LASSE F. WOLFF ANTHONY

Quant Developer ⋄ Danish National

+45 81 19 71 00 ⋄ `lfwa@proton.me` ⋄ `lfwa.github.io` ⋄ `github.com/lfwa` ⋄ `linkedin.com/in/lfwa`

EDUCATION

ETH Zürich Sep 2020 – Mar 2023
MSc ETH in Computer Science *Zürich, Switzerland*
Major in Machine Intelligence and minor in Programming Languages and Software Engineering.
Thesis: "Exploring Data Collection Dynamics Through Data Valuation."

University of Copenhagen Sep 2017 – Jun 2020
BSc in Computer Science *Copenhagen, Denmark*
Specialization in Data Science with coursework focused on machine learning.
Thesis: "The Carbon Footprint of Training Deep Learning Models."

EXPERIENCE

UBS Jun 2023 – Present
Quant Developer *Zürich, Switzerland*
· Design and develop big data tools and solutions for Treasury Risk Control's balance sheet analytics.
· Lead developer for a library calculating cash flows from position-level data, enhancing risk management through detailed sensitivity analysis using automatic differentiation.
· Drive code infrastructure improvements, including CI/CD pipelines and migration to Databricks and Spark, enhancing data processing speed and reliability.
· Implement machine learning models for predictive analytics and risk assessment, resulting in more accurate forecasting and better-informed risk management decisions.

Alexandra Institute Apr 2023 – Jun 2023
AI / Machine Learning Specialist *Copenhagen, Denmark*
· Dual role in applied research and expert consultancy in machine learning, focusing on natural language processing and utilizing pretrained transformers.

University of Copenhagen Jan 2020 – Jul 2020
Teaching Assistant *Copenhagen, Denmark*
· Assisted in teaching the Data Science course, covering databases, machine learning, and data pipelines.

Nykredit Oct 2018 – Jan 2020
Software Developer *Copenhagen, Denmark*
· Built financial software for internal advisors in an agile C# development team.
· Developed and maintained financial software for mortgage loans in .NET, reducing processing time.
· Implemented continuous deployment pipelines with Jenkins and BitBucket, fully automating integration testing and deployment, which improved deployment efficiency and reliability.

PUBLICATIONS

[1] L. F. W. Anthony, B. Kanding, and R. Selvan, "Carbontracker: Tracking and predicting the carbon footprint of training deep learning models," in *ICML Workshop on Challenges in Deploying and monitoring Machine Learning Systems*, Jul. 2020.

[2] R. Selvan, N. Bhagwat, L. F. W. Anthony, B. Kanding, and E. B. Dam, "Carbon footprint of selecting and training deep learning models for medical image analysis," in *International Conference on Medical Image Computing and Computer-Assisted Intervention – MICCAI 2022*, 2022.

HIGHLIGHTED PROJECTS

Carbontracker
`github.com/lfwa/carbontracker`
Open-source tool for tracking and predicting the energy consumption and carbon emissions of training deep learning models in Python. The tool is freely distributed under the MIT License. Corresponding publication [1]. It has been downloaded >75k times on the Python Package Index (PyPI) as of writing.

Datadynamics
`github.com/lfwa/datadynamics`
Open-source library and environment for simulating data collection dynamics in multi-agent settings, primarily targeting the exploration of data valuation approaches. The library is freely distributed under the BSD 3-Clause License.

Reinforced Graph Neural Networks for Collaborative Filtering
`github.com/lfwa/reinforced-gnn`
Introduced a novel architecture to generate predictive compatibility scores for never-before-seen content in recommendation systems. The architecture combines the strength of graph-extracted embeddings in a graph neural network with the generalization power of a deep feed-forward network and adds "reinforcements" providing additional information to the network.

Static Taint Analysis For Ethereum Contracts
`github.com/lfwa/vulnerable-ethereum-contracts`
Designed and implemented a static taint analyzer in Datalog for Ethereum smart contracts. The analyzer detects vulnerable contracts that may be deleted from the blockchain and have all remaining cryptocurrency transferred to an untrusted address.

Supporting Alternative SMT Solvers in Viper
`github.com/viperproject`
Added support for multiple SMT solvers, such as cvc5, in the symbolic-execution based automated verification backend written in Scala for the program verification tool chain and infrastructure, Viper.

RELEVANT COURSEWORK

Machine Learning & Big Data	Mathematics	Software Engineering
Advanced Machine Learning	Statistics & Probability Theory	Program Verification
Causal Representation Learning	Discrete Mathematics	Program Analysis for System
Natural Language Processing	Linear Algebra	↳ Security and Reliability
Probabilistic AI	Modelling & Analysis of Data	Concepts of Object-Oriented
Reliable & Trustworthy AI	Algorithms & Data Structures	↳ Programming
Computational Intelligence	Randomized Algorithms	Computer & Network Security
Big Data		

SKILLS

Programming Languages	Python, C#, SQL, Rust, Scala, F#, Java, C, Datalog
Databases	PostgreSQL, Oracle
Frameworks and Tools	PyTorch, TensorFlow, Gym(nasium), PettingZoo, NumPy, pandas, scikit-learn, Matplotlib, Git, Spark, Hadoop, Neo4j, QuantLib