Einführung und Administratives

VL Big Data Engineering
(vormals Informationssysteme)

Prof. Dr. Jens Dittrich

bigdata.uni-saarland.de

14. April 2022

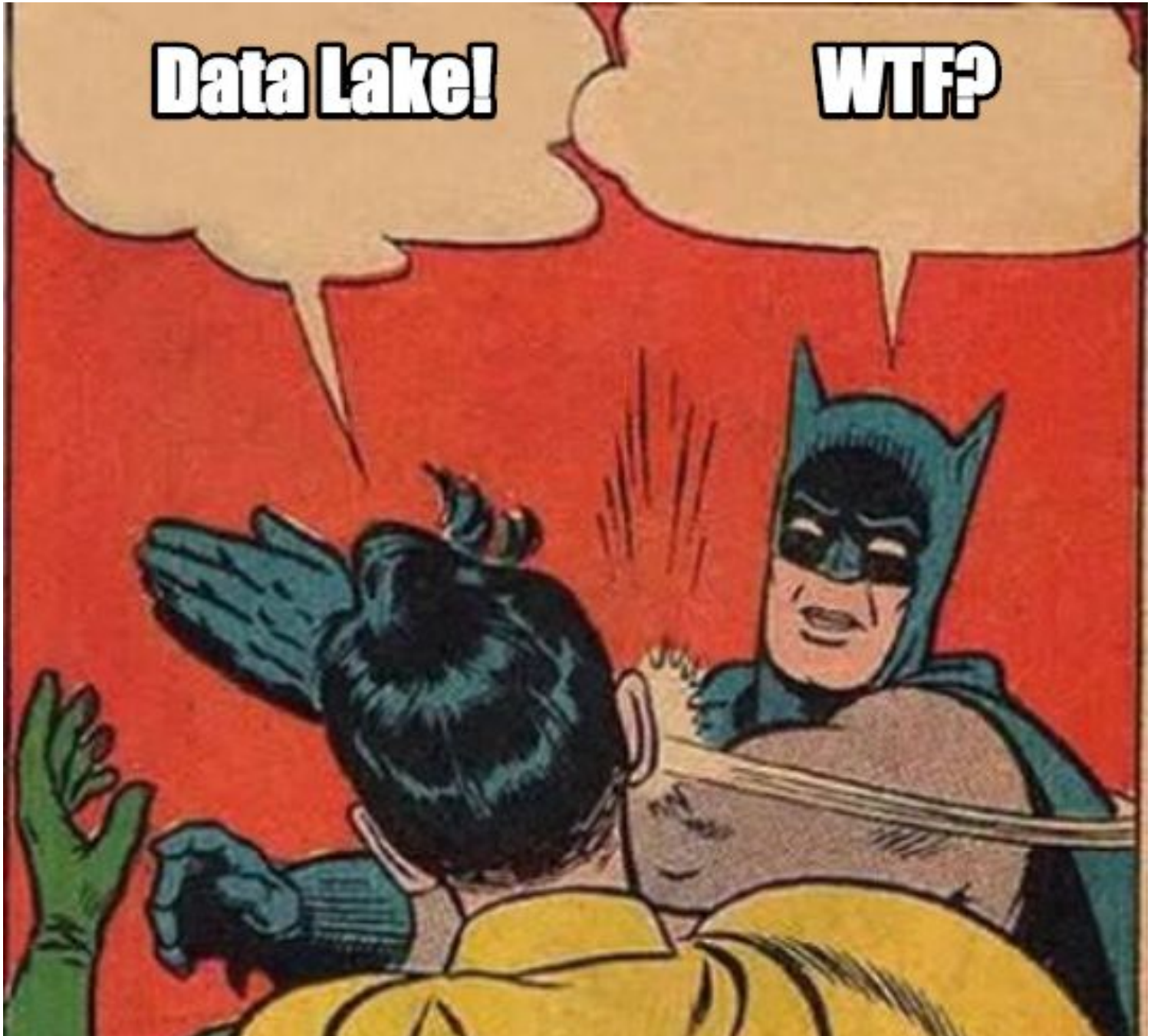
Übersicht über die Vorlesung

- Begriffsbildung
- Inhalt, Konzept
- Lernziele
- Übungen und Übungsgruppen
- Klausuren
- Office Hours
- Python

