Master’s Project Market · FS 2021

Nathan Labhart
Academic Coordinator

1. Dario Stähelin
2. Bruno Rodrigues
3. Christian Killer
4. Jürgen Bernard
5. Chat Wacharamanotham
6. Dhivya Ramasamy
7. Enrico Fregnan
8. Dan Olteanu
9. Ahmet Kara
Rules: The Master’s Project…

… is a **group project**: at least 2 members needed.
   → Chance of denial for individual projects: 99%

… can only be started **after** all members have successfully completed their Master’s Basic Module (only for Major).
   → Best time: During semester break; **max. 1 year** to complete.

… must be done **with an IfI professor**.

… yields 18 ECTS Credits.
Master’s Project: Procedure

1. Find a project (e.g., here at the Master’s Project Market, on the IfI website for MSc [https://www.ifi.uzh.ch/en/studies/msc-info.html](https://www.ifi.uzh.ch/en/studies/msc-info.html), in OLAT [http://t.uzh.ch/yi](http://t.uzh.ch/yi), …)

2. Build groups (find peers here, on Gather, in OLAT, …)

3. Meet with supervisor and submit the application form.

4. Start.
Master’s Project Market: Procedure

1. Listen to today’s presentations
2. Exit this Zoom meeting
3. Find peers in Gather: http://tiny.cc/master-project-FS2021
4. Join the Zoom meeting(s) of the project(s) you want to discuss
Master’s Project presentations (max. 2 minutes each):

1. Dario Stähelin: Tackling a health epidemic with an e-health application in Lesotho
2. Bruno Rodrigues: Data Correlation and Analysis of Multi-source Wireless Trackers
5. Chat Wacharamanotham:
   - Effect size ruler: A literature research and statistical meta-analysis on effect sizes for text-entry studies
   - Mona gaza: An online experiment on the influence of lens distortion on gaze perception in remote collaboration
7. Enrico Fregnan:
   - Identifying prominent classes in code review
   - Automatic classification of functional review changes
8. Dan Olteanu (video):
   - Middleware for fast incremental view maintenance of ML models
   - Understanding and implementing feature engineering for Kaggle competitions
Community-based Chronic Disease Care Lesotho

Tackling a health epidemic with an e-health application in Lesotho

Master’s Project Market
31.03.2021
The Situation

- Health burden of non-communicable diseases overtakes infectious diseases (e.g. HIV)
- Long travels to hospitals
- 5 doctors, 62 nurses per 100’000 people
- Village health workers provide primary care in communities

Photo credit: Alain Amstutz
The Task

• Review existing eHealth apps & conversational agents (i.e., chatbots)
• Identify needs and requirements of the users
• Develop a working prototype
• Start: June 2021
• Prototype: September 2021
• Duration: 6 Months, part-time
The Team: Native android experience wanted

Natalie Hennig

Louis Bienz

Kevin Kindler
(Bachelor’s Thesis)

Nitharsan Yoganathan

Patrick Luchsinger

Insert name here
The Peers

Swiss TPH – Project Lead & Medical Knowledge

Tech4All – Product Owner & Tech Partner

NUL – Student Collaboration

Ministry of Health – Government Stakeholder & EMR provider
The Perks

• Participate in a development cooperation project between Switzerland and Lesotho
• Potentially travel to Lesotho (if save)
• Cross-cultural and interdisciplinary collaboration
• Development in an resource restricted context (e.g., no stable internet connection)
• Very cool team
An Engine for Analysis and Correlation of a Multi-source Tracking Data

**Problem**
How to gain knowledge on an audience to improve (the delivery of) products and services in a live setting?

**Goal**
Explore the different characteristics of multiple (passive) sources to increase the overall tracking accuracy.

**Approach**
Temporal and spatial data correlation of multiple sensors

Contact: Bruno Rodrigues <rodrigues@ifi.uzh.ch>
Provotum: A Decentralized Voting System

