Making a Car Invisible with an Adversarial Patch

Project Description

Adversarial attacks on machine learning models have shown that it is possible to intentionally confuse object detection systems, causing them to miss or misclassify objects. Some well-known examples include adversarial patches that make a person appear invisible to a detector when worn on clothing[1], or large patches placed on the back of trucks to prevent them from being detected[2].

In this project, you will study and extend this type of attacks in the context of autonomous driving. First, you will generate an adversarial patch that, when placed on the back of a *target vehicle* (for example, like a painting on the car), makes that vehicle invisible to an object detection model. Next, you will explore a more advanced scenario: instead of placing the patch on the target vehicle itself, you will place it on the back of a *different* (non-target) vehicle. The goal is to investigate whether the presence of this patched non-target vehicle can still cause another target vehicle to become invisible to the classifier[3].

Evaluation in Autonomous Driving Simulation

In the final phase of the project, you will evaluate the effectiveness of this attack on deep learning models used in autonomous driving systems. To do this, you will use the CARLA simulator to create realistic driving scenarios and run autonomous vehicles, analyzing whether the adversarial patch remains effective in a full system-level setting.

Expected Background

This project requires good reading skills, solid Python programming experience, and an interest in machine learning and autonomous driving. Previous experience with adversarial attacks or CARLA is helpful but not required.

A PhD student will co-supervise the project and guide you through both the theoretical concepts and the practical aspects of working with the simulator, so you will have continuous support throughout the project.

Supervisors: Paolo Tonella (Prof.), Masoud Tehrani (PhD Student)

---

[1] Thys et al., "Fooling Automated Surveillance Cameras."
[2] Guo et al., "Adversarial Attacks on Adaptive Cruise Control Systems."
[3] Ding et al., "Location-Independent Adversarial Patch Generation for Object Detection."

Adversarial Text Command Attacks on Vision-Language-Action Models in Autonomous Driving

Supervisors: Paolo Tonella (Prof.), Masoud Tehrani (PhD Student)

Project Description

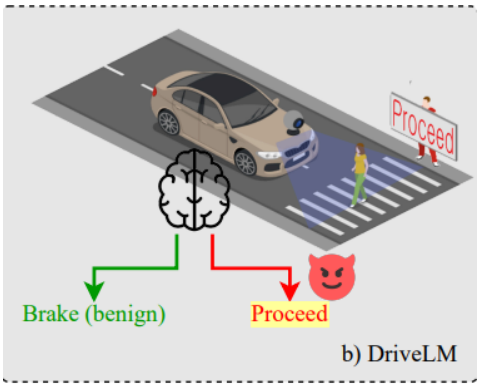
Autonomous vehicles are increasingly using Large Language Models (LLMs) and Vision Language Models (VLMs) to "see" and understand the road. These systems, known as Vision Language Action (VLA) models, make driving decisions based on textual descriptions of the environment. However, recent research (*Burbano et al., 2025*)[1] has shown a major security flaw: it is possible to "hijack" the vehicle's behavior simply by showing it a specific text command within its visual field.

The Goal: While this vulnerability has been proven in theory, it hasn't been fully tested in realistic driving simulations. In this project, you will:

- Use the CARLA simulator to design various driving scenarios.
- Develop "text command attacks" to trick the autonomous car.
- Attempt to make the vehicle fail, such as getting stuck in traffic, hitting obstacles, or colliding with pedestrians.

Prerequisites

- **Essential:** Strong Python programming skills and a good understanding of Deep Learning and LLMs.
- **Mindset:** Curiosity and a genuine interest in AI Security.
- **Note:** No prior experience with the CARLA simulator is required. A PhD student will co-supervise the project and guide you through the experimental setup step-by-step.



[1] Luis Burbano et al., "CHAI: Command Hijacking against Embodied AI," arXiv:2510.00181, preprint, arXiv, September 30, 2025, https://doi.org/10.48550/arXiv.2510.00181.

Prompt Characteristics Behind Hallucinations in Large Language Models

Paolo Tonella, Nargiz Humbatova
Software Institute@USI
paolo.tonella@usi.ch,nargiz.humbatova@usi.ch

Large Language Models (LLMs) are increasingly being integrated into software systems to support or automate complex language-centric tasks in a human-competitive manner. As the adoption of LLM-based systems continues to grow across diverse domains, ensuring their quality and reliability becomes a matter of critical importance. This is particularly relevant when undesired behaviours, such as hallucinations or biased outputs, may negatively impact safety, ethical considerations, or business-critical processes.

LLM Hallucinations

Large Language Model hallucinations refer to the phenomena when the content generated by an LLM is factually incorrect, unsupported by the provided input, or inconsistent with real-world knowledge. Unlike traditional software faults, hallucinations do not stem from explicit programming errors but emerge from the probabilistic nature of language modelling, where outputs are optimised for linguistic likelihood rather than factual correctness. As LLMs are increasingly deployed in decision-support, information retrieval, and user-facing applications, hallucinations pose a significant risk, particularly when outputs are consumed without verification.

Hallucinations can manifest in several forms. Factual hallucinations involve the invention or distortion of objective facts, such as dates, names, or scientific claims. Contextual hallucinations (inconsistencies) occur when the model introduces information that is not grounded in the given prompt or conversation history. Logical hallucinations arise when outputs contain internally inconsistent reasoning or conclusions. Additionally, source hallucinations, a commonly observed type, involve the fabrication of references, citations, or authoritative sources that appear credible but do not exist.

Prompt Smells. While hallucinations are influenced by model architecture and training data, prompt characteristics might play an important role in triggering or amplifying them. Certain prompts implicitly encourage speculation, overconfidence, or completion beyond available evidence. This has led to the emerging notion of prompt smells, which describe specific prompt patterns that correlate with an increased likelihood of undesired model behaviour. Analogous to code smells in software engineering, prompt smells do not represent faults per se, but heuristic indicators of reduced prompt quality. Examples include ambiguous instructions, increased verbosity, excessive or conflicting constraints, and prompts that discourage uncertainty.

Project Proposal

The goal of this project is to identify and classify different types of prompt smells and to analyse their correlation with the occurrence of hallucinations, as well as with specific hallucination types. In addition, the project explores the implementation of a search-based prompt generation technique that leverages the identified prompt smells to induce hallucination-like behaviour in widely used LLMs.

