DHBW Mannheim

Big Data Analytics

Projekt

Frank Schulz

Projektbeschreibung

Projektgruppen

- Gruppe aus 3-4 Teilnehmern
- **Ziel**: Prototypische Entwicklung einer **Data Pipeline**, die das Konsumieren, Speichern, Verarbeiten und Visualisieren von Daten umfasst

Deliverables

(1) Projektbericht

- Ein Bericht pro Gruppe, ca. 15 Seiten (± 3 Seiten).
- Jeder Teilnehmer schreibt einen oder mehrere Abschnitte, dabei soll in einem Anhang vermerkt werden, wer welchen Abschnitt geschrieben hat.
- Einreichen per Moodle bis **Sonntag, 24. Dezember 2023**

(2) Coding auf Github oder Gitlab

(3) Präsentation am Donnerstag, 14. Dezember 2023

- 10-15 Minuten Präsentation, jedes Gruppenmitglied soll einen Teil präsentieren
- 5 Minuten Diskussion

Projektbericht

Project Report

- Wenig formale Anforderungen:
  kein Abstrakt, kein Inhaltsverzeichnis, kein Tabellen- oder Abbildungsverzeichnis nötig
- Aber ein vollständiges Literaturverzeichnis mit Referenzen zu allen verwendeten externen Quellen (Papers, Webseiten, Blogs, Github Repositories, ...)

Ziel

Der Projektbericht soll zwei Punkte adressieren:

- Darstellung der Anwendung: kurze Beschreibung der verwendeten Datenquellen, der Fragestellung und der Ergebnisse (Beantwortung der Fragen anhand der Daten)
- Kurze Anleitung zum Nachvollziehen und Wiederholen der Arbeit. Keine allgemeinen Informationen, sondern spezifische Details der eigenen Implementierung.
  - Welche Tools und Packages wurden ausgewählt?
  - Welche Schritte wurden ausgeführt (eventuell mit Snippets von wichtigen Kommandos oder Coding)?
  - Welche Schwierigkeiten sind aufgetreten und wie wurden sie gelöst?

Projektpräsentation

Die Projektpräsentation soll enthalten

- Beschreibung der Datenquellen
- Motivation aus Anwendungssicht: welche Fragestellungen werden mit Hilfe der Daten beantwortet?
- Hauptteil: Beschreibung und Erläuterung der erstellten Datenpipeline, idealerweise mit einer Live Demo
- Zusammenfassung: Beantwortung der Fragen aus Anwendungssicht

Ziel

Die Präsentation soll einen Überblick über den erstellten Prototype geben:

- Architektur: Aus welchen Komponenten besteht die Pipeline und wie sind sie miteinander verbunden?
- Wurde eine spezielle Konfiguration verwendet?
- Welche Schwierigkeiten sind aufgetreten und wie wurden sie gelöst?

Projektaufgaben

Data Ingestion

- Verschiedene Optionen
  - API Call
  - Data Stream (Wikipedia live changes,...)
  - Daten aus File Upload (falls andere Optionen nicht realisierbar sind)

