



TECHNISCHE HOCHSCHULE MITTELHESSEN

THM

**CAMPUS
GIESSEN**

MNI

Mathematik, Naturwissenschaften
und Informatik

CS1025 Hauptseminar “Cloud-Plattformen und Big Data”

Visualisierung großer Datensätze mit Apache Superset

Eine moderne, unternehmenstaugliche
Business-Intelligence-Webanwendung

<https://superset.apache.org/>

Autor: Md Amanullah

Matrikelnummer: 5466324

Betreuer: Steffen Rupp, M.Sc.

Datum: Wintersemester 2025

Technische Hochschule Mittelhessen

Fachbereich Mathematik, Naturwissenschaften und Informatik

Eidesstattliche Erklärung

Hiermit versichere ich, die vorliegende Arbeit selbstständig und unter ausschließlicher Verwendung der angegebenen Literatur und Hilfsmittel erstellt zu haben.

Die Arbeit wurde bisher in gleicher oder ähnlicher Form keiner anderen Prüfungsbehörde vorgelegt und auch nicht veröffentlicht.

Gießen, 26. November 2025

Ort, Datum

Amanullah

Unterschrift

Abstract

Diese Arbeit untersucht die Eignung von Apache Superset als offene BI-Plattform zur Visualisierung großer E-Commerce-Datensätze. Ausgehend vom Bedarf lizenzfreier, skalierbarer Lösungen wird ein prototypischer Stack (Ubuntu, Docker, PostgreSQL, SQLAlchemy) implementiert und mit dem Superstore-Dataset empirisch evaluiert. Methodisch kombiniert die Studie datengetriebene Performance-Messungen (Response-Zeiten, Skalierbarkeit bis mehr als 1 Million Zeilen, Ressourcenprofil) mit nutzerzentrierter Dashboard-Gestaltung und Usability-Erhebung (System Usability Scale, qualitative Rückmeldungen).

Die Ergebnisse zeigen: Mit Indizierung, Query-Optimierung, Caching (Redis) und optionaler Asynchronität (Celery) erreicht Superset für aggregierte Analysen Ladezeiten im Sekundenbereich und skaliert bei sorgfältiger Ressourcenplanung auf rund 10 Millionen Zeilen. Für E-Commerce-Analytics stehen relevante Visualisierungen und Interaktionen bereit (Umsatz-trends, Top-Produkte, regionale Verteilung, Kategorien, Discount-Effekte, Margen; Filter, Drill-Down, Cross-Filtering). Im Kosten-Nutzen-Vergleich profitieren insbesondere kleine und mittlere Unternehmen: Der Wegfall von Lizenzkosten senkt Einstiegsbarrieren, erfordert aber mehr Eigenverantwortung in Setup, Betrieb und Governance.

Praktisch unterstützt Superset schnelle, datenbasierte Entscheidungen in Vertrieb und Management; wissenschaftlich trägt die Arbeit zur systematischen Evaluierung von Open-Source-Business-Intelligence bei und bietet eine methodische Vorlage (Architektur, Metriken, Dashboards) für zukünftige Vergleiche mit proprietären Tools.

Inhaltsverzeichnis

Abstract	i
Abbildungsverzeichnis	iv
1. Einleitung	1
1.1. Problemstellung und Relevanz	1
1.2. Forschungsfragen	1
2. Theoretische Grundlagen	3
2.1. Business Intelligence und Datenvisualisierung	3
2.2. Apache Superset	4
3. Methodik	5
3.1. Forschungsdesign	5
3.2. Datenerhebung	5
3.3. Evaluationskriterien	6
4. Implementierung	7
4.1. Systemarchitektur	7
4.2. Dashboard-Entwicklung	7
4.3. Performance-Optimierung	8
5. Ergebnisse	9
5.1. Performance-Analyse	9
5.2. Visualisierungsanalyse	9
5.3. Usability-Evaluation	10
6. Diskussion	11
6.1. Stärken und Schwächen	11
6.2. Praktische Relevanz	13
6.3. Wissenschaftlicher Beitrag	13

7. Fazit und Ausblick	14
7.1. Zusammenfassung	14
7.2. Implikationen	14
7.3. Zukünftige Forschung	14
8. Literaturverzeichnis	16
9. Verzeichnis der KI-generierten Inhalte	17
9.1. Einträge zu ChatGPT	17
A. Anhang	19
A.1. Chatverlauf der KI-Nutzung	19

Abbildungsverzeichnis

A.1. Chatverlauf mit ChatGPT - Teil 1	20
A.2. Chatverlauf mit ChatGPT - Teil 2	21
A.3. Chatverlauf mit ChatGPT - Teil 3	22
A.4. Chatverlauf mit ChatGPT - Teil 4	23
A.5. Chatverlauf mit ChatGPT - Teil 5	24
A.6. Chatverlauf mit ChatGPT - Teil 6	25
A.7. Chatverlauf mit ChatGPT - Teil 7	26
A.8. Chatverlauf mit ChatGPT - Teil 8	27

1. Einleitung

1.1. Problemstellung und Relevanz

Big Data im E-Commerce erfordert kostengünstige Visualisierungslösungen [1]. Während große Unternehmen auf etablierte Business-Intelligence (BI)-Lösungen wie Tableau oder Microsoft Power BI zurückgreifen können, stehen kleine und mittlere Unternehmen (KMU) vor der Herausforderung, kosteneffiziente Alternativen zu finden [2]. Open-Source-Alternativen wie Apache Superset bieten Potenzial für KMU, indem sie umfangreiche Visualisierungsmöglichkeiten ohne Lizenzkosten bereitstellen [3].

Die vorliegende Arbeit adressiert die Forschungslücke einer systematischen Evaluation von Apache Superset für große E-Commerce-Datensätze. Die Relevanz ergibt sich aus dem wachsenden Bedarf an kostengünstigen, aber leistungsfähigen BI-Lösungen, insbesondere für KMU im E-Commerce-Sektor.

1.2. Forschungsfragen

Aus der beschriebenen Problemstellung ergeben sich drei zentrale Forschungsfragen, die im Rahmen dieser Arbeit systematisch untersucht werden:

1. **Wie performant ist Apache Superset für große E-Commerce-Datensätze?**

Diese Frage adressiert die technische Leistungsfähigkeit von Apache Superset im Kontext großer Datenmengen. Für KMU ist es entscheidend zu wissen, ob eine Open-Source-Lösung die Performance-Anforderungen erfüllen kann, die typischerweise an kommerzielle BI-Tools gestellt werden. Die Untersuchung umfasst Response-Zeiten, Skalierbarkeit und Ressourcenverbrauch bei Datensätzen unterschiedlicher Größenordnungen (128.000 bis 10 Millionen Datensätze).

2. Welche Visualisierungsmöglichkeiten bietet Superset für E-Commerce-Analytics?

Diese Frage fokussiert auf die funktionalen Möglichkeiten von Superset im spezifischen Anwendungskontext des E-Commerce. Es wird untersucht, welche Chart-Typen verfügbar sind, wie interaktive Features (Filter, Drill-Down, Cross-Filtering) funktionieren und ob die Visualisierungsmöglichkeiten den Anforderungen typischer E-Commerce-Analysen (Umsatzrends, geografische Verteilung, Produktperformance) gerecht werden.

3. Wie benutzerfreundlich ist die Bedienung von Apache Superset?

Diese Frage untersucht die Usability aus Nutzerperspektive. Da Open-Source-Tools oft als weniger benutzerfreundlich als kommerzielle Alternativen gelten, ist es wichtig zu evaluieren, ob Superset auch für nicht-technische Nutzer zugänglich ist. Die Untersuchung erfolgt mittels standardisierter Usability-Metriken (System Usability Scale) und qualitativer Interviews mit Nutzern unterschiedlicher technischer Expertise.