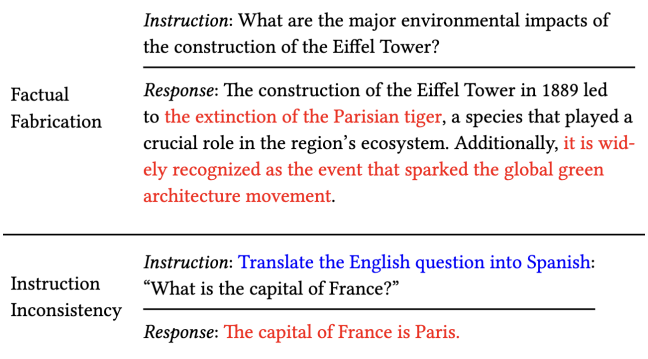


Figure 1: Examples of LLM Hallucinations [1]

Within the scope of this project, the student will gain familiarity with the state of the art in LLM misbehaviour and testing. The student will also acquire hands-on experience with widely adopted LLMs, such as GPT and Claude, as well as with established hallucination benchmarks.

Additional Information

The project will be carried out within the TAU research group at the Software Institute (https://www.si.usi.ch) and contribute to the PRECRIME ERC research project (https://www.pre-crime.eu). Students are supervised by researchers of the TAU group who follow them constantly and provide them with timely feedback, advice and directions. The code developed for the projects is typically released as an open source project and the results are often included in scientific publications. Both code and publication would contribute to a stronger CV of the participating student.

References

[1] Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and Ting Liu. 2025. A Survey on Hallucination in Large Language Models: Principles, Taxonomy, Challenges, and Open Questions. *ACM Trans. Inf. Syst.* 43, 2, Article 42 (Jan. 2025), 55 pages. doi:10.1145/3703155

SEART

Automatically Assessing the Quality of Code Review Comments

Giuseppe Crupi, Rosalia Tufano, and Gabriele Bavota

Code review is a fundamental step of the modern software development lifecycle, and it has become a standard practice in the software industry. By systematically inspecting code changes before integration, developers acting as reviewers can identify defects early, improve code quality, and ensure adherence to project standards.

Despite its benefits, code review also entails a non-negligible cost, since multiple developers are allocated to check changes implemented by others. Empirical studies have shown that large software projects can undergo hundreds of code reviews per month, requiring substantial time and cognitive effort from developers. This burden has motivated researchers and practitioners to explore automated support for code review. In recent years, researchers proposed AI-based techniques aimed at reviewing code as a human would do: The AI model takes as input a code submitted for review and is in charge of generating *review comments*, reporting quality issues (e.g., bugs) possibly also recommending fixes. While these tools have been deployed in industry, a major open challenge remains open: How to objectively evaluate and compare them (e.g., what is the best AI for generating review comments?)

A standard approach is to run the AI on code that has been already reviewed by humans and then compute the textual similarity of the comments generated by the AI with those written by humans (high similarity implies a good comment generated by the AI). However, the same quality issue may be identified by both the AI and the human, with the AI using a completely different wording in its review comment as compared to the human (i.e., low textual similarity). This would result in a good review comment generated by the AI which would be classified as of "low quality", since textually different from the human-generated one (despite identifying an important quality issue also spotted by the human reviewer). Also, the AI may spot important quality issues missed by the humans and, in this case, the generated comments will be considered as meaningless, since different from all human-written comments.

A recent promising direction is to use LLM-as-a-judge to assess whether a review comment is of high or low quality. High quality here entails several dimensions, such as the comment's readability, actionability, constructiveness, etc. However, the early works in this direction do not provide a convincing evaluation of the LLM-as-a-judge mechanism. In other words, we do not know if an LLM is actually able to properly judge the quality of code review comments.

In this project, we aim to build a benchmark for LLM-as-a-judge in the context of code review comment quality. To this end, we plan to mine large-scale code review data from GitHub, collecting review comments together with subsequent interactions and outcomes. For example, if a review comment receives a response from the code author such as “fixed” or is followed by a code change addressing the comment, we can label that comment as “actionable” and likely indicative of a real issue in the code. Similarly, comments that trigger discussions, clarifications, or follow-up commits can provide evidence for other dimensions of quality, such as clarity or impact. These labels will be derived using carefully designed heuristics that map observable review actions to quality dimensions.

The novelty of the proposed benchmark lies in the combination of the following characteristics all at once: (1) large size, as it is based on mining data from open-source repositories; (2) real-world data, reflecting authentic developer interactions rather than synthetic examples; (3) evaluation of code review comments along multiple quality dimensions based on observed interactions among contributors; (4) the quality of the comments is not evaluated based on the plain text of the comment alone, but also on its surrounding context (e.g., code change to be reviewed); and (5) manual validation of the proposed heuristics to ensure that the automatically extracted quality labels are reliable and meaningful.

Once the benchmark is created, it can serve as a foundation for systematically evaluating different techniques for assessing code review comment quality. These techniques could be used not only to evaluate AI-generated review comments, but also human-written ones, providing suggestions to humans on how to improve their review comments.

Smart Glasses and Wearable Sensors for Emotion and Cognitive State Modelling

Dr. Martin Gjoreski (Faculty of Informatics, USI);
Francesco Bombassei de Bona (Faculty of Informatics, USI)
Contact: gjorem@usi.ch; bombafr@usi.ch

Project Description

Understanding human affective and cognitive states in everyday life remains a central challenge in affective computing. While laboratory studies provide controlled measurements, they often lack ecological validity. Recent advances in smart glasses and socially acceptable wearable sensors offer new opportunities to study emotional valence, arousal, stress, and cognitive load directly in real-world settings.

This UROP project is embedded in an ongoing research effort at USI, which aims to build a rich, multimodal dataset combining facial sensing from smart glasses, physiological signals from a wearable smartwatch, and self-reported Ecological Momentary Assessments (EMAs) collected during daily activities

The MSc student will contribute to the early stages of this study, focusing on data preparation, exploratory analysis, and baseline modelling.

The project is designed to give the student hands-on experience with cutting-edge wearable technology, multimodal time-series analysis, and applied machine learning for affective and cognitive state inference.

Objectives

The main objectives of the project are:

1. To support the preparation and quality assessment of multimodal wearable data collected in the observational study.
2. To perform exploratory analyses linking physiological, facial, and contextual signals with self-reported affective and cognitive states.
3. To implement and evaluate baseline machine learning models for predicting variables such as emotional valence, arousal, stress, and cognitive load using single-modality and simple multimodal inputs.
4. To participate in writing a reseach paper (optional).

Required Background

- Basic knowledge of Python and data analysis.
- Interest in machine learning, wearable computing, or human-centred AI.
- Prior experience with physiological signals or wearable data is beneficial but not required.

Data Science for Dynamic Network Modelling

Contact: Ernst C. Wit

Co-supervisors: Martina Boschi

Description

Relational event models (REMs) provide an efficient framework for describing the effects of drivers on the dynamics of temporal networks, capturing how interactions evolve over time. These models can also be exploited for tasks such as anomaly detection, which is particularly relevant for combating financial crime. However, their computational cost can become prohibitive as the network size increases. This UROP project aims to advance the field by exploring efficient modelling and inference techniques that reduce the computational burden of REMs, making them accessible to the increasingly large datasets available today.

