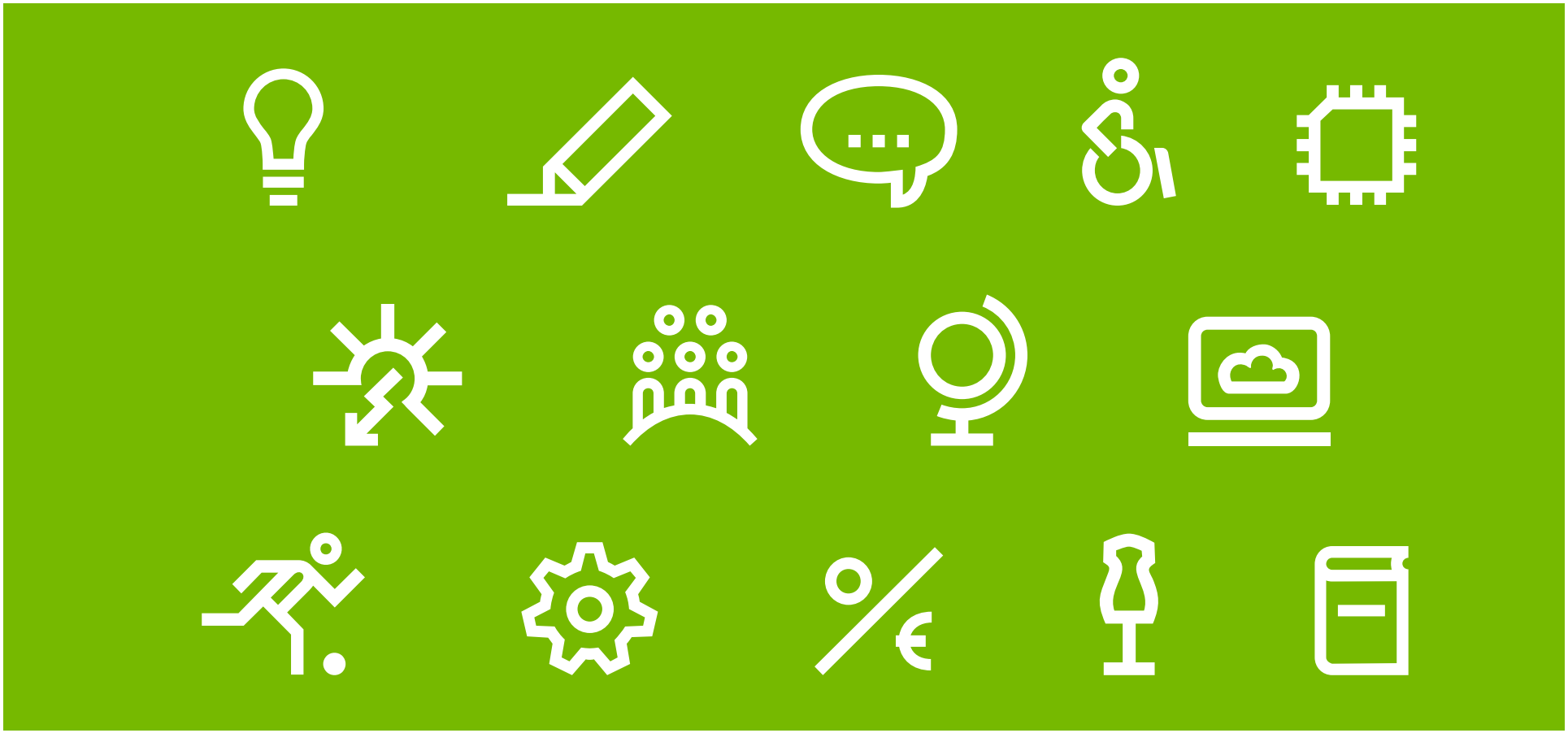


Hochschule für Technik
und Wirtschaft Berlin

University of Applied Sciences

USW
Projects

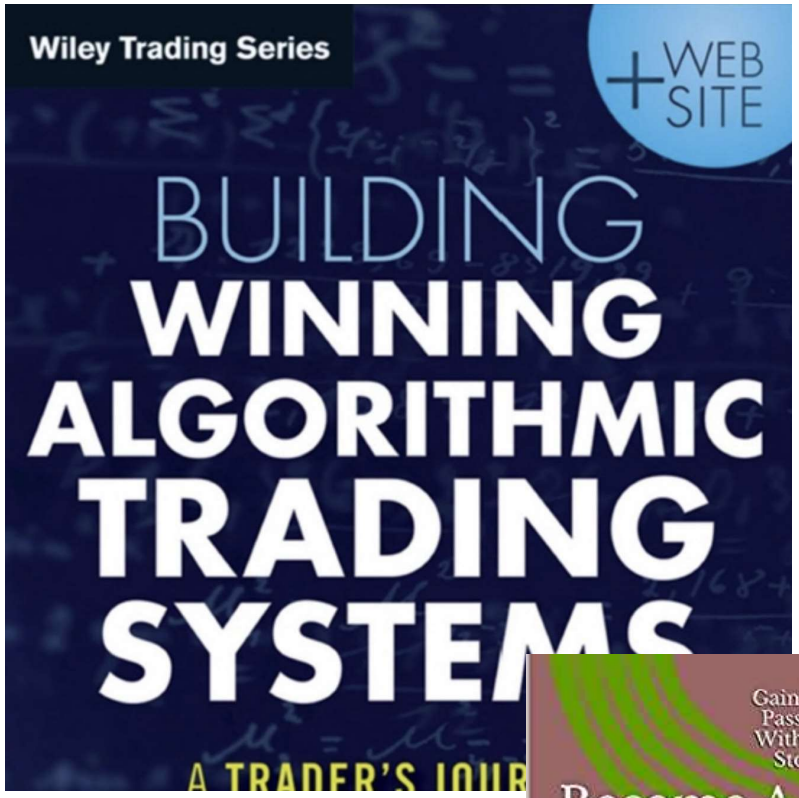
Hochschule für Technik
und Wirtschaft Berlin

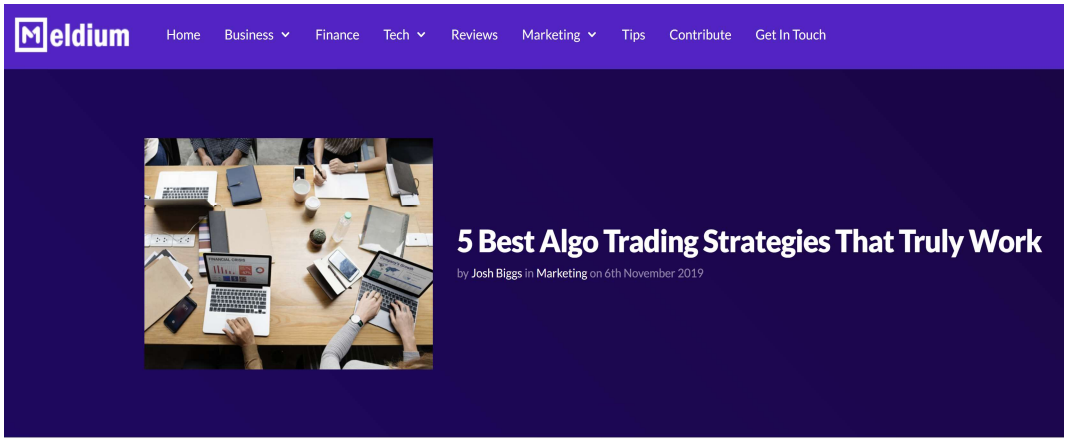
