. Hendrik Drachsler

Supervisors: Atezaz Ahmad,
Prof. Basit Shahzad,
Prof. Hendrik Drachsler

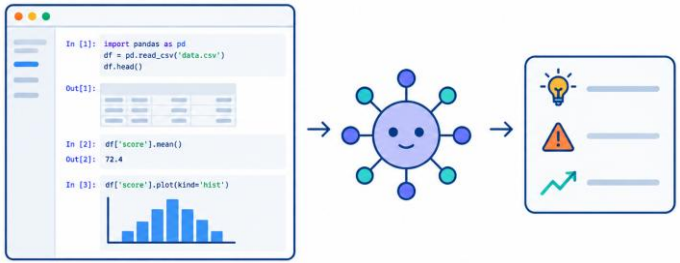
ahmad@sd.uni-frankfurt.de

Task. Design and develop a JupyterLab-based Educational Technology platform that enables educators to create **Data-Enriched Learning Activities**. Each activity should connect learning objectives and notebook tasks with telemetry events, learning indicators, intervention rules, and AI-supported feedback.

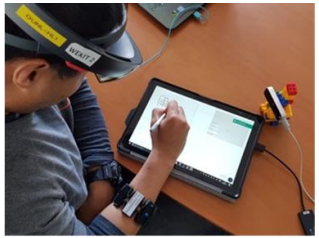
Expected Result. A functional JupyterLab extension or integrated Jupyter learning environment. An educator-facing authoring tool for configuring Data-Enriched Learning Activities. A telemetry and learning-analytics pipeline. An agentic orchestration component for monitoring, intervention selection, tutoring, and feedback verification. A student interface providing context-aware scaffolding and reflective prompts. A dashboard showing learning indicators, interventions, and activity progress. Several reusable EduTec Science learning activities. An evaluation of technical reliability, indicator validity, usability, pedagogical usefulness, and student perceptions.



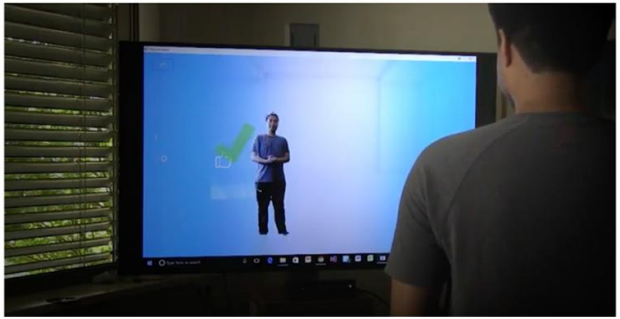
Skills. Python, TypeScript or JavaScript, JupyterLab extension development, learning analytics, LLMs and Agentic AI, prompt engineering, educational data modelling, dashboard development, Docker, databases, privacy-aware system design, usability evaluation, and experimental design.



Category 2: Multimodal AI for automated feedback

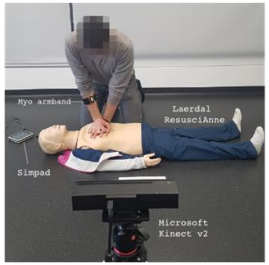












[2.1] Training oral presentation skills with motion capture



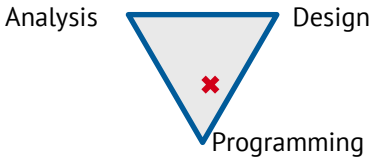
Suitable for: Bachelors & Masters

First examiner (Betreuer):
Prof. Dr. Hendrik Drachsler

Mentor: Jan Schneider

j.schneider@dipf.de

Evaluation: Design-driven evaluation



Background. There are different techniques that help people to read, understand, and remember the content of a text. One of these techniques consists of writing down the summary of a text's first paragraph in one sentence. Write in one sentence the summary of the second paragraph. For the next paragraphs, one needs to write two sentences, 1 summarising what happened previously, and one summarising the current paragraph. Current text readers are not designed for this practice.

Research Task. Develop a text reader designed for this practice. Explore to what extent LLMs can be used to assess if the user provided good summaries or not and help users to correct their summaries.

Expected result. An application that users can use to conduct an active reading approach that helps them to understand and remember the content of texts.

