## **Projektbeschreibung: Dynamische Datenanalyse mit LLM und Prompt-Evolution**

### **Datum: 29. September 2025**

### **1. Ziel der Untersuchung**

### Das Hauptziel dieses Projekts war es, die grundsätzliche Machbarkeit des Einsatzes von Großen Sprachmodellen (LLMs) für die flexible Analyse beliebiger Datenbanken ohne vorprogrammierte Algorithmen zu untersuchen und nachzuweisen.

### Wir wollten uns vom traditionellen Ansatz lösen, bei dem für jede neue Anfrage an eine Datenbank spezifischer Code (z. B. eine SQL-Abfrage) geschrieben werden muss. Stattdessen strebten wir ein System an, in dem ein Benutzer Fragen in natürlicher Sprache stellen kann und das Modell, mit dem richtigen Kontext versorgt, selbst "versteht", wie die Antwort zu finden und zu berechnen ist.

### Die Schlüsselhypothese war, dass es durch eine Evolution von Prompting-Ansätzen – von den einfachsten bis zu den komplexesten – möglich ist, ein universelles Sprachmodell so anzuleiten, dass es wie ein spezialisierter Analyst für einen bestimmten Datensatz agiert.

### **2. Vorgehensweise: Methodik des Experiments**

### Um unsere Hypothese zu überprüfen, wurde ein umfassendes Experiment konzipiert und durchgeführt, das die folgenden Schritte umfasste:

### **Werkzeugauswahl:** Wir testeten zwei Llama 3.1-Modelle unterschiedlicher Größe (8B und 405B), um zu bewerten, wie die Modellgröße die Fähigkeit zur Befolgung von Anweisungen beeinflusst. Als Testdatensatz wurde die MAZAK-Datenbank verwendet.

### **Entwicklung eines evolutionären Prompt-Systems:** Anstelle eines einzigen "perfekten" Prompts haben wir fünf aufeinander aufbauende Ansätze entwickelt, die die gedankliche Evolution bei der Lösung einer analytischen Aufgabe nachahmen:

### **Basic (Grundlegend):** Eine direkte Frage ohne Kontext. Dient zur Überprüfung des Basis-"Wissens" und der Neigung zu Halluzinationen.

### **Expert (Experte):** Das Modell erhält eine Rolle und minimalen Kontext. Dient zur Überprüfung, wie grundlegende Hinweise die Denkweise des Modells beeinflussen.

### **Enhanced (Erweitert):** Dem Modell werden detaillierte Kenntnisse über die Daten und statistische Richtwerte zur Verfügung gestellt. Ziel ist es, das Modell in einen nach Regeln arbeitenden Rechner zu verwandeln.

### **Systematic (Systematisch):** Es werden strenge, fast algorithmische Anweisungen und Verbote eingeführt, um maximale Kontrolle über das Ausgabeformat zu erlangen**.**

### **ML (Maschinelles Lernen):** Das Modell erhält keine Regeln, sondern muss diese selbst aus den bereitgestellten Beispielen ableiten (In-Context Learning).

### **Systematisches Testen:** Wir stellten beiden Modellen 9 präzise, überprüfbare Fragen und wendeten dabei jeden der fünf Ansätze an. Um die statistische Signifikanz zu gewährleisten, wurde jeder Test dreimal durchgeführt.