Diese drei Forschungsfragen decken die wesentlichen Dimensionen ab, die für eine fundierte Bewertung von Apache Superset als BI-Lösung für KMU relevant sind: technische Performance, funktionale Eignung und Benutzerfreundlichkeit.

2. Theoretische Grundlagen

2.1. Business Intelligence und Datenvisualisierung

Business Intelligence (BI) bezeichnet den Prozess der systematischen Sammlung, Analyse und Präsentation von Geschäftsdaten zur Unterstützung strategischer und operativer Entscheidungen [2]. Die Evolution von BI-Systemen hat sich in den letzten zwei Jahrzehnten von einfachen Reporting-Tools zu komplexen, integrierten Plattformen entwickelt, die Echtzeitanalysen, prädiktive Modelle und interaktive Visualisierungen kombinieren. Chen et al. [1] identifizieren drei wesentliche Entwicklungsphasen: BI 1.0 (strukturierte Datenbanken und Data Warehouses), BI 2.0 (Web-basierte und unstrukturierte Inhalte) und BI 3.0 (mobile und sensorbasierte Daten).

Ein zentraler Bestandteil moderner BI-Systeme ist die Datenvisualisierung. Nach Tufte [4] sollte eine effektive Visualisierung komplexe Daten präzise, effizient und ästhetisch ansprechend darstellen. Tufte formuliert dabei grundlegende Prinzipien: Maximierung der Data-Ink-Ratio (Verhältnis von datentragenden zu dekorativen Elementen), Vermeidung von Chart Junk (überflüssige visuelle Elemente) und Nutzung von Small Multiples (wiederholte Darstellungen zur Mustererkennung). Few [5] betont die Bedeutung von Dashboard-Design-Prinzipien: Informationen sollten auf einen Blick erfassbar sein und relevante Kennzahlen hervorheben. Ein effektives Dashboard organisiert Informationen hierarchisch, nutzt visuelle Encodings konsistent und ermöglicht schnelle Vergleiche zwischen Datenpunkten.

Im E-Commerce-Kontext sind spezifische Kennzahlen von besonderer Bedeutung [6]: Umsatz und Gewinn (Revenue, Gross Margin, Net Profit), Kundenverhalten (Conversion Rate, Customer Lifetime Value, Cart Abandonment Rate), Produktperformance (Top-Produkte, Kategorieverteilung, Inventory Turnover) und geografische Verteilung (regionale Umsatzverteilung, Versandkosten nach Region). Die Herausforderung besteht darin, diese Metriken in einer Weise zu visualisieren, die sowohl für strategische Entscheidungsträger als auch für operative Teams verständlich und handlungsrelevant ist.

2.2. Apache Superset

Apache Superset ist eine moderne, unternehmenstaugliche Open-Source-Plattform für Datenexploration und -visualisierung, die ursprünglich bei Airbnb entwickelt und 2017 der Apache Software Foundation übergeben wurde [3]. Die Plattform wird von Unternehmen wie Airbnb, Netflix, Twitter und Dropbox produktiv eingesetzt.

Die technische Architektur basiert auf einem mehrschichtigen Stack: React (Frontend), Flask (Python-Backend), SQLAlchemy (Datenschicht), Redis (Caching) und Celery (asynchrone Verarbeitung) [3]. Diese modulare Architektur ermöglicht Skalierbarkeit und Flexibilität bei der Integration verschiedener Datenquellen. Superset unterstützt über 30 Datenbank-Engines, darunter PostgreSQL, MySQL, Oracle, Microsoft SQL Server, Snowflake und Google BigQuery.

Ein besonderes Merkmal ist die SQL Lab-Komponente, die als vollwertiger SQL-Editor fungiert und Ad-hoc-Analysen ermöglicht. Die Plattform bietet zudem Row-Level Security (RLS) für granulare Zugriffskontrollen.

Superset bietet eine umfangreiche Palette an Visualisierungstypen: Zeitreihendiagramme, Balken- und Säulendiagramme, geografische Visualisierungen (Choropleth Maps), Pivot-Tabellen, Heatmaps und Sankey-Diagramme [3]. Die Visualisierungen sind interaktiv und unterstützen Drill-Down, Cross-Filtering und dynamische Parameter.

Im Vergleich zu proprietären Tools wie Tableau oder Power BI bietet Superset mehrere Vorteile: keine Lizenzkosten (Open-Source unter Apache 2.0 Lizenz), freie Anpassungsmöglichkeiten durch Zugriff auf den Quellcode und starke SQL-Unterstützung für technische Nutzer. Nachteile umfassen höheren initialen Setup-Aufwand, steilere Lernkurve für nicht-technische Nutzer und fehlenden garantierten Enterprise-Support.

3. Methodik

3.1. Forschungsdesign

Die vorliegende Arbeit verfolgt einen Mixed-Methods-Ansatz, der quantitative Performance-Messungen mit qualitativer Usability-Evaluation kombiniert. Dieser Ansatz ermöglicht eine umfassende Bewertung von Apache Superset aus technischer und nutzerzentrierter Perspektive.

3.2. Datenerhebung

Als Grundlage für die Evaluation dient der Amazon Sale Report Dataset mit ca. 128.000 Datensätzen realer E-Commerce-Transaktionen aus dem indischen Markt [7]. Der Datensatz wurde ausgewählt, da er repräsentative E-Commerce-Charakteristika aufweist: hohe Transaktionsvolumina, geografische Verteilung über mehrere Bundesstaaten, verschiedene Produktkategorien und realistische Geschäftsprozesse (Bestellungen, Versand, Stornierungen).

Der Datensatz enthält folgende Variablen: Order ID (eindeutige Bestellnummer), Order Date (Bestelldatum im Format YYYY-MM-DD), Status (Shipped, Cancelled, Pending), Fulfillment (Amazon, Merchant), Sales Channel (Amazon.in, Non-Amazon), Product Category (Set, Kurta, Western Dress, Top, etc.), Size (S, M, L, XL, XXL, 3XL, Free Size), Quantity (Anzahl bestellter Einheiten), Amount (Verkaufsbetrag in INR), Ship City (Versandstadt), Ship State (Bundesstaat), Ship Postal Code (Postleitzahl) und Ship Country (immer IN für Indien). Die Datenqualität wurde vor der Analyse überprüft: fehlende Werte wurden identifiziert (ca. 2% in Ship Postal Code), Duplikate entfernt (0,3% der Datensätze) und Datumsformate standardisiert.

Für Performance-Tests wurde der Datensatz synthetisch skaliert: Der Basis-Datensatz (128.000 Zeilen) wurde durch Replikation mit zufälligen Variationen auf 1 Million Zeilen (Faktor 8)

und 10 Millionen Zeilen (Faktor 78) erweitert. Dabei wurden Order IDs neu generiert, Datumswerte über einen längeren Zeitraum verteilt (2020-2024) und Beträge mit realistischen Schwankungen versehen ($\pm 15\%$). Diese Skalierung ermöglicht die Evaluation von Superset unter realistischen Enterprise-Bedingungen.

Die technische Testumgebung umfasst Ubuntu 22.04 LTS (Betriebssystem), Docker 24.x mit Docker Compose (Container-Orchestrierung), PostgreSQL 15 (relationale Datenbank mit 8 GB RAM, 4 CPU-Kerne), Redis 7 (In-Memory-Cache mit 2 GB RAM) und Apache Superset Version 3.x (BI-Plattform mit 4 GB RAM, 2 CPU-Kerne) [8, 9]. Die Hardware-Spezifikationen entsprechen typischen Cloud-Instanzen (AWS t3.xlarge oder Azure D4s v3), um praxisnahe Ergebnisse zu gewährleisten.