University of Applied Sciences

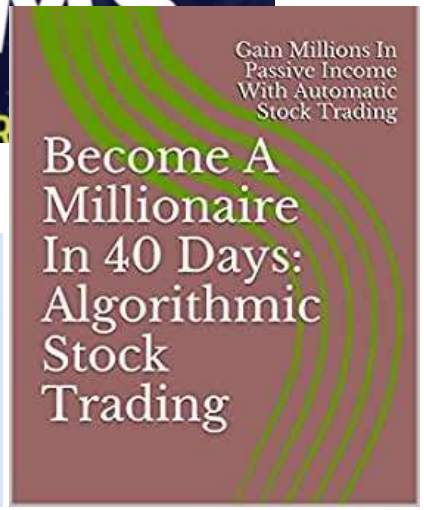
Algorithmic trading for equities/cryptocurrencies

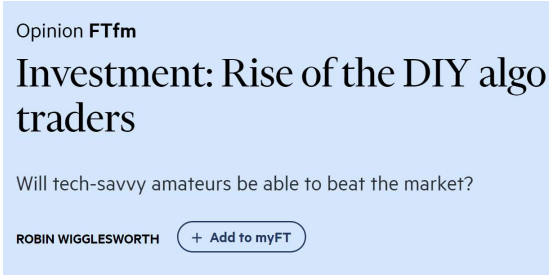
Hochschule für Technik und Wirtschaft Berlin
University of Applied Sciences

