Towards Higher Order Mutation Testing for Deep Learning Systems

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

Deep Learning (DL) components are getting increasingly integrated into software systems to automatically perform complex tasks in a human-competitive way. With the growing rate of DL systems in various areas of life, their quality assurance becomes a task of immense importance.

Mutation Testing

Mutation testing is a technique that deliberately seeds faults in form of small syntactic changes into the program under test to create a set of faulty programs called mutants. The general principle underlying this approach is the assumption that faults used by mutation testing represent the mistakes that programmers usually make. Mutation testing aims to assess the quality of a given test suite in terms of its capability to detect faults. For this, the test suite is executed on each of the generated mutants. If the result for a given mutant is different from the result of running the original program then the mutation is considered killed. The ratio of killed mutants to the overall number of generated mutants is called mutation score. The higher the mutation score, the better is the quality of the test suite.

The example in Figure 1 shows a method `subtract` that subtracts two integer values and returns the result. It has two mutations: in `Mutant 1` the subtraction is replaced with multiplication and in `Mutant 2` it is replaced with addition. If our test suite has only `test0`, none of the two mutations would be killed (as they all return the expected value `0`) and the mutation score is 0%. If we add test case `test1()` to our test suite, then `Mutant 2` gets killed and the mutation score becomes 50%. Once we add test case `test2()`, both mutations get killed and the mutation score achieves its maximum value of 100%.

Mutation Testing for DL Systems

In traditional software systems the decision logic is often implemented by software developers in the form of source code. In contrast, the behaviour of a DL system is mostly determined by the training data set and the training program, i.e. these are the two major sources of defects for DL systems. Thus, there should be a specific approach to mutation testing of DL systems. There currently exist two tools that are designed specifically for performing mutation testing for DL systems. However, one of the tools is a pre-training one, which means it injects the faults into system prior to the training and thus is computationally expensive, while the second one, a post-training mutation tool, injects faults that are random and not very likely to happen in real world. Such faults usually introduce slight noise or modifications to a randomly selected subset of weights or change a structure of an already trained DL model by adding/deleting its layers or replacing the activation function.

```
// Original Program
public int subtract(int a, int b) {
    return a - b;
}

public void test0() {
    assertEquals(0, subtract(0, 0));
}
```

```
// Mutant 1
public int subtract(int a, int b) {
    return a * b;
}

public void test1() {
    assertEquals(-4, subtract(-2, 2));
}
```

```
// Mutant 2
public int subtract(int a, int b) {
    return a + b;
}

public void test2() {
    assertEquals(1, subtract(2,1));
}
```

Figure 1: Mutation Testing Example

Project Proposal

There exists a mutation testing tool called DeepCrime[1], which is designed to perform mutation testing on DL systems and is based on real DL-specific faults. However, it injects faults into a DL system prior to the training following a realistic fault injection scenario and thus is computationally expensive. The goal of the project is to explore the behaviour of higher order DL mutants. In particular, the idea is to generate mutants by simultaneously injecting multiple faults to a DL system under test and to analyse whether first order mutants are redundant w.r.t. the more complex ones. The adoption of higher order mutants can potentially reduce costs associated with mutation testing and create novel, interesting patterns of faulty model behaviour.

In the frame of this project, the student will learn about state-of-the-art techniques in the domain of mutation testing for DL systems, their limitations and advantages. The student will practice with popular DL frameworks and widely-used models and datasets.

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.

[1] Humbatova, Nargiz, Gunel Jahangirova, and Paolo Tonella. "Deepcrime: mutation testing of deep learning systems based on real faults." https://dl.acm.org/doi/abs/10.1145/3460319.3464825

Large Language Model as a Mutation Testing tool for Deep Learning Systems developed in PyTorch

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

Deep Learning (DL) components are getting increasingly integrated into software systems to automatically perform complex tasks in a human-competitive way. With the growing rate of DL systems in various areas of life, their quality assurance becomes a task of immense importance.

Mutation Testing

Mutation testing is a technique that deliberately seeds faults in form of small syntactic changes into the program under test to create a set of faulty programs called mutants. The general principle underlying this approach is the assumption that faults used by mutation testing represent the mistakes that programmers usually make. Mutation testing aims to assess the quality of a given test suite in terms of its capability to detect faults. For this, the test suite is executed on each of the generated mutants. If the result for a given mutant is different from the result of running the original program then the mutation is considered killed. The ratio of killed mutants to the overall number of generated mutants is called mutation score. The higher the mutation score, the better is the quality of the test suite.