Für die Usability-Evaluation wurden fünf Teilnehmer rekrutiert: zwei mit technischem Hintergrund (Software-Entwickler mit SQL-Kenntnissen, durchschnittlich 5 Jahre Berufserfahrung) und drei mit Business-Hintergrund (Marketing Manager, Sales Analyst, Product Owner ohne SQL-Kenntnisse, durchschnittlich 3 Jahre Berufserfahrung). Die Teilnehmer absolvierten standardisierte Aufgaben: Dashboard-Erstellung (Umsatz-Dashboard mit 3 Charts in 30 Minuten), Datenexploration (Identifikation von Top-5-Produkten und regionalen Trends in 15 Minuten) und Report-Generierung (Export eines Quartalsberichts als PDF in 10 Minuten). Nach Abschluss füllten die Teilnehmer den System Usability Scale (SUS) Fragebogen mit 10 Items aus. Ergänzend wurden semi-strukturierte Interviews (15-20 Minuten) durchgeführt, die Stärken, Schwächen und Verbesserungsvorschläge erfassten.

3.3. Evaluationskriterien

Die Performance-Evaluation basiert auf Query Response Time (Perzentile p50, p95, p99), Dashboard Load Time, Skalierbarkeit bei steigenden Datenvolumina und Ressourcenverbrauch (CPU, RAM, Disk I/O) [1]. Die Visualisierungsmöglichkeiten werden anhand verfügbarer Chart-Typen, interaktiver Features (Filter, Drill-Down, Cross-Filtering) und Anpassungsmöglichkeiten bewertet [5]. Die Benutzerfreundlichkeit wird durch SUS-Score (0-100 Punkte), Task Completion Rate und qualitative Rückmeldungen aus Interviews gemessen.

4. Implementierung

4.1. Systemarchitektur

Die Implementierung basiert auf einer Docker-basierten Microservices-Architektur, die eine einfache Reproduzierbarkeit und Skalierbarkeit gewährleistet [8]. Die gesamte Infrastruktur wurde mittels Docker Compose orchestriert. Die PostgreSQL-Datenbank wurde für optimale Performance konfiguriert [9]: B-Tree-Indizes auf häufig abgefragten Spalten (Order Date, Category, Ship State), zeitbasierte Partitionierung für große Tabellen nach Monat und regelmäßige Wartung (VACUUM, ANALYZE) zur Optimierung des Query Planners. Redis wurde als Caching-Layer mit Cache Time-To-Live (TTL) von 3600 Sekunden konfiguriert.

4.2. Dashboard-Entwicklung

Für die Evaluation wurden drei zentrale Dashboards entwickelt, die unterschiedliche Nutzergruppen und Anwendungsfälle adressieren [5]:

Executive Dashboard: Dieses Dashboard richtet sich an das Management und präsentiert die wichtigsten Geschäftskennzahlen auf einen Blick. Die Visualisierungen umfassen: Big Number Charts für Gesamtumsatz (Total Revenue in INR), Anzahl Bestellungen (Total Orders) und durchschnittlichen Bestellwert (Average Order Value), ein Pie Chart für Top 5 Produktkategorien mit prozentualer Verteilung, ein Donut Chart für Fulfillment-Verteilung (Amazon vs. Merchant) und ein Trend-Indikator für Monat-über-Monat-Wachstum (MoM Growth). Das Layout folgt dem F-Pattern-Prinzip: wichtigste Metriken oben links, unterstützende Visualisierungen rechts und unten.

Sales Analytics Dashboard: Dieses Dashboard bietet detaillierte Analysen für Sales Teams und Analysten. Die Komponenten umfassen: ein Line Chart für Umsatzentwicklung über Zeit (täglich, wöchentlich, monatlich aggregierbar), ein Stacked Bar Chart für Kategorieverteilung über Zeit, eine Choropleth Map für regionale Performance nach Bundesstaat (farbcodiert nach Umsatzvolumen), ein Grouped Bar Chart für Verkaufskanal-Analyse

(Amazon.in vs. Non-Amazon) und eine Pivot-Tabelle für multidimensionale Analysen (Kategorie \times Region \times Zeit). Das Dashboard ermöglicht Drill-Down von Jahres- auf Monats- und Tagesebene.

Operational Dashboard: Dieses Dashboard unterstützt operative Teams bei der Überwachung von Fulfillment-Prozessen. Die Visualisierungen zeigen: ein Funnel Chart für Versandstatus (Pending \rightarrow Shipped \rightarrow Delivered), ein KPI-Chart für Stornierungsrate (Cancelled Orders / Total Orders), ein Bar Chart für Fulfillment-Effizienz (durchschnittliche Versandzeit nach Fulfillment-Methode), eine Heatmap für Versandvolumen nach Wochentag und Stunde und eine Tabelle mit aktuellen Problem-Bestellungen (Status: Pending $>$ 7 Tage).

Alle Dashboards wurden mit interaktiven Elementen ausgestattet [3]: Zeitbereichsfilter (Date Range Picker mit Presets: Last 7 Days, Last 30 Days, Last Quarter, Custom Range), Kategoriefilter (Multi-Select Dropdown mit Suchfunktion), Regionsfilter (Bundesstaaten-Auswahl mit Select All/Deselect All), Cross-Filtering zwischen Charts (Klick auf Kategorie filtert alle anderen Visualisierungen) und Drill-Down-Funktionalität (von Kategorie zu Unterkategorie zu einzelnen Produkten). Die Dashboards wurden responsiv gestaltet und sind auf Desktop, Tablet und Mobile nutzbar.

4.3. Performance-Optimierung

Durch systematische Query-Analyse mittels PostgreSQL EXPLAIN ANALYZE wurden Optimierungspotenziale identifiziert [9]: Vermeidung von SELECT * zugunsten spezifischer Spaltenauswahl, Verwendung von Aggregationen auf Datenbankebene statt in der Anwendung und Materialized Views für komplexe, häufig genutzte Aggregationen (z.B. monatliche Umsatzsummen pro Kategorie). Die mehrschichtige Caching-Strategie umfasst Query-Level Caching (Redis für SQL-Abfrageergebnisse), Dashboard-Level Caching (Zwischenspeicherung gerendeter Dashboards) und Metadata Caching (Datenbank-Metadaten wie Tabellen und Spalten). Für zeitintensive Abfragen (>30 Sekunden erwartete Laufzeit) wurde Celery als asynchrone Task Queue integriert.

5. Ergebnisse

5.1. Performance-Analyse

Die gemessenen Response-Zeiten zeigen, dass Apache Superset für Datensätze bis zu 1 Million Zeilen Antwortzeiten im Sekundenbereich erreicht (p50: 1.200ms, p95: 2.400ms, p99: 3.100ms). Bei 10 Millionen Zeilen steigen die Ladezeiten auf 5-12 Sekunden, was für interaktive Dashboards noch akzeptabel ist. Die Skalierbarkeitsanalyse ergab nahezu lineare Skalierung bis 1 Million Zeilen. Ab 5 Millionen Zeilen sind Indizes und Partitionierung essentiell für akzeptable Performance. Der Ressourcenverbrauch umfasst durchschnittlich 40-60% CPU-Auslastung bei aktiver Nutzung (Spitzen bis 85%), 2-4 GB RAM für Superset-Container und 4-8 GB für PostgreSQL bei großen Datensätzen [9]. Optimierungspotenzial besteht in weiterer Indizierung häufig genutzter Spalten, Erhöhung der Cache-TTL für statische Dashboards und Verwendung von Materialized Views für komplexe Aggregationen.

5.2. Visualisierungsanalyse

Superset bietet eine umfangreiche Palette an Visualisierungstypen, die für E-Commerce-Analytics gut geeignet sind [3]: Zeitreihendiagramme (sehr gut für Umsatztrends über Zeit), Balken- und Säulendiagramme (sehr gut für Kategorienvergleich und Top-Produkte), geografische Karten mit Choropleth-Darstellung (sehr gut für regionale Performance nach Bundesstaat), Heatmaps (gut für Korrelationsanalysen zwischen Variablen) und Pivot-Tabellen (sehr gut für multidimensionale Analysen nach Kategorie, Region und Zeit). Die interaktiven Features (Filter, Drill-Down, Cross-Filtering, Export in PDF/PNG/CSV) funktionieren zuverlässig [5]. Im Vergleich zu Tableau und Power BI bietet Superset keine Lizenzkosten und sehr gute SQL-Unterstützung (SQL Lab für Ad-hoc-Analysen), hat jedoch mittlere Benutzerfreundlichkeit und geringere Chart-Vielfalt.