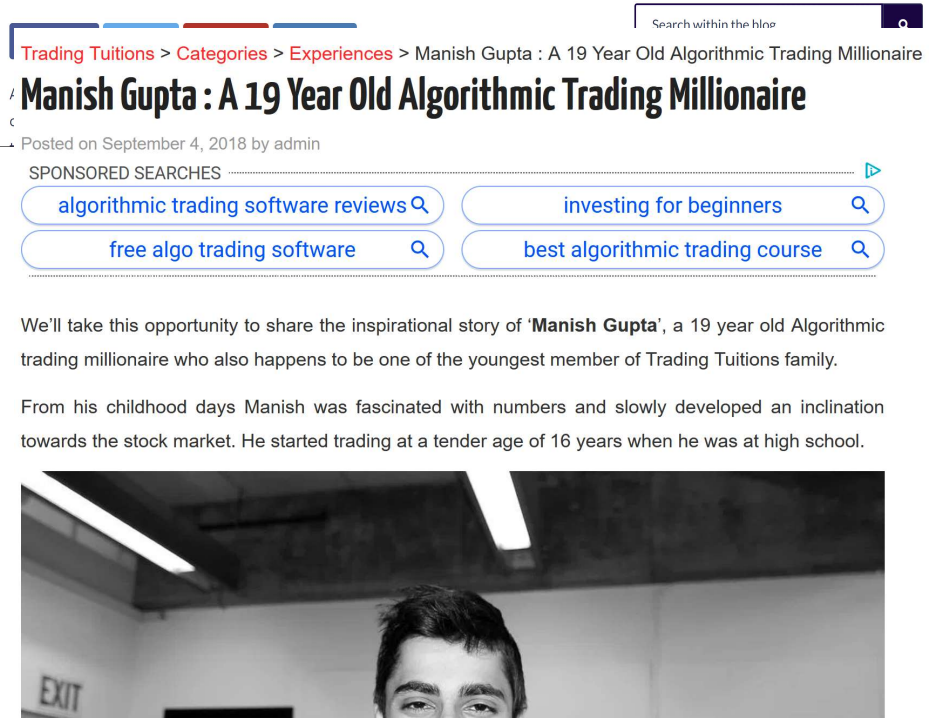
Algorithmic Trading











Hochschule für Technik und Wirtschaft Berlin

University of Applied Sciences

Research

THE JOURNAL OF FINANCE • VOL. XLVII, NO. 5 • DECEMBER 1992

Simple Technical Trading Rules and the Stochastic Properties of Stock Returns

WILLIAM BROCK, JOSEF LAKONISHOK, and
BLAKE LEBARON*

ABSTRACT

This paper tests two of the simplest and most popular trading rules—moving average and trading range break—by utilizing the Dow Jones Index from 1897 to 1986. Standard statistical analysis is extended through the use of bootstrap techniques. Overall, our results provide strong support for the technical strategies. The returns obtained from these strategies are not consistent with four popular null models: the random walk, the AR(1), the GARCH-M, and the Exponential GARCH. Buy signals consistently generate higher returns than sell signals, and further, the returns following buy signals are less volatile than returns following sell signals, and further, the returns following buy signals are less volatile than returns following sell signals. Moreover, returns following sell signals are negative, which is not easily explained by any of the currently existing equilibrium models.

ELSEVIER

Review of Financial Economics

Volume 23, Issue 1, January 2014, Pages 30-45

Predictability of the simple technical trading rules: An out-of-sample test

Jiali Fang, Ben Jacobsen, Yafeng Qin

Show more

https://doi.org/10.1016/j.rfe.2013.05.004

Abstract

In a true out-of-sample test based on fresh data we find no evidence that several well-known technical trading strategies predict stock markets over the period of 1987 to 2011. Our test safeguards against sample selection bias, data mining, hindsight bias, and other usual biases that may affect results in our field. We use the exact same technical trading rules that Brock, Lakonishok, and LeBaron (1992) showed to work best in their historical sample. Further analysis shows that this