### **3. Wichtigste Ergebnisse und Schlussfolgerungen**

### **Das Experiment bestätigte unsere ursprüngliche Hypothese vollständig und lieferte mehrere wichtige Erkenntnisse:**

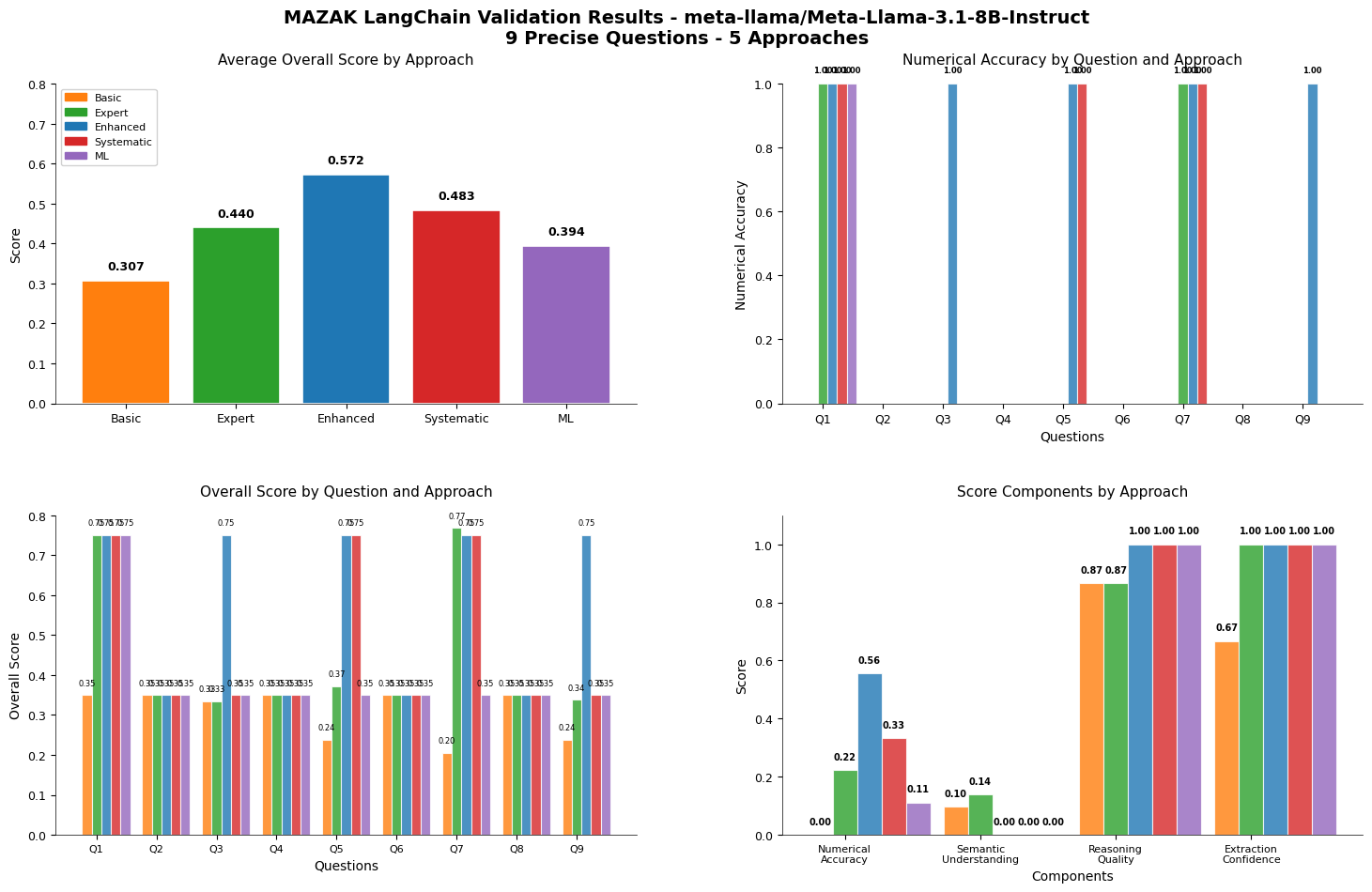
### **Machbarkeit des Ansatzes nachgewiesen:** Mit der richtigen Prompting-Strategie (Enhanced) zeigte das kleinere Llama 3.1 8B-Modell eine gute Genauigkeit und beantwortete 5 von 9 numerischen Fragen korrekt, während das größere Modell nur 2 von 9 bewältigte. Dies beweist, dass LLMs erfolgreich Daten extrahieren und berechnen können, ohne auf vordefinierte Algorithmen angewiesen zu sein.

### **"Größer" ist nicht immer "besser":** Das riesige 405B-Modell scheiterte im Test und gab nur 2 richtige Antworten. Dies deutet darauf hin, dass für Aufgaben, die eine präzise Befolgung von Anweisungen erfordern, ein richtig ausgewähltes und durch Prompts "trainiertes" kleineres Modell wesentlich effektiver sein kann.

### **Der Prompt als "temporärer Programmcode":** Der Erfolg des Enhanced-Ansatzes zeigt, dass ein Prompt als dynamisch erstellter Programmcode betrachtet werden kann. Indem wir dem Modell das benötigte Wissen und die Regeln in Textform zur Verfügung stellen, "programmieren" wir es im Grunde "on the fly" für die Lösung einer spezifischen Aufgabe.

### **Gesamtfazit:** Diese Untersuchung zeigt, dass die Technologie des evolutionären Prompt-Engineerings den Weg für die Schaffung universeller, flexibler und intuitiver Schnittstellen für die Interaktion mit beliebigen strukturierten Daten ebnet. Dies ermöglicht den Übergang von starr kodierten Systemen zu dynamischen, in denen jeder Benutzer durch das Stellen von Fragen in natürlicher Sprache zum Analysten werden kann.