The example in Figure 1 shows a method `subtract` that subtracts two integer values and returns the result. It has two mutations: in `Mutant 1` the subtraction is replaced with multiplication and in `Mutant 2` it is replaced with addition. If our test suite has only `test0`, none of the two mutations would be killed (as they all return the expected value `0`) and the mutation score is 0%. If we add test case `test1()` to our test suite, then `Mutant 2` gets killed and the mutation score becomes 50%. Once we add test case `test2()`, both mutations get killed and the mutation score achieves its maximum value of 100%.

```
// Original Program
public int subtract(int a, int b) {
   return a - b;
}

public void test0() {
   assertEquals(0, subtract(0, 0));
}
```

```
// Mutant 1
public int subtract(int a, int b) {
   return a * b;
}

public void test1() {
   assertEquals(-4, subtract(-2, 2));
}
```

```
// Mutant 2
public int subtract(int a, int b) {
   return a + b;
}

public void test2() {
   assertEquals(1, subtract(2,1));
}
```

Figure 1: Mutation Testing Example

Mutation Testing for DL Systems

In traditional software systems the decision logic is often implemented by software developers in the form of source code. In contrast, the behaviour of a DL system is mostly determined by the training data set and the training program, i.e. these are the two major sources of defects for DL systems. Thus, there should be a specific approach to mutation testing of DL systems. There currently exist two tools that are designed specifically for performing mutation testing for DL systems. However, one of the tools is a pre-training one, which means it injects the faults into system prior to the training and thus is computationally expensive, while the second one, a post-training mutation tool, injects faults that are random and not very likely to happen in real world. Such faults usually introduce slight noise or modifications to a randomly selected subset of weights or change a structure of an already trained DL model by adding/deleting its layers or replacing the activation function.

Project Proposal

There exists a mutation testing tool called DeepCrime[1], which is designed to perform mutation testing on DL systems and is based on real DL-specific faults. However, DeepCrime is only applicable to programs implemented using Keras DL framework. The goal of the project is to implement a DeepCrime-like mutation tool for DL systems developed in PyTorch. Unlike DeepCrime, the new tool will employ a Large Language Model (LLM) as a backend to efficiently inject various faults into the source code of a PyTorch program.

In the frame of this project, the student will learn about state-of-the-art techniques in the domain of mutation testing for DL systems, their limitations and advantages. The student will practice with popular DL frameworks and widely-used models and datasets.

The only requirements would be the knowledge of Python programming language and a motivation to learn more and contribute to the rapidly developing area of DL testing. Familiarity with PyTorch and Keras is welcome but not mandatory.

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.

[1] Humbatova, Nargiz, Gunel Jahangirova, and Paolo Tonella. "Deepcrime: mutation testing of deep learning systems based on real faults." https://dl.acm.org/doi/abs/10.1145/3460319.3464825

Spectral Analysis of Neural Activation Values on Failure-Inducing Inputs

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

Deep Learning (DL) components are getting increasingly integrated into software systems to automatically perform complex tasks in a human-competitive way. With the growing rate of DL systems in various areas of life, their quality assurance becomes a task of immense importance. This is especially true when misbehaviours in such systems have the potential to negatively affect safety, ethics or business critical activities.

Test Adequacy and Input Prioritisation

The testing of DNN-based software differs significantly from conventional software testing. Unlike conventional software, which relies on programmers to manually construct its logic, DNNs are built using a data-driven programming paradigm. As a result, having an ample supply of test data, along with oracle information, becomes crucial for identifying potential misbehaviours in DNN-based software. Obtaining appropriate test data for DL systems presents challenges, such as the complex and labour-intensive labelling process that often requires specialised domain knowledge, addressing bias and ensuring fairness, privacy concerns that limit access to sensitive data.

To address these challenges, a number of approaches were proposed to prioritise test inputs for the labelling by the likeliness of misbehaviour detection. Activation values of DL model's neurons have been used in the test adequacy criteria, such as neuron coverage (NC), k-multisection neuron coverage (KMNC) and surprise adequacy (SA). NC measures the number of neurons activated by a test set, where a neuron is considered activated if its output value is higher than a predefined threshold t. KMNC measures the coverage of k buckets into which neuron activation values are split. SA quantifies the novelty of each input with respect to the training data. Similarly to k-multisection neuron coverage, the surprise range is split into buckets that should be covered as much as possible by the test dataset. As a result, test inputs that get cover the range of neural activation better or are the most surprising when compared to training data are prioritised higher. However, these existing approaches either suffer from the effectiveness issue. More specifically, coverage-based test input prioritization, has been demonstrated to be not effective compared with confidence-based test input prioritization. The confidence/uncertainty based approaches prioritise an input higher if a DNN model outputsmore similar probabilities for all classes when classifying a test input, i.e. the confidence for classifying the test input is lower (or uncertainty is higher). This family of approaches has a limited application as it only considers classification systems. Therefore, the is a need for an effective, efficient and generalisable test input prioritisation approach.