[2.2] Presentable: AI feedback Presentation training software

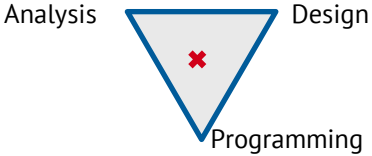


Suitable for: Bachelors & Masters

First examiner (Betreuer):
Prof. Dr. Hendrik Drachsler

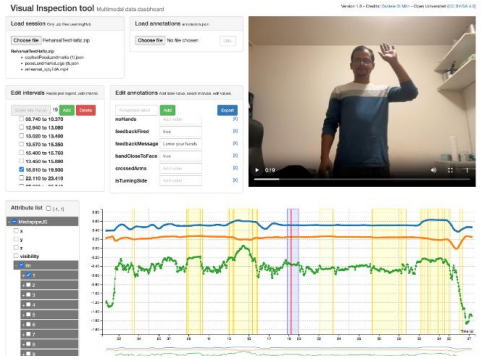
Mentor: Prof Dr Daniele Di Mitri
daniele.dimitri@german-uds.de

Evaluation: Design-driven + small Evaluation (MSc)

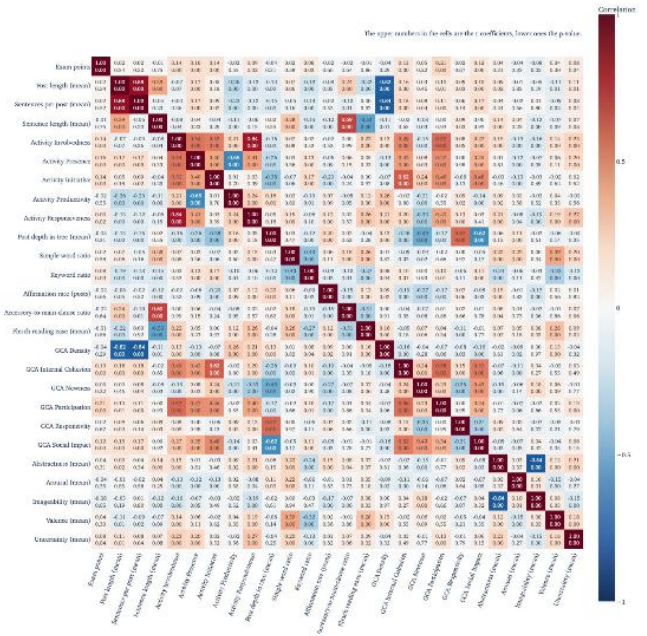


Background. Presentable (www.presentable.info) is a presentation training software developed at the DIPF in the context of the HyTea project. Is an advanced research software (collects data) designed to enhance users presentation skills through personalised AI-driven feedback on voice and body language and guidance on message composition. It provides educators with dashboards to monitor and support student progress, making it essential for improving presentation skills.
Research Task. Currently there several Master thesis ongoing in the context of Presentable which cover topics such as multimodal data analysis, voice analysis, game-based learning, speech transcription, MLOps, computer vision, mobile technologies etc.
Expected result. Dependent on your background/coding strength, we could find a relevant topic of research for you. You are expected to formulate a research question and develop a component which can be fully integrated in Presentable - should be compatible with the tech stack Next.js, django, MySQL, mediapipe. The thesis must be written in English.





Category 3: Natural Language Processing





[3.1] Rewriting Grading Rubrics for Better Clarity Using LLMs

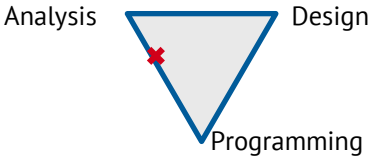


Suitable for: Bachelors & Masters

First examiner (Betreuer):
Prof. Dr. Hendrik Drachsler

Mentor: Sebastian Gombert,
s.gombert@dipf.de

Evaluation: Data Science-driven evaluation



Background. Rubrics are structured documents which define clear criteria used to assess all kinds of learner-produced content such as test answers, essays, or student-drawn diagrams. Recently, we started to experiment with AI and how well it can apply unseen rubrics to unseen tasks. In the ALICE project, we collected a dataset of student answers and corresponding rubrics. However, unfortunately, these rubrics are not always formulated in a clear way which we suspect hinders models to successfully apply them in all cases.

Research Task. The task at hand is to develop a method that can be used to identify and rephrase unclear or uncertain parts within grading rubrics so that they are more clear to follow. We suspect that this can be solved best with a pipeline implemented with the help of multiple LLM prompts and existing metrics for measuring text clarity, but are open to other approaches if discussed with us. We provide you with a German-language dataset that contains rubrics and student answers. Ideally, it must be shown that rewriting the prompts for more clarity improves downstream results when these are used to train short answer scoring models. This needs to be tested.