Schlagworte aus dem Bereich Datenanalyse/Big Data Engineering

Informationssysteme
Big Data
Künstliche Intelligenz
Machine Learning
Deep Learning
Data Mining
Cognitive Computing
NoSQL
SQL
DBMS
RDBMS
ODBMS
Datenbanken
Statistik
Lambda-Architektur
Cloud Computing
Data Warehousing
Data Science
Data Lake
Data Engineering
Data Cleaning
Data Curation
Spark
Hadoop
MapReduce
Data Streaming
IoT
Realtime Analytics
Big Data Analytics
Key/Value-Stores
Column Stores
Blockchain

Industrie vs Universität



The Data Science Cake

Ingredients:
50g statistics
120g linear algebra
200g programming
1kg visualisation
300g software engineering

Additional skills:
creativity
out of the box thinking
grit
team spirit

Die Sicht eines „Datenbänklers“ (1)

- Unsere Top-Konferenz heißt VLDB (Very Large Databases) seit 1975!
- Technisch ist das Verwalten und Anfragen großer Datenmengen längst gelöst (**falls** man weiß, was man tut).
- Performance-Probleme mit großen Datenmengen: Es liegt selten an Hardware/Software, das ist in 99,99% der Fälle ein Ausbildungsproblem (des Entwicklers/Informatikers).
- Die Kombination von Datenbanktechnologie mit anderen Teilgebieten von Data Science ist hochspannend.
- Beispiel: angewandtes maschinelles Lernen und Datenbanken, diverse Projekte.
- Wichtige Datenbankthemen für Data Science: Datenmodellierung, Relationales Modell, SQL, ETL, ELT, Data Cleaning, Data Curation, Data Warehousing, Scalability, verteilte Datenbanken, permissioned Blockchain, ...

Die Sicht eines „Datenbänklers“ (2)

Auswirkungen des Speicher-/Datenbanksystems wird oft unterschätzt

„*Die Daten kommen irgendwie von da unten. Wichtig ist die Komplexität der Algorithmen!*”
Nein! Das ist für viele Systeme falsch. Auf moderner Hardware ist nicht mehr die CPU der Flaschenhals sondern die langsame „Anlieferung“ der Daten.

Beispiel:

- Ausgangssituation: Hadoop-Cluster mit Spark und Software in Scala
- Änderung von uns: anderes Storage-Layout + ein paar DB-Tricks
- Vorhersage unseres Kostenmodells: Faktor 10000 schneller
- statt teurem Hadoop-Cluster:
  Laptop oder Smartphone
- KIWI (kill it with iron)
  vs KIWI (kill it with intelligence)

Die Sicht eines „Datenbänklers“ (3)

Die Performanz der Datenbanktechnologie spielt oft keine Rolle.

Wann?

Wenn die Daten klein sind und die Hardware so schnell ist, dass es keinen Unterschied macht.

versus

Die Performanz der Datenbanktechnologie spielt oft eine große Rolle.

Wann?

Wenn die Daten „größer“ sind und die Hardware die Mängel der Software nicht löst.