Hochschule für Technik und Wirtschaft Berlin
University of Applied Sciences

Secretive Community

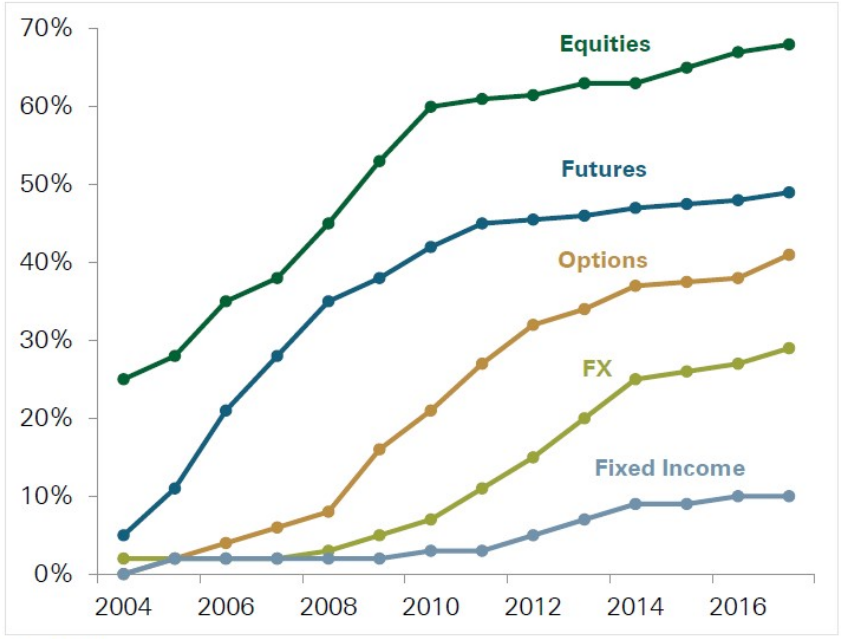
DOI:10.1145/2500117

The competitive nature of AT, the scarcity of expertise, and the vast profits potential, makes for a secretive community where implementation details are difficult to find.

BY PHILIP TRELEAVEN, MICHAL GALAS, AND VIDHI LALCHAND

Algorithmic Trading Review, Communications of the ACM, 2013

Exhibit 2: Market Share of Algorithmic Trading by Asset Class



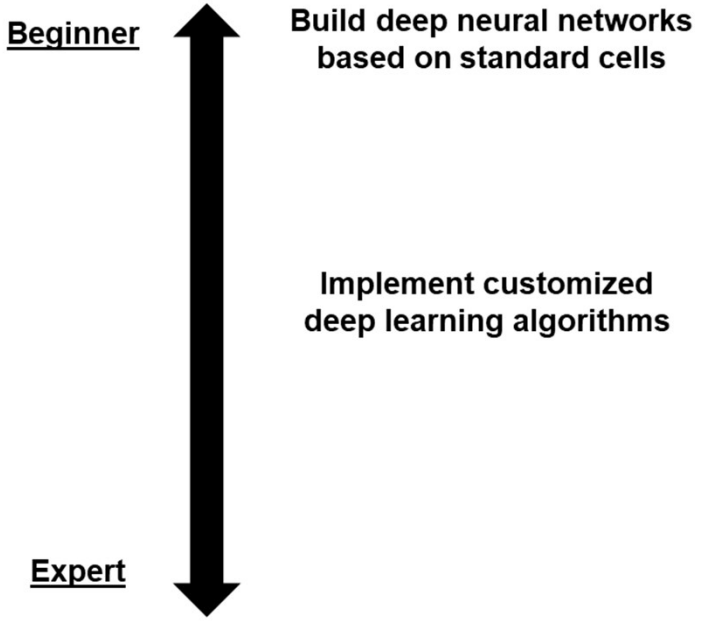
As of 2017.
Source: Goldman Sachs, AiteGroup

Hochschule für Technik und Wirtschaft Berlin
University of Applied Sciences

Papers

- Meta papers:
  - https://www.sciencedirect.com/science/article/abs/pii/S0957417421009441
  - https://www.mdpi.com/1911-8074/14/11/526

- Example:
  - http://proceedings.mlr.press/v95/li18c/li18c.pdf



Hochschule für Technik und Wirtschaft Berlin
University of Applied Sciences

Backtesting of Various Strategies

- Data acquisition:
  - Sources (Kaggle, etc.)
  - APIs (Yfinance, Alpaca, etc.)
  - https://www.cryptodatadownload.com/
- Features:
  - Open-High-Low-Close-Volume
  - Chart patterns (TAlib)
  - Correlated series (stocks, raw materials, etc.)
- Target:
  - Markets (NASDAQ, DAX, currencies, crypto, etc.)
  - Frequency (monthly, weekly, daily, hourly, every minute)
  - Individual stocks, industries, markets, indices
- Modeling architecture:
  - LSTM, RNN, Attention mechanism
  - Dimensionality (number of layers, number of neurons, etc.)
  - Activation functions (ReLU, Tanh, etc.)
  - Mechanisms (drop out, etc.)
  - Optimization (Adam, SGD, etc.)
- Performance criteria:
  - Return over period of time
  - Accuracy of prediction
- Baseline:
  - Market performance over period of time
  - Comparison with dummy strategy (randomly buy and sell, etc.)

Hochschule für Technik und Wirtschaft Berlin
University of Applied Sciences

Folder Structure Algorithmic Trading

- README.md
  - General description and goal of project
  - Summary and results across experiments
- Experiment 1
  - Data
  - Code
  - README.md
    - Short description
    - Data acquisition
    - Features
    - Target
    - Modeling architecture
    - Performance criteria
    - Baseline
    - Results
- Experiment 2
- Experiment 3
- ….