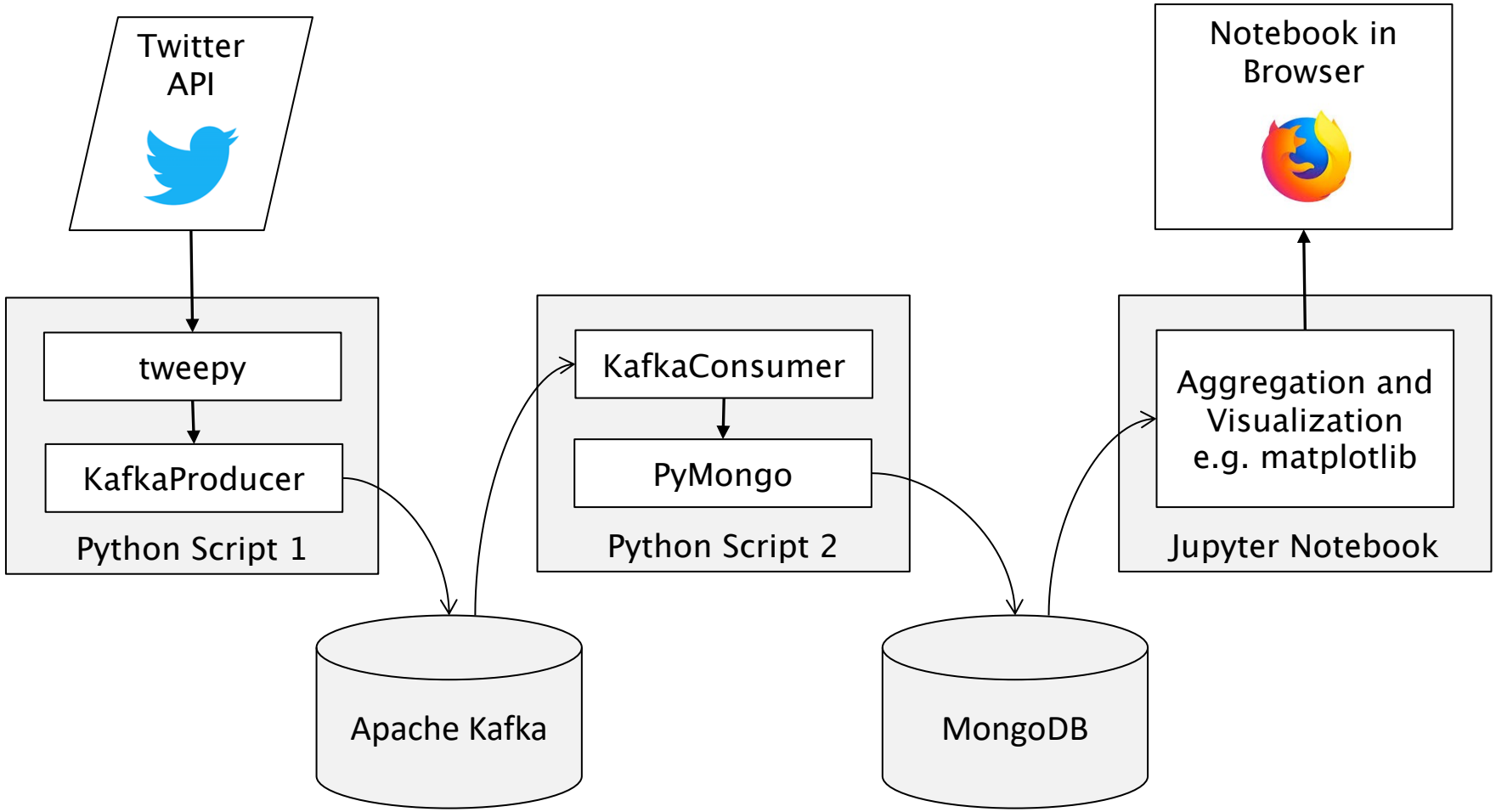
Data Storage

- Speicherung in NoSQL Datenbank (z.B. MongoDB), relationaler Datenbank, oder direkte Weiterverarbeitung (z.B. Spark Streaming)

Datenverarbeitung, Analyse und Visualisierung

- Darstellung der Ergebnisse mit einfacher Visualisierung (z.B. Jupyter oder Streamlit)

Beispiel für eine Data Pipeline



Datenquelle 1: Twitter / X

Twitter

- Access to the Twitter live stream
- Description see https://developer.twitter.com/en/docs/tweets/filter-realtime/api-reference/post-statuses-filter
- Using endpoint https://stream.twitter.com/1.1/statuses/filter.json

Deprecated

- Twitter API v2 erlaubt in der freien Variante nur noch sehr eingeschränkte Zugriffe

Datenquelle 2: Social Networks

Reddit

https://www.reddit.com/wiki/api

Youtube

https://developers.google.com/youtube/v3/

Instagram

https://developers.facebook.com/docs/instagram-basic-display-api

Facebook

https://developers.facebook.com/docs/graph-api/

Social Network Scraping

- https://github.com/JustAnotherArchivist/snscrape/blob/master/README.md

Datenquelle 3: Wikipedia

Wikipedia

- Data is available here (Download, API, Recent Changes Stream)
  https://meta.wikimedia.org/wiki/Research:Data