Spectral Analysis. All the approaches based on the analysis of neural activations are based on the assumptions that the faults made during the development process of DL systems (such as low-quality training data, suboptimal model architecture or hyperparameter values, etc.) are reflected in specific patterns in the weights or activation values of the neural network. This poses a threat to the validity of the proposed approaches as there is no guarantee that reaching a high coverage of activation patterns correlates with high fault exposure. A recent empirical study assessed the effect of different fault types that typically affect DNN systems on the DNN activation patterns. It characterised the behaviour of a DNN quantified at the neuron activation levels, by introducing the notion of the spectrum of a deep neural network (DNN), defined as the probability distribution of the activation values of its neurons. The results of the study showed that there exists a relationship between spectra of a DNN and fault types that affect the model, which makes the spectra an excellent representation of the DNN's behaviour for the purposes of fault localisation.

Project Proposal

The goal of the project is to analyse the difference in spectra of a DNN obtained on different sets of inputd. The analysis of the changes in spectra obtained on the misbehaviour-revealing and successfully passing test inputs, can provide more insights on how the failure-inducing features of inputs affect the activations of the neurons. Combined with the analysis of whether similar inputs produce similar spectra, it would reveal the potential of the use of spectra to evaluate the diversity of inputs and their usefulness for the DNN testing, which are two important factors for effective test input prioritisation. Additionally, the scope of the project consider limiting the calculation to the set of neurons which are found to have the most influence on the final output of a DNN and evaluating the effectiveness of such an approach.

In the frame of this project, the student will learn about state-of-the-art techniques in the domain of test adequacy criteria and input prioritisation for DL systems, their limitations and advantages. The student will practice with most popular DL frameworks and widely-used models and datasets.

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.

A “not-so-empathic" chatbot

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. The issue is that conversational AI often lacks a robust value system for determining which emotions should be validated and which should be soothed. 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 result in the reinforcement of those harmful beliefs.

The aim of the project is to incorporate a context-aware ethical framework that assesses the appropriateness of emotional validation based on the moral implications of the topic. For example, if a user expresses excitement about an immoral action, the system should recognize this and offer a neutral or corrective response instead of amplifying the excitement.

Tasks:

- Build a conversational chatbot prototype (starting from an existing framework).
- Integrate emotion recognition models into the chatbot architecture.
- Integrate models that can assess the morality/immorality of the user’s utterances.
- Design adaptive response strategies based on detected emotions and the morality of the user’s utterances.

The ideal candidate should have an interest in building conversational systems. This project offers the opportunity to the candidate to prepare and submit a scientific publication at the end of the internship.

References:

- Curry, A. C., & Curry, A. C. (2023, July). Computer says “no”: The case against empathetic conversational AI. In Findings of the Association for Computational Linguistics: ACL 2023 (pp. 8123-8130).
- Hendrycks, D., Burns, C., Basart, S., Critch, A., Li, J., Song, D., & Steinhardt, J. Aligning AI With Shared Human Values. In International Conference on Learning Representations.
- Reinig, I., Becker, M., Rehbein, I., & Ponzetto, S. P. (2024, August). A Survey on Modelling Morality for Text Analysis. In Findings of the Association for Computational Linguistics ACL 2024 (pp. 4136-4155).

Empathic chatbot

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. For example, in response to a user’s emotion, an empathic response can show understanding of the user emotional status and convey feelings that support the user emotion and engage with the user to find a more positive view of the user’s problem.

In this project, we aim to test how different empathic responses lead to changes in the emotion of the user. We want to test if positive empathic responses can lead to an improvement in the emotion of the user and measure such effectiveness.

Tasks:

- Build a conversational chatbot prototype (starting from an existing framework).
- Integrate emotion recognition models into the chatbot architecture.
- Design responses based on the user’s emotion and have the chatbot modify the conversational strategy accordingly.
- Conduct user assessment through the Positive Affect Negative Affect Scale (PANAS) to measure the users’ affect before and after the interaction with the chatbot.

The ideal candidate should have an interest in building an emphatic conversational system. This project offers the opportunity to the candidate to prepare and submit a scientific publication at the end of the internship.