Hochschule für Technik und Wirtschaft Berlin
University of Applied Sciences

Experiment 1

- Short description:
  - Using attention-based multi-input LSTM for predicting the price increase (percentage) of a stock one day ahead.
- Data acquisition:
  - Daily adjusted close stock prices from 1985-2020 from all stocks listed on NASDAQ throughout this period.
  - Time between 1985-2015 is used for training and the rest for testing
- Features:
  - For a given stock the following features are calculated and normalized over the last 10 days: HLOC, volume, close of 10 most correlated stocks
- Target:
  - Price increase (percentage) of given stock one day ahead.
- Modeling architecture:
  - Attention-based multi-input LSTM
- Performance criteria:
  - MSE of true vs. predicted stock price increase for test data
- Baseline:
  - MSE of true vs. average price increase over all stocks for test data
- Results

Hochschule für Technik und Wirtschaft Berlin
University of Applied Sciences

Experiment 2

- Short description:
  - Using attention-based multi-input LSTM for predicting the price increase (percentage) of a stock one day ahead.
- Data acquisition:
  - Daily adjusted close stock prices from 1985-2020 from all stocks listed on NASDAQ throughout this period.
  - Time between 1985-2015 is used for training and the rest for testing
- Features:
  - For a given stock the following features are calculated and normalized over the last 10 days: HLOC, volume, RSI-30, MACD, close of 10 most correlated stocks
- Target:
  - Price increase (percentage) of given stock one day ahead.
- Modeling architecture:
  - Attention-based multi-input LSTM
- Performance criteria:
  - MSE of true vs. predicted stock price increase for test data
- Baseline:
  - MSE of true vs. average price increase over all stocks for test data
- Results

Hochschule für Technik
und Wirtschaft Berlin

University of Applied Sciences

GenAI App

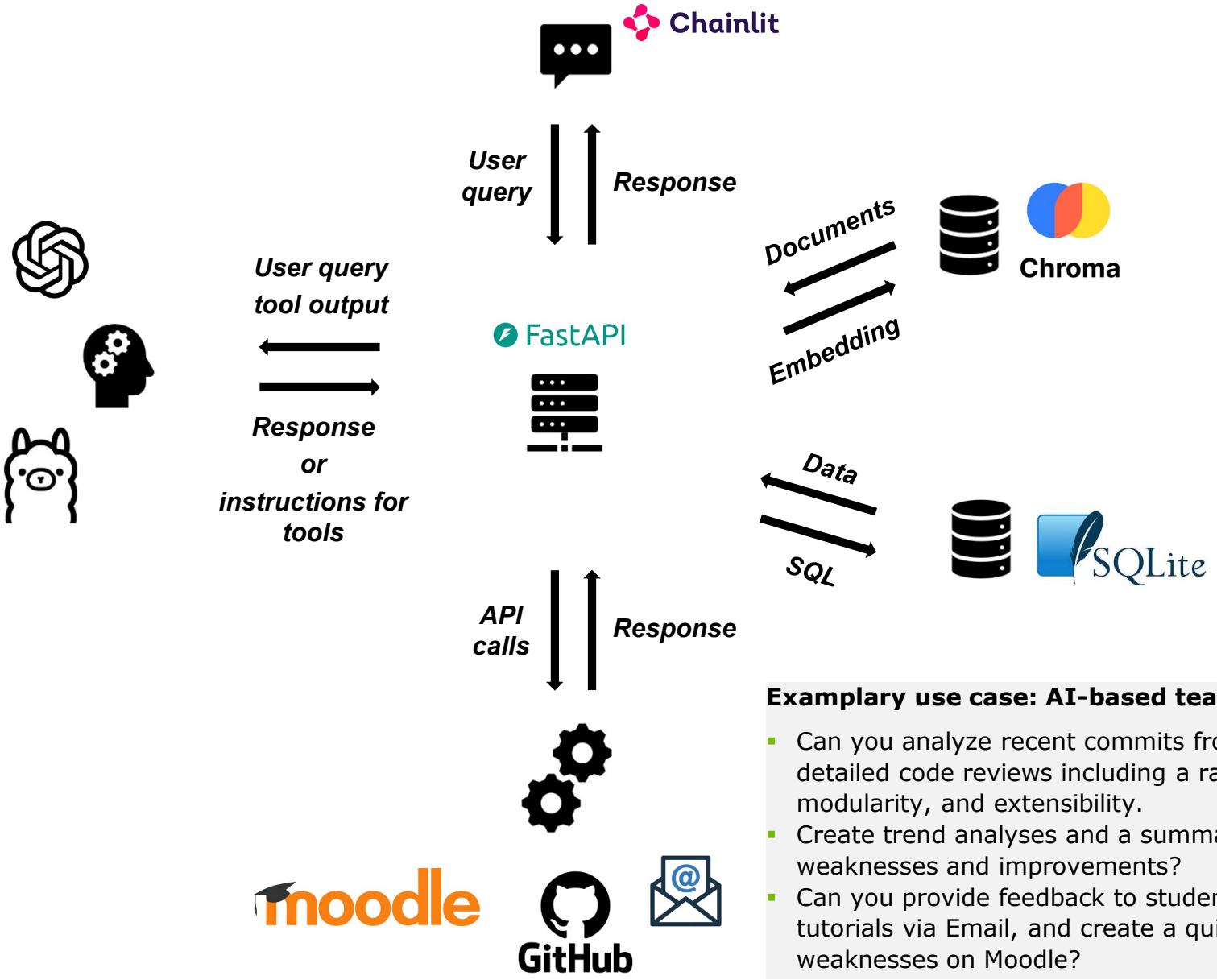
Hochschule für Technik und Wirtschaft Berlin
University of Applied Sciences