Long-Term Privacy
Quantum-Safe Cryptography

Remote Postal Voting
End-to-End Verifiability

Voting Protocols
Applied Cryptography

Distributed Ledgers
Permissioned Consensus, Network Security

Security & Privacy
Sidechannel Attacks, Voter Coercion

Contact: Christian Killer <killer@ifi.uzh.ch>
Visual Analysis of Complex Event Sequences

Master's Project

Jürgen Bernard
Interactive Visual Data Analysis Group

University of Zurich
Event Sequences (ES)

Time-stamped events

Analysis challenge 1:
- sequence may be long
Analysis challenge 2: - many sequences may exist
Is this relevant?

- **Software engineering** (commit behavior in repositories)
Is this relevant?

- **Software engineering** (commit behavior in repositories)
- **Medical histories** (heart beats, medication, sleeping, etc.)
Is this relevant?

- **Software engineering** (commit behavior in repositories)
- **Medical histories** (heart beats, medication, sleeping, etc.)
- **Geological phenomena** (earthquake aftershocks, etc.)
Is this relevant?

- **Software engineering** (commit behavior in repositories)
- **Medical histories** (heart beats, medication, sleeping, etc.)
- **Geological phenomena** (earthquake aftershocks, etc.)
- **Personal workouts** (dates of exercises)
Is this relevant?

- **Software engineering** (commit behavior in repositories)
- **Medical histories** (heart beats, medication, sleeping, etc.)
- **Geological phenomena** (earthquake aftershocks, etc.)
- **Personal workouts** (dates of exercises)
- **Internet of things logs** (door open events)
Is this relevant?

- **Software engineering** (commit behavior in repositories)
- **Medical histories** (heart beats, medication, sleeping, etc.)
- **Geological phenomena** (earthquake aftershocks, etc.)
- **Personal workouts** (dates of exercises)
- **Internet of things logs** (door open events)
- **Earth observation** (thunderstorms, cloudiness, etc.)
Is this relevant?

- **Software engineering** (commit behavior in repositories)
- **Medical histories** (heart beats, medication, sleeping, etc.)
- **Geological phenomena** (earthquake aftershocks, etc.)
- **Personal workouts** (dates of exercises)
- **Internet of things logs** (door open events)
- **Earth observation** (thunderstorms, cloudiness, etc.)
- **Finance** (days with rising stock prices, crashes)
Is this relevant?

- **Software engineering** (commit behavior in repositories)
- **Medical histories** (heart beats, medication, sleeping, etc.)
- **Geological phenomena** (earthquake aftershocks, etc.)
- **Personal workouts** (dates of exercises)
- **Internet of things logs** (door open events)
- **Earth observation** (thunderstorms, cloudiness, etc.)
- **Finance** (days with rising stock prices, crashes)
- **Social media streams** (tweets)
Is this relevant?

- **Software engineering** (commit behavior in repositories)
- **Medical histories** (heart beats, medication, sleeping, etc.)
- **Geological phenomena** (earthquake aftershocks, etc.)
- **Personal workouts** (dates of exercises)
- **Internet of things**
- **Earth observation**
- **Finance** (days with rich transactions)
- **Social media streams** (tweets)
- **Communication** (telephone call or Email protocols)
Is this relevant?

- **Software engineering** (commit behavior in repositories)
- **Medical histories** (heart beats, medication, sleeping, etc.)
- **Geological phenomena** (earthquake aftershocks, etc.)
- **Personal workouts** (dates of exercises)
- **Internet of things logs** (door open events)
- **Earth observation** (thunderstorms, cloudiness, etc.)
- **Finance** (days with rising stock prices, crashes)
- **Social media streams** (tweets)
- **Communication** (telephone call or Email protocols)
Vision:
Combining the strengths of humans and machines!