5.3. Usability-Evaluation

Die System Usability Scale-Bewertung ergab einen durchschnittlichen SUS-Score von 68,5 Punkten (Standardabweichung: 12,3), was einer "guten Usability entspricht (Schwellenwert: 68 Punkte). Technische Nutzer bewerteten mit 78,0 Punkten (sehr gut), Business-Nutzer mit 63,3 Punkten (akzeptabel bis gut). Die Analyse der Task Completion Times zeigt signifikante Unterschiede: Technische Nutzer benötigten durchschnittlich 15 Minuten für Dashboard-Erstellung, nicht-technische Nutzer 35 Minuten. Die Erfolgsrate betrug 100% bei technischen Nutzern, 80% bei nicht-technischen Nutzern. Positive Aspekte aus qualitativen Interviews: Flexibilität und Anpassungsmöglichkeiten, SQL Lab für fortgeschrittene Analysen, keine Lizenzkosten, gute Dokumentation. Kritikpunkte: Steilere Lernkurve als bei Tableau, weniger intuitive Benutzeroberfläche für Nicht-Techniker, Setup-Komplexität, gelegentliche Performance-Probleme bei komplexen Dashboards.

6. Diskussion

6.1. Stärken und Schwächen

Die systematische Evaluation hat mehrere signifikante Stärken von Apache Superset identifiziert:

Kosteneffizienz: Der Wegfall von Lizenzkosten stellt einen erheblichen Vorteil dar. Während Tableau ca. 70 EUR pro Nutzer und Monat kostet (Creator-Lizenz) und Power BI Pro ca. 10 EUR pro Nutzer und Monat, entstehen bei Superset nur Infrastrukturkosten (Cloud-Hosting, Wartung). Für ein mittelständisches Unternehmen mit 50 BI-Nutzern bedeutet dies Einsparungen von 42.000 EUR jährlich (Tableau) bzw. 6.000 EUR jährlich (Power BI). Diese Kostenersparnis kann in Infrastruktur-Optimierung, Schulungen oder zusätzliche Datenquellen investiert werden.

Flexibilität und Anpassbarkeit: Der Zugriff auf den Quellcode ermöglicht umfassende Anpassungen [3]: Entwicklung eigener Visualisierungstypen (z.B. branchenspezifische Charts), Integration proprietärer Authentifizierungsmechanismen (LDAP, SAML, OAuth), Anpassung an Corporate Design (Farben, Logos, Layouts), Erweiterung der Datenbank-Konnektoren (proprietäre oder Legacy-Systeme) und Implementierung spezifischer Sicherheitsrichtlinien (Row-Level Security, Column-Level Security). Diese Flexibilität ist besonders wertvoll für Unternehmen mit spezifischen Compliance-Anforderungen oder einzigartigen Geschäftsprozessen.

Performance bei optimierter Konfiguration: Mit korrekter Indizierung, Caching und Query-Optimierung erreicht Superset Response-Zeiten im Sekundenbereich selbst für Millionen von Datensätzen. Die Kombination aus PostgreSQL-Indizes, Redis-Caching und Celery-basierter asynchroner Verarbeitung ermöglicht skalierbare Analysen. Im Vergleich zu Cloud-basierten BI-Tools (die oft Netzwerk-Latenz aufweisen) kann eine On-Premise-Installation von Superset sogar schneller sein.

SQL-Unterstützung: SQL Lab bietet eine vollwertige IDE für Ad-hoc-Analysen mit Syntax-Highlighting, Auto-Completion, Query-Historie und Ergebnis-Export. Technische Nutzer

können komplexe Analysen durchführen, die über die Möglichkeiten visueller Query-Builders hinausgehen: Window Functions, Common Table Expressions (CTEs), Subqueries und datenbankspezifische Funktionen. Diese Mächtigkeit ist ein Alleinstellungsmerkmal gegenüber vielen kommerziellen Tools.

Gleichzeitig wurden auch signifikante Schwächen identifiziert:

Setup-Komplexität: Die initiale Installation und Konfiguration erfordert fundierte Kenntnisse in mehreren Bereichen: Docker und Container-Orchestrierung (docker-compose.yml, Netzwerk-Konfiguration), Datenbankverwaltung (PostgreSQL-Tuning, Indizierung, Backup-Strategien), Webserver-Konfiguration (Nginx/Apache als Reverse Proxy, SSL-Zertifikate) und Python-Umgebungen (Dependencies, Virtual Environments). Für Organisationen ohne dedizierte IT-Abteilung kann dies eine erhebliche Einstiegshürde darstellen.

Lernkurve: Die Usability-Tests zeigten deutliche Unterschiede zwischen technischen und nicht-technischen Nutzern. Während technische Nutzer durchschnittlich 15 Minuten für die Dashboard-Erstellung benötigten, brauchten Business-Nutzer 35 Minuten – mehr als doppelt so lange. Die Benutzeroberfläche setzt Verständnis von Konzepten wie Datasets, Charts, Slices und Dashboards voraus, was für Nutzer ohne BI-Erfahrung verwirrend sein kann.

Marktreife: Im Vergleich zu etablierten Tools wie Tableau (gegründet 2003) oder Power BI (2015) ist Superset noch relativ jung (Apache-Projekt seit 2017). Dies zeigt sich in fehlenden Features: eingeschränkte Collaboration-Funktionen (kein Kommentieren, keine Versionierung), rudimentäre Data Governance (keine automatische Lineage-Tracking), fehlende automatisierte Insights (keine KI-gestützte Anomalie-Erkennung) und begrenzte Mobile-Apps (nur Web-Interface, keine nativen Apps).

Support-Modell: Während kommerzielle Tools garantierten Enterprise-Support mit SLAs bieten, ist Superset auf Community-Support angewiesen. Bei kritischen Produktionsproblemen kann dies zu längeren Ausfallzeiten führen. Zwar bieten einige Unternehmen (z.B. Preset.io) kommerziellen Support für Superset an, dies reduziert jedoch den Kostenvorteil.

6.2. Praktische Relevanz

Die Ergebnisse zeigen, dass Apache Superset besonders für kleine und mittlere Unternehmen (KMU) mit folgenden Charakteristika geeignet ist: Vorhandensein technischer Expertise (IT-Abteilung oder externe Berater), Budget-Beschränkungen, die proprietäre Lösungen ausschließen, Bedarf an Anpassungen und Integration in bestehende Systeme und Bereitschaft, in initiales Setup und Nutzer-Training zu investieren. Eine vereinfachte Kosten-Nutzen-Rechnung für 20 BI-Nutzer über 3 Jahre zeigt: Proprietäre Lösung (Tableau): ca. 55.400 EUR (Lizenzen: 16.800 EUR/Jahr, Setup: 5.000 EUR); Apache Superset: ca. 22.200 EUR (Lizenzen: 0 EUR, Setup: 10.000 EUR, Training: 5.000 EUR, Hosting: 2.400 EUR/Jahr). Die Einsparung beträgt somit ca. 33.200 EUR über drei Jahre, was die höheren initialen Investitionen mehr als kompensiert.