AI App Development: Examples

- Personal advisor
  - How can I save enough money to be able to buy an e-bike next year?
  - Access to bank statements, personal preferences extracted from social media, email etc., data from fitness tracker
- Business intelligence specialist
  - Do you see any irregularities in our sales performance over the last 5 years?
  - Access to sales data
- Student assistant
  - How can I get a good grade in USW?
  - Access to Moodle, Github etc.
- Teaching assistant
  - …

Hochschule für Technik und Wirtschaft Berlin
University of Applied Sciences

AI App Development: Vector DBs



Examplary use case: AI-based teaching assistant

- Can you analyze recent commits from student XY, create detailed code reviews including a rating of readability, modularity, and extensibility.
- Create trend analyses and a summary of strenghts, weaknesses and improvements?
- Can you provide feedback to student XY, send some helpful tutorials via Email, and create a quiz targeting students‘ weaknesses on Moodle?

Hochschule für Technik und Wirtschaft Berlin
University of Applied Sciences

Folder Structure GenAI

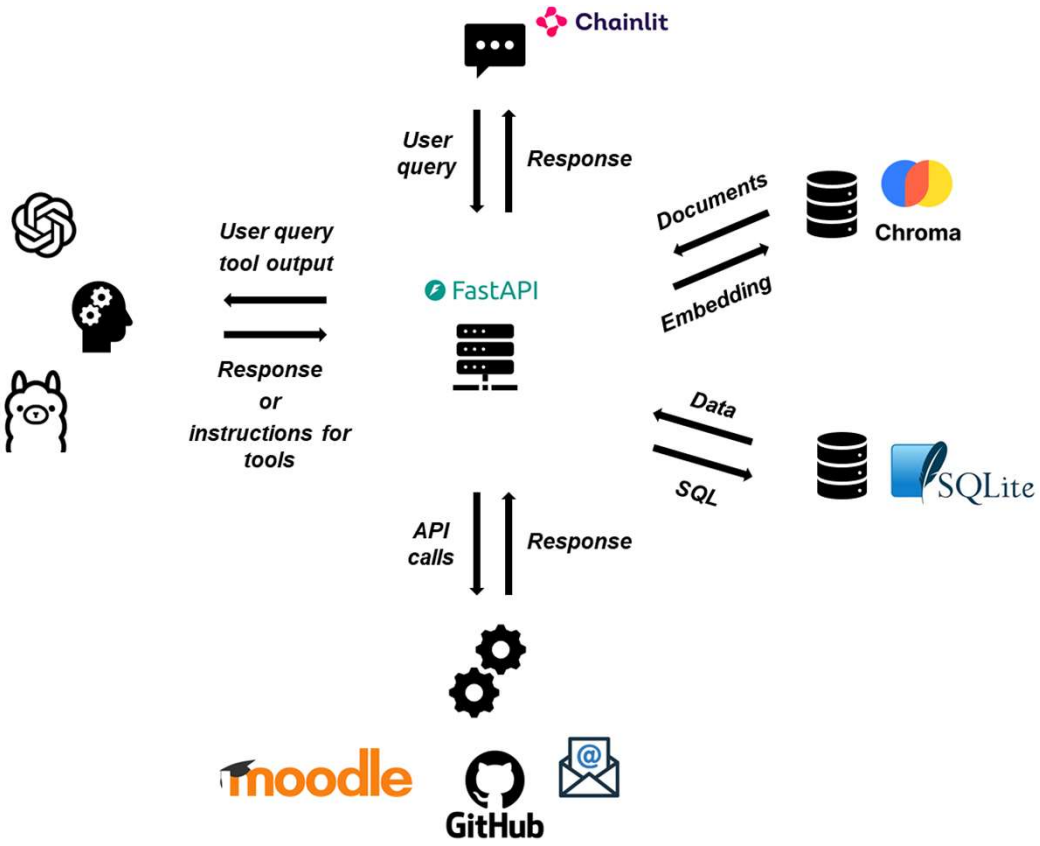
- README.md
  - General description and goal of project
- Use Case 1
  - Description of use case and tasks involved
  - Task 1
    - Experiment 1
      - Data
      - Code
      - Prompts
      - README.md
        - Short description
        - Use Case
        - Evaluation criteria incl. metrics
        - Baseline
        - Detailed description of approach (models, message flow, prompts, function, embeddings, etc.)
        - Results (incl. plots)
    - Experiment 2
    - …
  - Task 2
  - …
- Use Case 2
- …