Lernziele dieser Vorlesung

1. Grundlegende Techniken im Bereich „Big Data Engineering“ konzeptuell lernen:
   Folien, Übungsaufgaben
2. Grundlegende Techniken im Bereich „Big Data Engineering“ anwenden lernen:
   Python, SQL, Jupyter
3. Ihnen helfen, später nicht das Rad neu zu erfinden:
   Lernen, neue Probleme auf existierende Probleme abzubilden und mit etablierten Techniken zu lösen.
4. Für Probleme wichtiger Anwendungen sensibilisieren:
   Privatheit, Deanonymisierung,
   ethische Fragestellungen
5. Für Lösungen wichtiger Anwendungen sensibilisieren:
   Aufwand, Performanz, Robustheit,
   Erweiterbarkeit, Wartbarkeit

Konzept dieser Veranstaltung: Learning by Application

Geplante Struktur für jeweils zwei Wochen Vorlesung:

1. Konkrete Anwendung: XY
2. Was sind die Datenmanagement und -analyseprobleme dahinter?
3. Grundlagen, um diese Probleme lösen zu können
   (a) Folien
   (b) Jupyter/Python/SQL Hands-on
4. Transfer der Grundlagen auf die konkrete Anwendung

Geplante Struktur jedes Übungszettels:

1. 2 Aufgaben mit Bezug zu Grundlagen:
   Folien
2. 1 Aufgabe mit Bezug zu Grundlagen:
   Jupyter/Python/SQL Hands-on
3. 1 Aufgabe mit Bezug zum Transfer
   der Grundlagen auf die Anwendung

Wochenplan: Geplante Themen&Anwendungen

Thema	**Lernziele**
Python (kurze Übersicht, plus Videos)	Grundlagen, Funktionen, funktionale Programmierung, Objektorientierung und automatisches Testen
IMDb (Teil 1)	Datenmodellierung, relationales Modell
IMDb (Teil 2)	Relationale Algebra
NSA (Teil 1)	SQL Einführung
NSA (Teil 2)	Analytisches SQL, Big Data-Arithmetik, Big Data vs Privatheit, Gegenmaßnahmen
Anfrageoptimierung (Teil 1)	Automatische Anfrageoptimierung, Physische Operatoren, Heuristische Optimierung
Anfrageoptimierung (Teil 2)	kostenbasierte Optimierung, Joinreihenfolge, Planvarianten, Pipelining, Physische Optimierungen
Handel, Banken, Ticketsystem (Teil 1)	Datenbankmanagementsysteme (DBMS), Transaktionen, Serialisierbarkeitstheorie
Handel, Banken, Ticketsystem (Teil 2)	Two-Phase Locking (2PL), Isolationsstufen
Datenjournalismus (Teil 1)	Pivottabellen, Graphdaten, SQL vs Graphdatenbanken, WITH RECURSIVE, Cypher
Datenjournalismus (Teil 2)	SQL-Injection, Ablegen von Passwörtern, Grundlegende Sicherheitsmaßnahmen
Data in the Wild	Uni vs Realität
Zusammenfassung	

Abgrenzung zur Stammvorlesung Database Systems

Diese (Grund-)Vorlesung “Big Data Engineering”

Fokus auf Prinzipien, Entwurfsmuster und Anwendung von Big Data-Technologien

Stammvorlesung “Database Systems”

tieferer Einstieg in die zugrundeliegenden Techniken

Dozent: Prof. Dr. Jens Dittrich

Forschung:

- Big Data Analytics, Scalable Data Management, Data Science
- ACM SIGMOD, (P)VLDB, CIDR, ...

Lehre:

- Busy beaver awards 2011, 2013, 2018, 2021
- YouTube: https://www.youtube.com/user/jensdit
- Programmkoordinator BSc und MSc Data Science and Artificial Intelligence (seit WS 19/20)

Industrie:

- Data Science Startup Daimond GmbH, https://daimond.ai

https://bigdata.uni-saarland.de/people/dittrich.php

Tutorinnen und Tutoren

Joris Nix (Cheftutor, Doktorand)

Angelina Göbl

Patrick Gräfe

Franziska Granzow

Florian Kneip

Janine Lohse

Niklas Mück

Simon Rink

https://cms.sic.saarland/bde22/tutors/

Vorlesung

Infos zur Vorlesung

- Jeden Donnerstag 10:15 – 12:00 in Gebäude E2.2, Hörsaal 0.01 (Günter-Hotz-Hörsaal) (Kalender)
- Aufzeichnung der Vorlesung anschließend auf YouTube online