Expected Result. An answer to the question whether rewriting grading rubrics for better clarity using LLMs benefits downstream performance of scoring models when used with these rubrics.

[3.2] Implementing an LLM-based Annotation Recommender

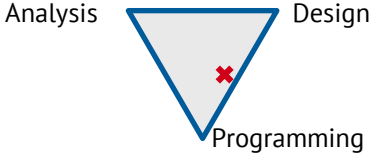


Suitable for: Bachelors & Masters

First examiner (Betreuer):
Prof. Dr. Hendrik Drachsler

Mentor: Sebastian Gombert,
s.gombert@dipf.de

Evaluation: Design-driven evaluation (BA), Design-driven + small Evaluation (MA)



Background. In our working group, we use the tool INCEpTION to annotate all kinds of textual data sets. This tool offers the option to interface annotation recommender systems which learn from users' input and then provide annotation recommendations to users. We want to explore to which degree prompting LLMs with few-shot prompts can provide us with useful annotation recommendations.

Research Task. For this purpose, we need a system which provides both a simple, but comprehensive user interface that can be used to design prompts which can be used for recommendations. This system should interface with INCEpTION over the corresponding REST API and with an LLM, e.g., gpt-4o, and use the latter to provide recommendations within the prior. In this context, it is important that there is a UI which helps users to flexibly design prompts, and that there are placeholders for few-shot examples that are filled by the system with already annotated passages retrieved from INCEpTION when issued to an LLM. While the exact tech stack is up to the candidate, the system should use MariaDB or PostgreSQL as database and run in a Docker container.

Expected Result. A tool which can be used to design flexible annotation recommenders using LLM prompts and interfaces with INCEpTION. For an MA thesis, this tool should also be evaluated in a small-scale user study with our HiWis who will then use the tool on an almost daily basis, if successful.

[3.3] Recommender Systems for Open Data

Background. GovData.de serves as Germany's central portal for open government data, providing access to datasets from a wide range of public authorities. The primary method for users to discover and access these valuable datasets is through the portal's search function. While functional, the current search system on GovData.de offers significant potential for improvement. For instance, there is currently no system in place to recommend similar or related datasets to users, limiting discoverability.

Research Task. The primary objective of this thesis is to investigate, conceptualize, and potentially prototype improved methods for data discovery on GovData.de.

- Researching state-of-the-art recommendation techniques applicable to dataset portals.
- Identifying relevant criteria for generating meaningful dataset recommendations
- Designing and evaluating different recommendation approaches.

Expected Result.

1. **State-of-the-Art Review:** A comprehensive analysis of current research and best practices in Recommender systems for data portals.
2. **Feasibility Analysis:** An investigation into potential recommendation criteria and an assessment of the technical possibilities and constraints
3. **Conceptual Design:** Development of a clear concept for the proposed improvements.
4. **Proof of Concept (PoC):** Implementation and demonstration of a working prototype showcasing the core ideas and functionalities of the proposed solution.

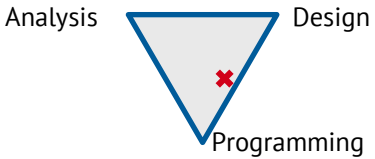
This thesis project will be conducted in close collaboration with the **GovData Product Management team and the technical service provider **SEITENBAU GmbH**, offering valuable insights into real-world operations and development*

Suitable for: Bachelors & Masters

First examiner (Betreuer):
Prof. Dr. Hendrik Drachsler

Mentor: Dr. Daniel Biedermann,
daniel.biedermann@fitko.de*

Evaluation: Design-driven evaluation (BA), Design-driven + small Evaluation (MA)

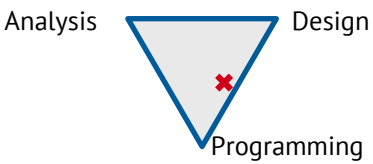


[3.4] Synthesizing student answers for short answer scoring

Prerequisites

1. Interests and experience in NLP
2. Familiarity with transformer and the attention architectures
3. Experience with NLP libraries such as transformers and pytorch.