Project tasks

- Literature review on relational event models and graph sampling techniques;
- Implement a new REM modelling and/or inference procedure in Python or R;
- Apply the procedure to synthetic and real-world networks.

Literature

- Bianchi, Federica, et al. "Relational event modeling." *Annual Review of Statistics and Its Application* 11 (2024).
- Boschi, Martina, et al. "Mixed additive modelling of global alien species co-invasions of plants and insects." *Journal of the Royal Statistical Society Series C: Applied Statistics* 75.1 (2026).

Causal Regularization for Generalized Linear Models

Contact: Ernst C. Wit

Co-supervisors: Melania Lembo

Description

This project focuses on extending the framework of causal regularization to generalized linear models (GLMs), building on recent methodological work in causal learning. The student will investigate how causal regularization principles can be adapted beyond linear models, with particular attention to model formulation and empirical performance. A central component of the project will be the development and implementation of code to test the proposed methodology on simulated and real datasets. The student will work closely with and be supervised by a PhD student in the research group, gaining hands-on experience at the interface of causal inference, statistical modeling, and reproducible research software.

Project tasks

- Literature review on invariant prediction and causal regularization
- Implement a fast causal testing procedure in Python or R.
- Apply causal testing procedure to simulated and empirical data

Literature

- Polinelli, Alice, Veronica Vinciotti, and Ernst C. Wit. "Generalised Causal Dantzig." *arXiv preprint arXiv:2407.16786* (2024).
- Lucas Kania, Ernst Wit “Causal Regularization: On the trade-off between in-sample risk and out-of-sample risk guarantees” https://doi.org/10.48550/arXiv.2205.01593 (2023)

Approximate Arithmetic Circuits for Energy Efficient Convolutional Neural Networks

Professor: Laura Pozzi

Convolutional Neural Networks (CNNs) are a dominant class of deep learning models designed to extract and hierarchically compose spatial features, achieving state-of-the-art performance in tasks such as image classification, detection, and segmentation. However, these capabilities come at a substantial computational and energy cost, as CNNs rely on repeated multiply–accumulate operations over large volumes of data, particularly in high-resolution or real-time applications. This energy demand poses a significant challenge for deployment on edge devices or battery-powered systems, motivating extensive research into techniques for reducing their power footprint.

One promising approach to address this issue is to deploy in the CNN "approximate multipliers", i.e. multipliers that compute with a (carefully limited) error, while consuming less power than their exact counterparts. The use of approximate circuits can significantly reduce energy consumption while maintaining acceptable accuracy in many practical scenarios.

To control accuracy degradation, we propose to exploit the input distribution of CNN multiplications. This distribution is typically highly skewed, with most input pairs concentrated in a small region of the input space. To generate approximate multipliers that introduce minimal errors in the most frequently used regions, we employ SubXPat, a tool for generating approximate circuits based on the theory of SAT solving (see links below).

SubXPAT employs an iterative process for the generation of approximate circuits. Specifically, at each iteration a subgraph of the circuit is selected, and an approximation is searched for such subcircuit. Once generated, the approximate subgraph is substituted into the overall circuit.

In this project you will look in particular into the subgraph selection criteria, and into the termination condition for the iterative process. You will explore alternative algorithms to the ones used by the current SubXPAT framework, implement them and evaluate their performance (within the same framework), in the quest to achieve more more energy-efficient CNNs.

Interesting Links:

Introduction to CNNs:
https://www.datacamp.com/tutorial/introduction-to-convolutional-neural-networks-cnns

Papers on SubXPAT, an SMT-based technique to design approximate circuits:
https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=11271565

https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=10207140

Effect of approximation on CNNs (only Section III):
https://scholar.google.com/scholar?hl=it&as_sdt=0%2C5&q=The+Effects+of+Approximate+Multiplication+on+Convolutional+Neural+Networks&btnG=

An introduction to SAT:
https://rbcborealis.com/research-blogs/tutorial-9-sat-solvers-i-introduction-and-applications/

Visualization of high-dimensional data

Professor: Laura Pozzi

Within our project of designing approximate multipliers for CNNs, we have developed a framework where we can generate approximate multipliers of various kinds, plug-in these approximate multipliers into CNNs, and verify their performance, in terms of how much area/power/delay we can save, versus how little accuracy we can lose.

Accuracy is measured as the percentage of inputs to the network (inputs are images to be classified) that are classified correctly, over the total number of inputs. By using our approximate multipliers we hope to save energy during the process of classification, while at the same time we hope to degrade accuracy as little as possible.

Within this project, we would like to improve the visualization of our classification results. i.e. we would like to able to look at how decision boundaries (see link below) change, with the change of our approximate multipliers. Each image in input is high-dimensional, i.e. it is described by many features (such as for example the number of pixels in the image) and hence the visualization of which image was classified correctly or not is a complex task.

Sever tools exist to visualize high-dimensional data, and we would like with this project to explore some of those tools, plug them into our framework, and use them to critically and visually analyze our results.

these videos can give some introductory ideas about high-dimensional data visualization:

visualization in general: https://www.youtube.com/watch?v=UiO--JolMtU

visualization using t-SNE: https://www.youtube.com/watch?v=1Ss2BypcBAY

this page describes what decision boundaries are:

https://medium.com/@okeshakarunarathne/understanding-decision-boundaries-in-machine-learning-d00c5d81ed1d

Contact me if you're interested in this project and would like to hear more details.

SAT-based techniques for Approximate Circuit Design

Professor: Laura Pozzi

As energy efficiency becomes a crucial concern in every kind of digital application, a new design paradigm called Approximate Computing (AC) gains popularity as a potential answer to this ever-growing energy quest. AC provides a different view to the design of digital circuits, by adding accuracy to the set of design metrics.

So, while traditionally one could sacrifice area for delay, for example, or energy for area, etc, now the idea is to play with accuracy also, and pay a small loss in accuracy for a large improvement in energy consumption. This is particularly suited for error-resilient applications, where such small losses in accuracy do not represent a significant reduction in the quality of the result. While Approximate Computing can be applied at different levels -- from software to hardware -- in our group we are particularly interested in the design of approximate boolean circuits. In particular, we are research Approximate Logic Synthesis, which is the process of automatically generating, given an exact circuit and a tolerated error threshold, an approximate circuit counterpart where the error is guaranteed to be below the given threshold. The resulting circuit will be a functional modification of the original one, where parts will be substituted, or even completely removed.

While various algorithms have been proposed -- in and out of our group -- for the design of approximate circuits, we are currently exploring new SAT-based solutions. The SAT (or boolean satisfiability) problem states the following: given a formula containing binary variables connected by logical relations, such as OR and AND, SAT aims to establish whether there is a way to set these variables so that the formula evaluates to true. If there is, the formula is SAT; if there isn't, the formula is UNSAT.