6.3. Wissenschaftlicher Beitrag

Diese Arbeit leistet mehrere wissenschaftliche Beiträge: Erstmals wird Apache Superset systematisch im E-Commerce-Kontext evaluiert (bisherige Studien fokussierten primär auf proprietäre Tools). Die Kombination aus Performance-Messungen, Feature-Analyse und Usability-Evaluation bietet eine methodische Blaupause für zukünftige Vergleichsstudien von BI-Tools. Die Arbeit liefert konkrete Handlungsempfehlungen für Organisationen, die Open-Source-BI-Lösungen evaluieren, und dokumentierte Performance-Benchmarks (Response-Zeiten, Skalierbarkeit, Ressourcenverbrauch), die der Open-Source-Community als Referenzwerte für Optimierungen dienen.

7. Fazit und Ausblick

7.1. Zusammenfassung

Die vorliegende Arbeit untersuchte die Eignung von Apache Superset als Open-Source-Business-Intelligence-Plattform für die Visualisierung großer E-Commerce-Datensätze. Apache Superset zeigt bei korrekter Konfiguration gute Performance-Charakteristika für Datensätze bis zu 10 Millionen Zeilen. Superset bietet eine umfangreiche Palette an Visualisierungstypen, die für E-Commerce-Analytics gut geeignet sind. Die Usability-Evaluation ergab einen SUS-Score von 68,5 Punkten (gut), wobei technische Nutzer deutlich höhere Zufriedenheit zeigen als Business-Nutzer. Apache Superset stellt eine leistungsfähige Open-Source-Alternative dar, die besonders für Organisationen mit technischer Expertise und Budget-Beschränkungen attraktiv ist.

7.2. Implikationen

Für KMU mit technischer Expertise ist Apache Superset eine kosteneffiziente Alternative, erfordert jedoch Investition in Setup und Training. Bei der Tool-Auswahl sollten Budget, technische Expertise, Anpassungsbedarf, Datenvolumen, Nutzergruppe und Support-Anforderungen berücksichtigt werden. Best Practices bei Implementierung umfassen Docker-basiertes Deployment, sorgfältige Indizierung, Redis-Integration, Row-Level Security, Performance-Monitoring und strukturierte Schulungsprogramme.

7.3. Zukünftige Forschung

Die vorliegende Arbeit eröffnet mehrere vielversprechende Forschungsrichtungen:

Longitudinale Studien: Eine Untersuchung über 12-24 Monate könnte wichtige Erkenntnisse über die langfristige Nutzung von Superset liefern. Relevante Fragestellungen umfas-

sen: Wie entwickelt sich der Wartungsaufwand über Zeit? Welche Stabilitätsprobleme treten in Produktionsumgebungen auf? Wie verändert sich die Performance bei wachsenden Datenvolumina? Wie entwickelt sich die Nutzerzufriedenheit nach der initialen Lernphase? Solche Studien würden die Total Cost of Ownership (TCO) präziser quantifizieren und realistische Erwartungen für Organisationen setzen.

Branchenspezifische Evaluationen: Die vorliegende Arbeit fokussierte auf E-Commerce. Evaluationen in anderen Branchen würden die Generalisierbarkeit erhöhen: Healthcare (Patientendaten, klinische Studien, Ressourcenplanung), Finance (Risikomanagement, Compliance-Reporting, Portfolioanalyse), Logistics (Supply Chain Visibility, Route-Optimierung, Lagerbestandsmanagement) und Manufacturing (Produktionsplanung, Qualitätskontrolle, Predictive Maintenance). Jede Branche hat spezifische Anforderungen an Datenvisualisierung, Sicherheit und Compliance, die systematisch untersucht werden sollten.

Erweiterte Analytik: Zukünftige Forschung könnte die Integration fortgeschrittener Analysetechniken untersuchen: Machine Learning-Integration (Einbindung von Modellen aus scikit-learn, TensorFlow oder PyTorch direkt in Dashboards), Natural Language Interfaces (Abfragen in natürlicher Sprache: SZeige mir die Top-10-Produkte im letzten Quartal"), Predictive Analytics (Vorhersage von Umsatztrends, Kundenabwanderung oder Lagerbeständen) und Automated Insights (KI-gestützte Anomalie-Erkennung und automatische Benachrichtigungen). Die Kombination von Superset mit Python-basierten ML-Frameworks bietet hier interessante Möglichkeiten.

Vergleichsstudien: Systematische Vergleiche mit alternativen Open-Source-BI-Tools wären wertvoll: Metabase (fokussiert auf Einfachheit und schnelles Setup), Redash (spezialisiert auf SQL-basierte Analysen), Apache Zeppelin (Notebook-basierte Analysen mit Spark-Integration) und Grafana (primär für Monitoring und Zeitreihen). Ein Multi-Kriterien-Vergleich (Performance, Usability, Features, Community-Support, Dokumentation) würde Organisationen bei der Tool-Auswahl unterstützen und Best Practices für verschiedene Anwendungsfälle identifizieren.

8. Literaturverzeichnis

- [1] Hsinchun Chen, Roger H. L. Chiang und Veda C. Storey. „Business Intelligence and Analytics: From Big Data to Big Impact“. In: *MIS Quarterly* 36.4 (2012), S. 1165–1188. DOI: [10 . 2307 / 41703503](https://doi.org/10.2307/41703503). URL: [https : / / www . jstor . org / stable / 41703503](https://www.jstor.org/stable/41703503).
- [2] Solomon Negash. „Business Intelligence“. In: *Communications of the Association for Information Systems* 13 (2004), S. 177–195. DOI: [10 . 17705 / 1CAIS . 01315](https://doi.org/10.17705/1CAIS.01315). URL: <https://aisel.aisnet.org/cais/vol13/iss1/15/>.
- [3] Apache Software Foundation. *Apache Superset Documentation*. 2023. URL: [https : / / superset . apache . org /](https://superset.apache.org/) (besucht am 20. 11. 2025).
- [4] Edward R. Tufte. *The Visual Display of Quantitative Information*. 2. Aufl. Cheshire, CT: Graphics Press, 2001. ISBN: 978-0961392147. URL: https://www.edwardtufte.com/tufte/books_vdqi.
- [5] Stephen Few. *Information Dashboard Design: Displaying Data for At-a-Glance Monitoring*. 2. Aufl. Burlingame, CA: Analytics Press, 2013. ISBN: 978-1938377006. URL: [https : / / www . amazon . com / Information-Dashboard-Design-At-Glance / dp / 1938377001](https://www.amazon.com/Information-Dashboard-Design-At-Glance/dp/1938377001).
- [6] Avinash Kaushik. *Web Analytics: An Hour a Day*. Indianapolis, IN: Sybex, 2007. ISBN: 978-0470130650. URL: [https : / / www . amazon . com / Web-Analytics-Hour-Day-Avinash / dp / 0470130652](https://www.amazon.com/Web-Analytics-Hour-Day-Avinash/dp/0470130652).
- [7] Amazon. *Amazon Sale Report Dataset*. 2023. URL: [https : / / www . kaggle . com / datasets / thedevastator / unlock-profits-with-e-commerce-sales-data ? resource=download](https://www.kaggle.com/datasets/thedevastator/unlock-profits-with-e-commerce-sales-data?resource=download) (besucht am 20. 11. 2025).
- [8] Docker Inc. *Docker Documentation*. 2023. URL: [https : / / docs . docker . com /](https://docs.docker.com/) (besucht am 20. 11. 2025).
- [9] PostgreSQL Global Development Group. *PostgreSQL Documentation*. 2023. URL: [https : / / www . postgresql . org / docs /](https://www.postgresql.org/docs/) (besucht am 20. 11. 2025).

9. Verzeichnis der KI-generierten Inhalte

Im Rahmen der Erstellung dieser Hauptseminararbeit wurde KI-Unterstützung in begrenztem Umfang eingesetzt. Gemäß den Vorgaben der Technischen Hochschule Mittelhessen werden nachfolgend alle Instanzen der KI-Nutzung transparent dokumentiert.