Suitable for: Masters.
First examiner (Betreuer):
Prof. Dr. Hendrik Drachsler
Mentor: Zhifan Sun,
z.sun@dipf.de
Evaluation: Data Science-Driven evaluation



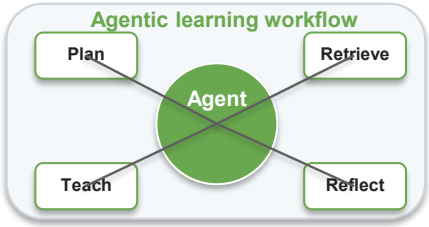
Background One of the biggest challenges in automatic Short Answer Scoring (ASAS), and possibly all the ML tasks, is that collecting human labeled data is expensive. Hence, using LLMs to generate synthesized data have always been appealing in the NLP/ML practices and subjected to intensive research. However, there is no previous work investigating the feasibility of LLM-synthesized data for ASAS.
ASAS poses unique challenges for data synthesis: 1) LLMs are trained to produce the "perfect" response, yet realistic student response can be noisy.
2) Student answers show high variability depending on on the question type, context and the level of the answer.

Research Task:

- Conduct literature review on LLM-data synthesis and student simulation via LLMs.
- Conduct linguistic/semantic analysis of the real student answers and compare with the llm-synthesized ones
- Based on the semantic analysis of student answers, design various LLM-prompts to simulate student responses
- Train an LLM to simulate student responses via SFT.
- **(BONUS) Train student response simulators via RL.**
- Train ASAS models with the synthesized data and compare with those trained by the actual human data

Expected outcome:
Either successfully implementing a student response simulator and/or discuss the limitation of training data synthesis by LLM for ASAS.

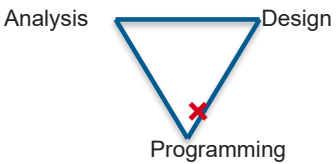
[3.5] Agentic Learning Assistants for Computer Science Education



Suitable for: Bachelors & Masters
First examiner (Betreuer):
Prof. Dr. Hendrik Drachsler

Supervisors: Atezaz Ahmad, Prof. Basit Shahzad
ahmad@sd.uni-frankfurt.de

Evaluation: Data Science-Driven evaluation



Background. Agentic AI can plan learning activities, retrieve resources, generate practice, assess responses, and adapt feedback. In specialized computer science domains, its value depends on correctness, pedagogical quality, traceability, efficiency, and appropriate safeguards.

Research Task. Design and implement an agentic learning assistant for a bounded topic in Software Engineering, Machine Learning, or a related area.

- Define the workflow, tools, knowledge sources, guardrails, and logging strategy.
- Create an expert-verified task set and rubric; compare against a non-agentic baseline.
- Optionally compare several LLMs or agent architectures under the same workflow.

Expected Result. A functional application and reproducible evaluation covering answer correctness, feedback quality, response time, robustness, and resource use; an MA project may include a small user study.

Skills. Python, LLM APIs/local models, RAG or agent frameworks, software engineering, educational design, and evaluation.

[3.5] Agentic Sprint-Planning Simulator and Learning Coach



Suitable for: Bachelors & Masters
First examiner (Betreuer):
Prof. Dr. Hendrik Drachsler

Supervisors: Atezaz Ahmad,
Prof. Basit Shahzad
ahmad@sd.uni-frankfurt.de

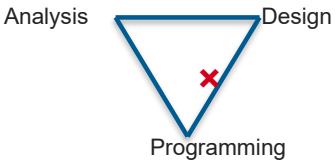
Background. Agile project courses often teach estimation and planning through simplified exercises that do not expose students to cascading dependencies. The original brief predicts which connected tasks may delay a sprint using dependency-network features. The revised topic creates a simulation-based learning environment in which students make planning decisions and receive agentic feedback on the consequences.

Research Task. Develop a sprint-planning simulator that imports or generates a backlog, models dependencies among issues, visualizes delay propagation, and allows students to assign priorities, estimates, developers, or mitigation actions. An agentic coach should analyze the learner's plan, test alternative scenarios, explain bottlenecks, and prompt reflection rather than simply revealing an optimal answer.

Expected Result. An interactive backlog simulator, dependency visualization, risk-analysis component, coaching agent, and learning analytics for comparing decisions across scenarios.

Skills. Web development, graph analysis, Python or JavaScript, project-management data, LLM agents, visualization, and educational evaluation.

Evaluation. Design-driven plus small-scale comparative evaluation.