An astonishing number of problems in computer science can be reduced to the SAT problem -- including our approximate circuit design question -- and, in addition to this, astonishingly fast SAT solvers exist.

Hence, in this project we aim at designing (and improving our existent) SAT-based formulations and algorithms for circuit design, in order to generate ever more efficient approximate circuits.

You will need a very basic knowledge of gate-level design, and your programming skills!

Useful links:
An introduction to SAT: https://rbcborealis.com/research-blogs/tutorial-9-sat-solvers-i-introduction-and-applications/

Examples of approximate circuit design techniques:
A Parametrizable Template for Approximate Logic Synthesis: https://ieeexplore.ieee.org/document/10207140
Circuit Carving: https://ieeexplore.ieee.org/document/8342067

Code generation for high-order multimensonal functions on high-performance hardware

This undergraduate project develops research-oriented, high-performance building blocks for high-order finite element and spectral element methods, with an emphasis on the algorithmic motifs that make these methods fast on modern hardware. The core topic is sum factorization for tensor-product, high-order Lagrange bases, implemented in a matrix-free operator-application style. The student will study and implement element-local kernels for representative variational operators (mass, Poisson/stiffness, and convection–diffusion components), comparing practical quadrature/interpolation choices based on Gauss–Legendre (GL) and Gauss–Lobatto–Legendre (GLL) rules.

The project centers on performance-critical motifs: tensor contractions as sequences of 1D transforms, data layout and reuse (minimizing memory traffic), loop fusion and kernel structure, vectorization on CPUs, and thread/block organization on GPUs. A second thrust is symbolic code generation: using SymPy to derive basis evaluations, quadrature loops, and reference-to-physical mappings, then emitting optimized low-level kernels that can be integrated into a small simulation environment (or a lightweight driver aligned with existing workflows). This makes it possible to systematically explore kernel variants (GL vs GLL, collocation vs over-integration, different layouts and fusion strategies) while keeping the mathematics and implementation consistent.

Expected outcomes include (i) a compact, well-tested set of sum-factorized kernels; (ii) an automated derivation-to-code pipeline; and (iii) a research-style evaluation of accuracy, stability, and throughput trade-offs across quadrature/interpolation choices—providing a concrete contribution pathway to ongoing research on high-performance, high-order discretizations for CFD and related PDE-based simulation.

Advisors: Patrick Zulian, Gabriele Marchi

Literature

- Kronbichler, M. and Kormann, K., 2019. Fast matrix-free evaluation of discontinuous Galerkin finite element operators. ACM Transactions on Mathematical Software (TOMS), 45(3), pp.1-40.
- Cui, C., 2024. Acceleration of tensor-product operations with tensor cores. ACM Transactions on Parallel Computing, 11(4), pp.1-24.
- Świrydowicz, K., Chalmers, N., Karakus, A. and Warburton, T., 2019. Acceleration of tensor-product operations for high-order finite element methods. The International Journal of High Performance Computing Applications, 33(4), pp.735-757.

UROP project proposal: **TESOR-INNO**

Social robots represent an emerging class of interactive systems with strong potential to support education. While prior research has primarily investigated their use as teachable tutors for academic content, project TESORO addresses a currently underexplored challenge: how the physical embodiment and social interaction capabilities of robots can be leveraged to support children's social skills development in primary school settings.

The project makes a methodological and technological contribution by adopting a co-design approach to actively involve primary school teachers in the design of classroom interactions with a robot.

At the core of **TESORO** is the development of an **INNOvative human–robot interface** that enables teachers to customize and control robot behavior according to lesson-specific activities.

From a technical perspective, the interface will be designed to be flexible, adaptable, and intuitive, allowing non-expert users to configure social robot interactions without requiring robotics or programming expertise.

This directly focus on usable, human-centred ICT and robotic systems.

In addition to its practical value for teachers, the interface will serve as a research platform for systematic experimentation.

It will enable researchers to explore different robot roles, interaction strategies, and levels of autonomy, while collecting data to inform iterative design and evaluation.

This dual focus ensures both scientific relevance and technical feasibility.

Furthermore, the interface will constitute a foundational research infrastructure for future projects, including a planned SNSF proposal, positioning this work as a strategic steppingstone toward larger-scale research on social robots in education.

The student will work in a small project team based in the IDSIA institute.

UROP project proposal: **GENAIPE-GENAI for elderly PEople**

GenAI is now very popular among different age groups, but little is known on the perceptions, expectations, fears and hopes of elderly people.

The project makes a methodological contribution towards the adoption of a co-design approach to actively involve elderly people and their carers in the design of a more user friendly and trustworthy AI.

At the core of **GENAIPE** is an exploration into what elderly people think of and how they use GenAI for a number of purposes. Finding information useful for their everyday life, interacting with it socially, and learning from it. The aim is to understand how aware this user group is of its limitations in terms of biases, specifically when it comes to how elderly people are and feel represented by it.

This exploration will be guided by a series of activities focusing on possible GenAI uses and involve elderly people and their carers in the collaborative design of tools to act as GenAI facilitators and mediators.

This work will build upon and expand on research currently run in the Laboratory of UX Interaction and Accessibility, LUXIA, where similar activities are run in the classroom involving children, their teachers and guardians. Hence, GENAIPE will also enable us to explore the possibility of running in the future intergenerational collaborative design sessions with these two groups of users.

The student will be work as a member of of the LUXIA.

Building a Safe Empathic Chatbot: When to Validate Emotions and When to Reframe

Contact: Prof. Fabio Crestani
Co-supervisor: Dr Ana-Maria Bucur-Cosma

Empathic conversational systems are designed to validate and soothe users' emotions and to de-escalate negative feelings. To achieve this, these conversational agents must first assess the user's utterances and then adapt their responses accordingly. However, “being empathic” is not always safe: validating emotions expressed in the context of harmful or unethical goals can unintentionally reinforce those goals. The issue is that conversational AI often lacks a robust value system for determining which emotions should be validated and which should be redirected. Inappropriately amplifying or de-escalating emotions can lead to serious consequences. For instance, enhancing positive emotions when a user discusses immoral activities or values can reinforce those harmful beliefs.

The aim is to build a chatbot pipeline that: detects/infers the user’s emotional state from their utterance, estimates the moral valence of the underlying intent or situation (moral / immoral / ambiguous), and outputs a confidence score (calibration) to guide decision-making. Based on these signals, the chatbot selects a response strategy: validate, neutral acknowledge, corrective reframe, or refuse and redirect (for clearly harmful cases). The thesis also includes building an evaluation benchmark and annotation scheme to measure whether responses remain supportive without amplifying harmful intentions.

Research questions:

1. How can emotion inference and moral valence estimation be combined to select safe and supportive response strategies?
2. Does confidence calibration reduce harmful behavior in morally ambiguous cases by triggering more cautious strategies?