Hochschule für Technik und Wirtschaft Berlin
University of Applied Sciences

Description Of Project

Examplary use case: AI-based teaching assistant

- This project aims at developing an AI-based teaching assistant that supports the teacher in any task simply by using a natural language interface and where necessary providing insightful reports.
- The architecture of the implemented system is depicted in figure 1. Detailed description of used components can be found in explanations of use cases, tasks, and experiments.
- Exhaustive experiments around the following use cases were performed:
  1. Can you analyze recent commits from student XY, create detailed code reviews including a rating of readability, modularity, and extensibility.
  2. Create trend analyses and a summary of strenghts, weaknesses and improvements?
  3. Can you provide feedback to student XY, send some helpful tutorials via Email, and create a quiz targeting students‘ weaknesses on Moodle?



Hochschule für Technik und Wirtschaft Berlin
University of Applied Sciences

Description Of Use Case 1

- Use Case: Teacher asks AI assistant to give a summary of recent development of particular student
- 1. task: AI assistant collects all submissions from student
- 2. task: AI assistant evaluates each submission
- 3. task: AI assistant analyzes time-based trends based on evaluations
- 4. task: AI assistant returns summarizing response to teacher

Hochschule für Technik und Wirtschaft Berlin
University of Applied Sciences

1. Task: AI Assistant Collects All Submissions From Student

Experiment 1: Usage of Llama 3.1

- Input: query from teacher with student ID
- Output: all submissions from student
- Approach: function calling
- Message flow:
  - Query of teacher is submitted to model
  - Model decides to use function „get_submissions_from_moodle" with student ID as argument
  - Model decides to use function „get_github_name" with student ID as argument
  - Model decides to use function „get_commits" with student's Github name as argument
  - Model returns all Moodle submissions and all Github commits
- Tools: Moodle, Github
- Models: Llama 3.1
- Prompt technique: chain-of-thought prompting, step-back prompting
- Evaluation criteria
  - Correctness: are correct submissions returned
  - Completeness: are all submissions returned
- Results
  - Table with prompt alternatives and corresponding values for evaluation criteria

Hochschule für Technik und Wirtschaft Berlin
University of Applied Sciences

1. Task: AI Assistant Collects All Submissions From Student

Experiment 2: Usage of OpenAI

- Input: query from teacher with student ID
- Output: all submissions from student
- Approach: function calling
- Message flow:
  - Query of teacher is submitted to model
  - Model decides to use function „get_submissions_from_moodle" with student ID as argument
  - Model decides to use function „get_github_name" with student ID as argument
  - Model decides to use function „get_commits" with student's Github name as argument
  - Model returns all Moodle submissions and all Github commits
- Tools: Moodle, Github
- Models: OpenAI
- Prompt technique: chain-of-thought prompting, step-back prompting
- Evaluation criteria
  - Correctness: are correct submissions returned
  - Completeness: are all submissions returned
- Results
  - Table with prompt alternatives and corresponding values for evaluation criteria

Hochschule für Technik und Wirtschaft Berlin
University of Applied Sciences

Evaluation

- Human evaluation
- LLM as a judge: https://arxiv.org/pdf/2306.05685

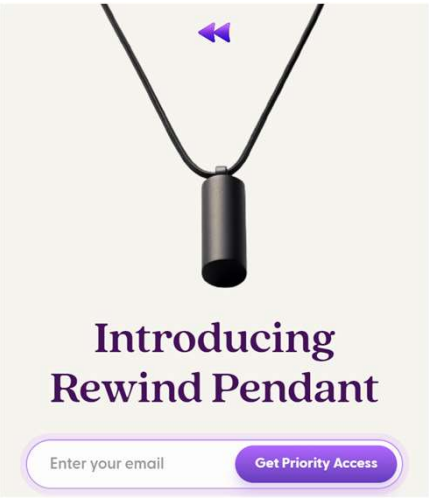
Hochschule für Technik und Wirtschaft Berlin
University of Applied Sciences