Algorithmic support is needed
- Very good if data analysis problem can be solved automatically

Human supervision is needed
- Very good if data analysis problem can NOT be solved automatically
Analysis approach

Frontend
- Data visualization
- User interaction

Backend
- Data processing
- Machine learning
- Statistics

Visual Representation
Feedback Loop
Analysis approach

Key techniques

- **Metrics and Features**: to characterize event sequences
- **Motif simplification**: to substitute sequences by motifs
- **Grouping**: to assign similar sequences to clusters
- **Ranking and Filtering**: to enable changing data focus
- **Application**: to apply techniques to real-world datasets
Your application

- Email to bernard@ifi.uzh.ch
- Tell me about your
  - Skills in Python programming
  - Knowledge (data science, interactive visualization, interactive data analysis
  - Interest and motivation in this project (machine learning, statistics, backend, frontend, data, application, user-centered, visualization design, interaction design)
- Attach your CV
Effect size ruler
for text-entry studies

How do you know if a new technique or device for text entry compared to others?

You:
- Love reading research papers
- Love working with data and statistics, especially in R
- Like to learn statistical techniques and do hands-on data analysis
- Ideally, have taken or are taking the QuantHCI course

Task: You will compile findings from research papers that have experiments on typing and run statistical meta-analysis.

Results: Reference for researchers to design and interpret results for future experiments

Supervisor: Chat Wacharamanootham
Mona Gaza

Remote meetings, eye contact perception, and lens distortion

How does lens distortion influence eye contact in both directions of remote meetings?

You:

• Like to work on empirical research problems
• Ideally, have knowledge about distortion (from photography or computer graphics)
• Ideally, have taken or are taking the QuantHCI course

Create video stimuli and conduct an online experiment

Results: A model that predict the perception of eye contact in remote meetings

Supervisor: Chat Wacharamanotham
CrowdAlytics
Large-Scale Human-Machine Systems for Data Science (SNSF project)

**Goal:** Framework for the Semantic Annotation of Data-Driven Discoveries

*Arguments, Hypotheses, Scientific Workflows in Publications, Code, and Raw Data*

**DDIS Team:**
Dhivyabharathi Ramasamy
Florian Ruosch
Rosni Kottekulam Vasu
Cristina Sarasua

**Start Date:** July 2021

**Interested?** Write us an email ddis-theses@ifi.uzh.ch
Identify prominent classes in code review

Not all classes have the same importance

Machine learning to find most relevant classes for review

Contacts:
ZEST (zest@ifi.uzh.ch)
Automatic classification of functional review changes

Different changes happen during code review

Is it possible to automatically classify them?

Contacts:
ZEST (zest@ifi.uzh.ch)
Middleware for fast incremental maintenance of machine learning models

Prof. Dan Olteanu

DaST
Data•(Systems+Theory)

March 31, 2021
Many data-intensive applications require computation over the same dataset

- **Our scope**: Applications that manage machine learning models
  - Manage: Train, explain, check fairness, …
  - Low-level computation: Batches of similar aggregates

- **Our scope**: Dataset subject to high-throughput updates
  - Inserts and deletes of tuples to input relations

**State of the art**: Each application re-invents the wheel

- It codes the same or very similar computation
- Lots of redundant computation prone to error
Project Idea

One middleware Mflow between the applications and the dataset

- Applications can (de)register aggregates to Mflow
- Mflow decomposes them into units that can be shared across aggregates
- Mflow computes and maintains the aggregates and their units

Project goals:

1. Understand the problem and design space for possible solutions
2. Approach to decomposing aggregates into shareable units
3. Approach to computing and maintaining aggregates and their units
4. Implement Mflow on top of F-IVM
   https://github.com/fdbresearch/FIVM
5. Benchmark Mflow

If interested: Send details of team members and their CVs to dast@ifi.uzh.ch
Understanding and implementing feature engineering in Kaggle competitions

Prof. Dan Olteanu

DaST
Data • (Systems+Theory)