References:

- Tian, Z., Wang, Y., Song, Y., Zhang, C., Lee, D., Zhao, Y., ... & Zhang, N. L. (2022). Empathetic and Emotionally Positive Conversation Systems with an Emotion-specific Query-Response Memory. In Findings of the Association for Computational Linguistics: EMNLP 2022 (pp. 6364-6376).
- Watson, D., Clark, L. A., & Tellegen, A. (1988). Development and validation of brief measures of positive and negative affect: the PANAS scales. Journal of personality and social psychology, 54(6), 1063.

Adaptive Personality-Based Chatbot

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

Research indicates that individuals connect more effectively with others who share similar personality traits. Chatbots that can adapt to or imitate users' personalities, may enhance user experience and engagement. Personality-aware chatbots usually have a fixed personality from the start of the conversation.

This project aims to develop an innovative conversational AI system that can dynamically adapt its personality to align with the traits of individual users. Leveraging psychological principles, particularly the Big Five personality traits—Openness, Conscientiousness, Extroversion, Agreeableness, and Neuroticism—this chatbot will create a more engaging, relatable, and personalized user experience.

Tasks:

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

The ideal candidate should have an interest in building conversational systems. This project offers the opportunity to the candidate to prepare and submit a scientific publication at the end 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.

Program Analysis for Python

Motivation

Over the last two years more than half a million Python programs were executed within the more than 300 different programming activities available in the educational online programming environment developed in our research group. To improve the environment and to develop new tools and pedagogies that support students in their learning, we need answers to the many questions we have about this large collection of novice programmer code:

- What were the common syntax, name, type, and other errors made in the code?
- What were the common APIs used?
- Which Python language features were used how often?
- How often were type annotations used, where, and how?
- Were the variables used mutable or immutable?
- Were the functions defined pure or impure?
- …

Manually reading half a million programs to answer these questions is impractical. In this UROP project you are going to develop automated program analyses that can answer such questions.

Prerequisites

For this project, you need strong programming skills, good algorithmic thinking, and an interest in the foundations of programming languages.

More Information

If you are interested in this project, or in related projects available in our group, please contact Matthias.Hauswirth@usi.ch to discuss the details.

UROP 2025 project proposals, Prof. Langheinrich

Title: Design and Validation of a Privacy-Preserving Dashboard for Outputs in Employee Well-Being
Contact: Prof. Marc Langheinrich
Co-supervisors: Mohan Li, Daniil Kirilenko

The modern workplace increasingly relies on advanced machine learning methods, such as Federated Learning (FL), to derive insights into employee well-being and productivity while ensuring privacy and confidentiality. These methods enable decentralized data analysis, protecting sensitive information. However, there is a critical need for intuitive tools that allow users—both employees and employers—to interact with these insights while maintaining trust, transparency, and control. A well-designed dashboard can bridge this gap by providing actionable insights, allowing for the management of privacy preferences, and clarifying the reasoning behind model outputs through explainability methods.

The task involves designing and validating a user-centered dashboard for presenting FL model outputs related to well-being and productivity. The dashboard will include features for managing privacy preferences and generating intuitive, actionable visualizations and explanations of model results. The project requires basic computer science skills, including web development (e.g., React, Python), and an understanding of user interface design principles. The expected outcome is a functional, validated dashboard that enhances user trust and control while maintaining privacy, along with a research report documenting the design and validation process.

Title: Self-Supervised Learning for Smart Glasses Data
Contact: Prof. Marc Langheinrich
Co-supervisors: Francesco Bombassei De Bona, Martin Gjoreski

Affective computing is an interdisciplinary field that develops systems capable of recognizing, interpreting, and simulating human affect. A fundamental assumption is that different mental states (e.g., emotions and stress) manifest through physiological and behavioral changes. These changes can be captured through various wearable technologies, including smart glasses. Smart glasses offer a unique, unobtrusive platform for monitoring physiological signals like facial expressions [1].

Early affect-recognition systems relied on traditional machine learning (ML) methods paired with hand-crafted features. However, modern systems increasingly utilize deep learning, which can be further improved with techniques like self-supervised and unsupervised learning [2, 3]. These approaches have shown promise in domains like image-based ML and video analysis and are beginning to show potential with smart glasses in semi-controlled environments. The utility of smart glasses for affective computing in real-world settings is an active area of research.

This project will explore personalization and domain-adaptation techniques to address important challenges in wearable computing: noisy data, limited data, and domain shifts in the labels and the sensor data due to subjectivity. Existing processing pipelines and the deep learning architectures will be augmented with the latest unsupervised and/or self-supervised learning techniques. These advanced techniques should produce more robust and data-efficient models (i.e., requiring fewer person-specific labels). Domain adaptation will be explored within datasets (e.g., from one user to another) and across datasets and devices.