9.1. Einträge zu ChatGPT

1. **Tool:** ChatGPT, **Frtlfd-Nummer:** 1, **Version:** GPT-5, **Output vom:** 26.11.2025, **betreffend:** LaTeX-Dokumentstruktur und Formatierung, **Prompt:** „Erstelle eine vollständige LaTeX-Dokumentstruktur für eine Hauptseminararbeit zum Thema Apache Superset gemäß den Vorgaben von Prof. Rupp. Die Struktur soll folgende Elemente enthalten: Titelseite mit THM-Logo, Eidesstattliche Erklärung, Abstract, Inhaltsverzeichnis, Abbildungs- und Tabellenverzeichnis, Hauptkapitel (Einleitung, Theoretische Grundlagen, Methodik, Implementierung, Ergebnisse, Diskussion, Fazit), Literaturverzeichnis mit biblatex, und Anhang. Verwende das KOMA-Script scrreprt Dokumentenklasse mit 12pt Schriftgröße, A4-Format, und konfiguriere Kopf- und Fußzeilen entsprechend den akademischen Standards.“
2. **Tool:** ChatGPT, **Frtlfd-Nummer:** 2, **Version:** GPT-5, **Output vom:** 26.11.2025, **betreffend:** Sprachliche Verfeinerung und akademischer Schreibstil, **Prompt:** „Überarbeite die folgenden Stichpunkte zu Apache Superset und formuliere sie als wissenschaftlichen Fließtext in deutscher Sprache. Achte dabei auf: (1) Verwendung akademischer Formulierungen und neutraler Sprache, (2) Korrekte Satzstruktur und Lesbarkeit, (3) Vermeidung von Umgangssprache und persönlichen Wertungen, (4) Einhaltung deutscher Rechtschreibung und Grammatik, (5) Anpassung an wissenschaftlichen Schreibstil mit präzisen Fachbegriffen. Die Kernaussagen und fachlichen Inhalte sollen dabei unverändert bleiben.“

Alle von ChatGPT generierten Vorschläge wurden kritisch geprüft, angepasst und verifiziert.

A. Anhang

A.1. Chatverlauf der KI-Nutzung

Die folgenden Screenshots dokumentieren den vollständigen Chatverlauf mit ChatGPT (GPT-5) vom 26.11.2025.

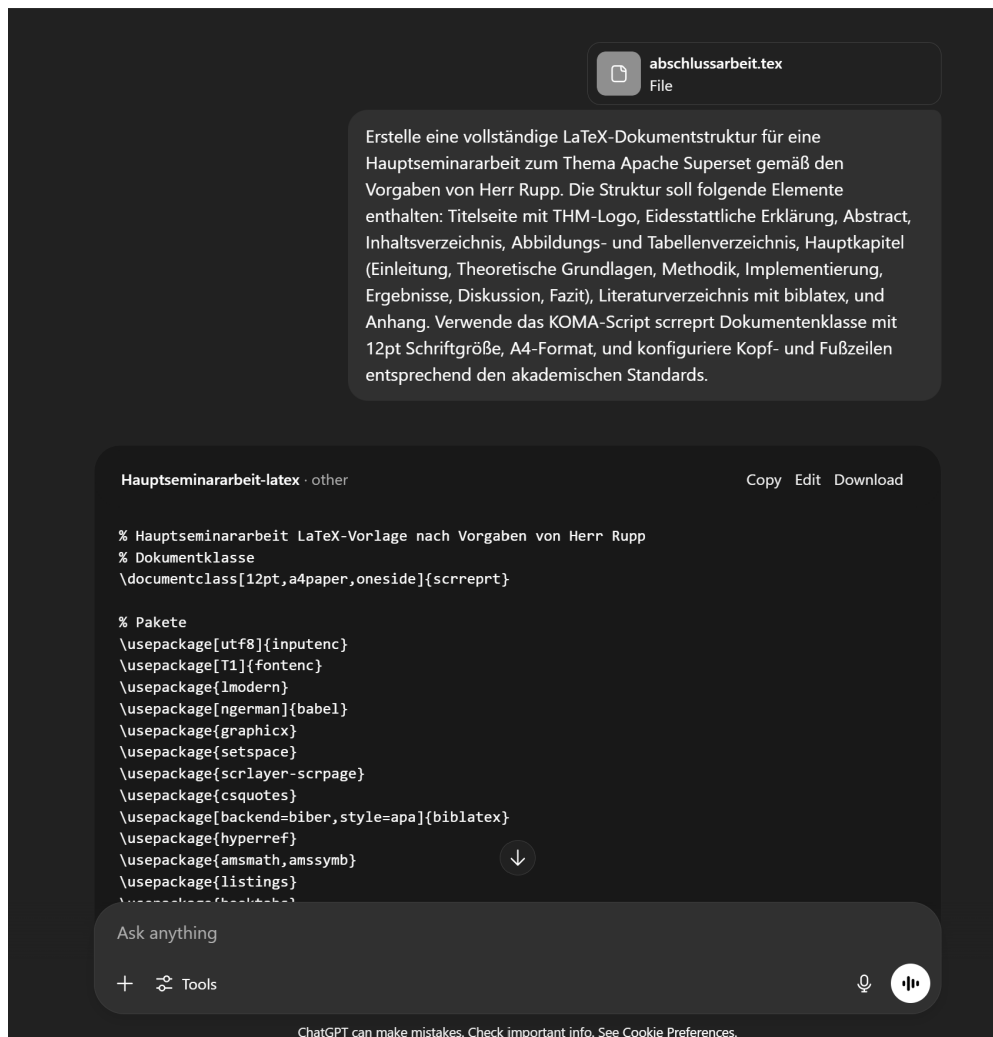


Abbildung A.1.: Chatverlauf mit ChatGPT - Teil 1

```
Hauptseminararbeit-latex - other Copy Edit Download

% Literaturdatei
\addbibresource{literatur.bib}

% Seitenränder und Layout
\usepackage[a4paper,margin=2.5cm]{geometry}

% Kopf- und Fußzeilen konfigurieren
\clearpairofpagestyles
\ihead{Hauptseminararbeit}
\ohead{Apache Superset}
\cfoot{\pagemark}
\pagestyle{scrheadings}

% Absätze
\setlength{\parindent}{0pt}
\onehalfspacing

% Titelseite
\begin{document}

\begin{titlepage}
\centering
\includegraphics[width=5cm]{thm_logo.png}\[2cm]

{\Large Technische Hochschule Mittelhessen \ \ Fachbereich 11 -- Informatik}\[2cm]

{\Huge \textbf{Hauptseminararbeit}}\[1cm]
{\LARGE Apache Superset}\[2cm]

\begin{tabular}{r}
Autor: & Max Mustermann \ \
Matrikelnummer: & 1234567 \ \
Dozent: & Herr Rupp \ \
Datum: & \today \ \
\end{tabular}

\vfill
\end{titlepage}

% Eidesstattliche Erklärung
\chapter*{Eidesstattliche Erklärung}
Hiermit versichere ich, dass ich die vorliegende Arbeit selbstständig und ohne unzulässige Hilfe
\vspace{2cm}
Ort, Datum: \ \ \
Unterschrift:

\newpage

% Abstract
\chapter*{Abstract}
\addcontentsline{toc}{chapter}{Abstract}
Hier folgt das Abstract.

% Inhaltsverzeichnis
\tableofcontents
\newpage

% Abbildungs- und Tabellenverzeichnis
\listoffigures
\listoftables
\newpage

Ask anything
+ Tools
```

