DAI-Labor
TU Berlin

Technische Universitaet Berlin
Fakultaet IV - Elektrotechnik und Informatik
Fachgebiet AOT
Prof. Dr. Sahin Albayrak

Bachelorthesis proposal

An intuitive user interface for the automated machine learning project CODA

Robin Ruth
Matrikel–Nummer 316672

Betreuer Researcher Christian Geissler Dipl.-Inform.

Inhaltsverzeichnis

1 Motivation **2**

2 Objective **3**
2.1 Architecture 3
2.2 Workflows 4
2.3 Evaluation 4

3 Work packages **5**
3.1 Architecture 5
3.2 Development 5
3.3 Think aloud interviews 6

4 Schedule **7**

5 Organizational **8**

6 Appendix **9**

1 Motivation

With the ongoing digitalization and the generation of big data in most aspects of society, data driven approaches to every day problems become more and more viable. One way to evaluate these large datasets is with a machine learning approach. Machine learning, though powerful, has its own challenges. Not every machine learning approach is applicable for every problem, not every algorithm will perform in reasonable time on every data, the list goes on.

A lot of intimate knowledge and intuition about machine learning algorithms is needed to select a good approach to a given problem. Knowledge and intuition that is only developed in few experts. Even then, to optimize for a given problem still requires a lot of time, as most of the time the best hyperparameter settings can only be found by trial and error.

To automate this process, DAI-Labor (Distributed Artificial Intelligence) and GT-ARC (German Turkish Advanced Research Center for ICT) have started CODA, a fundamental research project in algorithm selection and hyperparameter optimization. [Siv]

This project creates an interface to the CODA project, allowing machine learning enthusiasts and specialists to use the developed solutions and giving the team an interface to demonstrate CODAs capabilities.

The provided user interface may be used in demonstrations as well as by persons with a background in machine learning. The focus is set on a clean workflow, good user guidance and a visual appealing display.

2 Objective

The overall objective of Sparked is to create a web application and a supporting backend to interface with the CODA backend.

To develop Sparked an architecture has to be chosen. This includes which languages and technologies will be used in the development process.

2.1 Architecture

The architecture has to support several use cases:

- Demonstration startup

  It should be easy to spin up a clean Sparked instance with empty databases for demonstration purposes. Sensible defaults should apply, to make such an instance immediately usable even by an untrained individual.

- Configurability

  Changing the CODA Backend should be possible without recompile.

- Docker deployment

  Sparked has to run in docker containers on a linux system.

Some common characteristics of web applications are not relevant in this project:

- User accounts

  All Users will operate on the same data. There will not be any user accounts. Any data uploaded to Sparked is visible to all other users.

- Login

  The UI will be open and not contain any login features or any other form of access control.

Coda will run on a separate machine, connected via a Kafka bus system. Kafka access is not part of the Sparked project. An encapsulation library written in Java will be provided.

2.2 Workflows

Sparked contains several workflows:

- Order creation

  An order can be created. This includes the selection of all values needed to create an order, classifier, metric, test dataset and validation method with their respective parameters.

  After an order is created, it will be split in tasks and put into a queue for processing. This includes persisting them to have a keep them in data on restart.

- Test data

  While creating an order, the user may select from a pool of test data but may also chose to upload their own sets. Uploaded files have to be redirected to CODA to allow usage in the evaluation process.

  The data will not be validated in Sparked. Correctly formatted data will be assumed.

- Order overview

  View all orders, with their tasks and allows them to be paused or continued if not finished. A Sparked instance will connect to a single backend.

- Order evaluation

  Display an evaluation page for order and task results.

It is not possible to access multiple CODA backends at the same time.

Sparked does not contain any evaluation logic or machine learning support itself. As such it is dependent on the data returned from CODA via Kafka. Neither does it contain an error logic. If an error occurs in CODA, Sparked will not provide help. Additionally Sparked will assume correct configuration and service setup.

2.3 Evaluation

To evaluate the usability and user interface decisions, a think-aloud interview will be conducted. This method helps to understand the expectations a user has at a given point in a workflow and shows if the user is able to use the application in a successful manner without prior knowledge of the program or if help is needed.

The results of the think aloud interview will be discussed in this thesis, but not influence the actual development. Ideally a think aloud interview would be followed by a secondary development block to address the found factors. Instead the results will be collected for possible future projects.

The think aloud interview is not supposed to be a statistical evaluation, the number of participants does not have to be statistically significant.