Literature:

1. Kiprijanovska, I., Stankoski, S., Broulidakis, M. J., Archer, J., Fatoorechi, M., Gjoreski, M., ... & Gjoreski, H. (2023). Towards smart glasses for facial expression recognition using OMG and machine learning. Scientific Reports, 13(1), 16043.
2. Meegahapola, L., Hassoune, H., & Gatica-Perez, D. (2024). M3BAT: Unsupervised Domain Adaptation for Multimodal Mobile Sensing with Multi-Branch Adversarial Training. Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies, 8(2), 1-30.
3. Meegahapola, L., Hassoune, H., & Gatica-Perez, D. (2024). M3BAT: Unsupervised Domain Adaptation for Multimodal Mobile Sensing with Multi-Branch Adversarial Training. Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies, 8(2), 1-30.

Conditionally Positive Definite Kernels in Approximation

Supervisors: S. Avesani, M. Multerer

Abstract

The goal of this project is to extend the `FMCA`[1] library by integrating theory and numerical methods related to conditionally positive definite (CPD) kernels. We will investigate their matrix representations and employ Uzawa's method in solving saddle point problems arising from kernel-based formulations.

Project Description

A kernel function k is said to be conditionally positive definite of order m if, for any finite set of distinct points $\{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ in the domain and any set of coefficients $\{\alpha_i\}$ satisfying

$$\sum_{i=1}^{n} \alpha_i \, p(\mathbf{x}_i) = 0 \quad \text{for all polynomials } p \text{ of degree } \leq m-1, \tag{1}$$

we have

$$\sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \, \alpha_j \, k(\mathbf{x}_i, \mathbf{x}_j) \geq 0.$$

These kernels are crucial in meshless approximation methods because they allow polynomial terms to be enforced as constraints, ensuring polynomials reproduction.

We focus on solving the interpolation problem with polynomial constraints (1). The interpolation system reads as follows:

$$\begin{pmatrix} K & P \\ P^T & 0 \end{pmatrix} \begin{pmatrix} \boldsymbol{\alpha} \\ \boldsymbol{\beta} \end{pmatrix} = \begin{pmatrix} \mathbf{f} \\ \mathbf{0} \end{pmatrix}, \tag{2}$$

where $K = \big[k(\mathbf{x}_i, \mathbf{x}_j)\big]_{1 \leq i,j \leq n}$ is the CPD kernel matrix, P encodes polynomial constraints, $\boldsymbol{\alpha}$ contains the main interpolation coefficients, $\boldsymbol{\beta}$ enforces the polynomial conditions and $\mathbf{f}$ is the input data. The zero block at the bottom-right reflects the saddle point nature of this system.

Solving saddle point systems as (2) directly can be challenging, as these matrices are not strictly positive definite. *Uzawa's method* provides an iterative algorithm that updates the primal and dual variables in separate steps, which can be adapted to exploit the block structure efficiently. By designing specialized preconditioners (e.g., block diagonal or approximate Schur complement), we can achieve stable and efficient convergence.

Proposed Tasks

The student will:

- Investigate the theoretical properties of CPD kernels, focusing on polynomial constraints and their role in numerical stability.
- Extend the `FMCA` library to handle matrix assembly for CPD kernels, explicitly forming the augmented system that captures polynomial reproduction constraints.
- Implement Uzawa's method to solve the resulting saddle point problems, integrating suitable preconditioners for improved performance.
- Validate the methods on benchmark examples, comparing results with those obtained from strictly positive definite kernels or other relevant approaches.

[1] `https://github.com/muchip/fmca`

Università
della
Svizzera
italiana

Faculty
of Informatics

Foundation Model for Electrodermal Activity Data

UROP project proposal at the Università della Svizzera Italiana (USI), Lugano, Switzerland