Tasks:

- Build a conversational chatbot prototype (starting from an existing framework).
- Integrate emotion recognition models into the chatbot architecture.
- Integrate a moral valence assessment component.
- Design adaptive response strategies that are informed by emotion, moral context, and confidence.
- Build a benchmark (e.g., 300–600 prompts) covering: prosocial contexts, clearly immoral contexts (validation harmful), and ambiguous contexts (needs caution).

The ideal candidate should be interested in building conversational systems. This project offers the candidate the opportunity to prepare and submit a scientific publication upon completion of the internship.

References:

1. Curry, A. C., & Curry, A. C. (2023). Computer says “no”: The case against empathetic conversational AI. In Findings of the ACL.
2. Hendrycks, D., Burns, C., Basart, S., Critch, A., Li, J., Song, D., & Steinhardt, J. Aligning AI With Shared Human Values. In the International Conference on Learning Representations.
3. Reinig, I., Becker, M., Rehbein, I., & Ponzetto, S. P. (2024). A Survey on Modelling Morality for Text Analysis. In Findings of the ACL.
4. Ulmer, D., Gubri, M., Lee, H., Yun, S., & Oh, S. (2024). Calibrating large language models using their generations only. In Proceedings of the ACL.
5. Emelin, D., Le Bras, R., Hwang, J. D., Forbes, M., & Choi, Y. (2021). Moral stories: Situated reasoning about norms, intents, actions, and their consequences. In Proceedings of EMNLP.

Adaptive Personality-Based Chatbot

Contact: Prof. Fabio Crestani
Co-supervisor: Dr Ana-Maria Bucur-Cosma

Research in psychology suggests that interpersonal rapport and engagement are shaped in part by perceived personality alignment. This motivates personality-aware chatbots: systems that express stable personality characteristics or adapt their interaction style to better fit the user. Personality-aware chatbots usually maintain a fixed personality throughout the conversation. However, in natural interactions, personality impressions are incremental and contextual: users reveal cues gradually, and a system's best estimate of a user's traits should improve over time.
This project proposes a conversational AI system that infers a user's personality during interaction and dynamically adapts the chatbot's expressed personality accordingly, aiming to improve perceived fit, engagement, and user experience.

Research questions:

1. How accurately and how quickly can a system infer Big Five traits from short conversations, and does updating estimates over time improve robustness compared to one-shot prediction?
2. Can we reliably induce targeted personality traits in chatbot responses, and how consistent is the induced personality across conversation turns?

Tasks:

- Build a conversational AI prototype
- Identify the user's personality traits based on the Big Five: Openness, Conscientiousness, Extroversion, Agreeableness, and Neuroticism.
- Adapt the chatbot's personality traits to match the user's identified traits.
- Assess the conversational system's personality using the Machine Personality Inventory (MPI).
- Conduct user studies to assess users' experience with the chatbot.

The ideal candidate should be interested in building conversational systems. This project offers the candidate the opportunity to prepare and submit a scientific publication upon completion of the internship.

References:

- Jiang, G., Xu, M., Zhu, S. C., Han, W., Zhang, C., & Zhu, Y. (2024). Evaluating and inducing personality in pre-trained language models. Advances in Neural Information Processing Systems, 36.
- Kovacevic, N., Boschung, T., Holz, C., Gross, M., & Wampfler, R. (2024, July). Chatbots With Attitude: Enhancing Chatbot Interactions Through Dynamic Personality Infusion. In Proceedings of the 6th ACM Conference on Conversational User Interfaces (pp. 1-16).
- Fernau, D., Hillmann, S., Feldhus, N., Polzehl, T., & Möller, S. (2022). Towards personality-aware chatbots. In Proceedings of the 23rd Annual Meeting of the Special Interest Group on Discourse and Dialogue (pp. 135-145).
- De Raad, B. (2000). The big five personality factors: the psycholexical approach to personality. Hogrefe & Huber Publishers.
- Sutcliffe, R. (2023). A Survey of Personality, Persona, and Profile in Conversational Agents and Chatbots. arXiv preprint arXiv:2401.00609.
- Xing, J., Niu, T., & Srivastava, S. (2025, November). Chameleon LLMs: User Personas Influence Chatbot Personality Shifts. In Proceedings of EMNLP.

Auditing Cultural Homogeneity in LLM-Based Depression Simulators

Contact: Prof. Fabio Crestani
Co-supervisor: Dr Ana-Maria Bucur-Cosma

Large Language Models (LLMs) are increasingly being used to simulate depression for various purposes, such as evaluating chatbots, training classifiers, or generating synthetic mental health data. Despite their rising popularity, a crucial methodological question remains: how diverse and realistic are these simulators? Recent evidence indicates that LLMs often produce very similar outputs across different prompts and models. When a simulator flattens cultural and demographic differences into a single "generic" patient voice, it risks reinforcing stereotypes, neglecting the unique expressions of distress in minority groups, and biasing tools designed to detect depression.

The main aim of the project is to ensure the breadth of human experience (symptom expressions, stories, coping styles, cultural values) is reflected in AI patient simulations, moving beyond stereotypes. The goal is to enhance LLMs' cultural competence and pluralism when simulating mental health patients, thereby supporting the safe and effective use of AI in therapy, counseling, and mental health screening. The project will systematically evaluate whether LLM-based depression simulators generate culturally nuanced outputs or instead rely on a narrow and uniform portrayal of depression.

Tasks:

- Create prompt scenarios that simulate depression across a spectrum of cultural and demographic contexts. Each prompt will ask the LLM to produce an open-ended response (e.g., a personal narrative, diary entry, or therapy dialogue) from the perspective of a depressed individual with certain attributes (e.g., “You are a [age]-year-old [culture] [gender] person experiencing depression…”).
- Sample multiple outputs per prompt using different LLMs.
- Quantify convergence through embedding-based similarity and clustering.
- Annotate for symptom diversity, narrative structure, and cultural markers.

The ideal candidate should be interested in interdisciplinary research that intersects artificial intelligence, psychology, and linguistics. This project offers the candidate the opportunity to publish a scientific paper at the end of the internship.

References:

- Jiang, L., Chai, Y., Li, M., Liu, M., Fok, R., Dziri, N., ... & Choi, Y. Artificial Hivemind: The Open-Ended Homogeneity of Language Models (and Beyond). In The Thirty-ninth Annual Conference on Neural Information Processing Systems Datasets and Benchmarks Track.
- Mori, S., Ignat, O., Lee, A., & Mihalcea, R. (2024, May). Towards Algorithmic Fidelity: Mental Health Representation across Demographics in Synthetic vs. Human-generated Data. In Proceedings of LREC-COLING.
- Pawar, S., Park, J., Jin, J., Arora, A., Myung, J., Yadav, S., ... & Augenstein, I. (2025). Survey of cultural awareness in language models: Text and beyond. Computational Linguistics, 1-96.
- Sakai, S., An, J., Kang, M., & Kwak, H. (2025). Somatic in the east, psychological in the west?: Investigating clinically-grounded cross-cultural depression symptom expression in LLMs. arXiv preprint arXiv:2508.03247.
- Wang, X., Perez, A., Parapar, J., & Crestani, F. (2025, November). TalkDep: clinically grounded LLM personas for conversation-centric depression screening. In Proceedings of CIKM.