- Event stream of recent changes
  https://wikitech.wikimedia.org/wiki/EventStreams
  https://www.mediawiki.org/wiki/API:Recent_changes_stream

  Endpoint for reading data: https://stream.wikimedia.org/v2/stream/recentchange

- Examples
  http://rcmap.hatnote.com/#en
  http://listen.hatnote.com/

Datenquelle 4: New York City Administration

New York Cabs

- Data on the taxi trips in New York
  http://www.nyc.gov/html/tlc/html/about/trip_record_data.shtml

- Monthly PARQUET files in three categories
  - Yellow — around 800-900 MB each
  - Green — around 100 MB each
  - FHV (For Hire Vehicle) — around 400-800 MB each

- Contains geographical data (Pickup / Dropdown Zone)
  => Possibility of visualization on a map

More open data from New York City Government

- https://opendata.cityofnewyork.us/

Datenquelle 5: Benzinpreise

Fuel prices

- "Tankerkönig" offers access to current fuel prices of all fuel stations in Germany. The fuel stations are obliged to report their prices to the "Markttransparenzstelle für Kraftstoffe" (MTS-K): http://www.tankerkoenig.de/

- Historical data for download
- API for accessing current data: https://creativecommons.tankerkoenig.de/
  - Name, address and geographical coordinates of the fuel station
  - Current prices for different types of fuel
  - Opening times and information whether currently open or closed

Datenquelle 6: Nachrichten

- **New York Times** offers extensive APIs for querying news articles and related information: https://developer.nytimes.com/apis
  - Article search
  - Most popular articles
  - Geographical information
  - User comments

- **FiveThirtyEight**
  - Opinion polls and other news
  - https://data.fivethirtyeight.com/

Datenquelle 7: Wetter

- Open Weather
  - Free access to current weather and limited forecast, at most 1000 calls/day
  - https://openweathermap.org/api
  - https://openweathermap.org/api/one-call-api

- Weather API
  - https://www.weatherapi.com/

Datenquelle 8: Börse

- Alpha Vantage
  - Free real-time stock data
  - https://www.alphavantage.co
- IEX Cloud
  - Real-time data for financial applications
  - https://iexcloud.io/
- Twelve Data
  - https://twelvedata.com/docs#getting-started
- Quantopia
  - Quantitative finance data including real-time stock prices
  - https://www.quantopian.com

Datenquelle 9: Crypto

- **Coincap**
  - Free, no registration required (rate limiting of 200 queries / minute)
  - https://docs.coincap.io/
- **Coingecko**
  - Free, no registration required (rate limiting of 50 queries / minute)
  - https://www.coingecko.com/en/api

Datenquelle 10: Filme und Serien

- The Movie Database (TMDb)
  - Access to movie and series metadata
  - https://www.themoviedb.org/documentation/api

- Python client library
  - https://pypi.org/project/tmdbsimple/
  - and others

Datenquelle 11: Verkehr

- Pedestrians in Germany cities
  - Free access to number of people passing a specific point
  - https://hystreet.com/ (requires registration)
  - Python client: https://github.com/JohannesFriedrich/hystReet
- Bikesharing
  - NextBike: https://api.nextbike.net/maps/nextbike-live.json
  - Capital Bikeshare: https://www.capitalbikeshare.com/system-data
- Deutsche Bahn
  - https://data.deutschebahn.com/dataset.groups.apis.html
- Airplanes
  - FlightRadar24: https://www.flightradar24.com/
  - Only aggregate data can be downloaded for free
- Ships
  - MarineTraffic: https://www.marinetraffic.com/
  - No free download

Datenquelle 12: Bundesstelle Open Data

Diverse APIs für Daten zu Meldungen und Warnungen von allgemeinem öffentlichem Interesse

https://github.com/bundesAPI

Weitere Datenquellen:
https://rapidapi.com/search/