### 



### 

### **Analyse der Ergebnisse: Llama-3.1 8B vs. 405B**

### Diese Analyse vergleicht die Leistung von zwei Sprachmodellen, meta-llama/Meta-Llama-3.1-8B-Instruct (ein kleineres 8-Milliarden-Parameter-Modell) und meta-llama/Meta-Llama-3.1-405B-Instruct-FP8 (ein sehr großes 405-Milliarden-Parameter-Modell), bei der Beantwortung von neun präzisen Fragen zum MAZAK-Datensatz.

#### **1. Analyse des 8B-Modells**

### **Beste Strategie:** Der Enhanced-Ansatz ist mit einer durchschnittlichen Gesamtbewertung von 0.572 der klare Gewinner. Diese Strategie, die dem Modell detailliertes Fachwissen und statistische Richtwerte "beibringt", funktioniert für dieses Modell am besten. Der Systematic-Ansatz ist mit 0.483 eine solide zweite Wahl.

### **Numerische Genauigkeit:** Das Modell zeigt eine beeindruckende Leistung. Mit dem richtigen Prompt (meist Enhanced) beantwortet es 7 von 9 Fragen (Q1, Q3, Q4, Q5, Q7, Q8, Q9) mit perfekter numerischer Genauigkeit (1.0). Bei den Fragen Q2 und Q6 (Anzahl der Programme) versagt es jedoch vollständig. Dies deutet auf ein "Alles-oder-Nichts"-Verhalten hin.

### **Komponenten der Bewertung:** Die Stärken des Modells liegen in der Reasoning Quality (Qualität der Argumentation) und der Extraction Confidence (Sicherheit bei der Extraktion von Zahlen), die durchweg hoch sind. Die größte Schwäche ist das Semantic Understanding (semantisches Verständnis), was bedeutet, dass das Modell zwar die richtige Zahl liefert, aber nicht unbedingt die relevanten Fachbegriffe im Antworttext verwendet.

#### **2. Analyse des 405B-Modells**

### **Beste Strategie:** Überraschenderweise ist hier der Systematic-Ansatz mit einer durchschnittlichen Bewertung von 0.439 am besten. Noch überraschender ist, dass die Bestleistung deutlich unter der des 8B-Modells liegt (0.439 vs. 0.572).

### **Numerische Genauigkeit**: Hier zeigt sich die größte Schwäche des 405B-Modells. Es beantwortet nur 2 von 9 Fragen (Q1 und Q7) numerisch korrekt. Bei allen anderen Fragen, bei denen das 8B-Modell erfolgreich war, scheitert das größere Modell komplett.

### **Komponenten der Bewertung:** Ähnlich wie das 8B-Modell hat es hohe Werte bei Reasoning Quality und Extraction Confidence. Die Werte für Numerical Accuracy und Semantic Understanding sind jedoch extrem niedrig. Das bedeutet, das Modell gibt gut formatierte Antworten, aber die darin enthaltenen Zahlen sind fast immer falsch.

### **Vergleichende Analyse und wichtigste Erkenntnisse**

### **"Größer" ist nicht immer "besser".** Das überraschendste und wichtigste Ergebnis ist, dass das wesentlich kleinere 8B-Modell das massive 405B-Modell bei dieser spezifischen Datenanalyseaufgabe deutlich übertrifft. Es ist sowohl in der durchschnittlichen Gesamtbewertung als auch, was entscheidend ist, in der numerischen Genauigkeit weitaus besser.

### **Numerische Präzision als entscheidender Faktor.** Der Hauptgrund für den Sieg des 8B-Modells ist seine Fähigkeit, die in den Prompts gegebenen Regeln und statistischen Hinweise korrekt anzuwenden und präzise Berechnungen durchzuführen. Das 405B-Modell scheitert genau an dieser Aufgabe, obwohl es theoretisch über weitaus mehr "Wissen" verfügt. Dies könnte darauf hindeuten, dass sehr große Modelle für kreative und komplexe logische Aufgaben optimiert sind, aber bei einfachen, regelbasierten Berechnungen "zu viel nachdenken" oder die Anweisungen ignorieren.