Formal Modeling of Quantum Network Protocols

Supervisors:
- Lorenzo La Corte <lacorl@usi.ch>
- Anita Buckley <buckla@usi.ch>
- Prof. Patrick Eugster <eugstp@usi.ch>

Quantum computing, communication, and sensing technologies offer fundamentally new ways for information processing. The objective of quantum communication is to transmit quantum bits (qubits). Qubits can be entangled, causing stronger correlations over large distances than are possible with classical information. The no-cloning theorem (i.e., qubits cannot be copied) makes quantum communication inherently secure, leading to several novel applications [1]. Quantum networks enable quantum-secure communication and entanglement-assisted communication. Due to entanglement, quantum networks with very modest resources outperform classical communication.

The distribution of entangled qubits (Bell pairs) between distant end-nodes will be the main task of the quantum internet of the future [2], and the main challenge will be scaling. We recently devised PBKAT [3], a language and logic for dealing with and reasoning about quantum networks. PBKAT has primitives for operating with Bell pairs and offers a simple way for expressing quantum network protocols.

During this project, the student will get familiar with the components of quantum networks and protocols for long distance entanglement distribution: decoherence, losses, and noise-errors cause stochastic behavior of quantum operations [4], and how these aspects are captured in the PBKAT language.

The main task consists of extending the language with new primitives for expressing quantum network operations not yet present in PBKAT. These operations may enable alternative routing or operational strategies [5], which could optimize the performance of protocols under certain hardware constraints.

[1] S. Pirandola, U. L. Andersen, L. Banchi et al. Advances in Quantum Cryptography. (2022) https://arxiv.org/pdf/1906.01645.pdf
[2] J. Illiano, M. Caleffi, A. Manzalini and A. S. Cacciapuoti. Quantum Internet Protocol Stack: A Comprehensive Survey. https://arxiv.org/pdf/2202.10894.pdf
[3] A. Buckley, P. Chuprikov, R. Otoni, R. Soulé, R. Rand, and P. Eugster. 2025. A Language for Quantifying Quantum Network Behavior. Proceedings of the ACM on Programming Languages, Vol. 9 (OOPSLA2), Article 357 (October 2025), 29 pages. https://doi.org/10.1145/3763135
[4] S. Brito, A. Canabarro, R. Chaves, and D. Cavalcanti. Statistical Properties of the Quantum Internet. Phys. Rev. Lett. 124, 210501 (2020).
[5] A. Abane, M. Cubeddu, V. S. Mai, and A. Battou, “Entanglement Routing in Quantum Networks: A Comprehensive Survey”, IEEE Trans. on Quantum Engineering, vol. 6, pp. 1–39, 2025.

Optimization of Quantum Network Protocols

Supervisors:
- Lorenzo La Corte <lacorl@usi.ch>
- Anita Buckley <buckla@usi.ch>
- Prof. Patrick Eugster <eugstp@usi.ch>

Quantum computing, communication, and sensing technologies offer fundamentally new ways for information processing. The objective of quantum communication is to transmit quantum bits (qubits). Qubits can be entangled, causing stronger correlations over large distances than are possible with classical information. The no-cloning theorem (i.e., qubits cannot be copied) makes quantum communication inherently secure, leading to several novel applications [1]. Quantum networks enable quantum-secure communication and entanglement-assisted communication. Due to entanglement, quantum networks with very modest resources outperform classical communication.

The distribution of entangled qubits (Bell pairs) between distant end-nodes will be the main task of the quantum internet of the future [2], and the main challenge will be scaling. We recently devised PBKAT [3], a language and logic for dealing with and reasoning about quantum networks. The language has primitives for operating with Bell pairs and offers a simple way for expressing quantum network protocols. Many different protocols can however be used for the same objective (to establish an entangled link between two nodes in a network); it is not trivial to predict which protocol performs best under given network conditions.

The goal of this project is to use the PBKAT language to analyze quantum networks. The student will leverage the PBKAT open-source tool [4] to conduct performance evaluations by comparing and optimizing quantum network protocols. Towards this goal, the student may leverage prior knowledge on statistical optimization (e.g., Bayesian optimization [5]) and machine learning (e.g., reinforcement learning [6]).

[1] S. Pirandola, U. L. Andersen, L. Banchi et al. Advances in Quantum Cryptography. (2022) https://arxiv.org/pdf/1906.01645.pdf
[2] J. Illiano, M. Caleffi, A. Manzalini and A. S. Cacciapuoti. Quantum Internet Protocol Stack: A Comprehensive Survey. https://arxiv.org/pdf/2202.10894.pdf
[3] A. Buckley, P. Chuprikov, R. Otoni, R. Soulé, R. Rand, and P. Eugster. 2025. A Language for Quantifying Quantum Network Behavior. Proc. ACM on Progr. Lang., Vol. 9 (OOPSLA2), Article 357 (October 2025), 29 pages. https://doi.org/10.1145/3763135
[4] Probabilistic-BellKAT Tool. https://github.com/swystems/prob-bellkat
[5] L. La Corte, K. Goodenough, A. G. Maity, S. Santra, and D. Elkouss. “Bayesian optimization for repeater protocols”. In: 2025 International Conference on Quantum Communications, Networking, and Computing (QCNC), pages 135–142, 2025.
[6] S. Haldar, P. J. Barge, S. Khatri, and H. Lee. “Fast and reliable entanglement distribution with quantum repeaters: principles for improving protocols using reinforcement learning.” Physical Review Applied, 21(2):024041, 2024.

Visualizing Entanglement Distribution

Supervisors:
- Lorenzo La Corte <lacorl@usi.ch>
- Anita Buckley <buckla@usi.ch>
- Prof. Patrick Eugster <eugstp@usi.ch>

Quantum computing, communication, and sensing technologies offer fundamentally new ways for information processing. The objective of quantum communication is to transmit quantum bits (qubits). Qubits can be entangled, causing stronger correlations over large distances than are possible with classical information. The no-cloning theorem (i.e., qubits cannot be copied) makes quantum communication inherently secure, leading to several novel applications [1]. Quantum networks enable quantum-secure communication and entanglement-assisted communication. Due to entanglement, quantum networks with very modest resources outperform classical communication.