Background

Large foundations models are machine or deep learning models trained on a vast and diverse range of datasets, usually unlabeled and from one or multiple modalities, and can be applied to a wide range of tasks (Bommasani et al., 2021). Foundation models have revolutionized various fields, from natural language processing (Naveed et al., 2023), e.g., ChatGPT, to images (Alom et al., 2018; Dosovitskiy, 2020). Recently, researchers have created foundation models using physiological data (Abbaspourazad et al., 2023; Narayanswamy et al., 2024). Physiological data refers to the measure of biological or physical traits in individuals, such as heart rate, blood pressure, body temperature, and many others. Physiological data is used in a wide range of applications, from the medical domain, e.g., (Orphanidou, 2019), to commercial wearable applications, e.g.,(Gopala et al., 2018). Wearable devices can be equipped with various sensors that record physiological and behavioural data, e.g., accelerometer. One of the most common is the Photoplethysmogram (PPG) sensor, which shines light, at specific wavelengths, through the skin to measure changes in blood volume pulse, used to estimate, among others, the heart rate (Yuan et al., 2024; Dhekane and Ploetz, 2024). Due to its abundance on commercial devices, recent research has shown that foundation models can be trained on accelerometer and PPG data (Haresamudram et al., 2020; Abbaspourazad et al., 2023). For example, Abbaspourazad et al. (2023) show that, using a large amount of PPG data, foundation models can be trained and used successfully on a wide range of applications, e.g., age estimation.

However, less common sensors are not as widely studied as PPG or accelerometer. For example, Electrodermal Activity (EDA) is still uncommon in commercially available devices, with Fitbit Sense 2 being the only one with the sensor available commercially[1]. EDA measures the conductivity of the skin, which is used as a proxy for autonomic nervous system (ANS) arousal (Boucsein, 2012). EDA data collected from wearable devices have been used to detect, for example, laughter episodes (Di Lascio et al., 2019), cognitive load (Alchieri et al., 2024), stress (Liu and Du, 2018), and many others. Some preliminary work exists on exploring EDA data to train large foundation models. Researchers have used one component of EDA, called Skin Conductance Level (SCL) , in conjunction with other physiological markers, e.g., heart rate, to train a foundation model (Narayanswamy et al., 2024). However, SCL is only one component of the EDA signal and, since the approach from Narayanswamy et al. (2024) is multimodal, it is not clear the contribution of EDA compared to the other modalities, e.g., heart rate.

With this work, our aim is to explore the use of large foundation models on EDA data only. Even if there are limited datasets that contain, alone, a big enough number of EDA data that would allow to train such a model, there exists a wide range of smaller, specialized datasets that contain EDA data. In this project, we have two objectives: categorize datasets, available to researchers, that contains EDA data; and train a large foundation model using these data.

[1] https://store.google.com/us/product/fitbit_sense_2?hl=en-US

Expected Outcomes

The expected outcomes of this project are as follows:

1. Review of papers provided by the advisor and co-advisor, which provide an understanding of the topics for this work, e.g., similar works like (Abbaspourazad et al., 2023).

2. Categorize datasets that contain EDA data, according to parameters provided by the advisor and co-advisor. These datasets would be both publicly available as well as datasets collected by our research group.

3. Implement a unified pre-processing pipeline to all applicable datasets, according to the review and categorization performed. The pre-processing will use libraries already developed in our research group, and that are publicly available.

4. Finally, the student will implement a large foundation model, mostly based on contrastive loss as in (Abbaspourazad et al., 2023), using the processed EDA data. The student will try various model architectures and compare how different parameters affect the performance of this model, e.g., data availability.

5. Write a project report, with a summary of the work performed and the findings.

Supervisors and contact information

Prof. Dr. Silvia Santini, silvia.santini@usi.ch

Leonardo Alchieri, leonardo.alchieri@usi.ch

References

Abbaspourazad S, Elachqar O, Miller AC, Emrani S, Nallasamy U, Shapiro I (2023) Large-scale training of foundation models for wearable biosignals. arXiv preprint arXiv:231205409

Alchieri L, Abdalazim N, Alecci L, Gashi S, Gjoreski M, Santini S (2024) Lateralization effects in electrodermal activity data collected using wearable devices. Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies 8(1):1–30

Alom MZ, Taha TM, Yakopcic C, Westberg S, Sidike P, Nasrin MS, Van Esesn BC, Awwal AAS, Asari VK (2018) The history began from alexnet: A comprehensive survey on deep learning approaches. arXiv preprint arXiv:180301164

Bommasani R, Hudson DA, Adeli E, Altman R, Arora S, von Arx S, Bernstein MS, Bohg J, Bosselut A, Brunskill E, et al. (2021) On the opportunities and risks of foundation models. arXiv preprint arXiv:210807258

Boucsein W (2012) Electrodermal activity. Springer Science & Business Media

Dhekane SG, Ploetz T (2024) Transfer learning in human activity recognition: A survey. arXiv preprint arXiv:240110185

Di Lascio E, Gashi S, Santini S (2019) Laughter recognition using non-invasive wearable devices. In: Proceedings of the 13th EAI International Conference on Pervasive Computing Technologies for Healthcare, pp 262–271