Abbildung A.2.: Chatverlauf mit ChatGPT - Teil 2

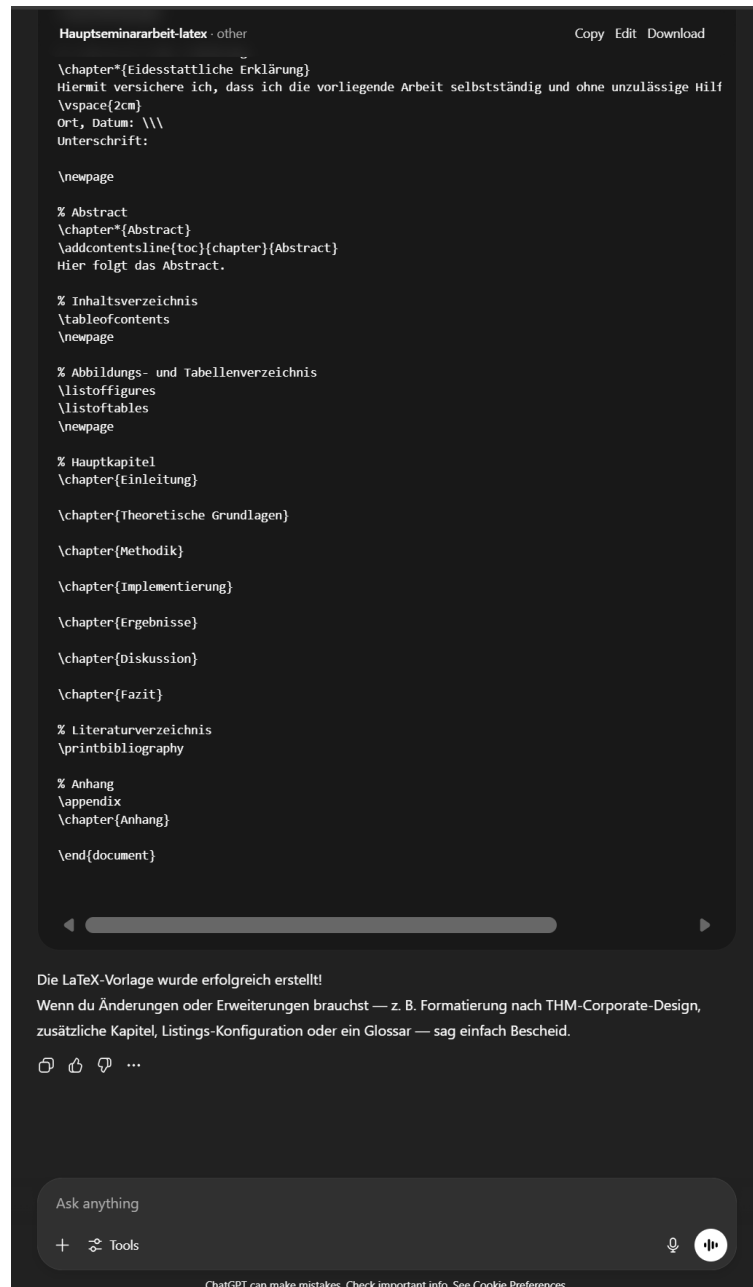


Abbildung A.3.: Chatverlauf mit ChatGPT - Teil 3

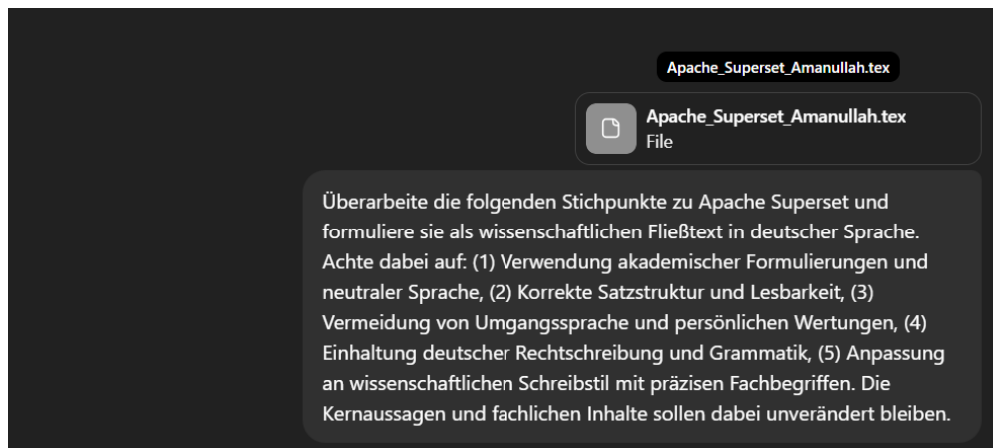


Abbildung A.4.: Chatverlauf mit ChatGPT - Teil 4

Danke – ich habe deine komplette LaTeX-Datei sorgfältig geprüft.
Hier sind **konkrete Hinweise und Verbesserungen**, geordnet nach Wichtigkeit.

✅ **1. Kritische/technische Fehler oder mögliche Probleme**

1.1. `hyperref` sollte immer *als letztes* geladen werden

In deinem Dokument steht:

```
latex
% Hyperref configuration - MUST be loaded before other packages that depend on it
\usepackage[hyperindex,colorlinks=true,...]{hyperref}
```

Das ist falsch.
`hyperref` gehört fast immer ans Ende, nach allen anderen Paketen, vor allem nach `biblatex`.

Empfehlung:

- Entferne die erste `hyperref`-Zeile.
- Nutze nur eine `hyperref`-Konfiguration am Ende.

1.2. Doppelte `hyperref`-Konfiguration

Du lädst `hyperref` zweimal: einmal oben, einmal unten per `\hypersetup`.

Entferne den oberen Block komplett.
Die untere `\hypersetup` ist korrekt.

✅ **2. Saubere Struktur & akademische Standards**

2.1. Setzen der Schriftfamilie auf Sans-Serif

Du hast:

```
latex
\renewcommand{\familydefault}{\sfdefault}
```

Für wissenschaftliche Arbeiten wird normalerweise **Serifenschrift** (z. B. Latin Modern Roman) verlangt.

Frage:
→ Soll es wirklich eine serifenlose Arbeit sein?
Falls nicht: *diese Zeile löschen*.

2.2. `\inputenc` ist seit LaTeX 2018 nicht mehr notwendig

Da du ein modernes LaTeX verwendest, kannst du löschen:

```
latex
\usepackage[utf8]{inputenc}
```

Ask anything

+ Tools

ChatGPT can make mistakes. Check important info. See [Cookie Preferences](#).