### **Die beste Prompt-Strategie hängt vom Modell ab**. Für das 8B-Modell funktioniert der Enhanced-Ansatz am besten. Das "Lehren" von Fachwissen hilft dem kleineren Modell, die richtigen Schlüsse zu ziehen. Für das 405B-Modell funktioniert der Systematic-Ansatz am besten. Dies deutet darauf hin, dass größere Modelle von extrem strengen und unmissverständlichen Anweisungen profitieren, die keinen Raum für Interpretationen lassen.

### **Gemeinsame Schwachstellen.** Beide Modelle haben Schwierigkeiten mit den Fragen Q2 und Q6, die sich auf die Zählung der spezifischen Programme beziehen. Dies könnte auf eine Unklarheit im Prompt für diese speziellen Fragen hindeuten. Ebenso zeigen beide Modelle ein schwaches semantisches Verständnis, was für rein numerische Aufgaben jedoch weniger kritisch ist.

### **Fazit und Empfehlung**

### Für Aufgaben, die eine präzise numerische Extraktion und Berechnung basierend auf bereitgestellten Regeln und Kontext erfordern, ist das Llama-3.1-8B-Modell in Kombination mit dem Enhanced-Prompt die eindeutig überlegene, effizientere und kostengünstigere Wahl.

### Das Llama-3.1-405B-Modell ist trotz seiner enormen Größe für diese Art von faktenbasierter, instruktionsgesteuerter Analyseaufgabe nicht geeignet und liefert unzuverlässige Ergebnisse. Es ist möglich, dass seine Stärken in anderen Bereichen wie kreativem Schreiben, komplexer Argumentation oder breiterem Allgemeinwissen liegen, die in diesem Test nicht bewertet wurden.

### 

## **Strategische Weiterentwicklung des Projekts zur LLM-basierten Datenanalyse**

Zusammenfassung:

Die initiale Testphase hat eine fundamentale Erkenntnis geliefert: Für präzise, faktenbasierte Analyseaufgaben ist das kleinere Llama-3.1-8B-Modell in Kombination mit einem durchdachten Enhanced-Prompt dem wesentlich größeren 405B-Modell deutlich überlegen. Dieser Erfolg bildet die Grundlage für die nächste Phase des Projekts. Mit der nun gesicherten Finanzierung und dem Zugang zu Fachexperten schlagen wir eine strategische Weiterentwicklung in vier Kernbereichen vor, um aus dem Proof-of-Concept ein robustes, skalierbares und wertschöpfendes Analysewerkzeug zu entwickeln.

### **1. Optimierung und Industrialisierung der bestehenden Lösung (Production)**

**Ziel:** Die aktuelle, erfolgreiche Kombination (8B-Modell + Enhanced-Prompt) stabilisieren, beschleunigen und für den produktiven Einsatz vorbereiten.

* **1.1. Fine-Tuning des 8B-Modells**
  + **Problem:** Der Enhanced-Prompt ist sehr effektiv, aber auch lang und damit bei jedem API-Aufruf kosten- und zeitintensiv.
  + **Lösung:** Ein Team aus ML-Experten und Fachexperten sollte einen hochwertigen Datensatz mit Hunderten von Beispielen erstellen (Frage, Kontext, korrekte numerische Antwort). Mit diesem Datensatz wird das Llama 3.1 8B-Modell gezielt nachgeschult (Fine-Tuning).
  + **Ergebnis:** Das Modell "internalisiert" das Wissen über die MAZAK-Datenstruktur. Dies ermöglicht wesentlich kürzere Prompts, schnellere Antwortzeiten und potenziell eine noch höhere Genauigkeit.
* **1.2. Entwicklung einer robusten API**
  + **Beschreibung:** Das bestehende Skript wird in einen stabilen, dokumentierten Webservice (API) überführt. Diese API dient als zentrale Schnittstelle für alle zukünftigen Anwendungen.
  + **Anforderungen:** Fehlerbehandlung, Logging, Caching-Mechanismen und ein strukturierter JSON-Output sind essenziell.
* **1.3. Aufbau einer automatisierten Validierungs-Pipeline**
  + **Beschreibung:** Implementierung eines automatisierten Testsystems. Jede neue Version des Modells oder eines Prompts durchläuft automatisch die bewährte Test-Suite (die 9 Fragen und erweiterte Tests), um eine gleichbleibend hohe Qualität sicherzustellen und Regressionen zu vermeiden.