University of Zurich

March 31, 2021
Context

**Kaggle** (https://www.kaggle.com): Exciting competitions in data science

- Winning solutions typically invest a lot in engineering the features
- Features defined by feature extraction queries over the given dataset

State of the art

- Jupyter notebooks using Pandas used to compute the features
- This can take non-trivial time
Efficient shared computation of all feature extraction queries

- Feature engineering may share computation across different features
- LMFAO: In-house prototype for shared computation across aggregates for training machine learning models

https://github.com/fdbresearch/LMFAO

Project goals:

1. Systematic study of feature engineering in a handful of Kaggle competitions

2. Speed up feature engineering for winning solutions in Kaggle competitions by computing the feature extraction queries using LMFAO

3. Benchmark the prototype against original Pandas implementations

If interested: Send details of team members and their CVs to dast@ifi.uzh.ch
Implementation of an Efficient Algorithm for the Dynamic Evaluation of Hierarchical Queries

Master’s Project Market

Dr. Ahmet Kara

March 31, 2021

• **Static query evaluation**: Provides algorithms that *compute* the query result as fast as possible
Static and Dynamic Query Evaluation

- **Static query evaluation**: Provides algorithms that compute the query result as fast as possible.

- Nowadays, data is subject to frequent updates.

![Diagram showing query flow through a database](image)
• **Static query evaluation**: Provides algorithms that **compute** the query result as fast as possible

• Nowadays, data is subject to frequent updates

• **Dynamic query evaluation**: Provides algorithms that **refresh** the query result after each update as fast as possible
Goals of This Project

Our research group has designed a dynamic algorithm for hierarchical queries

- **Improves** previous results
- Maintains classes of queries in **worst-case optimal** time
- Allows for flexible **trade-offs** between update time and answer time
Goals of This Project

Our research group has designed a dynamic algorithm for hierarchical queries

- Improves previous results
- Maintains classes of queries in worst-case optimal time
- Allows for flexible trade-offs between update time and answer time

Goals of this project

- Understanding of the main techniques in dynamic query evaluation
  - Reading the main paper describing our algorithm
  - Looking into related papers
- Implementation of the dynamic algorithm for hierarchical queries
  - From scratch or on top of an existing system (DBToaster)
- Benchmarking against existing solutions
  - Exploring different datasets
What expects you?

Prerequisites

- Good programming skills in C++
- Conversant with data structures and algorithms; databases
What expects you?

Prerequisites

- Good programming skills in C++
- Conversant with data structures and algorithms; databases

Outcome

- Design and implementation of a novel maintenance algorithm
- Benchmarking against existing solutions
What expects you?

Prerequisites

- Good programming skills in C++
- Conversant with data structures and algorithms; databases

Outcome

- Design and implementation of a novel maintenance algorithm
- Benchmarking against existing solutions

You will work with

Prof. Dan Olteanu
Dr. Ahmet Kara
Haozhe Zhang
References

Papers


- Maintaining Triangle Queries under Updates. Ahmet Kara, Hung Q. Ngo, Milos Nikolic, Dan Olteanu, Haozhe Zhang. TODS 2020

- Incremental View Maintenance with Triple Lock Factorization Benefits. Milos Nikolic, Dan Olteanu. SIGMOD 2018

Systems

- F-IVM: https://github.com/fdbresearch/FIVM
- DBToaster: https://dbtoaster.github.io

If interested: Send details of team members and their CVs to dast@ifi.uzh.ch
End of Master’s Project presentations
Master’s Project Market: Procedure

1. Listen to today’s presentations
2. Exit this Zoom meeting
3. Find peers in Gather: http://tiny.cc/master-project-FS2021
4. Join the Zoom meeting(s) of the project(s) you want to discuss

See Zoom links in your e-mail, or in the Forum at http://t.uzh.ch/yi
Interested in a project? Talk to representatives and form groups!

http://t.uzh.ch/yi

Good luck with your Master’s Project!