Project: Simulate Social Platform With Generative Agents

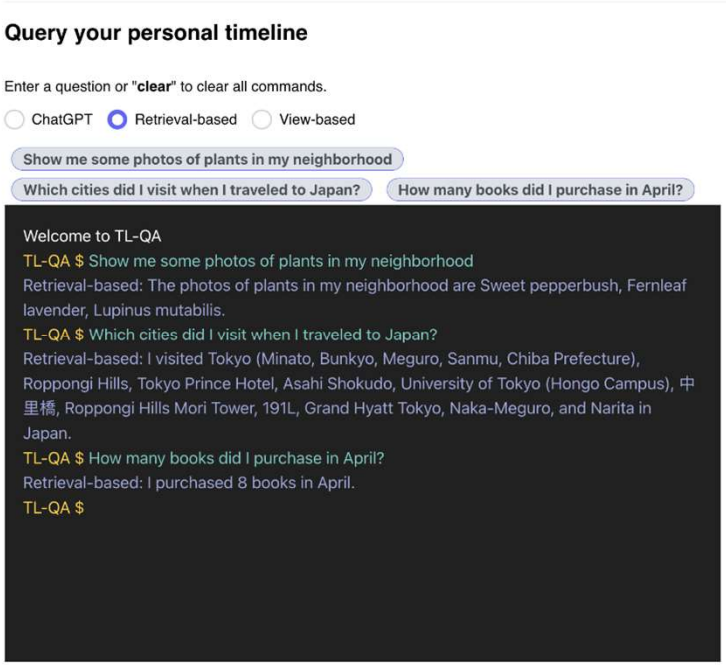
- Provide initial description of agent (possibly hobbies, interests, CV, goals, etc.)
- For each member, decide to have conversation (on a daily basis)
- Feed conversation into memory stream
- Generate summary and feed into memory stream
- Rate each entry how interesting it was
- Generate higher-level reflections per chat partner and across chat partners
- Retrieve relevant content for conversations (history for particular member, higher-level reflections from other conversations, etc.)
- Allow user feedback/input
- Goal could be to generate as deep conversations as possible (evaluation via LLM)

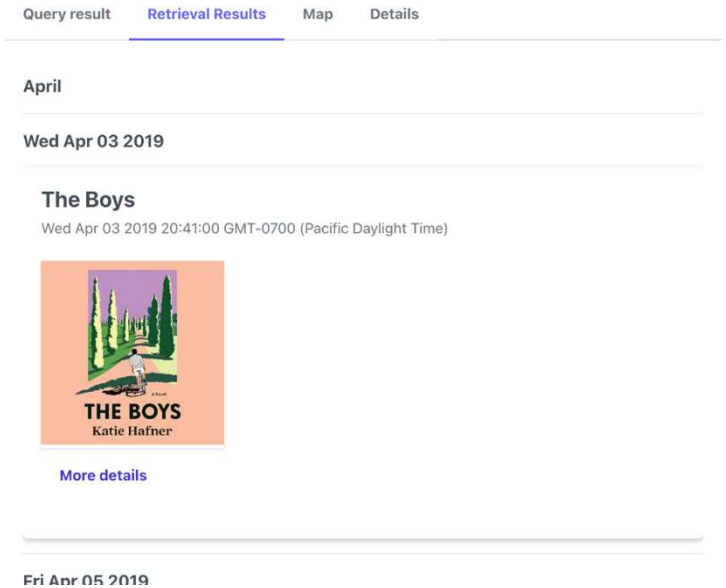
Hochschule für Technik und Wirtschaft Berlin
University of Applied Sciences

Project: Digital Twin



- Build a digital version of yourself
- Connectors to email, X, LinkedIn, Moodle, Whatsapp, Instagram, FB, bank account etc.
- Personal files
- Microphone transcript (record meetings, Zoom, etc.)
- Meta Smart Glasses (record images, videos)
- Let LLM extract additional features (e.g. personality traits, high-level reflections)
- Also check Quivr (https://www.quivr.app/), Rewind (https://www.rewind.ai/), Augment (https://www.augment.co/)
- Local version with ChromaDB
- UI to navigate personal data (https://github.com/facebookresearch/personal-timeline/tree/main)





Hochschule für Technik und Wirtschaft Berlin
University of Applied Sciences

Project: Recursively Breaking Down And Executing Complex Tasks Using Tools

- Example: writing a book
- Use LLM to generate list of high-level book characteristics (topic, genre, length, main characters, attributes such as funny, plot twist, sad etc.)
- Give detailed description for some characteristics or let LLM choose
- LLM generate synopsis
- LLM breaks down synopsis into chapters
- LLM breaks down chapters into paragraphs
- Let LLM take role of book critic to generate feedback per paragraph, chapter, synopsis
- Let LLM rework pieces based on feedback
- Use search API to do research and enrich paragraphs, chapter, synopsis
- How to ensure that everything fits together?
  - Summarization
  - Retrieval
- Other areas:
  - Games
  - Applications
  - Theses