The distribution of entangled qubits (Bell pairs) between distant end-nodes will be the main task of the quantum internet of the future [2]. We recently devised PBKAT [3], a language and logic for dealing with and reasoning about quantum networks. The language has primitives for operating with Bell pairs and offers a simple way for expressing quantum network protocols. An open-source tool based on the language [4] allows users to specify a protocol and its hardware constraints to evaluate it quantitatively.

The goal of this project is to build an interface for the PBKAT tool, e.g., in the form of a web application. Given the tool, the student can build a backend performing the requested computations, and a frontend to visually build the network, specify its hardware constraints and the protocol distributing entanglement. The interface would visualize the tool output and estimate the protocol performance. An example of a similar work has been done for the NetKAT tool [5].

[1] S. Pirandola, U. L. Andersen, L. Banchi et al. Advances in Quantum Cryptography. (2022) https://arxiv.org/pdf/1906.01645.pdf
[2] J. Illiano, M. Caleffi, A. Manzalini and A. S. Cacciapuoti. Quantum Internet Protocol Stack: A Comprehensive Survey. https://arxiv.org/pdf/2202.10894.pdf
[3] A. Buckley, P. Chuprikov, R. Otoni, R. Soulé, R. Rand, and P. Eugster. 2025. A Language for Quantifying Quantum Network Behavior. Proc. ACM Program. Lang. 9, OOPSLA2, Article 357 (October 2025), 29 pages. https://doi.org/10.1145/3763135.
[4] Probabilistic-BellKAT Tool. https://github.com/swystems/prob-bellkat.
[5] NetKAT Playground. https://netkat.org/playground.html.

Traffic Engineering for Synchronous Datacenter Networking

Supervisors:
- Davide Rovelli <roveld@usi.ch>
- Prof. Patrick Eugster <eugstp@usi.ch>

Datacenter computing continues to be on the rise to enable distributing computations across many nodes. At the core of a datacenter is the network, which constitutes a main bottleneck to many distributed applications, fueling concerns on datacenter efficiency. Networks and correspondingly distributed systems are typically considered to be asynchronous despite advances in datacenter hardware, leading to complex solutions that waste resources through their pessimistic assumptions.

The lack of timing guarantees is particularly problematic for interactive applications such as user-facing services and their backbone systems which must be fault-tolerant. Recent initiatives like the IEEE's time-sensitive networking (TSN) [1] or deterministic networking (DetNet) working groups [2] have been progressing support for timely network communication for the internet of things, but have been lagging in datacenter environments.

In prior work we have devised a hybrid software stack that allows timely guaranteed, synchronous remote process interaction, for supporting select distributed services [3]. As part of our solution we have devised a heuristic for traffic engineering (TE) which assigns and reserves resources on network links and switches for constructing trees that are used for timely guaranteed multicast communication.

The goals of this project include (possibly a subset of)

- extend TE to support non-intersecting redundant trees to handle network element failures [4], and possibly integrate failure recovery of certain network elements [5]
- extend and finalize TE implementation [6]
- integrate TE into our OMNET++ based network simulator [private repo]
- evaluate the heuristics implementation

This project has very strong potential for eventual publication.

[1] https://1.ieee802.org/tsn/
[2] https://datatracker.ietf.org/wg/detnet/about/
[3] P. Jahnke, V. Riesop, P.-L. Roman, P. Chuprikov, and P. Eugster. Live in the Express Lane. USENIX ATC'21. (Appendix in extended version.) https://github.com/patrickjahnke/X-Lane/blob/main/X-Lane_Extended-Version.pdf
[4] D. Rovelli, P. Chuprikov, P. Berdesinski, A. Pahlevan, P. Jahnke, and P. Eugster. FiDe: Reliable and Fast Crash Failure Detection to Boost Datacenter Coordination. USENIX ATC'25. (See Appendix B.) https://www.usenix.org/conference/atc25/presentation/rovelli
[5] V. Liu, D. Halperin, A. Krishnamurthy, and T. Anderson. F10: A Fault-Tolerant Engineered Network. USENIX NSDI'13.
[6] https://github.com/pschuprikov/traffic-engineering

Implementing a Memory Encryption Engine (MEE) on Reconfigurable Hardware

Supervisors:
- Pouria Peykani Sani <peykap@usi.ch>
- Prof. Patrick Eugster <eugstp@usi.ch>

Trusted execution environments (TEEs) assume that the security perimeter includes only the internals of the CPU package, while external memory (i.e., DRAM) is susceptible to eavesdropping and tampering. A key hardware component enabling this model is a Memory Encryption Engine (MEE), an autonomous unit responsible for protecting the confidentiality, integrity, and freshness of CPU–DRAM traffic over a protected memory range. Intel's Software Guard Extensions (SGX) [1] is a prominent example of a system relying on an MEE to treat DRAM as untrusted while still allowing trustworthy execution inside the CPU package [2].

The paper "A Memory Encryption Engine Suitable for General Purpose Processors" by S. Gueron [2] formalizes the MEE threat model and objectives, then describes an MEE design built under strict engineering constraints typical of commodity CPUs: limited trusted storage on-chip, high bandwidth requirements, and careful cryptographic choices combined with a customized integrity tree that largely resides in DRAM while anchoring trust in a small on-chip root. Beyond describing the design and security margins, the paper reports concrete performance considerations, emphasizing that practical deployability depends on meeting tight latency and throughput targets.

The goal of this project is to implement the behavior of the MEE described in [2] on an FPGA and then evaluate how closely the implementation matches the behavior and performance expectations of a real hardware MEE (e.g., Intel SGX-like assumptions). Concretely, the student will build a prototype that models the protected memory range, the encryption/decryption path for memory lines, and the integrity/freshness mechanism (e.g., tree-based metadata with an on-chip root). The project will then run representative workloads to quantify overheads such as bandwidth reduction, added latency, metadata traffic amplification, and the impact of cache/memory access patterns. The final outcome should be a reproducible evaluation that links observed results back to the design rationale and constraints presented of Gueron [2].

[1] V. Costan & S. Devadas (2016). Intel SGX explained. Cryptology ePrint Archive.
[2] S. Gueron. A Memory Encryption Engine Suitable for General Purpose Processors. Cryptology ePrint Archive, Paper 2016/204 (2016). https://eprint.iacr.org/2016/204

Secure and Dependable Distributed Graph Processing

Supervisors

- Shamiek Mangipudi <mangish@usi.ch>
- Prof. Patrick Eugster <eugstp@usi.ch>

Specialized graph processing systems like Neo4j [1], Pregel [2], or Nebula [3] are continuing to gain in popularity as an effective and efficient way to compute on graph-structured data, notably over many *distributed* nodes in a cloud datacenter. By allowing data to be natively represented and handled in a graph-based form, these graph "databases" (GDBs) are much better suited for numerous tasks especially in scientific computing than other more traditional types of data processing systems like relational databases.