Materialien aus der Vorlesung

- Folien: CMS
- Code: GitHub

Die Uni in Zeiten von Corona

- FAQ der Uni
- Robert-Koch Institut

Office Hours

Reguläre Office Hour

- Jeden Mittwoch um 12:15 – 14:00 (Kalender)
- Die Office Hours finden in E1.1, R 3.06 statt.
- Fragen zu Übungen und Konzepten aus der Vorlesung.

Office Hour Prof

- Direkt nach jeder Vorlesung.
- Fragen zur Vorlesung.

Vagrant Support

- Mittwoch 20.04. von 12:15 – 14:00 in E1.1, R 3.06
- Unterstützung beim Einrichten der Vagrant VM.

Übungsgruppen

Übungsgruppen finden montags und dienstags vor Ort (Seminarraum E1.1, R 3.06) und online (Discord) statt. (Terminübersicht).

Prinzip: als LAB gestaltet

1. 15 Minuten Lösungshinweise zur vergangenen, abgegebenen Übung
2. 75 Minuten Teamarbeit: einfache Übungsaufgaben lösen

In jedem Tutorium geht es thematisch jeweils um das Material einer 90 Minuten Vorlesungseinheit.

Teilnahme

- Bis 14.04. 23:59 an der Umfrage bezüglich Tutorien online oder vor Ort im Forum teilnehmen.
- Bis 20.04. 23:59 auf Ihrer persönlichen Statusseite im CMS präferierte Zeitslots wählen.
- Anschließend teilen wir Sie den Übungsgruppen zu.
- Sollten Sie mal verhindert sein, können Sie gerne an einer anderen Übungsgruppe teilnehmen.

Übungen

Infos zu Übungen

- Ausgabe: am Donnerstagabend nach der Vorlesung im CMS
- Abgabe: vor Beginn der nächsten Vorlesung im CMS
- Abgabe in Gruppen von 2 bis 3 Studierenden
- Genauere Infos zur Abgabe auf der jeweiligen Übung
- Musterlösungen werden im CMS zur Verfügung gestellt.

Klausurzulassung

- In Summe müssen mindestens 50% der Punkte erreicht werden, um zur Abschluss- und Wiederholungsklausur zugelassen zu werden.
- Im Semester maximal zwei Übungszettel mit 0 Punkten oder unbearbeitet.

Klausuren

Klausurtermine

- Klausur: Donnerstag, 28.07.2022, 10 bis 12 Uhr
- Nachklausur: Donnerstag, 22.09.2022, 10 bis 12 Uhr

Bestehen und Note

- Zum Bestehen müssen mindestens 50% der Punkte in Klausur oder Nachklausur erreicht werden.
- Die Note wird zu 100% vom besseren Ergebnis aus Klausur und Nachklausur bestimmt.

Python

Wir verwenden in dieser Vorlesung Python und insbesondere Jupyter Notebooks um Konzepte zu erklären.

Python Basics

- Wir stellen Notebooks und Videos mit den Grundlagen zur Verfügung.
- Wir gehen davon aus, dass Sie Prog 1 und Prog 2 gehört haben.

Vagrant VM

- Da wir neben Python auch verschiedene Systeme verwenden (Apache Spark, Neo4j, PostgreSQL) stellen wir eine virtuelle Maschine zur Verfügung.
- Eine Anleitung, wie Sie die VM aufsetzen, finden sie hier.
- In der ersten Übungsgruppe und in der speziellen Office Hour helfen wir Ihnen bei Problemen.
- Bitte versuchen Sie es zunächst selbständig!

Virtualisierung

- Abstraktionsschicht zwischen Anwendung und Hardware
- Virtualisierungssoftware, z.B. VirtualBox