Dosovitskiy A (2020) An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:201011929

Gopala K, Varatharajan R, Manogaran G, Priyan MK, Sundarasekar R (2018) Wearable technology applications in healthcare: A literature review. HIMSS Greater Kansas City Chapter URL https://gkc.himss.org/resources/wearable-technology-applications-healthcare-literature-review

Haresamudram H, Beedu A, Agrawal V, Grady PL, Essa I, Hoffman J, Plötz T (2020) Masked reconstruction based self-supervision for human activity recognition. In: Proceedings of the 2020 ACM International Symposium on Wearable Computers, pp 45–49

Liu Y, Du S (2018) Psychological stress level detection based on electrodermal activity. Behavioural brain research 341:50–53

Narayanswamy G, Liu X, Ayush K, Yang Y, Xu X, Liao S, Garrison J, Tailor S, Sunshine J, Liu Y, et al. (2024) Scaling wearable foundation models. arXiv preprint arXiv:241013638

Naveed H, Khan AU, Qiu S, Saqib M, Anwar S, Usman M, Akhtar N, Barnes N, Mian A (2023) A comprehensive overview of large language models. arXiv preprint arXiv:230706435

Orphanidou C (2019) A review of big data applications of physiological signal data. Biophysical Reviews 11(1):83–87

Yuan H, Chan S, Creagh AP, Tong C, Acquah A, Clifton DA, Doherty A (2024) Self-supervised learning for human activity recognition using 700,000 person-days of wearable data. NPJ digital medicine 7(1):91

Accelerating Network Computing for Large-Scale Applications Using General-Purpose Accelerators

Supervisors:
- Dr Sina Darabi <darabs@usi.ch>
- Prof. Patrick Eugster <eugstp@usi.ch>

This project aims to significantly enhance the performance of distributed applications in rapidly evolving fields such as machine learning (ML) and bioinformatics. As these applications grow in complexity and scale, traditional network infrastructures face increasing challenges, resulting in bottlenecks that hinder performance and increase latency. This project seeks to address these limitations by leveraging general-purpose accelerators, particularly GPUs, to offload and accelerate specific computational tasks. By doing so, the project aspires to improve the overall efficiency of network computing, enabling faster data processing and analysis in large-scale environments.

Goals & Scope

To achieve this, the project will begin with a literature review [1,2,3] that explores recent advancements in network computing, specifically focusing on the performance implications for large-scale data analysis tasks. An analysis of current network infrastructure will identify existing bottlenecks and limitations in processing power, particularly in routers. By understanding these constraints, the project will be better positioned to propose effective solutions that utilize GPU acceleration to alleviate network congestion and enhance processing speed.

The implementation phase will involve developing a framework that integrates GPU acceleration into network devices. This will include identifying tasks that are well-suited for GPU offloading, such as data preprocessing and feature extraction, and designing a system that can execute these tasks in a parallelized manner. Performance evaluation will be a key aspect of this phase, with metrics such as latency reduction, overall throughput, and resource utilization being measured against traditional processing methods. By benchmarking the framework with both synthetic and real-world datasets, the project will provide valuable insights into the practical benefits of this approach.

Expected Outcomes

Ultimately, this project aims to demonstrate the feasibility and effectiveness of using GPUs within network devices to enhance network computing for large-scale applications. By showcasing significant performance improvements in terms of latency and throughput, the work will contribute to the growing body of knowledge surrounding accelerator use in distributed network computing. This research has the potential to inform future developments in the field, paving the way for more efficient and responsive solutions that can meet the evolving demands of complex applications in ML and bioinformatics.

References

[1] https://dl.acm.org/doi/abs/10.1145/3132747.3132764
[2] https://doi.org/10.1145/3318464.3389698
[3] https://www.usenix.org/conference/nsdi21/presentation/sapio

Design/Implementation of Secure Algorithms in a New IFC+DP Language

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

Traditional systems struggle with enforcing fine-grained security guarantees while supporting practical declassification of sensitive data in a secure manner [1, 2]. We are developing a new programming language integrating information-flow control (IFC) with differential privacy (DP)-aware budgeted declassification [3, 4], ensuring security while allowing controlled, quantifiable information release.