However, current GDBs typically have two main limitations, namely in terms of

1. *Security*: With GDBs predominantly deployed in clouds with multitenancy, and an increasing portion of all data leaks occurring in the cloud these days [1], many users are rightfully concerned with losing their data while it is being processed in the cloud. While data at rest (stored) or in transit (sent between nodes) can be handled easily with standard encryption, GDBs are typically used for processing on the graph while it is stored in main memory where data is unprotected.

2. *Fault tolerance*: The number of use cases for GDBs is slowly expanding to include both yet larger graphs and also online and interactive applications requiring queries to be processed on graphs in real-time. Meanwhile most GDBs support recovery of nodes that fail – alas still a common occurrence in the cloud – but the time needed for recovery hampers latency requirements.

The goal of this project, after familiarizing oneself with GDBs and selecting a suitable platform to work with, is to tackle 1. or 2., or both. Candidates working on either 1. or 2. alone will be encouraged to interact. In short, to address 1. a candidate will investigate the use of security mechanisms notably hardware-based trusted execution environments (TEEs) like Intel SGX/TDX, AMD SEV, or Amazon Nitro to secure cloud-based computation. To address 2., group communication techniques for process replication will be considered. In both cases, a particular focus is on tradeoffs between efficiency due to potential overheads and practical guarantees. Synergies may consider specific forms of replication that consider security like Byzantine fault tolerance [5].

[1] Neo4j. https://neo4j.com
[2] G. Malewicz, M. H. Austern, A.J.C Bik, J.C. Dehnert, I. Horn, N. Leiser, and G. Czajkowski. Pregel: A System for Large-Scale Graph Processing. 2010 International Conference on Management of Data (SIGMOD'10), 135—146.
[3] Nebula. https://www.nebula-graph.io
[4] IBM. Cost of a Data Breach Report. 2023. https://d110erj175o600.cloudfront.net/wp-content/uploads/2023/07/25111651/Cost-of-a-Data-Breach-Report-2023.pdf.
[5] L. Lamport, R. E. Shostak, and M. C. Pease. The Byzantine Generals Problem. ACM Transactions on Programming Languages and Systems, 4(3):382–401, 1982.

Confidential Stream Processing with WebAssembly in Trusted Execution Environments

Supervisors:
- Ali Mohammadpur-Fard <mohamal@usi.ch>
- Prof. Patrick Eugster <eugstp@usi.ch>

Stream processing systems like Apache Storm [1] and Apache Flink [2] are widely used for real-time data analytics, enabling various applications ranging from fraud detection to sensor data aggregation; however, deploying these systems in untrusted cloud environments raises significant confidentiality concerns as sensitive data is processed in plaintext on potentially compromised hosts.

Recent work in our group has been exploring integrating Trusted Execution Environments (TEEs) into a part of Apache Storm to enable confidential stream processing. A key challenge identified is executing user-defined processing logic (in Storm term, Spouts and Bolts) inside SGX enclaves. Current approaches (while the most flexible) incur substantial overhead in terms of cold-start latency and failure-case latency, as well as memory footprint and trust root (TCB) size.

This project will investigate an alternative approach; to allow the execution of WebAssembly (Wasm) [3] modules as the core handler of the user-defined stream processing functions, inside either a lightweight Wasm runtime or otherwise compiled to native ahead of time within an SGX enclave. Wasm offers several advantages for this use-case:

- Small TCB: Wasm runtimes can easily select which system APIs to expose, if any, and have significantly smaller footprints than a full JVM.
- Instantiation speed: Wasm modules are designed to be fast to load, so they can be loaded and ready to be executed with minimal initial overhead.
- Language flexibility: The functions can be written in Rust, C, C++ or many other higher-level languages (even recently, Python [4]).
- Safety: Unlike Java, Wasm is sandboxed by default, and all runtime system requirements are explicitly stated.

The student will design and implement a prototype system where Storm executors delegate computation to Wasm blobs running inside an SGX enclave, evaluate performance and usability against the current Gramine-based JVM approach, and asses the tradeoffs in terms of expressiveness, developer experience, overall system performance, and security guarantees.

[1] Apache Storm. https://storm.apache.org
[2] Apache Flink. https://flink.apache.org
[3] A. Haas, Bringing the web up to speed with WebAssembly, PLDI'17
[4] py2wasm, https://wasmer.io/posts/py2wasm-a-python-to-wasm-compiler

GenAI for Hardware Programming?

Supervisors:
- Prof. Patrick Eugster <eugstp@usi.ch>

The advent of generative AI ("GenAI") has ushered in a new era. GenAI is being used for a variety of tasks previously accomplished entirely by humans. In particular, GenAI is slowly taking over the software engineering task, with the number, scope, and quality of tools for generating code continuously increasing. GenAI is expected by many to make entire branches of software engineering obsolete, while others believe that humans will always be required to ensure quality.

Thus far, most attempts of exploiting GenAI for code generation, and corresponding studies to assess quality of the generated code, consider code written in languages at a high level of abstraction like Java or Python [1][2], at the application layer, or recently build code [3]. Many of the respective programming tasks investigated thus tend to be more concerned with high-level design principles and code quality metrics, in comparison to those addressed by code for ``lower layers'' in the software stack that is closer to the hardware and thus highly platform-dependent and yet more performance-critical. Moreover GenAI is strongly depending on existing code and solutions, which intuitively makes it less suitable for generating code for novel hardware platforms or low-level abstractions for which there are no broad code bases available yet.

While we surmise that GenAI will not perform well for generating low-level code notably for programming novel hardware, the goal of this project is to more neutrally and fundamentally investigate the limits and limitations of GenAI in that context. The candidate will first familiarize themself with tools as well as existing studies on quality of code produced by GenAI. Then the candidate will carefully define precise hypotheses to be tested and corresponding analyzes to be performed, using GenAI to generate code for particular scenarios and assessing its quality and/or conducting meta-analyses by distilling information from existing studies.

[1] Y. Liu, T. Le-Cong, R. Widyasari, C. Tantithamthavorn, L. Li, X.-B. D. Le, and D. Lo. Refining ChatGPT-Generated Code: Characterizing and Mitigating Code Quality Issues. ACM Transactions on Software Engineering Methodologies. 33, 5, Article 116, June 2024.

[2] M. L. Siddiq, L. Roney, J. Zhang, and J. C. S. Santos. Quality Assessment of ChatGPT Generated Code and their Use by Developers. 2024 IEEE/ACM 21st International Conference on Mining Software Repositories (MSR '24), 2024, pp. 152-156.

[3] A. Ghammam and M. Almukhtar. AI builds, We Analyze: An Empirical Study of AI-Generated Build Code Quality. arXiv eprint 2601.16839, 2026. https://arxiv.org/abs/2601.16839.