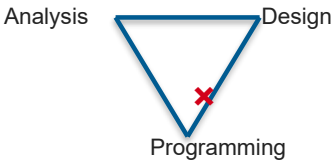


[3.5] SecureVibe: A VS Code Learning Assistant for Security-Aware and Energy-Efficient AI-Assisted Programming



Suitable for: Bachelors & Masters
First examiner (Betreuer):
Prof. Dr. Hendrik Drachsler

Supervisors: Atezaz Ahmad,
Prof. Basit Shahzad
ahmad@sd.uni-frankfurt.de



Background. Vibe coding allows users to generate software through natural-language instructions, but LLM-generated code may contain security vulnerabilities, functional errors, hard-coded credentials, sensitive-data exposure, insecure dependencies, and injection risks. Novice developers may also send confidential code or personal data to external LLMs without recognizing the privacy implications. A learning-oriented development tool is therefore needed to validate generated code and teach users secure, privacy-aware, and resource-efficient AI-assisted programming practices.

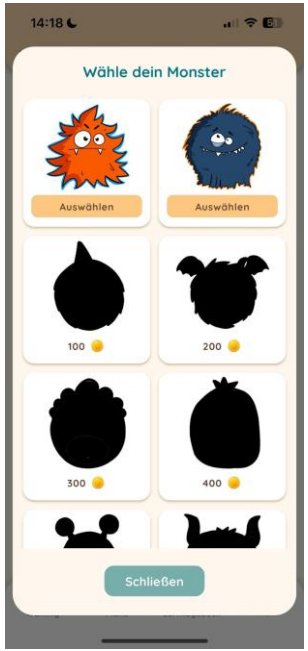
Research Task. Design and develop a VS Code extension that analyses LLM-generated code using static analysis, security rules, automated tests, and optional LLM-based review. The extension should detect vulnerabilities such as SQL injection, cross-site scripting, command injection, insecure authentication, exposed credentials, and potential privacy leaks. It should provide inline explanations, secure-code recommendations, and improved prompts containing relevant security requirements. The extension should also reduce unnecessary prompt context and track token usage, latency, estimated cost, and, where technically measurable, energy consumption.

Expected Result. A functional VS Code extension providing: Real-time security, privacy, and correctness diagnostics. Detection of vulnerable code, exposed secrets, and potentially sensitive data. Explanations and secure-code quick fixes for detected problems. A prompt coach that generates concise, security-aware prompts. Token, latency, cost, and optional energy-usage monitoring. A learning dashboard showing recurring mistakes, resolved vulnerabilities, and prompt-efficiency improvements. An evaluation using labelled vulnerable programs, baseline security scanners, and a user study with programming students.

Skills. TypeScript or JavaScript, VS Code Extension API, secure software development, static code analysis, LLM APIs or local models, prompt engineering, automated testing, data-flow analysis, dashboard development, usability evaluation, and basic experimental design

Category 4: Mobile Learning App for Children







JETZT BEI Google Play

Laden im App Store

[4.1] Mini-Games to Promote App Engagement



Suitable for: Masters

First examiner (Betreuer):
Prof. Dr. Hendrik Drachsler

Mentors:
Dr. Daniel Biedermann
Dr. Jasmin Breitwieser,
d.g.biedermann@gmail.com
j.breitwieser@dipf.de

Evaluation: Design-driven + small Evaluation

Background. In this interdisciplinary research project, we develop the mobile app "PROMPT – Die Lernplan-App" to promote self-regulated learning (SRL) in young students (5th-6th grade). The app supports children in setting learning goals, strategically planning learning times and monitoring and evaluating their progress in the context of vocabulary learning. The app includes a structured training, a learning diary, and targeted prompting. To be effective, the app must not only be pedagogically sound but also engaging for its young users. To this end, we draw on gamification principles to enhance motivation and sustain interest over time. The current version of the app includes basic gamification mechanics, such as collecting coins and unlocking avatars.

Task. The goal of this thesis is to develop an interactive mini-game that can be integrated into the app as a gamification element. The challenge is to develop a game that is child-friendly, motivating and functionally linked to the learning process of the app. Ideally, the game also provides us with data on children's cognitive abilities (e.g., working memory).

Expected Result. A mini-game programmed in Flutter and a concept of how this game should ideally be integrated into the app. This game should also be evaluated in a small-scale user study (e.g., focus groups) with children.

Skills. Programming skills are a must, as well as experience and interest in game development and gamification concepts, as well as an interest in application-oriented, education-related software development.