Goals & Scope

This project focuses on developing and evaluating algorithms that leverage the language's IFC and DP capabilities while also contributing to its design and implementation. The primary goal is to explore how well the language supports real-world privacy-sensitive applications while identifying areas where its type system, static analysis, and runtime support can be improved. Possible options include:

1. Algorithm Development: Design and implement DP-secure algorithms in domains such as multi-level security (MLS) systems, governmental and financial data handling, and selective access control in healthcare and intelligence-sharing scenarios.
2. Applying IFC & DP Mechanisms: Explore how IFC and DP principles impact algorithm behavior, ensuring effective enforcement of security policies while enabling controlled data sharing [2, 4].
3. Language & Tooling Enhancements: Identify and address any limitations in the language's type system, static analysis, and compilation pipeline to better support secure algorithm development.
4. Validating Language Usability: Evaluate how expressive and practical the language is for writing privacy-aware algorithms, refining its design based on real-world use cases [2, 3].

Expected Outcomes

- A set of implemented algorithms demonstrating the language's security guarantees in real-world scenarios.
- Insights into the practical use of IFC and DP mechanisms for privacy-aware computations.
- Contributions to refining the language's type system, analysis capabilities, and usability to support algorithm development more effectively.

The student will gain experience in privacy-preserving algorithms, security-aware programming, and programming language design, including type systems and static analysis.

References

[1] apacheranger [n. d.]. Apache Ranger. https://ranger.apache.org/.

[2] Anindya Banerjee, Roberto Giacobazzi, and Isabella Mastroeni. 2007. What You Lose is What You Leak: Information Leakage in Declassification Policies. Electronic Notes in Theoretical Computer Science 173 (April 2007), 47–66. https://doi.org/10.1016/j.entcs.2007.02.027

[3] Gilles Barthe, Boris Köpf, Federico Olmedo, and Santiago Zanella-Béguelin. 2013. Probabilistic Relational Reasoning for Differential Privacy. ACM Transactions on Programming Languages and Systems 35, 3 (Nov. 2013), 1–49. https://doi.org/10.1145/2492061

[4] David Elliot Bell and Leonard J. LaPadula. 1973. Secure Computer Systems: Mathematical Foundations and Model. Technical Report MTR-2547. MITRE Corp.

Data Science for Causal Networks

Contact: Ernst C. Wit

Co-supervisors: Martina Boschi, Melania Lembo, Francisco Richter

Description

Relational event models provide an efficient framework for describing the effect of drivers on the dynamics of temporal networks, capturing how interactions evolve over time. Data used to fit these models have always been purely observational. Although sociologists have always been interested in testing hypothesis-driven effects, the observational experimental designs have limited direct causal interpretation of the results. This UROP project aims to advance the field by exploring explicit causal implementations of relational event models, leveraging the concept of invariance causal prediction.

Project tasks

- Literature review on invariant prediction and relational event models
- Implement a fast causal testing procedure in Python or C.
- Apply causal testing procedure to real world networks

Literature

- Polinelli, Alice, Veronica Vinciotti, and Ernst C. Wit. "Generalised Causal Dantzig." *arXiv preprint arXiv:2407.16786* (2024).
- Bianchi, Federica, et al. "Relational event modeling." *Annual Review of Statistics and Its Application* 11 (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)

SAT-based techniques for Approximate Circuit Design

Professor
Laura Pozzi

Abstract

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! Drop me an email if you are interested or if you want to have more information.
Useful links:

An introduction to SAT: https://www.borealisai.com/en/blog/tutorial-9-sat-solvers-i-introduction-and-applications/

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

A survey of approximate circuit design
techniques: https://www.inf.usi.ch/phd/scarabottolo/papers/ALS_survey.pdf

Studying the Relationship Between Genome and Disease

Professor
Laura Pozzi

Abstract

Genome Wide Association Studies (GWAS) aim at comparing the genome of several individuals affected by a certain trait or disease, in order to analyze their similarity and hence potentially uncover the genes correlated to such disease.
The human genome consists of 3 billion base pairs, and is 99.9% identical among any non-related pairs of humans. The remaining 0.1% (approximately 5 million base pairs) differs, and hence makes up the difference among all individuals of our species. The genome positions at which these differences are found are called SNPs (single nucleotide polymorphisms).
A GWAS, then, analyses the SNPs of individuals affected by a disease, and reports the SNPs that are different from the "norm", but in common among the individuals under test.
Computer scientists aim at abstracting away the biological details pertaining to these studies, while still trying to understand the algorithms underlying these natural processes.
In this project, we will start with the following tutorial to collect and then to process GWAS data,
https://pmc.ncbi.nlm.nih.gov/articles/PMC6001694/
https://github.com/MareesAT/GWA_tutorial/
and we will then use Satisfiability (SAT) techniques
https://www.borealisai.com/en/blog/tutorial-9-sat-solvers-i-introduction-and-applications/
and Python Programming in order to study aspects of the relationship between genome and disease.