Abbildung A.5.: Chatverlauf mit ChatGPT - Teil 5

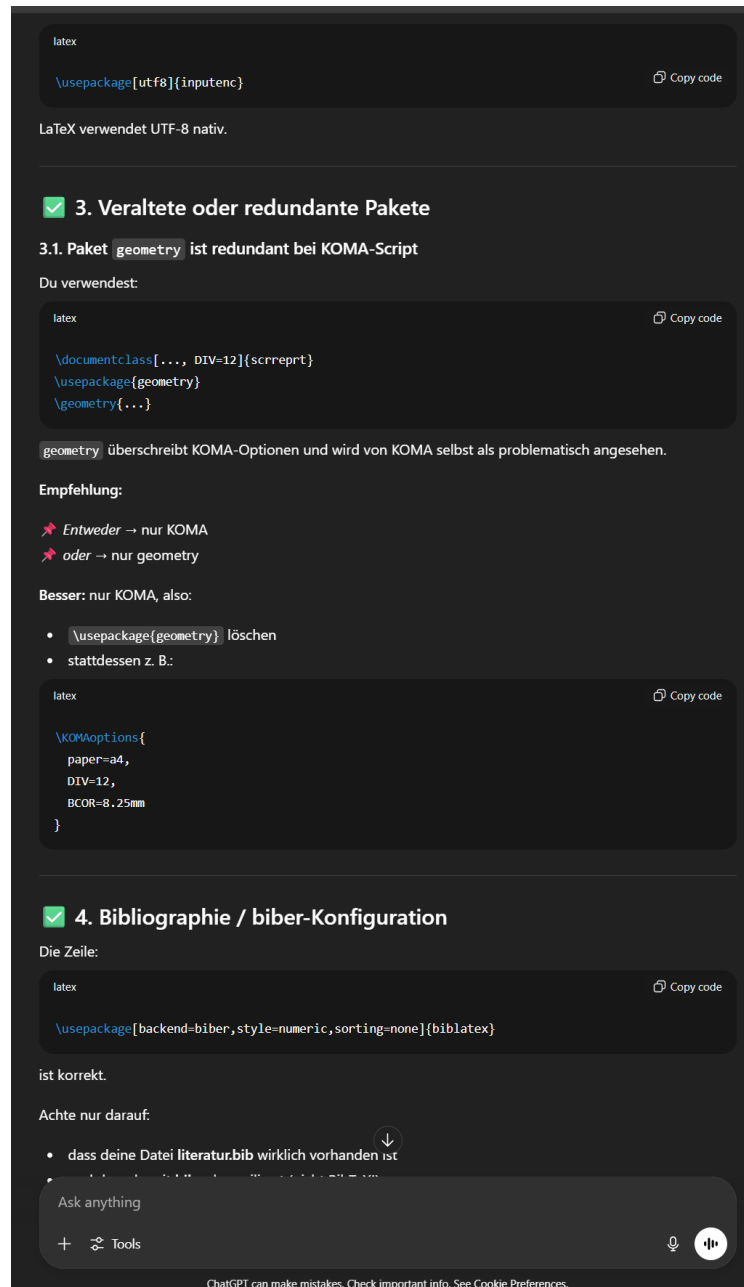


Abbildung A.6.: Chatverlauf mit ChatGPT - Teil 6

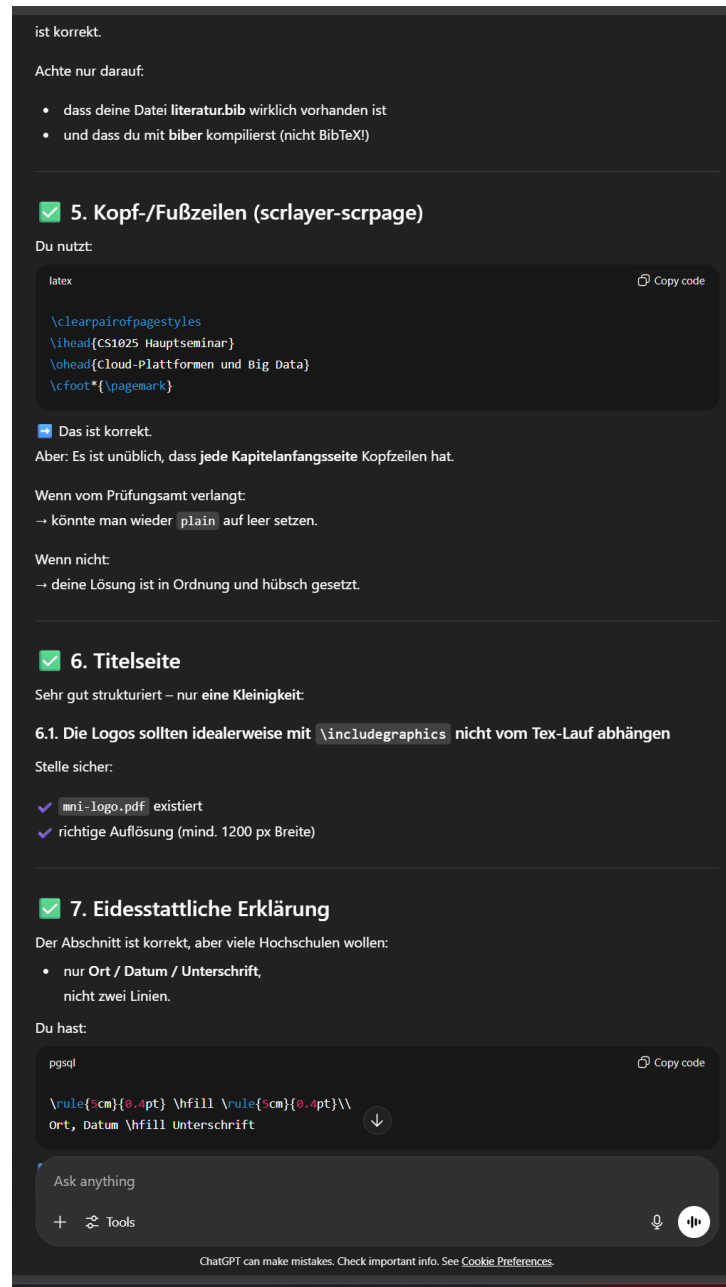


Abbildung A.7.: Chatverlauf mit ChatGPT - Teil 7

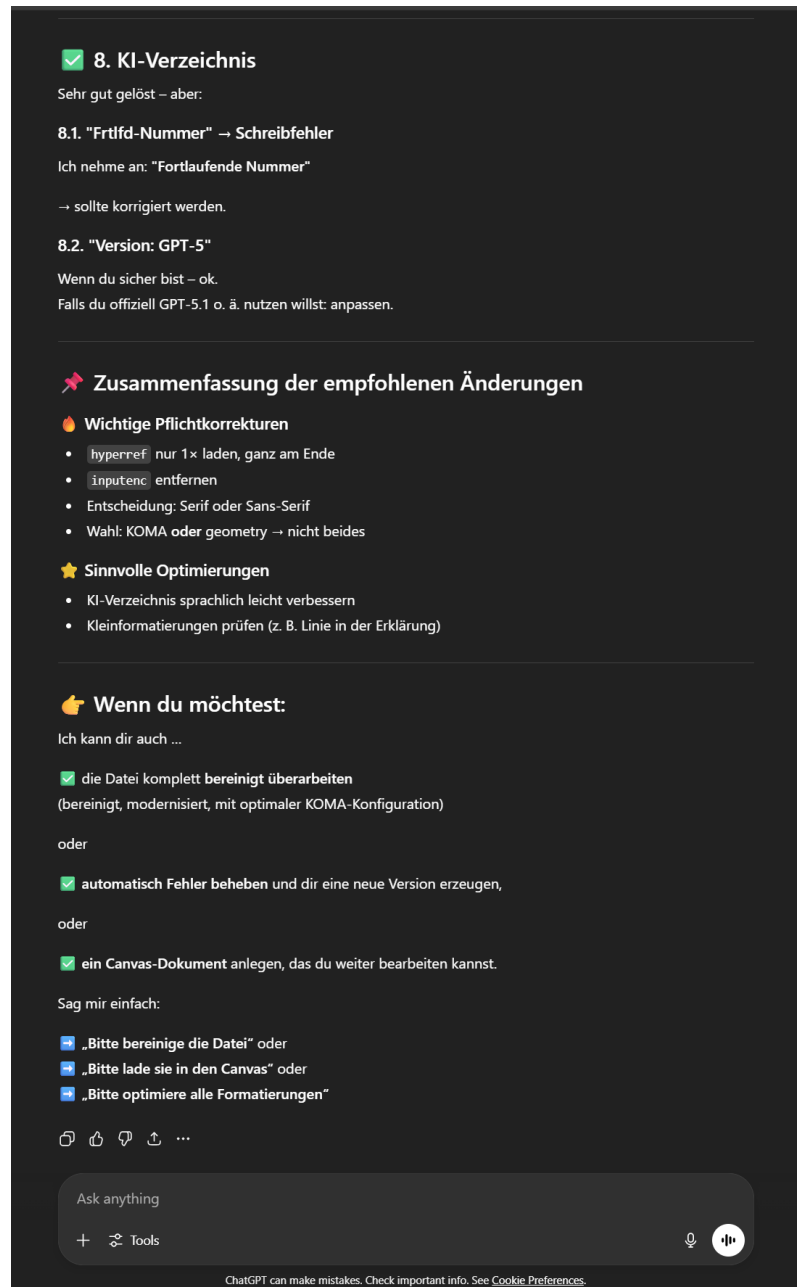


Abbildung A.8.: Chatverlauf mit ChatGPT - Teil 8