### **2. Vertiefte Forschung und Entwicklung (R&D)**

**Ziel:** Ein tieferes Verständnis für das Verhalten der Modelle entwickeln und die technologischen Grenzen erweitern.

* **2.1. Tiefenanalyse des Scheiterns des 405B-Modells**
  + **Fragestellung:** Warum war das wesentlich größere Modell nicht in der Lage, einfache Rechenanweisungen zu befolgen?
  + **Vorgehen:** Ein Expertenteam sollte gezielte Experimente entwerfen, um die Fähigkeit des Modells zum "Instruction Following" bei unterschiedlichen Komplexitätsgraden zu testen. Die Erkenntnisse aus dieser Analyse sind nicht nur für dieses Projekt, sondern für die gesamte LLM-Community wertvoll.
* **2.2. Benchmarking alternativer Modelle**
  + **Beschreibung:** Das 8B-Modell hat gewonnen, aber ist es das Optimum? Es sollte ein systematisches Benchmarking anderer Modelle im 7B-15B-Parameterbereich (z. B. von Mistral, Gemma) durchgeführt werden, um das beste Modell in Bezug auf Genauigkeit, Geschwindigkeit und Kosteneffizienz zu identifizieren.
* **2.3. Weiterentwicklung zum dialogfähigen Agenten**
  + **Beschreibung:** Das System sollte über Einzelanfragen hinausgehen und kontextbezogene Folgefragen beantworten können.
  + **Beispiel:**
    - *Nutzer:* "Wie hoch ist die Gesamtzahl der aktiven Datensätze?"
    - *Modell:* "40.908."
    - *Nutzer:* "**Und wie verteilt sich diese Zahl auf die drei beliebtesten Programme?**"
  + **Technologie:** Dies erfordert die Implementierung eines Conversation Memory.

### **3. Erweiterung des Funktionsumfangs und der Anwendungsfälle (Features)**

**Ziel:** Den Geschäftswert des Tools durch neue, intelligente Funktionen steigern.

* **3.1. Automatisierte Berichterstellung**
  + **Beschreibung:** Anstatt nur auf eine Frage zu antworten, soll das Modell in der Lage sein, kurze, prägnante Zusammenfassungen und Berichte in natürlicher Sprache zu erstellen.
  + **Beispiel-Prompt:** "Erstelle einen Performance-Bericht für die letzte Woche und nenne die wichtigsten Kennzahlen: Anteil des Automatikmodus, das meistgenutzte Programm und die gesamte aktive Maschinenlaufzeit."
* **3.2. Anomalieerkennung und proaktive Insights**
  + **Beschreibung:** Das System soll nicht nur reaktiv antworten, sondern proaktiv auf ungewöhnliche Muster in den Daten hinweisen.
  + **Beispiel-Output:** "Hinweis: Der Anteil des manuellen Modus für Programm 'XYZ' ist diese Woche um 30 % gestiegen, was untypisch ist. Eine Überprüfung wird empfohlen."
* **3.3. Integration in BI-Systeme (z. B. Tableau, Power BI)**
  + **Beschreibung:** Entwicklung eines Plugins, das es Anwendern ermöglicht, direkt in ihren Dashboards Abfragen in natürlicher Sprache zu stellen. Das LLM generiert im Hintergrund die entsprechende Datenvisualisierung.

### **4. Ressourcen und Teamstruktur**

**Ziel:** Ein schlagkräftiges, interdisziplinäres Team aufbauen und die notwendige Infrastruktur bereitstellen.

* **4.1. Empfohlene Teamstruktur**
  + **ML Engineer:** Verantwortlich für Modellauswahl, Fine-Tuning und Prompt Engineering.
  + **Software Engineer (Backend):** Entwickelt und wartet die API und die Infrastruktur.
  + **Fachexperte (Produktion/CNC):** Definiert Business-Anforderungen, validiert die Korrektheit der Antworten und identifiziert neue Anwendungsfälle.
  + **Projektmanager:** Steuert die Roadmap und priorisiert die Aufgaben.
* **4.2. Infrastruktur-Investitionen**
  + Mit der vorhandenen Finanzierung sollte in dedizierte GPU-Server (On-Premise oder Cloud) investiert werden. Dies ermöglicht kosteneffizientes Fine-Tuning und Hosting des Modells, reduziert die Latenz und verringert die Abhängigkeit von externen API-Anbietern.

### **Fazit**

Das Projekt hat bereits einen entscheidenden Durchbruch erzielt. Die vorgeschlagenen Schritte werden es ermöglichen, diesen Erfolg zu industrialisieren, die technologische Basis zu vertiefen und ein intelligentes Werkzeug zu schaffen, das von der reinen Datenabfrage zur proaktiven Gewinnung von Geschäftseinblicken übergeht.