3 Work packages

The project is roughly separated in 3 parts:

- Architecture
- Development
- Evaluation

3.1 Architecture

- UX Design
- UI Design
- Technology evaluations
  - Database
  - Backend language and technologies
  - Frontend language and technologies
  - Frontend chart library
- System architecture
- Evaluating similar programs

3.2 Development

- Basic setup
- Order creation
- Data persistence: Database access module
- Basic user interface
- Test data upload
- Order overview page

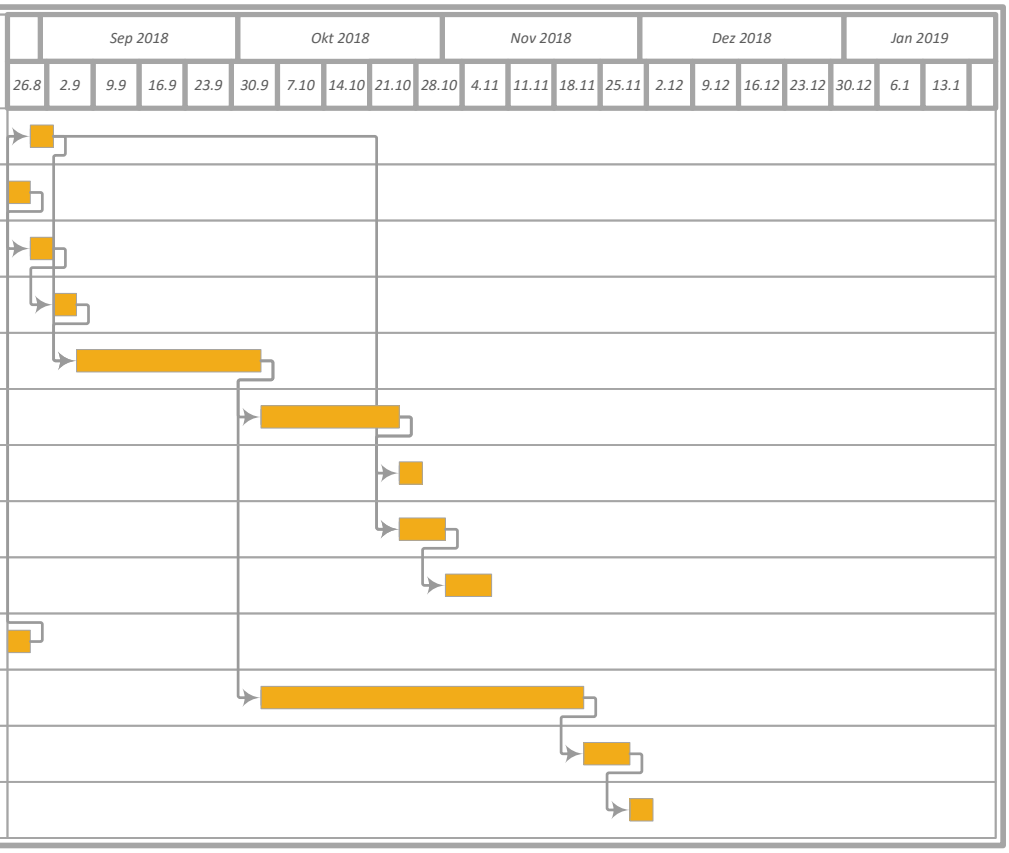
- Integrating Kafka access library / CODA connection
- Configuration page
- Order evaluation
- UI optimization
- Bug fixing
- Docker setup
- Final deployment on DAI server

3.3 Think aloud interviews

- Find test group
- Create interview questionnaire
- Hold the interview
- Evaluate the statements

4 Schedule

ID	Aufgabenname	Dauer
1	UI & UX Design	,5w
2	Technology Evaluation	,5w
3	System architecture	,5w
4	Basic Setup	,5w
5	Basic Development	4w
6	UI Optimization	3w
7	Bugfixing	,5w
8	Interview Preparation	1w
9	Interview and Interviewevaluation	1w
10	Evaluating Similar Solutions	,5w
11	Writing	7w
12	Feedback, Corrections	1w
13	Presentation	,5w



5 Organizational

- Language of this Bachelor thesis is english.
- The thesis will be written with pdflatex.
- Choosing programming languages and technologies are not defined and part of the development process.
- Supervisors is Christian Geißler
- Evaluators are Prof. Dr. Albayrak and Prof. Kao

6 Appendix

Literatur

[Siv] Dr. Fikret Sivrikaya. *Automatisierung von Machine Learning Workflows, CODA Projekt-Steckbrief*. URL: http://www.gt-arc.com/wp-content/uploads/2017/08/Projekt_GT-ARC_CODA_de.pdf.